

2005

# Building and Maintaining Alliances in SPI Projects: Implications for Organizing Effective SPI

Jacob Norbjerg

*Copenhagen Business School, jno.itm@cbs.dk*

Ojelanki Ngwenyama

*Ryerson Polytechnic University, ojelanki@ryerson.ca*

Follow this and additional works at: <http://aisel.aisnet.org/ecis2005>

## Recommended Citation

Norbjerg, Jacob and Ngwenyama, Ojelanki, "Building and Maintaining Alliances in SPI Projects: Implications for Organizing Effective SPI" (2005). *ECIS 2005 Proceedings*. 143.

<http://aisel.aisnet.org/ecis2005/143>

This material is brought to you by the European Conference on Information Systems (ECIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ECIS 2005 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact [elibrary@aisnet.org](mailto:elibrary@aisnet.org).

# BUILDING AND MAINTAINING ALLIANCES IN SPI PROJECTS: IMPLICATIONS FOR ORGANIZING EFFECTIVE SPI

Jacob Nørbjerg, Department of Informatics, Copenhagen Business School, Njalsgade 80, DK-2300 Copenhagen S, Denmark, [jan.inf@cbs.dk](mailto:jan.inf@cbs.dk)

Ojelanki Ngwenyama, Information Technology Management, Ryerson University, 350, Victoria Street, Toronto, ON, Canada, MSB 2K3, [ojelanki@ryerson.ca](mailto:ojelanki@ryerson.ca)

## Abstract

*Software Process Improvement projects face a double challenge: On the one hand are these projects often major organizational change projects requiring considerable investments in terms of time and money. The group charged with conducting an SPI project has, on the other hand, little formal authority to influence or force software professionals to engage in SPI work and define and implement changes. In this research in progress paper, we present first results from our analysis of how an SPI group can use alliances to obtain influence and induce change. Our analysis shows that the concept of alliances is useful both to understand organizational influence and to provide practical advice for practitioners involved in organizational processes such as SPI.*

# 1 INTRODUCTION

Since the early 1990s many software development organizations have turned to Software Process Improvement (SPI) as a means to improve the predictability, quality and productivity of their software processes (Aaen & Arent & Mathiassen & Ngwenyama 2001, Mathiassen & Pries-Heje & Ngwenyama 2001, Herbsleb & Zubrow & Goldenson & Hayes & Paulk 1997, Paulk & Curtis & Chrissis 1995). An SPI approach involves deliberate, planned, and ongoing changes to all parts of a software organization: day-to-day and long term planning and management, the organization of projects, the processes used in software development, and the tools needed to support software development and management. Successful SPI can produce several benefits for a software development organization, but it is also a very complex and time-consuming organizational change process (Herbsleb et. al. 1997, Mathiassen et. al. 2001). A survey of SPI projects has shown that implementation of stable project management practices takes 18-24 months, and getting from there to shared organizational software development process standards another 12-18 months (Herbsleb et. al. 1997). Studies have also shown a considerable failure rate for SPI projects (Ngwenyama & Nielsen 2003).

Experience from SPI projects underline the importance of high and sustained commitment from all levels of management as well as the software developers. Organizational politics, turf guarding, and cynicism towards change processes are, on the other hand, considered the most important threats to SPI (Herbsleb et. al. 1997). It is, however, less clear how an organization undertaking SPI can secure and maintain the necessary support from all stakeholders, and how it can identify and overcome the potential threats.

The SPI literature generally recommends that responsibility for planning and conducting SPI be organized around a separate organizational entity such as a dedicated SPI group or department (c.f. McFeeley 1996). The group's responsibilities include: production and dissemination – including training – of software development process descriptions, implementation of development and management tools, and establishment of procedures for collection and use of process and product metrics. The group is, however, not part of the management hierarchy and has no direct or formal way to make changes happen; nor does it control critical organizational resources such as, time, personnel or careers. The SPI group is therefore without the means normally associated with organizational power and influence (Porter, Angle and Allen 2003a). The formation and staffing of an SPI group does signal management's interest in SPI and consequently lends some authority to the group, but initial resource allocation is not the same as sustained and, more importantly, continued management commitment and support to a long term effort such as SPI (Abrahamsson 2001). Furthermore, important stakeholders, e.g., software professionals and project managers may support or resist SPI for reasons of their own (Abrahamsson 2001, Nielsen & Nørbjerg 2001).

