



# Relatório Técnico

**Núcleo de  
Computação Eletrônica**

## Collaborative Improvement of Software Processes

Renata Mendes de Araujo  
Marcos R. S. Borges

NCE - 29/99

**Universidade Federal do Rio de Janeiro**

## Collaborative Improvement of Software Processes

Renata Mendes de Araujo  
COPPE - Programa de Engenharia de  
Sistemas e Computação – UFRJ  
Caixa Postal 68511, Rio de Janeiro, RJ  
21945-970, Brasil  
renata@cos.ufrj.br

Marcos R. S. Borges  
NCE & IM / UFRJ  
Caixa Postal 2324  
Rio de Janeiro, RJ  
20001-970, Brasil  
mborges@nce.ufrj.br

### Abstract

*To achieve high levels of product quality, organizations dealing with software development have massively invested in ways to improve their development processes. Many studies reporting factors that influence the success of such initiatives point out to the relevance of social issues to succeed in adopting process improvement. This paper discusses the possibilities for collaboration support in helping organizations to reduce social barriers to process improvement and to establish a process culture among developers. The paper explores this idea by proposing a collaborative environment that aims to promote SPI initiatives. The environment is focused on the role of awareness support, since we believe it is a primary resource for software process improvements to achieve higher levels of group/collaboration and process consciousness.*

**Keywords:** *software process improvement, CSCW, workflow, awareness support.*

### 1 Introduction

For the last years, organizations have been facing the challenge of improving the quality of their working processes, as a strategy to remain safe and competitive. In spite of these efforts, process modeling, process assessment and improvement techniques are unused and sometimes unknown [GIBS97]. In many organizations, where a systematic software development process has been adopted, the process culture is still immature.

The social issue has been pointed out as a relevant factor that influences the success of software process improvement<sup>1</sup> initiatives [PAUL93, SPIC98, GOLD95]. For improvement initiatives to be successful people need to be engaged, accept the initiative being proposed and, most importantly, must understand the cultural process. The great obstacle for the introduction of improvement initiatives is the nonexistence of a “process culture” inside organizations.

People involved in the development process rarely know about how and why processes are defined (if they are defined). Besides, people have a tendency to react to changes because they don't have ways to understand the impacts of changes in their own work. Also, software development processes, although intrinsically collaborative in their nature, are not clearly recognized as a collaborative work. Collaboration and interaction between participants are not explicitly defined, that may prevent participants to recognize possibilities for improvement and to suggest changes.

---

<sup>1</sup> from now on referred as SPI

Based on this scenario, SPI opens a myriad of possibilities for studying and applying collaborative technologies. The aim of this paper is to explore these ideas by proposing and describing a collaborative environment, which can promote SPI initiatives. Among the many aspects concerning collaborative support [ARAU97], our proposal focuses on the role of process perception as a primary resource for participants to achieve higher levels of group/collaboration and process consciousness.

The main objective of this environment is to aid participants in recognizing and understanding their working processes and how their work influence other member's work. By providing this knowledge, its expected that participants would be able to identify potential bottlenecks in their working processes and be encouraged to suggest changes or alternatives.

The next section discusses the issues that influence SPI initiatives. Section 3 outlines the potential of collaborative technologies to promote these initiatives inside organizations. Section 4 presents an overview of the proposed environment, focusing on the definition of the resources offered to provide process and group collaboration awareness. The last section concludes the paper.

## **2 Software Process Improvement**

As noted by Kitson [KITS97], IBM was the first organization to conduct software process evaluations in a systematic way. Following this initiative, in 1987, the Software Engineering Institute (SEI) defined what became the American model for software process evaluation - the Capability Maturity Model or CMM [PAUL93]. The diffusion of this model and the definition of other models and approaches in Europe and Asia motivated the definition of an international model for SPI. This new model aims to incorporating the best aspects of the existent models and to define a framework for continuous improvements. This model, being proposed as an ISO norm - ISO 15504 - is currently being defined and studied by the SPICE project [ELEM98].

The majority of approaches for SPI is based on these reference models and use top-down strategies to their execution. They start from what are considered to be the best practices (defined by the reference models) and try to conduct the improvement based on these models [SPIC98]. On the other hand, inductive approaches define bottom-up strategies for process improvement. These approaches argue that it is necessary to previously understand which processes exist in the organization and to determine which cause problems.

