The Journal of Systems and Software 83 (2010) 1662–1677

Contents lists available at ScienceDirect

The Journal of Systems and Software

journal homepage: www.elsevier.com/locate/jss

Using Scrum to guide the execution of software process improvement in small organizations

Francisco J. Pino[a,*], Oscar Pedreira[b], Félix García[c], Miguel Rodríguez Luaces[b], Mario Piattini[c]

[a] IDIS Research Group, Electronic and Telecommunications Engineering Faculty, University of Cauca, Calle 5 # 4–70 Popayán, Colombia
[b] Databases Laboratory, Computer Science Department, Facultade de Informática, University of A Coruña, Campus de Elviña, 15071 A Coruña, Spain
[c] ALARCOS Research Group, Information Systems and Technologies Department, UCLM–INDRA Research and Development Institute, University of Castilla–La Mancha, Paseo de la Universidad 4, 13071 Ciudad Real, Spain

ARTICLE INFO

Article history:
Received 25 May 2009
Received in revised form 31 March 2010
Accepted 31 March 2010
Available online 15 June 2010

Keywords:
Software process improvement
Small software companies
Model to guide improvements
Agile methods
Scrum
SPI

ABSTRACT

For software process improvement – SPI – there are few small organizations using models that guide the management and deployment of their improvement initiatives. This is largely because a lot of these models do not consider the special characteristics of small businesses, nor the appropriate strategies for deploying an SPI initiative in this type of organization. It should also be noted that the models which direct improvement implementation for small settings do not present an explicit process with which to organize and guide the internal work of the employees involved in the implementation of the improvement opportunities. In this paper we propose a lightweight process, which takes into account appropriate strategies for this type of organization. Our proposal, known as a "Lightweight process to incorporate improvements", uses the philosophy of the Scrum agile method, aiming to give detailed guidelines for supporting the management and performance of the incorporation of improvement opportunities within processes and their putting into practice in small companies. We have applied the proposed process in two small companies by means of the case study research method, and from the initial results, we have observed that it is indeed suitable for small businesses.

1. Introduction

Software process improvement – SPI – is a planned, managed and controlled effort which aims to enhance the capability of the software development processes of an organization (Krasner, 2001). SPI aims to understand the software process within an organization and, using this knowledge base, it sets out to drive the implementation of change so that specific goals can be achieved (Coleman and O'Connor, 2008) (i.e. to meet its particular purpose). According to the systematic review on SPI in small companies presented in Pino et al. (2008), for some years now the Software Engineering community has been showing an ever-increasing interest in tackling SPI in small companies. They see this to be a strategy which can be used to increase software product quality, as well as to improve the productivity of software development.

Even though there are different SPI standards in existence at present, there is evidence that the majority of small software organizations are not adopting these standards (Coleman and O'Connor, 2008; Laporte et al., 2008). The reasons for this are diverse, but according to Laporte et al. (2008) one important factor is the nature of the standards, which do not provide adequate guidance for use in small software organizations. We might also observe that these standards, along with many proposals explicit to small companies, focus on offering models related to the best practices of processes (process reference models) and the manner of assessing process (process assessment models). They do not provide the explicit guidelines needed for the small companies to drive their SPI activities (improvement models suitable for small companies). An interesting finding of the systematic review presented in Pino et al. (2008) is the percentage of the use of these models in the primary studies analyzed: 71% used some process reference model, 42% used some assessment model and 24% used some improvement model. That is, small organizations have used models that direct the implementation of SPI activities in their initiatives for process improvement to a lesser extent. Regarding this issue, we consider it important to look into improvement models that may be suitable for small organizations because such a model should: (i) help to understand what to do when a small organization is interested in carrying out SPI initiatives within their organization; and (ii) provide the guidelines that are needed to organize all the activities related to process improvement as well as to connect the other models and elements involved.

* Corresponding author.
E-mail addresses: fjpino@unicauca.edu.co (F.J. Pino), opedreira@udc.es (O. Pedreira), Felix.Garcia@uclm.es (F. García), luaces@udc.es (M.R. Luaces), Mario.Piattini@uclm.es (M. Piattini).

0164-1212/$ – see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jss.2010.03.077

With all that in mind, we have developed an *Improvement framework* in the context of the COMPETISOFT project (Oktaba et al., 2007). COMPETISOFT offers a strategy for supporting software process improvement in small organizations, and its *Improvement framework* provides improvement practices, strategies and tools to guide the execution of improvement initiatives in small companies. Initially, we defined a process to manage and lead the SPI initiatives in small organizations step-by-step, called PmCOMPETISOFT (Pino et al., 2009a), as the backbone of this *Improvement framework*. This process describes five activities: (a) *initiating the cycle*, (b) *diagnosing the process*, (c) *formulating improvements*, (d) *executing improvements* and (e) *revising the cycle*. PmCOMPETISOFT offers a set of well-defined process entities (such as activity diagram, activities, roles, work products and templates) to the whole process improvement lifecycle. From an early application of this process in a pilot case study we observed that it was a good proposal for guiding SPI initiatives in small organizations, but that it was not yet sufficient. This first application (and other previous studies that we have performed (Hurtado et al., 2008)) led us to conclude that:

- For small companies it is not easy to incorporate into their processes the improvement opportunities that are uncovered in the *diagnosing of process* activity.
- Small organizations have difficulties in managing and carrying out improvement activities in which the greatest amount of effort falls mainly upon the organization (in PmCOMPETISOFT these activities are *formulating improvements* and *executing improvements*).
- It is necessary to develop a proposal which offers more detailed guidelines to address specifically those activities which have to do with the formulation and execution of improvement opportunities. Furthermore, what is proposed should be suitable for small organizations.

We have developed a proposal to tackle the issue described above. What we have proposed takes into account agile methods, for as according to Coleman and O'Connor (2008), Hurtado and Bastarrica (2006) and Abrahamsson et al. (2002) the strategies offered by these methods are appropriate for small organizations. Our proposal thus describes a process called PfemCOMPETISOFT, which uses the Scrum agile method, seeking to provide detailed guidelines for supporting the management and performance of the improvement iteration (which is made up of the activities of *formulating* and *executing improvement* from PmCOMPETISOFT). The objective of PfemCOMPETISOFT is to support the incorporating of improvement opportunities (found in the *diagnosing of process* activity) within the processes of the small organizations. We have used the Scrum agile method for the definition of this process, because it provides support for project management, emphasizes management values and practices and focuses on small teams (Abrahamsson et al., 2002). A discussion on why we have used agile methods, and especially Scrum, in our proposal, is presented in the next section.