The challenge for the SPI group is therefore how it can influence other members of an organization to support the SPI effort by e.g.; following new management and development processes, implementing quality assurance measures etc. In the research presented here, we use the concept of *alliances* (Cohen & Bradford 2003) to analyze how an SPI group manages to influence key organizational members to support and further the SPI project. The research is based on data collected during three years of action research in SPI projects in two different organizations. For reasons of space limitations this paper only discusses findings from one of the organizations.

Our research contributes to SPI practice and research as well as to organizational research more generally. First, there is a need, as discussed above, to learn about the factors contributing to SPI success (or failure) as well as practical advice for SPI professionals. Second, research on organizational power and influence processes has been mainly concerned with *vertical* (i.e. up and down the formal hierarchy) influence processes, largely ignoring the way groups or individuals influence others *laterally*. There is, however, increased recognition of the need to understand lateral

influence processes as organizations change towards flatter hierarchies and a more networked structure (Porter et. al. 2003b). A study of alliances in SPI; how they are formed, sustained and (perhaps) dissolve, is a valuable contribution in these regards.

In the next section we briefly discuss organizational influence and power, and we examine the concept of *alliances* as a way to obtain organizational influence. Section 3 describes the research setting and approach and section 4 gives a short overview of the SPI project under study. Section 5 analyses how the SPI group on the one hand managed to build and use alliances to reach its ends and on the other how – as events unfolded – alliances were eventually dismantled. Section 6 discusses the findings and section 7 concludes the paper.

## 2 ORGANIZATIONAL INFLUENCE AND ALLIANCES

It is not simple to define organizational power, let alone determine how it is obtained and used to influence others (Salancik & Pfeffer 2003). In their collection of theoretical and empirical studies of organizational power and influence processes Porter et. al. portray power as a resource, “a reservoir of force” that a person can use to change the behavior of another (2003a, p. 3). In line with the strategic consistency theory of power (Hickson & Hinings & Lee & Schneck & Pennings 2003) they go on to relate *power* closely to *control* over and *dependence* on resources: If a person, B, depends (in some sense) on A, then A is in a position to exert influence over B. Key to this is A’s control over *resources* that B needs and cannot – or only with considerable difficulty – get access to without A’s consent (Porter et. al. 2003a). Note that resources are understood broadly to include visible resources such as a job, money, promotion or time, and less tangible such as advise, knowledge or appreciation. Thus, organizational power is not linked to formal management hierarchies but can be based on; e.g. (expert) knowledge, pivotal roles in organizational processes or prestige (Porter et. al. 2003a, Salancik et. al. 2003).

Above, power and influence is seen as a one-way process; i.e. a person with access to resources can influence another person (or group) to do something. But how can a person that does not have that control over resources make change happen, especially when the persons (or groups) whose effort is needed are busy or do not agree with the need for change? In such cases Cohen & Bradford (2003) propose to see influence not as a one-way but mutually beneficial *alliance* between organizational members. Alliances depend on the idea of reciprocity; i.e. that organizational transactions are built upon the exchange of something for something else. The alliance partners must therefore agree on the “goods” or “currencies” to exchange and their value for each. Currency must be understood in broad terms to include more than material goods or tangible resources. Examples of currencies are being allowed to excel in what one knows well, appreciation, time and effort, increased visibility, information, friendship, and inclusion in networks. Building and sustaining alliances require mutual recognition and understanding and – not the least – that the party wanting to build an alliance is able to build trust with the potential ally.

A newly formed SPI group does not have immediate access to the resources usually related to power. The group has no managerial authority over the software developers and managers whose help they need to identify software process problems, find solutions and – usually the most difficult task – change the way software projects are conducted and managed. The members of the group may be experts in software engineering and software project management but such knowledge is not necessarily scarce among software professionals and/or those professionals may believe themselves to be (at least) as knowledgeable in matters of software development, as the members of the SPI group. Furthermore, the software professionals may not be convinced that the benefits of SPI; i.e. better process control, and higher productivity and quality, are worth their investment in effort and time, or they may fundamentally disagree with the assumptions and goals of the SPI project (Nielsen et. al. 2001). In this situation, as discussed above, alliance building may be a fruitful way for the SPI group to gain influence.