We believe that both approaches are necessary to succeed in software process improvement. While we need reference models to guide de improvement, they must somehow be adapted to each organization context and culture. However, regardless of the approach used, it is a consensus that participants must be involved in the initiative since the very beginning.

### **2.1 Process Culture**

Among the risks in adopting process improvement initiatives, many are related to their acceptability by members of the organization. Along with the implementation of improvement initiatives in an organization, changes are expected to occur not only in the

process but also in attitudes, values and behavior of members involved. The cultural context of an organization turns out to be a key element to plan and to succeed in improvements, influencing and being influenced by such initiative.

The interim report of the second phase of the SPICE project [SPIC98] shows the results of the use of the ISO-15504 in process evaluations in various organizations. One of the contributions of this report is the identification of the factors that impact the success of improvement initiatives and which of them are critical to this success. The results can be summarized in Table 1.

<b>Organizational factors:</b>	
SPI goals well understood	critical
Technical staff involved in SPI	critical
Senior management monitoring of SPI	
SPI people well respected	
Compensated SPI responsibilities	
Staff Time/Resources dedicated to process improvement	
<b>Process factors:</b>	
Creating an action plan	
Creating process action teams	critical
Piloting process changes	

Table 1- Factors that influence SPI success

Many critical factors that influence improvement success deal with those issues related to team involvement into the improvement initiative. The SPICE project mentions that it is important to remove any possible barriers that prevent participants from an effective communication and cooperation [SPIC98]. It also explains that it is needed to communicate and discuss the improvement results before planing or suggesting any recommendations or action plans.

Although collaboration is not explicit to developers, software process is collaborative in its nature. In these processes, activities involve the participation and interaction between many participants. The difficulty to recognize cooperation in process activities may lead participants to ignore the process as a whole, restricting the possibilities to suggest changes and useful improvements.

Some proposals argue that the possibility to suggest and participate on the definition work processes may induce employees to high satisfaction and confidence and increase work quality [BORG99]. Putting participants as “co-responsible” for the process makes them to rethink the way they work, the way

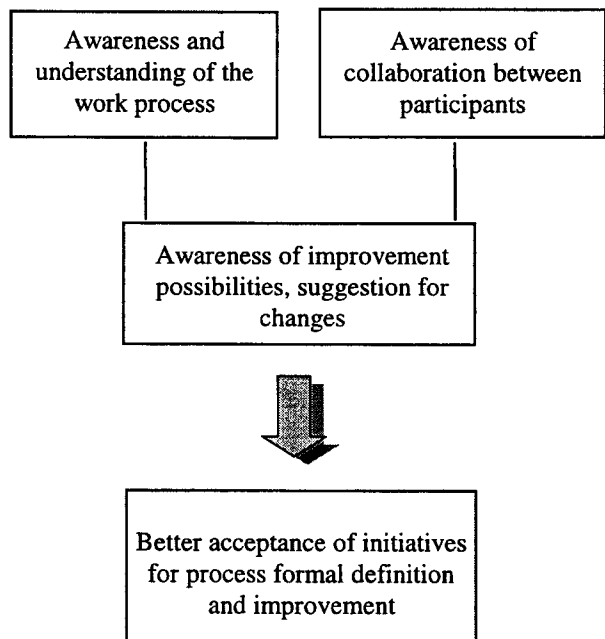


Figure 1 - The role of awareness in SPI

their activities influences others' activities and how they can be improved. The process turns out to be a common place in organization; an object belonging to everybody's domain, ready to be examined, questioned and improved.

In summary, by enforcing collaboration in SPI contexts, it may be expected a change in the organizational "process culture", increasing people's acceptance to improvements. By providing the awareness of processes, putting group collaboration explicit and involving participants in the initiative, it is expected that process actors may better recognize, understand, discuss, formalize, trace and improve their working processes. This idea is depicted in Figure 1.

### 3 A Collaborative Environment for SPI

Our proposal is to provide software process participants with a collaborative environment that enables collaboration and makes working processes more explicit in order to facilitate the establishment of a process culture in an organization. This environment aims to offer special awareness resources to software process participants. During process execution, the environment will provide mechanisms that help participants to observe their work and interactions with other participants and to gradually construct their notion about the process being defined.

As shown in Figure 2, the environment lies over a workflow system where the software process can be modeled and enacted. The use of workflow technology is the first step to provide collaboration in software development and to provide awareness of the process, as we will discuss later. In a higher layer, users are provided with additional awareness mechanisms, not provided by the majority of workflow systems, that are important for helping participants to recognize the way they work.