The purpose of this paper is to present the PfemCOMPETISOFT process. It also describes our experience of the application of the proposed process in two small organizations, through case studies. These organizations were involved in an improvement project, which used the *Improvement framework* of COMPETISOFT. They used PfemCOMPETISOFT to guide the incorporation into their processes of the improvement opportunities that were uncovered in the diagnosis activity. The paper is organized into six sections. Following this introduction, Section 2 of the paper presents the background on characteristics of small companies and SPI success factors in these organizations; why we used agile methods is set out, and the related works are also described. Section 3 describes the *Improvement framework* of COMPETISOFT and gives an overview of the PmCOMPETISOFT process. The PfemCOMPETISOFT process and how Scrum supports the activities of formulating and executing improvement is shown in Section 4. Section 5 presents two case studies where this process was applied and finally, Section 6 presents our conclusions and future lines of work.

2. Background

To develop process improvement proposals suitable for small software organizations, the special characteristics of this type of organization and the success factors of process improvement in small organizations should be taken into account.

In (Laporte et al., 2008; Horvat et al., 2000; Richardson and Wangenheim, 2007; Hofer, 2002) the special characteristics of small companies are analyzed and discussed. From these studies we have extracted some characteristics of the small organizations, which include the following:

- Their software development is driven by light processes which are strongly human-oriented and there is constant communication between the project members and the customer.
- They are usually dynamic and flexible, with a flat organizational structure (not traditional), in a free-flow management style that boosts enterprising and innovative spirits.
- Typically, they carry out their management process through informal mechanisms, based on face-to-face relationships (communication, decision-making, problem resolution, etc.).
- They do not have enough staff to develop specialized functions, and they have little or no room in their budget for buying the required expertise.
- These companies are economically vulnerable and they have limited economic resources.

The characteristics of these organizations, such as specific procedures of work and specific relationship between employees, require appropriate management of SPI projects (Horvat et al., 2000). Another feature is that the success of process improvement in small organizations is also directly related to the manner in which the improvement project is managed. Some of these success factors, which can be found in Pino et al. (2008), are: guiding the implementation (or deployment) of SPI by means of specific processes and the combination of different approaches, the monitoring and supervision of the SPI project, the attainment of a rapid return on investment, the definition of what SPI objectives are feasible with the resources available, the participation of employees in SPI (button-up), suitable communication between SPI participants, and that SPI should be based upon learning, not upon control.

2.1. Small organizations and agile methods

According to Coleman and O'Connor (2008), small organizations argue that each project and situation is new to them and that creativity and flexibility are important capabilities; they also affirm that these capabilities can be supported by agile methods. Furthermore, according to Coleman and O'Connor (2008), agile methods, with their advocacy of self-empowered teams and shared ownership, are more associated with the style of management of small organizations. This is because this type of company considers that agile practices can be applied in their software process basically due to the small initial investment required and because it allows them to take advantage of competitiveness in their personnel (Hurtado and Bastarrica, 2006). In our opinion, the agile methods also can support some success factors involved in process improvement

in small organizations, as well as several special characteristics of these companies.

As Abrahamsson et al. (2002) point out, some agile methods that focus on small teams are: Agile modeling (Ambler, 2002), Extreme programming (Beck, 1999), Pragmatic programming (Hunt and Thomas, 2000) and Scrum (Schwaber, 1995). However, of the four methods described previously, only the Scrum agile method provides a framework for managing projects, whilst the others center on describing practices, activities and work product techniques to do with software development. As stated in Schwaber (2009), Scrum is not a process or a technique for building products; rather, it is a framework within which you can employ various processes and techniques. This feature allows combining the framework for managing projects described by Scrum with technical processes of a specific knowledge area, in our context, for instance, the SPI area.

Bearing in mind (i) the aspects previously described as regards agile methods and Scrum, and (ii) the particular needs set out in the introduction about developing a proposal to support the incorporation of improvement opportunities into the processes of small organizations, we believe that Scrum is suitable for this kind of organization and that it should be borne in mind in the development of our proposal. This conviction comes from the fact that Scrum focuses on small teams, it gives most emphasis to management values and practices, it can be combined with other processes (e.g. PmCOMPETISOFT), and that it is the method which can support success factors for process improvement such as those we have set out before.

2.2. Related work

With regard to research on models that direct improvement implementation for small organizations, various proposals have emerged in recent years. These include, amongst others, MESOPyME (Calvo-Manzano et al., 2002), IMPACT (Scott et al., 2001), PROCESSUS (Horvat et al., 2000), ASPE-MSC (Wangenheim et al., 2006) and the application of the IDEAL model to small and medium enterprises. The PROCESSUS framework is based on the process modeling paradigm, in which each procedure is dealt with as a process, which is defined, established, implemented and maintained. The IMPACT framework is based on the idea that the process is an abstraction of the practices carried out in many different projects by many different people. It therefore permits continual learning and the improvement of the process, precisely through this experience that has been gained by a lot of people in a lot of projects. MESOPyME has as its focal point the reduction of time and effort in the implementation of SPI by using the concept of action packages as a base. ASPE-MSC integrates and adapts existing approaches on establishment of software process to the characteristics of small companies. The goal is to support a cost-efficient and effective establishment process in these companies as part of a SPI initiative. However, none of these proposals presents an explicit process with which to guide the internal work of the employees involved, to enable them to manage and carry out the deployment of the improvement activities in the small organization.