This raises the following questions with respect to SPI projects:

- Who are the potential allies?
- What does the SPI group need from these allies?
- What can the SPI group offer in exchange?
- How can the SPI group initiate an alliance?
- How are alliances sustained?
- What causes alliances to fail or fall apart?

Cohen and Bradford (2003) do not describe the processes of forming and dissolving alliances. In the following we will, however, use the concept of alliances, understood as mutually beneficial exchange of “currencies” between organizational members or groups, to analyze data collected during a longitudinal action research project in a Danish software development organization. The analysis and subsequent discussion serves as (1) a first answer to the questions raised above; (2) exploration of the usefulness of the alliance concept in understanding and guiding SPI projects. The results further serve as a basis for further exploration of lateral influence processes in organizations.

### **3 RESEARCH SETTING AND METHOD**

This research is based on an action research project in a Danish software organization. The research project followed the guidelines for *Collaborative Practice Research* (Mathiassen 2002). The project took place in the period 1997-1999 during which one of the authors participated in the organization’s SPI activities together with other researchers, external consultants and SPI professionals from the company. The researchers and consultants participated in regular meetings in the SPI group and worked with software developers and project managers engaged in specific improvement initiatives.

The research was documented through tape-recordings of meetings, meeting minutes, taped interviews with key stakeholders, diaries, e-mails, field notes and internal material from the company.

For the present research, e-mails, minutes, interview transcripts, field-notes and diaries produced during the three-year period have been analyzed by one of the authors. The coding has been based on Cohen and Bradford’s (2003) idea of alliances and aimed to answer questions such as: What alliances were formed? How did the SPI group identify potential allies? What currencies were used to establish and sustain alliances and what caused alliances to break down?

The first analysis was documented as an extended narrative, which was discussed among the authors. The present paper presents a condensed version of the narrative and a first discussion of the findings.

### **4 THE SPI PROJECT AT B&K**

B&K develops leading edge measurement instruments and systems. A typical product consists of a measuring device sometimes with embedded software, connected to a handheld or desktop computer with analysis and presentation software. Development projects usually have both hardware and a software part, but they are run as integrated projects under one project manager. The SPI project at B&K was only concerned with software development projects.

During the SPI project there were around 7-10 active development projects at any one time, typically running 6-18 months each with 5-10 software developers. A technical director headed the development department. A new director had been appointed shortly before commencement of the SPI project. He was replaced about half way through the project.

The SPI group consisted of external researchers and consultants, and one full time and 2-3 part-time software professionals and project managers from B&K. The researchers and consultants spent 1-2 days/month at B&K and participated in information gathering, workshops and training sessions, and took active part as coaches and consultants in specific improvement activities.

The project resulted in both successful and failed software process improvement initiatives. The following description of the project divides it roughly into three main phases, identified by the main focus of the project:

- Identification of improvements: January – October 1997
- Improvement initiatives: November 1997 – mid 1999
- Re-assessment and planning of new improvements (mid to end 1999)

#### **4.1 Identification of improvements**

The SPI group used the first meetings to get acquainted and to search for an approach to the SPI project. B&K's own SPI professionals suggested to link improvement initiatives to ongoing projects, so that new (improved) software processes could grow from practice, rather than be written directly into the quality system. The approach was generally accepted as the best way to ensure acceptance among the three important groups of stakeholders: developers, project managers and top management.

To implement this approach the SPI group needed precise knowledge of the character of current processes and potential improvement areas. The SPI group therefore decided to launch a series of interviews with project managers to identify problems and collect suggestions for solutions. The group conducted 7 interviews with project managers over a 3-month period. The interviews focused on present practices, software process problems identified by the project manager, and the project manager's suggestions on how to solve the problems (Iversen & Nielsen & Nørbjerg 1999).