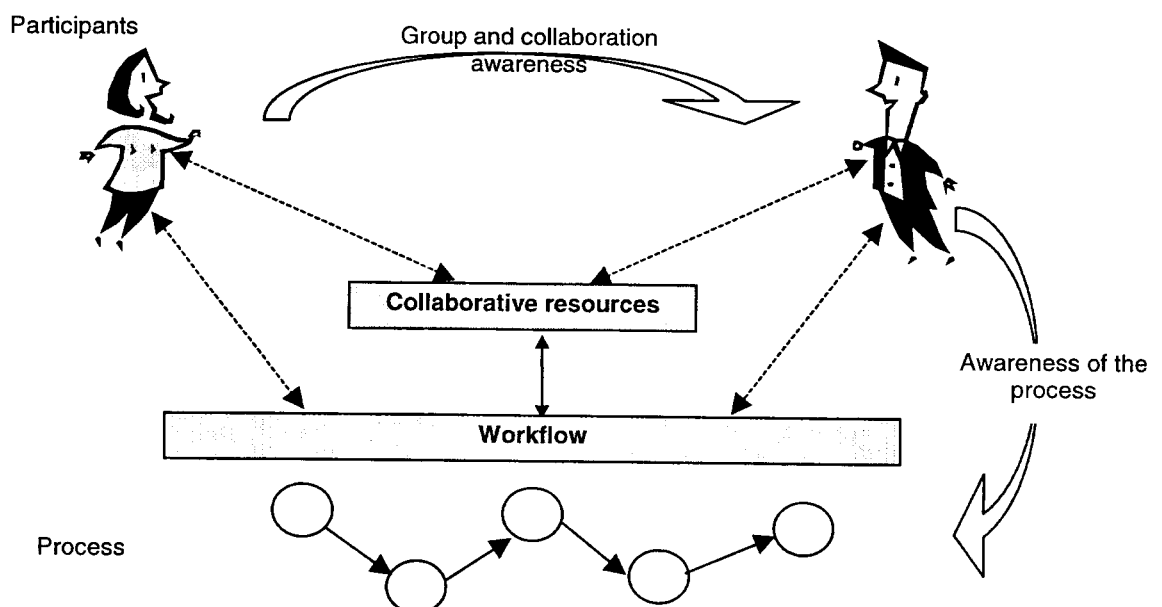


Figure 2 - Overview of the proposed environment

### 3.1 Awareness of the Work Processes

Most organizations define their software processes informally, using some kind of documentation that is rarely referenced by developers. In mature organizations, where the importance of a process definition has already been recognized, this definition is expressed by means of formal documentation but still, this documentation is often neglected by participants and sometimes far from day-to-day practices [CAIN96].

To achieve a better consciousness about the idea of software processes, the first requirement is to let the “process” be a common object, visible by everybody, naturally referenced by all involved members. To induce teams to accept the idea of process tractability and improvement it is necessary to explicitly introduce the concept of process as a work element and to create resources that will aid teams to perceive the process in which they are involved. An ideal culture would be the one where its members are able to know about the existence of a process that direct their activities and:

- *Understand how the process was defined:* this means to have access to process definition and understand its representation, recognizing the elements that comprise its definition.
- *Execute their activities based on this definition:* to execute the tasks as defined by the process, being aware of their own responsibilities.
- *Situate themselves and the group during the process execution:* recognize the status of process execution, which activities have been finished, being executed and to be executed by himself and by the overall group.