When considering the application and adjustment of the IDEAL model to small and medium enterprises, it is worth noting the studies presented in Casey and Richardson (2004) and Kautz et al. (2000). The first study was carried out in a medium-sized enterprise, and it identified that the IDEAL model's acting phase should be adapted to the organization's characteristics. In the acting phase the improvements discovered are created, piloted, and deployed throughout the organization. The study also highlighted that if the business had carried out this phase as the IDEAL model had indicated that it should, then it would have had a negative effect upon the organization's process improvement initiative. The reason for this is that it did not take into account either the time or the resources needed to carry out the tasks. On the other hand, the second study took place in a small enterprise, and discovered that to improve processes in small organizations a structured model needs to be used to organize the process, to adjust the model to the particular conditions of the organization, and to perform the improvement activities as a project with clearly assigned and documented roles, responsibilities and resources. What is more, both studies demonstrate that, if the improvement initiatives suggested by the IDEAL model are to be applied, there is a need for at least one full-time specialist in process improvement. That does not seem to be suitable for the characteristics of small companies, as many of these organizations do not have enough staff to develop specialized functions and their budget forbids buying this required expertise. Moreover, neither of these studies explicitly presented a process in detail which was adapted to the IDEAL model and which could be applied to others small organizations.

With regard to works related to models that direct improvement implementation and agile methods, we should mention those presented in Salo and Abrahamsson (2007) and Kähkönen (2005). This first work proposes an iterative improvement process for carrying out SPI within agile software development teams, and the second one presents a lifecycle model for an SPI project which deploys an agile method in a single project. That is to say, these works focus upon process improvement in projects or teams which work with (or implant) agile methods. Our proposed process, however, makes use of an agile method such as Scrum to support the management and carrying out of the improvement process, following any type of process reference paradigm.

The aspects set out above justify the definition of a *Lightweight process for improvement incorporation*, which would add more details to the activities of *formulating improvement* and *executing improvements* of the PmCOMPETISOFT process (see Fig. 1). We have used the Scrum agile method to manage and carry out the activities of formulation and execution of improvement, in which many of the organization's employees are directly involved. As discussed in Section 2.1, our reason for doing so is that this method provides support for project management and it is suitable for small teams. Furthermore, apart from those advantages, Scrum emphasizes management values and practices, without including practices on technical issues. This allows Scrum to be integrated with other methods and processes, thus offering them more agile management or deployment. Taking into account this aspect, the incremental and iterative process proposed by Scrum has been integrated and tailored to the activities of *formulating improvement* and *executing improvements* of the PmCOMPETISOFT. This means we can offer a strategy which is useful and practical and through which all the employees will become involved and take part in the process improvement to which they are related within the organization.

3. Improvement framework of COMPETISOFT

The *Improvement framework*, along with the *Process reference model* and the *Evaluation model*, are the three components that make up the Methodological Framework of COMPETISOFT; see Fig. 1.

This Methodological Framework is the strategy developed by the COMPETISOFT project (Oktaba et al., 2007) for supporting SPI in small organizations, since the vast majority of the companies that make up the Latin American software industry are small-sized, with fewer than 50 employees (MBI, 2004). In this respect, the framework that has been developed took into account the characteristics peculiar to this type company, thereby offering the Latin American software industry a process improvement approach which has been adapted to its particular context. The COMPETISOFT project seeks to increase the level of competitiveness of small software organi-

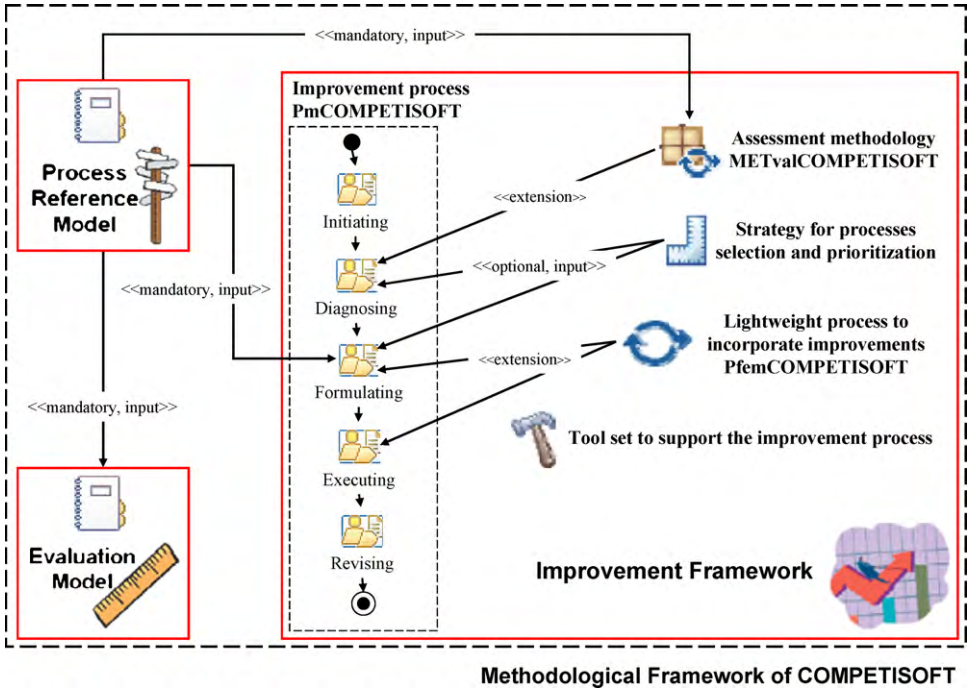

Fig. 1. Overview of the *improvement framework* and methodological framework of COMPETISOFT.

zations through the improvement and certification of the software processes of these organizations, using the proposed components and by means of the Methodological Framework. Regarding components of the Methodological Framework, a brief description of these is presented below:

- The *Process reference model* provides a suitable and useful process group for increasing the process capability in small organizations. In this respect, each of these processes offers best software practices, which should be considered by this kind of organization if they are to improve their processes.
- The *Evaluation model* defines the necessary elements in the execution of a formal assessment, the objective being to give the small organizations a rating of their process capability and organizational maturity. In this sense, we suggested that the defined *Evaluation model* must use and be in conformance with ISO/IEC 15504 Part 2 (ISO, 2004a) and Part 7 (ISO, 2008).
- The *Improvement framework* provides improvement practices, strategies and tools to support SPI initiatives in small companies (Pino et al., 2009b). This framework defines: (i) an improvement process called PmCOMPETISOFT (Pino et al., 2009a), which is the backbone as well as the component integrator of the *Improvement framework*; (ii) a methodology for software process assessment called METvalCOMPETISOFT (Pino et al., 2010), which extensively describes the activity of diagnosing the software processes in small organizations and which is conformance with ISO/IEC 15504-2; (iii) a *Strategy for process selection and prioritization*, which supports the selection of critical processes during the implementation of a process improvement project in small companies (Pino et al., 2009c); (iv) a *Lightweight process for improvement incorporation* called PfemCOMPETISOFT, which provides a extension of the activities of formulating and executing improvements of PmCOMPETISOFT; and (v) a *Tool set to support the improvement process*, which provides software tools to support the dissemination, the knowledge, the management and the implementation of the components of the *Improvement framework*.