The interview series resulted in a report summarizing the SPI group's interpretation of B&K's process problems and a list of suggested improvements; e.g. an iterative software process model, requirements specification, or reuse. The report was presented to the project managers and top management at a workshop in October 1997. At the end of the workshop, the project managers chose which improvement they would work with in their current or coming projects.

#### **4.2 Improvement initiatives**

The improvement initiatives resulting from the workshop ran from December 1997 through the first 4 months of 1999. Experiences from successful initiatives were summarized in reports and the new processes spread through 'word-of-mouth' and by inclusion into the quality system.

Four improvement projects were launched:

- Iterative development process (December 1997 - March 1999)
- Requirements specification (second quarter 1998 - February 1999)
- Reuse (November December 1998; failed)
- Project tracking (November 1997 - second quarter 1998, failed)

Each of these initiatives was associated with one or more development projects. Members of the SPI formed a support group for the development projects involved in improvement initiatives.

The support group met regularly with the development group to coach them in usable processes, techniques, and tools, and to collect information about the projects' progress and the experience with the new processes. Suggestions for changes to practice or for adjusting the process description as and when needed were also made by the support group. The resulting processes were ultimately described and included in B&K's quality system.

#### **4.3 Re-assessment and planning new improvements**

With the conclusion of ongoing improvement projects it was time to take stock and plan future activities. A Bootstrap assessment (with outside assessors) confirmed that improvements had been made in some areas but also pointed to new or so far unaddressed problems. The approach used

previously (work with individual development projects) was tried once more but without much success. The SPI group realized that it did not quite know how to proceed and decided to make a second round of interviews with both project managers and top management to get an evaluation of the SPI project and ideas for future activities. During and after the interview series the SPI group attempted to involve top management in the planning of future SPI initiatives but was unsuccessful.

With key SPI persons leaving B&K and dwindling support from top management, the end of 1999 effectively terminated B&K's SPI project.

## **5 CREATING, MAINTAINING AND LOOSING ALLIANCES AT B&K**

The organizational model of SPI at B&K anchored the improvement initiatives in development projects. Members of the SPI group helped pilot project teams define and apply new processes. The experiences gained in the pilot projects were used to define organization wide processes that were distributed via B&K's intranet.

The SPI group identified top management, project managers and project teams as the most important allies. In the course of the project the group managed to form working alliances with project managers and team members. The relationship with top management was, on the other hand, never very successful.

The following analysis focuses on the building of alliances with project managers and teams, and the less successful alliances with top management. The analysis will concentrate on the currencies used and the outcome of the alliances

### **5.1 Committing project managers: obtaining knowledge**

B&K's project managers were a powerful group whose support was crucial for the success of the SPI project. The SPI group therefore quickly decided to focus its attention on project managers and software developers while in practice downplaying the relations with upper management.

At the outset B&K's project managers were skeptical towards SPI, and they were not ready to adopt new processes for which they saw no immediate use. The SPI group was convinced, however, that there were a number of problems with B&K's software processes such as delays, faulty products, and poor planning and oversight of projects. To the project managers, on the other hand, were these problems largely caused by unrealistic expectations from the sales and marketing departments and overly bureaucratic planning processes (Nielsen et. al. 2001). Thus, they did not necessarily see a need to change their own practices.

Furthermore, engaging in improvement initiatives is not without costs and risks for the projects: There is an up front investment in time needed to learn and adapt new processes and techniques and there is always a risk that the new practices are no better, or even worse, than the old.

The SPI group therefore had to convince the project managers, both that there was room for improvements in development projects, and that collaboration with the SPI group would support these improvements. The SPI group soon realized that this would require detailed knowledge about B&K's software development practices and problems.

The main currency used by the SPI group in its approach to the project managers were: knowledge of software process improvement as well as of specific software processes and techniques that could be useful for the project managers. The project managers, on the other hand, would have to allocate time to SPI activities in exchange. The challenge for the SPI group was to make its software process knowledge useful for the potential partners; i.e. to convince project managers that spending time with the SPI group would give them useful knowledge.