### 3.2 The Role of Workflow

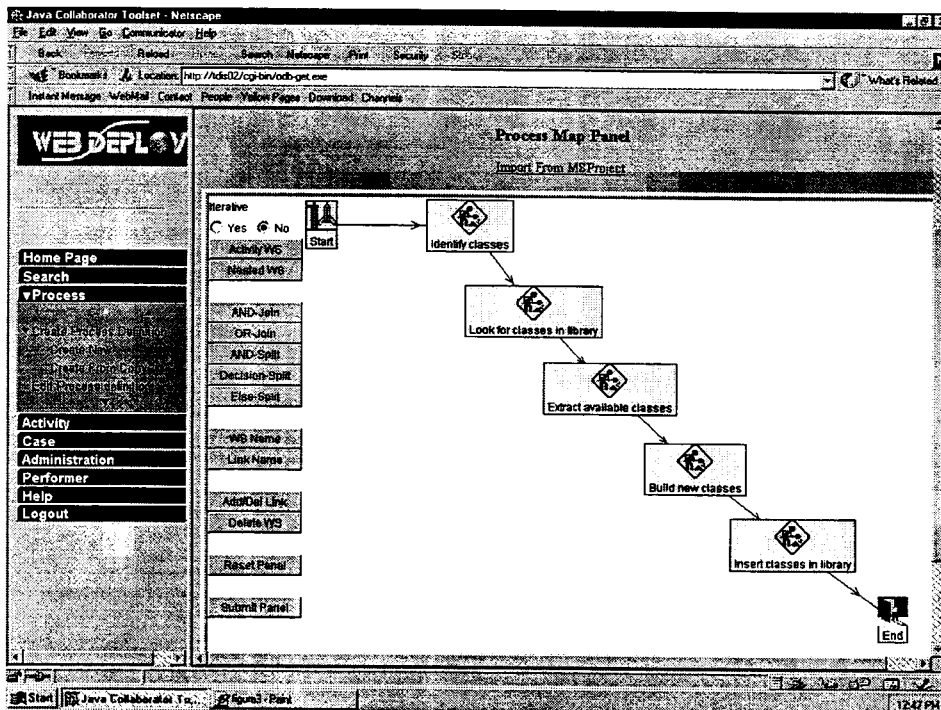


Figure 3 - Awareness of process definition

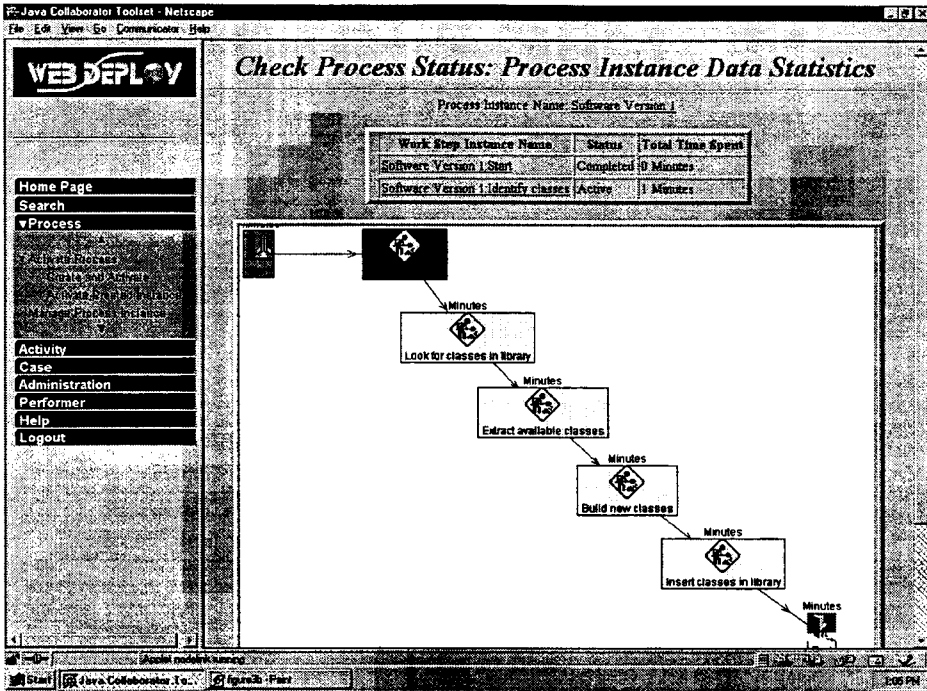


Figure 4 - Individual worklist

Recently, workflow systems have obtained success in business market due to their basic feature of providing greater visibility and control of the work process. This functionality is a starting point to support the dialogue between workers and their processes and is an interesting solution to provide the awareness of software processes.