The strategy for process selection and prioritization, METvalCOMPETISOFT and Tool set to support the improvement process of the Improvement framework are beyond the scope of this article, which focuses on the description of the Lightweight process for improvement incorporation. To help to make the proposed PfemCOMPETISOFT process easier to understand, we will now go on to give an overview of PmCOMPETISOFT (a complete description is presented in Pino et al., 2009a).

3.1. Improvement process – PmCOMPETISOFT

PmCOMPETISOFT is a light process influenced by the IDEAL, ISO/IEC 15504-4 (ISO, 2004b) and Scrum models, which guides the implantation of an improvement initiative in an iterative and incremental manner, and which plans to satisfy the following principles:

- *Early and ongoing achievement of improvement*: the generation of visible short-term results implies that the actors involved in this improvement process see the fruits of their work at an early stage and thus continue to be motivated. This also permits the minimization and control of risks in the improvement project.
- *Ongoing and rapid process diagnosis*: the ongoing diagnosis of the organization's processes and of the improvement process, in an effort to verify whether the improvement project is fulfilling its objective of raising the level of capability and efficiency of the organization's processes.
- *Elemental process measurement*: the carrying out of measurement activities through the systematic use of process base measures which are suitable for this type of organizations.
- *Effective group collaboration and communication*: the establishment of communication and collaboration strategies between the various actors involved in the improvement process project, highlighting the importance of each person's role within this project.
- *Ongoing learning*: offering training in software process improvement to those involved in the improvement project, making them aware that their support, contributions and active participation

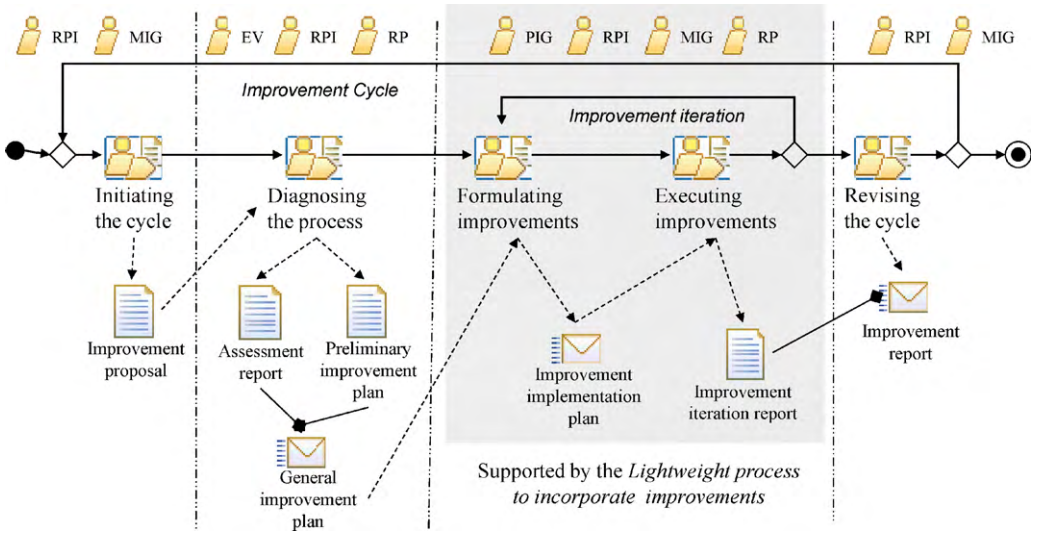

Fig. 2. PmCOMPETISOFT activity diagram.

are necessary for the project to be a success. These actors are also encouraged to reflect upon the work carried out; the lessons learnt are thereby reinforced and the improvement project can hence be improved and adjusted.

PmCOMPETISOFT's purpose is to improve the organization's processes as appropriate to its particular business objectives. It is also to assist it in carrying out its software improvement process by focusing on small enterprises, defining a guide with which to implement a step-by-step process improvement. Fig. 2 shows an overview of PmCOMPETISOFT by means of an activities diagram. From this diagram we can see that:

- There are five roles: Management improvement group (MIG), Responsible for process improvement (RPI), Process improvement group (PIG), Responsible for process or Participant (RP) and Evaluator (EV).
- Each of the improvement cycles is formed by one or more improvement iterations. An improvement cycle consists of five activities: *Initiating the cycle*, *Diagnosing the process*, *Formulating improvements*, *Executing of improvements* and *Revising the cycle*: the activities of *Formulating improvements* and *Executing of improvements* make up the improvement iteration.
- The work products are: Improvement proposal, General improvement plan, Improvement implementation plan and Improvement report.

4. Using Scrum to support the activities of formulating and executing of improvements

As a method, Scrum (Schwaber, 1995) emphasizes values and management practices and does not include technical practices, and this is just what gives it its capacity to complete other methods and processes. This agile method offers an incremental and evolutive process for the development of products. We could add that, as the PmCOMPETISOFT activities of *formulating and executing improvements* are iterative and incremental, initiated by prioritized improvement opportunities which are described in the *general improvement plan* (obtained from the diagnosing activity; see Fig. 2), the strategies proposed by Scrum are in our view suitable for the management and deployment of both these activities in small businesses. It is important to clarify that after carrying out these activities, the desired product is a process with a higher capability level and this is put into practice within the organization. That being the case, we used and adapted the incremental and evolutive process of Scrum to develop the *Lightweight process to incorporate improvements*, which gives a more detailed guideline for carrying out the activities of *formulating improvement* and *executing improvements* of PmCOMPETISOFT.