The series of project manager interviews helped the SPI group build its assets in terms of knowledge about B&K's software processes and the project managers' own improvement ideas and suggestions. When the SPI group presented the results of the interviews, it was clear that the project managers accepted the interpretation of their process problems and suggestions for improvements (Iversen et. al. 1999).

## **5.2 Working with teams: trading knowledge and time**

Improvement initiatives were anchored in ongoing development projects as described above. A project group would apply a new or changed process and the SPI group would help fine-tune the process and collect the project group's experiences. The end result would be a tested and verified software process ready to disseminate to the rest of the organization.

Looked at as an alliance, this process can be described as follows: the SPI group trains or otherwise supplies knowledge of useful processes to the project group. The project group uses the processes in its daily practices and gives the SPI group access to its experiences, which the SPI group systematizes and disseminates to the rest of the organization. The currencies used in the exchange are knowledge and training from the SPI group, and time from the project group

This produces some challenges for the SPI group wanting to form the alliance: The project managers had – as explained above – accepted the SPI group's suggestions for improvements, and they had all committed to participate in at least one improvement initiative. But the commitment was informal and the project managers were able to back out of the improvement initiative at any time without any sanctions. All the SPI group had to offer in the alliance was knowledge that might or might not be useful for the project group. The project group had to invest time, both to learn and adapt the new processes, and to provide feed-back to the SPI group without knowing if the new processes were better than the old, thus running the risk of project delays and/or decreased product quality. The SPI group did not, however, have the authority required to guarantee the projects the resources or "slack" they needed on this account.

The SPI group was able to strengthen its position in some cases. In the area of requirements engineering, for example, previous improvement initiatives in B&K had already demonstrated the usefulness of a number of specific techniques, and other project managers wanted to learn about these. The SPI group could, therefore, offer training in return for the projects' time and effort.

In other cases it was less obvious what the SPI group had to offer in return for the project team's time and effort. In the case of development process models, for example, it was clear that an iterative development process would fit B&K's needs better than the current waterfall process. It was less obvious, however, exactly what kind of iterative process would work. All the SPI group had to offer, therefore, was a general overview of iterative approaches to software development, not a ready-to-use process or set of techniques. It was up to the development project – assisted by the SPI group – to adapt the general iterative approaches to its own needs. The SPI group – on the other hand – needed frequent access to the project group in order to learn about the group's experiences with the iterative processes.

Not surprisingly, the project team working with the new development process questioned this arrangement in various ways. At the initial meeting with the SPI group, project team members wanted confirmation that the new processes would be better than the old. During subsequent meetings they would frequently ask for very detailed advice on a number of problems, ranging from specific planning and follow-up techniques to user interface and design architecture. The members of the support group felt that the developers wanted them to be consultants and teachers on topics unrelated to the improvement initiative. They were, therefore, not immediately prepared to meet all the developers' requests, leading to a series of crises in the collaboration.

When viewed from the perspective of alliance formation, this is an example shows how the SPI group and the project team members did not agree on the currencies to be exchanged in the alliance. In



exchange for its time and effort, the project team wanted knowledge and training beyond what the support group was prepared to offer. The turning point came when the support group was able to use the meetings with the project group for a combination of coaching and experience gathering. In these meetings, the developers informed the SPI group about their progress and experiences with the iterative development process, while the SPI group gave feed-back and advice on how to solve specific problems and avoid potential pitfalls. Thus the two groups managed to reach an agreement on what to exchange in the alliance.

### **5.3 (Not) dealing with management**

Management support for SPI projects is obviously important because management grants the SPI project the resources and 'rights' to work with the organization's software processes. From a management perspective, SPI is a long-term investment in improving productivity, predictability and product quality that often delivers no immediate results.

Although the newly appointed technical director of B&K stated that 'SPI was already on his table' [Interview with technical director 1999] when he assumed his position, it was never high on his list of priorities. The SPI group experienced his lack of interest in several ways: First, he allocated relatively few resources to the SPI project. Second, it was difficult to get a clear statement of his vision and purpose for the SPI project. Finally he would not explicitly allocate time to improvement initiatives within projects.