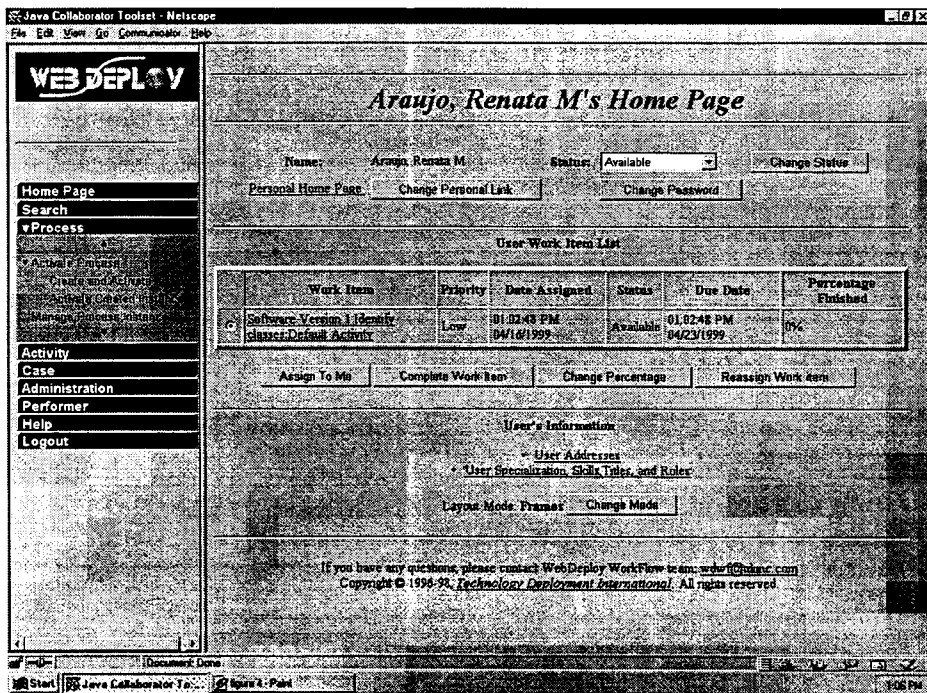


Figure 5 - Status of process enactment

The focus on process generality, in the flexibility and adaptability to changes, in the interoperability and heterogeneity offered by these systems make them promising technologies for establishing process culture [ARAU99, OCAM98]. Also, due to its cooperative nature, workflow systems integrate their executors in the process enactment, helping them to perceive their individual responsibilities in the process.

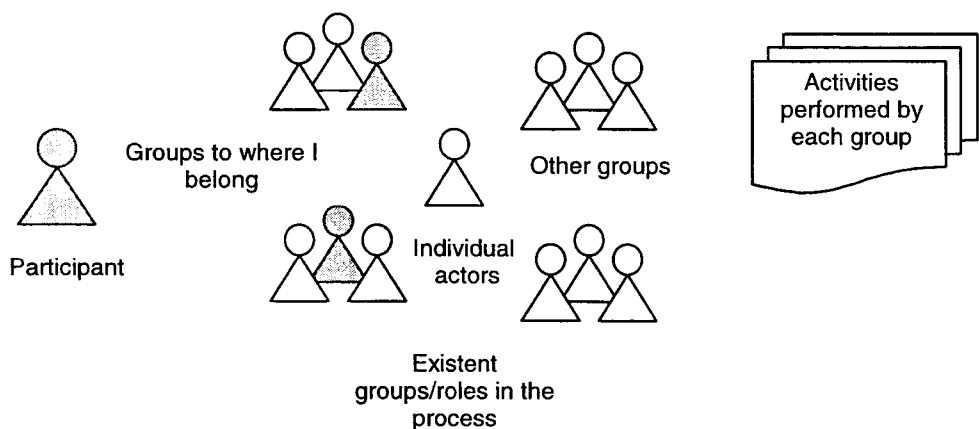
In the proposed environment, a workflow system [KHOS98] will be used as a software development environment, where processes will be defined and enacted by the development group. The workflow system will provide information about the process definition (Figure 3), the status of enactment (Figure 4) and the responsibilities of each actor through its worklist (Figure 5).

### 3.3 Awareness of Collaboration

During software processes, participants may interact directly with other participants or through the products they generate. Generally, interactions are costly, executed using deficient means and take time to produce a consistent final artifact. The lack of awareness of the possibilities for interaction with others may generate delays, inconsistencies and decreases the chances of building artifacts with better quality. Additionally, collaboration is not stimulated because process actors don't have an explicit perception of how they contribute with each other. Role definitions inside the group are sometimes confusing and obscure for common actors. Also, there is a lack of knowledge about the flow of interaction between these roles.

In order to achieve a better perception of the way processes are conducted, it is necessary to provide actors with mechanisms which will help them in constructing the notion about how the team is composed and the awareness of possible communication channels between them. Also, to enforce collaboration and to allow further improvements in the interaction process, participants must realize how interactions occur within the group.