4.1. The lightweight process to incorporate improvements

This section provides a detailed description of this proposed lightweight process following the process pattern established by COMPETISOFT, which includes purpose, objective, activity diagram, roles, activities, work products and tools support. Within the description of activities, roles and work products we show the Scrum elements (described in Schwaber (2009)) that are related to each of these. The goal is clarify how we took this method into account in the development of PfemCOMPETISOFT.

4.1.1. Purpose

To offer a guideline for managing and carrying out the activities of *formulating improvements* and *executing improvements* (i.e. an improvement iteration). This guideline would allow the employees involved in the SPI initiative of the small organizations (mainly to the *responsible for process improvement*) to manage, deploy and take part in the execution of the incorporation of the improvement opportunities found and with which they have some relationship.

4.1.2. Objective

To support the incorporating of improvement opportunities in the processes of small organizations, by means of the establishment of a break-down of the activities of *formulating improvements* and *executing improvements* (improvement iteration), following the philosophies proposed by Scrum.

4.1.3. Activity diagram

Fig. 3 shows the activity diagram of the proposed lightweight process.

4.1.4. Activities

One or more improvement iterations must be carried out to include the improvement opportunities within the processes to be improved (improved processes). To manage the complexity of the improvement opportunities found in the SPI initiative, these are

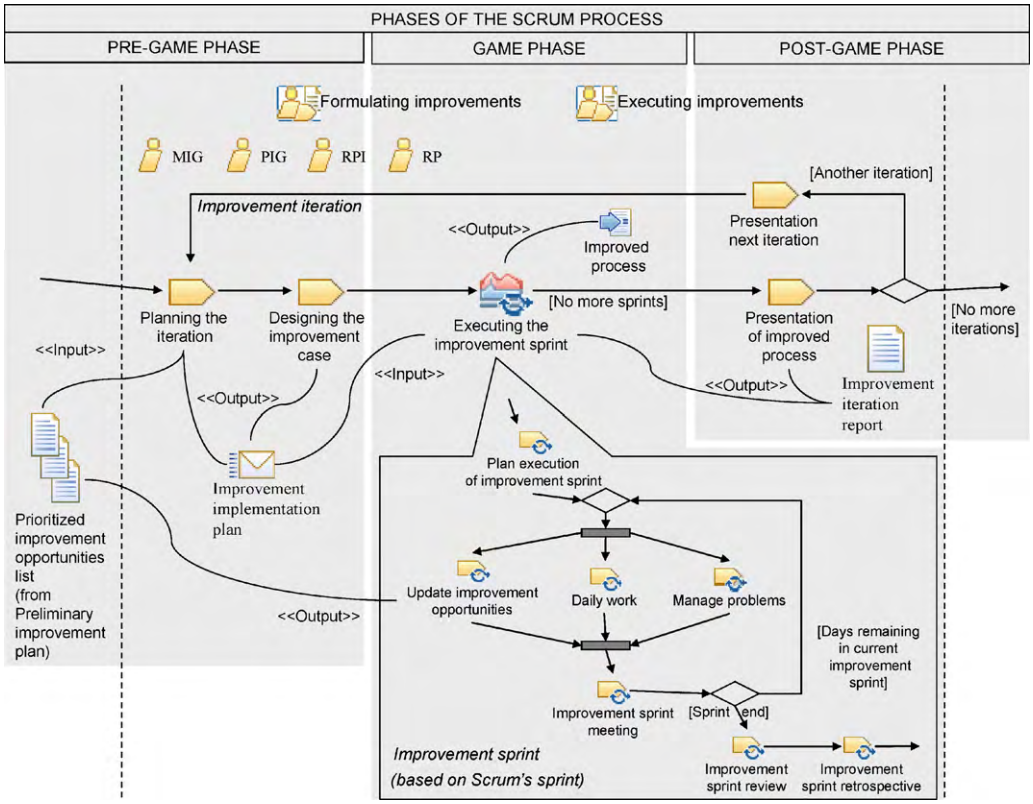

Fig. 3. Diagram activities of the lightweight process to incorporate improvements.

grouped in *improvement cases* according to specific improvement targets of the organization. An iteration allows the management and development of an *improvement case* to advance independently. In this respect, each *improvement iteration* creates an increment in the process capability by means of the incorporation of specific improvement opportunities. In order to reach this increment, iteration entails the definition and the putting into practice of the improved process. Each *improvement iteration* consists of five activities: *Planning the iteration*, *Designing the improvement case*, *Executing the improvement Sprint*, *Presentation of improved process* and *Presentation of next iteration*. These activities are presented below:

- *Planning the iteration*: the objectives of the current *improvement iteration* are described. The improvement opportunities set associated with each improvement case and the other items established in the *preliminary improvement plan* (which is part of the *general improvement plan*, see Fig. 2 and work products sub-section) are reviewed, validated and approved by the *management improvement group*. The strategy presented in Pino et al. (2009c) can be used to prioritize the improvement opportunities since this is a key task in this activity (this activity is related to Release planning meeting from Scrum).
- *Designing the improvement case*: the strategies with which to satisfy the improvement opportunities are identified. The activities associated with the strategies for carrying out the current process improvement are defined and an estimate (in hours or days) of these tasks is made. These activities are defined and assigned to the members of the work team in the organization and an agreement is reached with the work team regarding the development of the improvement iteration. The team responsible for this activity is the *process improvement group* (this activity is related to Sprint planning meeting and Sprint backlog).
- *Executing the improvement sprint*: bearing in mind the concept of Scrum's *sprint*, we consider an *improvement sprint* to be a set of improvement activities conducted over a pre-defined period, usually 1–4 weeks, which is used to implement and monitor the incorporation of the improvement opportunities within processes of the small organization. As can be seen from Fig. 3, we have tailored the tasks of the Scrum's sprint to the needs of the *improvement sprint*, thereby providing the people involved in this sprint with a work structure that would place the desired improvement opportunities within the processes. Through the implementation of several improvement sprints, the process is completely defined and put into practice in the organization. The team responsible for these tasks is made up of the person *responsible for process improvement* and the person *responsible for process or participant*. The tasks defined in the improvement sprint are compliant and consistent with those described by the Scrum's sprint, and they are:
  - *Plan execution of improvement sprint*: The improvement sprint's goal is defined. The first sprint must have as its goal to define an initial process version that incorporates the selected improvement opportunities and which belongs to the improvement case that is being addressed with the current iteration. In this respect, the best practices described in a process reference model should be borne in mind to define the process that is being improved. The next improvement sprints should focus on putting the latest process version into practice and/or on incorporating in the definition of this process the rest of the improvement opportunities that have not been addressed yet. The specific tasks to define the improved process or to put it in practice are identified, estimated and selected (this task is related to Sprint planning meeting and Sprint backlog).
  - *Daily work*, *Manage problems*, *Update improvement opportunities*: the specific tasks to reach the improvement sprint goal are performed by the work team (related to development work).