The SPI group had few assets in its dealing with the technical director. A potential asset of course, is a project plan and the following status reports that can help convince management that something is going on. The group discussed such plans during its first meetings, but when the interview series with project managers began, the SPI group more or less ceased to think about top management and concentrated on the interviews. The SPI group later learned that this was a nearly fatal decision. The technical director began to see the interviews conducted by the SPI group as merely 'studying the fish in the aquarium' [Interview with technical director 1999], making no real progress in any useful direction. He seriously considered closing down the SPI project. However, the SPI group misread the director, thinking that management was willing to leave SPI alone – if not actively support it. However, it was an SPI workshop that partly changed the technical director's opinion. During the workshop he was able to see purpose and direction in the SPI group's work. But this did not result in more interaction/attention between the SPI group and upper management.

The appointment of a new technical director in 1998 did not change much from the SPI group's point of view. B&K went through some major organizational changes (and lay-offs) and the SPI group was busy with improvement projects. Signals, from 'above' were, that the new director found SPI important, but attempts to establish closer contact with top management failed and during 1999 the SPI group became increasingly frustrated with the lack of management involvement and support.

Towards the end of the SPI project the new technical director acknowledged that the SPI work had produced positive results so far, but he was critical towards what he called the SPI group's 'scientific' theoretical approach [Interview with new technical director 1999]; what was needed were practical initiatives, tools etc. that would create immediate change based on the practical and immediate needs of the projects. He also found that the development organization was capable of defining and implementing improvement initiatives without an SPI group, and he proposed to replace a dedicated SPI group with interest groups and committees of project managers around different topics.

### **5.4 Running out of alliances**

The overall impression from the last part of the project was that the SPI group gradually lost influence in B&K, with top management as well as with software developers and project managers.

Concerning project managers, it appears that the changing nature of improvements required a change in alliance forming strategies. The successful initiatives, process models and requirements specification were focused on individual projects, but the remaining improvements; e.g. reuse and configuration management required coordinated action across projects. Project managers and developers would no longer be free to (individually) adapt processes to solve their own problems, but would have to coordinate with others; thus incurring higher risks for the individual project with less tangible (immediate) benefits.

With these new and changing conditions the SPI group's alliances with individual project managers which were based on the exchange of the project manager's time in return of useful knowledge and solutions to immediate project specific problems, needed to be reformulated. A new strategy for alliances had to be devised to account for the more complex arrangements involving groups of project managers. Furthermore, such an arrangement would require more visible management involvement to ensure (and possibly enforce) cross-project coordination. The SPI group did not acknowledge this; neither did it have an idea of how to approach the problem.

A possible approach could have been to strengthen the alliance with upper management in order to secure support for organizational wide initiatives. But, as discussed above, the SPI group had more or less ignored upper management (perhaps) trusting that the results obtained in the collaboration with the project managers spoke for themselves. Hence, the new technical management was not prepared to continue the support for a dedicated SPI group.

## **6 DISCUSSION**

The analysis of B&K's SPI project gives interesting insights into the dynamics of SPI projects but it also points to broader issues and questions regarding the alliance concept itself as a means to study and guide organizational influence processes.

First, the experience from B&K shows us that building and maintaining alliances is an ongoing process, which has to be continuously monitored and adjusted. Changes brought about by successful SPI alliances, as well as other events, may ultimately undermine the basis of such alliances. Thus, the SPI group must be prepared to change the conditions for alliances or to replace alliance partners. Identifying the conditions and circumstances leading to (changes in) choice of alliance partners and alliance approaches require further studies.

Second, the case demonstrates the challenges involved in initiating an alliance. The critical point for the SPI group in the B&K case was always how to get software developers and project managers ready to engage in SPI in the absence of formal management incentives or directives. As a rule, this requires that the software professionals are ready to accept an 'up front' time and effort investment while the SPI group develops the detailed knowledge necessary to be effective. A topic for further research is why would software professionals accept these conditions. Is it possible that the "currency" during this phase was their wish to excel professionally (Cohen et. al. 2003)? Do they see collaboration with the SPI group as one way to gain access to general knowledge about software processes that they need?