### 3.4 Awareness of Collaboration – Groups, Roles and Interaction



**Figure 6 - Group awareness information**

To best understand the process definition, participants should also be able to recognize the different groups assigned to contribute to the process, the roles defined in each group and what are the activities they perform (Figure 6). The environment will help each actor to



know, for example, what are the existent groups and roles assigned to the process; to which groups the participants belong; and which activities are assigned to each defined group. This information will help actors to have a notion about proximity, that is: who is collaborating with him in a same activity, who is performing a near task; who are far from their tasks, with whom he will interact in the future and with whom he interacted in the past.

To provide participants with perceptions about team collaboration, the environment proposed in this paper adopts a solution based on Cain and Coplien's approach for process modeling [CAIN96]. This approach focuses on modeling processes using roles and their relationship during process execution. The motivation for this approach is the fact that roles are more stable elements in an organization than activities. The approach aimed to analyze the interactions that exist between actor's roles and will be the observation of more stable and recurrent patterns of process execution inside organization.

To model roles, the approach uses a technique named CRC (Classes, Responsibilities and Collaborators) cards. In this technique, cards are assigned to each role (actor), its responsibilities are enumerated and their relationships (collaboration) with other cards identified. Each actor is given a card where he can write down what he thinks is his responsibility in the process and which other roles he collaborates with. A graph can be generated using the information provided by actors where each node of this graph is a card (an actor) and the arcs are the collected interactions among them. The graphs can be analyzed and compared helping to determine if a process is chaotic or if has already reached any level of organization and efficiency.

The experiences made by the authors showed that, when reviewing the generated graphs, the great majority of process participants felt surprised to verify that the model that they imagined was very different from the resulting model being presented. Also, most recognize the technique made them modify the way they view their work.

The proposed environment aims to use this approach for providing process actors an awareness resource for recognizing their interaction. Usually, workflow systems define processes based on the sequence of activities and on the performers of each one of these activities. They are not concerned about which interactions may occur during the execution of each activity. A process work step is defined in terms of the instructions of the activity, the documents that will be used or produced and what are the performers responsible for it. But, if this activity will be executed collaboratively, there is no indication about how this collaboration will take place. It is outside the process definition and, consequently, outside the awareness of current common actors. Also, as showed by Cain e Coplien's studies, the notion about the overall interaction between actors (roles), offers value information about how collaboration is viewed in the organization. This is important for process improvement purposes, since the analyses of such interaction may help to identify problems and possibilities for change.

The environment will extract from the process definition stored in the workflow system the existent groups and roles defined for the process and present them to the actors as CRC cards. Each actor may introduce their perceptions of interactions with other actors throughout the process. This information will be shared among all actors in order to help them perceive the overall group interaction throughout the process.

## 4 Conclusion

The aim of this paper was to emphasize the role of awareness support in collaborative improvement of software processes. Awareness support increases the acceptability of software process improvement initiatives in the organization. This is achieved by providing actors with means to recognize the process they work on, to promote collaboration between them and to let them share their own perceptions of the work being performed.

One motivation for this idea is the results collected by Cain and Coplien [CAIN96]. They show that, in productive organizations, even if there isn't a formal process definition, people are aware of what they do and about what works in the process. They mention that there probably exists a work paradigm change when organizations go from level 2 to level 3 of CMM. It is believed that organizations in levels 1 and 2 need a strong and present management, while organizations in higher levels are self-directed.

We claim that it is possible to construct an environment that aims at offering actors awareness resources about their processes and interactions. The continuous use of this environment would help its actors to gradually establish a proactive process improvement culture in the organization, decreasing the time for process redefinition and facilitating the introduction of new software engineering practices.

In this culture, participants will be able to identify bottlenecks in the execution of activities and in the process as a whole. Also, participants can analyze the interaction patterns within the group and identify problems in the number, type and way of executing these interactions. This may help them to detect possible obstacles in the process due to bad interactions. Participants are stimulated to outline the problems they find in the process, to comment the group work and to share their opinions with the overall group.

It is important to notice that the proposed environment is not restricted to a specific SPI model or methodology. Its main objective is to be an assistant in SPI, allowing actors to be more involved with the improvement initiative, increasing their learning space about the process and increasing the overall organization capacity to improve future processes.