- *Improvement sprint meeting*: an inspection of the progress towards the sprint goal is performed in order to regulate improvement tasks. Process improvement is an activity involving the whole organization, permeating all of the ongoing daily work of the software development in the company. It is also an activity that complements those day-to-day tasks and it requires an additional effort. That being so, we propose that the sprint meeting should take place once or twice a week, depending on the progress and needs of the improvement iteration (this task is related to daily Scrum).
- *Improvement sprint review*: the goal is to show the improvement work carried out during the current sprint. The work performed is assessed against the sprint goal. To carry out this review, it is important to consider that the objective of each sprint is to increase the improved process capability. This capability increase is given by incorporating best practices in the definition of the process and its implementation. These practices satisfy the improvement opportunities that have been selected for the current sprint. On the other hand, in this meeting, suggestions about the lessons learnt and problems found are also received, along with the most relevant results of the improvement sprints. This allows for feedback from these sprints, which must be taken into account for the following improvement iterations (this task is related to Sprint review).
- *Improvement sprint retrospective*: a retrospective analysis of the work carried out by team members that were involved in the last improvement sprint takes place. The feedback obtained supports ongoing learning and improvement of the sprint and iteration. In our context the ongoing learning and improvement involve: (i) to improve the process used for improving processes, and (ii) to identify new process improvement opportunities. This task entails revising and adjusting the improvement process used, in this case PfemCOMPETISOFT, to make it more effective and enjoyable for the next improvement sprint and iteration. Furthermore, actionable improvement identified from the daily work related to the process that is being improved or other processes is added to the improvement opportunities group (this task is related to retrospective sprint).

- *Presentation of improved process*: the improved process of the organization is presented, and the manner in which to institutionalize it within the entire organization is determined.
- *Presentation of next iteration*: the new improvement iteration is announced.

The work carried out in the activities *Planning the iteration* and *Designing the improvement case* should be registered in the *improvement implementation plan*. In addition, the work carried out in the other activities is registered in the *iteration improvement report*.

4.1.5. Roles and work products

Both process entities are derived from PmCOMPETISOFT. This process describes a competence set needed to perform the work of each role. These competences involve the skills necessary to improve software process and to increment their capability, such as improvement project execution and management, and assessment, definition and execution of software process. To use PfemCOMPETISOFT, it is necessary to consider additional competences related to Scrum for some roles. In this respect, the individual *responsible for process improvement* must have knowledge about the responsibilities of the ScrumMaster, and one person from the *management improvement group* must play the role of Product owner. On the other hand, in the following bullet points, we describe those work products which have a direct relationship with the proposed lightweight process:

- *General improvement plan*: this is the input work product to the lightweight process and it is a document is made up of two parts: (i) the *assessment report* which describes the state and analysis of the processes, and (ii) the *preliminary improvement plan* which defines: the improvement opportunities and their prioritization, the improvement cases, the number of improvement iterations, a draft of the general planning (it includes measures, training, risk management and chronogram sketch). The list of improvement opportunities is the requirements to be incorporated within processes that are being improved (related to Product backlog).
- *Improvement implementation plan*: a document which defines the high-level activities which must take place if the improvement cases are to be created, designed and executed (related to Sprint backlog). This document contains: the probable planning of the current iteration and a draft design of the improvement case (or improvement opportunities) that is addressed by this iteration. This design involves a definition of the process to be improved, which incorporates the selected improvement opportunities.
- *Improvement iteration reports*: this report includes relevant information on the performance and assessment of the current iteration and its sprints. In this respect, the specific tasks identified to meet the objectives of the different performed sprints are recorded (related to Sprint backlog). The analysis of the improvements brought into the organization's processes, the improved process documentation, the effort involved, achievements attained, lessons learnt, an iteration post mortem review, recommendations for improvement iteration adjustment, and conclusions about the improvement iteration from the point of view of the organization are recorded too.

4.1.6. Tools support

One factor that may assist small companies to successfully direct a process improvement initiative is that of providing technological support to such companies through software tools which will enable them to guide, implement and manage their improvement process (Pino et al., 2008). *Lightweight process to incorporate improvements* has therefore been described with the standard SPEM 2.0 and edited with the EPF Composer (Eclipse, 2007), in order to generate documentation in a standard format which is updated and which is available to organizations through the Web. It can be viewed in CYTED (2008). We have also developed a couple of tools to support the person *responsible for process improvement* during the improvement cycle, these are: (i) GENESIS (Hernández et al., 2008), which can be used for the management and implementation of all activities of an improvement cycle (including the formulation and execution of improvements), as well as in the administration of generated knowledge; and (ii) HEPALE! (Cruz et al., 2009), which is an educational tool to support dissemination-getting to know and understand the processes described by the *Process reference model* and the *Improvement framework*.

4.2. Correspondence between PfemCOMPETISOFT and Scrum

Based on the description presented in the previous section, an overview of the correspondence between the PfemCOMPETISOFT process entities and the Scrum elements is shown in Table 1.