Third, and connected to the first point above, the concept of alliances seems to be a useful vehicle for understanding the dynamics of SPI processes. As discussed previously, the SPI literature emphasizes management commitment as important for SPI success, but our research shows that under certain circumstances SPI projects may make progress without such commitment. Our analysis also demonstrates the risks involved in such a strategy. This leads to a further research question concerning how can SPI groups build and maintain alliances with top management.

## 7 CONCLUSION

In this paper we have described our research in progress aiming to understand how SPI groups build and maintain alliances in order to bring the SPI project forward. Our research so far has convinced us that the concept of alliances is a fruitful way to understand lateral influence processes of the type often required in SPI projects. We also believe that an improved understanding of alliances, their formation, maintenance, and termination can provide useful practical advice for SPI groups.

Our analysis has provided interesting insights but also raised questions for further research. To answer these questions we intend to develop the alliance concept further based on a study of the literature on lateral influence processes as well as on empirical analysis of other SPI projects.

## 8 REFERENCES

- Aaen, I., J. Arent, L. Mathiassen and O. Ngwenyama (2001). A Conceptual MAP of Software Process Improvement. *Scandinavian Journal of Information Systems* **13**: 81-102.
- Abrahamsson, P. (2001). Rethinking the Concept of Commitment in Software PProcess Improvement. *Scandinavian Journal of Information Systems* **13**: 37-61.
- Cohen, A. R. and D. L. Bradford (2003). Influence Without Authority: The Use of Alliances, Reciprocity, and Exchange to Accomplish Work. in *Organizational Influence Processes*. L. W. Porter, H. L. Angle and R. W. Allen (eds). Armonk, New York, M.E. Sharpe: 384-394.
- Herbsleb, J., D. Zubrow, D. Goldenson, W. Hayes and M. C. Paulk (1997). Software Quality and the Capability Maturity Model. *CACM* **40**(6): 31-40.
- Hickson, D.J., C.R Hinings, C.A.Lee, R. E. Schneck and J.M. Pennings (1983) A Strategic Contingencies' Theory of Intraorganizational Power. *Organizational Influence Processes*. R. W. Allen and L. W. Porter. Glenview, Illinois, Scott, Foresman and Company: 33-51.
- Iversen, J., P. A. Nielsen and J. Nørbjerg (1999). Situated Assessments of Problems in Software Development. *The Database for Advances in Information Systems* **30**(2): 66-81.
- Mathiassen, L., J. Pries-Heje and O. Ngwenyama, Eds. (2002). *Improving Software Organizations. From Principles to Practice*. The Agile Software Development Series. Boston, Addison-Wesley.
- Mathiassen, L. (2002). Collaborative Practice Research. *Information Technology and People* **15**(4): 321-345.
- Ngwenyama, O. and P. A. Nielsen (2003). Competing Values in Software Proces Improvement: An Assumption Analysis of CMM From an Organizational Culture Perspective. *IEEE Transactions on Engineering Management* **50**(1): 100-112.
- Nielsen, P. A. and J. Nørbjerg (2001). Assessing Software Processes: Low Maturity or Sensible Practice. *Scandinavian Journal of Information Systems* **13**(1-2): 23-36.
- Paulk, M. C., B. Curtis and M. B. Chrissis (1993). *Capability Maturity Model for Software, v. 1.1*. Pittsburgh, PA, Software Engineering Institute, Carnegie Mellon University.
- Porter, L. W., H. L. Angle and R. W. Allen (2003a). Influence, Power, and Politics in Organizational Settings. in *Organizational Influence Processes*. L. W. Porter, H. L. Angle and R. W. Allen. Armonk (eds), New York, M.E. Sharpe: 3-13.
- Porter, L. W., H. L. Angle and R. W. Allen (2003b). Lateral Influence. *Organizational Influence Processes*. L. W. Porter, H. L. Angle and R. W. Allen (eds). Armonk, New York, M.E. Sharpe: 275-281.
- Salancik, , G. R. and J. Pfeffer (1983). Who Gets Power - And How They Hold on to It: A Strategic-Contingency Model of Power. *Organizational Influence Processes*. R. W. Allen and L. W. Porter. Glenview, Illinois, Scott, Foresman and Company.