The proposed environment seems to be useful for organizations at any of the maturity levels. It can be both used as a tool to start process improvement in immature organizations, as well as a tool to maintain continuous process improvement in mature organizations. The process description, its continuous redefinition and the introduction of new software engineering practices will be facilitated, increasing the possibilities for process improvement.

Some undesirable reactions may appear with the introduction of an environment such as the one being proposed in this work. The impact of making people and working practices visible and explicit may cause some discomfort since not all individuals like the idea of being criticized or even to receive comments about the way they do their work. So, it is important to emphasize that the environment will be useful only if introduced as a helper tool in ongoing SPI initiatives, where efforts for explaining, convincing and training people about the benefits of improvement are already occurring in parallel.

## 5 Acknowledgments

The authors would like to thank the Brazilian National Research Council (CNPq) for financial support, Technology Deployment International Inc. and Computer Associates for the demo versions of software WebDeploy™ and Jasmine™ database.

## 6 References

- [ARAU97] ARAUJO, R.M., DIAS, M.S., BORGES, M.R.S., "A Framework for the Classification of Computer Supported Collaborative Design Approaches". In: *Proceedings of the Third CYTED-RITOS International Workshop on Groupware (CRIWG'97)*, pp. 91-100, San Lorenzo de El Escorial, Madrid, Spain, Oct. 1997.
- [ARAU99] ARAUJO, R.M., BORGES, M.R.S., "Sobre a Aplicabilidade de Workflow no Suporte a Processos de Software". In: *Proceedings of IDEAS'99*, pp. 417-428, San Jose, Costa Rica, Mar. 1999 (In Portuguese).
- [BORG99] BORGES, M., PINO, J. A., "PAWS: Towards a Participatory Approach to Business Process Reengineering". To be presented at the *Fifth International Workshop on Groupware (CRIWG'99)*, Cancun, Mexico, September, 1999.
- [CAIN96] CAIN, B.G., COPLIEN, J.O., "Social patterns in productive software development organizations", *Annals of Software Engineering*, vol.2, pp. 259-286, 1996.
- [ELEM98] EL EMAM, K., DROUIN, J., MELO, W., *SPICE - The Theory and Practice of Software Process Improvement and Capability Determination*, 1ed. California, IEEE Computer Society, 1998.
- [GIBS97] GIBSON, R. "Software Process Modeling and Assessment". In: *Proceedings of the 9th International Conference on Software Engineering & Knowledge Engineering (SEKE'97)*, pp. 105-110, Madrid, Spain, Jun. 1997.
- [GOLD95] GOLDENSON, D. R., HERBSLEB, J.D., *After the Appraisal: A Systematic Survey of Process Improvement, its Benefits, and Factors that Influence Success*. CMU/SEI-95-TR-009 ESC-TR-95-009, Software Engineering Institute, Carnegie Mellon University, Pennsylvania, 1995.
- [KHOS98] KHOSHAFIAN, S., "WebDeploy: Workflow White Paper", TDI Inc., <http://www.tdiinc.com/>
- [KITS97] KITSON, D.H. "An Emerging International Standard for Software Process Assessment". In: *Proceedings of the International Symposium on Software Engineering Standards (ISSES'97)*. California, USA, Jun. 1997.
- [KOCK97] KOCK, N., "Fostering Interdepartmental Knowledge Communication through Groupware: A process Improvement Perspective". In: *Proceedings of GROUP'97*, pp. 29-37, Phoenix, Arizona, 1997.
- [OCAM98] OCAMPO, C., BOTELLA, P., *Some reflections on applying workflow technology to software processes*, TR-LSI-98-5-R, Universitat Politecnica de Catalunya, Barcelona, Feb. 1998, <http://www-lsi.upc.es/~ocampo/home.html>
- [PAUL93] PAULK, M.C. et al, *Capability Maturity Model for Software, version 1.1*. CMU/SEI-93-TR-24, Carnegie Mellon University, Software Engineering Institute, Pittsburgh, Pennsylvania, Feb. 1993.

- [SCP98] SCP, *Supporting Collaborative Processes*,  
[http://www.sei.cmu.edu/scp/scp\\_welcome.html](http://www.sei.cmu.edu/scp/scp_welcome.html)
- [SPIC98] *SPICE Phase 2 Trials Interim Report v 1.00*. <http://www.iese.fhg.de/SPICE>, Jun. 1998.