The Scrum framework consists of a set of activities (the Release planning meeting, the Sprint planning meeting, the Sprint, the Daily Scrum, the Sprint review, and the Sprint retrospective), roles (the ScrumMaster, the Product owner and the Team) and work products (the Product backlog, the Release burndown, the Sprint backlog, and the Sprint burndown). From these elements, we have emphasized activities for defining PfemCOMPETISOFT, since these directly support the goal of this process, giving a break-down of the improvement iteration (activities of formulating improvements and executing improvements) in managing

Table 1
Relationship between PfemCOMPETISOFT process entities and Scrum elements.

PfemCOMPETISOFT		Related Scrum's element
Process entity	Key description related to Scrum	
Improved process	Process with higher capability level that is put into practice within the organization.	Product
Improvement opportunities set	Improvement opportunities are all best practices desired in the processes to be improved.	Product backlog
Planning the iteration	Goals and a draft plan (this is mid-term planning, by including the major risks, probable delivery date, work team and cost) of the *improvement iteration* are established. The highest priority improvement opportunities and cases are determined.	Release planning meeting
Designing the improvement case	Strategies are determined, along with their high-level activities to attempt the improvement sprints so as to satisfy the improvement opportunities. These activities are estimated and assigned to the members of the work team. An agreement is reached with the work team for the carrying out of the improvement iteration.	Sprint planning meeting Sprint backlog
Improvement sprint	Set of improvement activities conducted over a pre-defined period, which is used to implement and monitor the incorporation of the improvement opportunities within processes of the organization.	Sprint
Plan execution of improvement sprint	Goals and scope of the improvement sprint are defined. The specific tasks to define the improved process or to put it into practice are identified, estimated and selected. These tasks describe how the high-level activities perform. A short-term planning for these tasks is established.	Sprint planning meeting Sprint backlog
Improvement sprint meeting	A progress inspection towards the sprint goal is performed to regulate improvement tasks. In the context of a process improvement project we propose that this meeting take place once or twice a week.	Daily Scrum
Improvement sprint review	What was accomplished during the improvement sprint is shown. The success of the improvement sprint is assessed by contrasting the work performed against the sprint goal.	Sprint review
Improvement sprint retrospective	How the sprint went as regards team members, process and tools is analyzed. This feedback on the improvement sprint and iteration should support their ongoing learning and improvement.	Sprint retrospective
Responsible for process improvement	This person has the role of ScrumMaster.	ScrumMaster
Management improvement group	One single person from this group plays the role of Product owner.	Product owner
Preliminary Improvement Plan	This work product contains the improvement opportunities, their prioritization and grouping in improvement cases.	Product backlog
Improvement Iteration Reports	This work product contains the tasks identified to meet the sprints' objectives.	Sprint backlog

and carrying out the incorporation of improvement opportunities in the processes of the small organizations. Regarding the roles and work products, Table 1 establishes a relationship between these elements from PfemCOMPETISOFT and Scrum. For these elements, in our proposed process we have kept the same names as in PmCOMPETISOFT (without using the names from Scrum), in order to guarantee the coherence of these elements between both processes.

5. Application of the proposed lightweight process

The Action Research (A-R) and Case Studies methods have been used for the definition, refinement and application of the components developed in the context of the COMPETISOFT project. According to McKay and Marshall (2001) and Chiasson et al. (2009), A-R involves a *research cycle* and a *problem-solving cycle*, in which the knowledge is applied and discovered interactively between activities with different goals and outcomes. Each cycle includes at least the following activities: problem diagnosis, action intervention, and reflective learning (Avison et al., 1999). In this respect, the components of the COMPETISOFT's *improvement framework*, including PfemCOMPETISOFT (theoretical knowledge), were developed by means of the execution of various *research cycles*, and these components were applied by means of the execution of the *problem-solving cycles*. We have used the case study method to support the execution of the action intervention activity from the *problem-solving cycle*. The execution of the *problem-solving cycles* allowed us to discover practical knowledge, which was registered in the case study reports. This knowledge influenced the next research cycle, to discover new theoretical knowledge, which was included in the components of the *Improvement framework*, thus creating new, refined and improved versions of the components of this framework (including PfemCOMPETISOFT).

On the other hand, by using the A-R research method we divide the project participants into two groups: the first is made up of *researchers* from various universities and an national body of certification, and the second is called the *critical reference group* and includes computer professionals from small software organizations. As mentioned previously, the application of A-R is iterative and it has allowed us a continual feedback between the *researchers* and the companies involved, so we can polish up the components of the *Improvement framework*. With regards to the execution of the action intervention activity of the problem-solving cycle from A-R, the *researchers* applied the empirical variant (French and Bell, 1999). That is, PfemCOMPETISOFT was applied by the *researchers* in the small companies (*critical reference group*) by means of the use of the case study research method. In order to validate the proposed PfemCOMPETISOFT process we have conducted two case studies by following the protocol template for case studies presented in Brereton et al. (2008) (see Fig. 4).

It is important to highlight that in order to carry out the case studies of PfemCOMPETISOFT, it has also been necessary to use PmCOMPETISOFT, due to the fact that, to apply PfemCOMPETISOFT, a set of process improvement opportunities is needed and these opportunities are discovered by means of the diagnosis activity from PmCOMPETISOFT. That is to say, by means of the *diagnosing the process* activity of PmCOMPETISOFT, we determined the improvement opportunities, and these have been incorporated in the processes of the small organizations by using PfemCOMPETISOFT. Thus, that proposed process has been used to conduct the

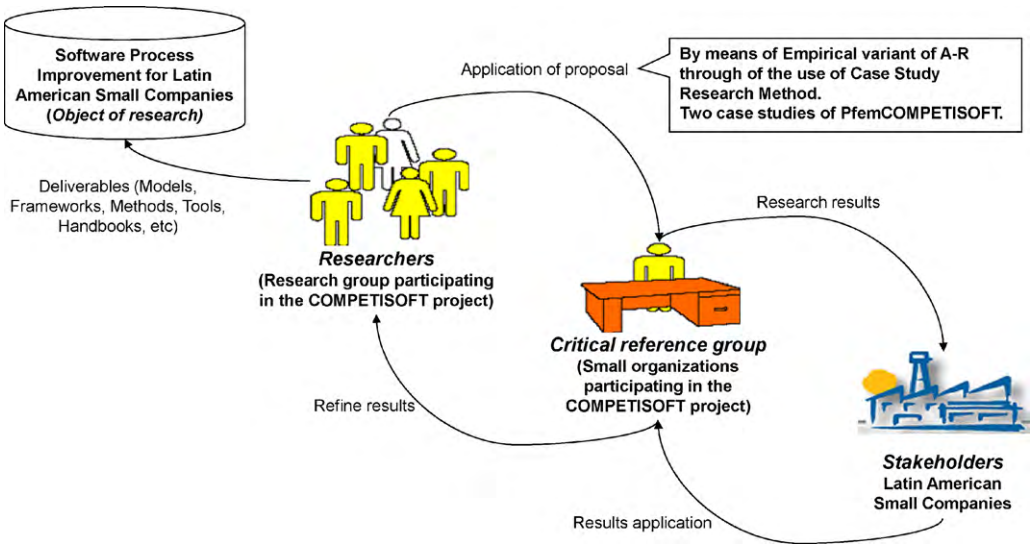

Fig. 4. Application of A-R and case studies to the COMPETISOFT project.

activities of *formulating improvements* and *executing improvements*. In this respect, this section shows how PfemCOMPETISOFT has been applied alongside PmCOMPETISOFT to give a complete vision of the improvement initiatives performed by the two small enterprises involved in the case studies. Although this section describes all the activities performed in the improvement initiatives, in Section 5.3 we emphasize the results of applying PfemCOMPETISOFT. Our next step in this paper will be to describe the case studies performed in terms of design, analysis unit, subjects, field procedure, data collection, and analysis.

5.1. Design, analysis unit and subjects

The *main research question* addressed is: Is the *Lightweight process to incorporate improvements* suitable (useful and practical) for driving the formulation and execution of improvements in small software enterprises? *Additional research questions* addressed by these case studies are: (i) Does the proposed lightweight process enable small companies to involve their employees in the execution of the improvement activities? (ii) Does the proposed lightweight process enable small companies to incorporate the improvement opportunities in their processes and put them in practice? (iii) Is the effort of applying the proposed lightweight process suitable for small companies? Taking into account the focus presented by Yin (2003), the *design type* of the case study in this work is multiple cases – holistic, since the strategy has been applied in the context of two small business. The *object of study* is the *Lightweight process to incorporate improvements*. The *measures* used to investigate the research questions are: (i) the effort involved in carrying out the activities associated with the SPI initiative, and (ii) the capability level of the processes under analysis (those which need to be improved). Furthermore, we also took into account the benefits described by the small organizations. In this vein, the *analysis units* are the improvement iteration composed of the activities of formulating and executing improvement, along with the processes to be improved by the organizations.

For the application and validation of the *study object*, we worked with two companies that form part of the *critical reference group* of the COMPETISOFT project, which were our research *subjects*. In order to protect their anonymity, in this paper we shall refer to these two companies as SmallComp1 and SmallComp2. Next, some features about these companies are described, including their products, business area and motivation to improve:

- *SmallComp1* is a small software development company with 5 years of experience at a nationwide level. The company's present staff is made up of seven people, six of whom are devoted to the development, operation, and maintenance of software products. The main work areas of the company include the development of organizational web portals, e-commerce solutions, geographic information systems, and support software for education centers. Although this company offers off-the-shelf products to its customers, most of its workload is given by projects for custom software development. Its average sales during the last 3 years have been $153,000. One of the main problems at *SmallComp1* was that the absence of visible and explicitly defined processes prevented the organization from growing accordingly to the increasing demands of its customers. That was the main reason for establishing a process improvement program. Furthermore, this company considers that software process improvement is important because "without it, the company has few possibilities of growth".
- *SmallComp2* is a small academic organization (in a Spanish university) with 13 years of experience at a nationwide level. It combines research and development activities with software development projects for other organizations (through contracts and trade agreements), which serve as a way of transferring research results to the industry. The group's staff currently consists of 21 people, 15 of whom work in the development, operation and maintenance of software products. As in the case of previous company, there were no defined processes for software development or project management. The organization took the strategy proposed by COMPETISOFT as the reference for improving the development and project management processes, which are critical due to the nature of its work. This organization considers that software process improvement is important because "if we know what we are doing, we can improve".

It is important to highlight that these two small companies are closely connected and have a strategic agreement to tackle certain software projects together. Depending on their workload at a given time, either of the two companies can request the development of a software product from its partner through outsourcing.

The top management groups of both companies have committed to process improvement as the support for the organization as well as for the systematic consolidation and growth of the companies. Neither of the two companies has previous experience in

software process improvement. However, at the present time it is important for them to increase the capability level of their processes, in the quest to gain clarity, organization and tracking of the processes involved in the development of their software products. There is also the desire to improve their processes, thus offering their customers higher quality products and services. This led to the commencement of a process improvement cycle in both companies last year, with the support of an advisor in process improvement from the group of *researchers* in the COMPETISOFT project. For this first improvement cycle we suggested to these companies that they should incorporate some process related to the Profile 1 of the *Process reference model* of COMPETISOFT, which includes: the Software development process – SD, the Software maintenance process – SM, and the Specific project administration process – SPA.

5.2. Field procedure and data collection

The PfemCOMPETISOFT process (see Fig. 3) that we have presented in this paper, along with the PmCOMPETISOFT process (see Fig. 2), were the guidelines for both improvement initiatives. From these diagrams we can see that the procedure for carrying out the field work is formed by one or more improvement cycles. An improvement cycle consists of five activities: initiating the cycle, diagnosing the process, formulating improvements, executing improvements and revising the cycle. In each improvement cycle, one or more improvement iterations must be carried out for the formulation and execution of the improvement opportunities encountered for the processes under intervention (processes to be improved). In this respect, we used PmCOMPETISOFT to conduct the initiating, diagnosing and revising activities and PfemCOMPETISOFT to guide the formulating and executing activities. This implies that the procedures governing the *field procedure* of the case studies and *data collection* plan are directly related to the activities, roles and work products described in PmCOMPETISOFT and PfemCOMPETISOFT. The data collected are those included in the work products, which are, as shown in Figs. 2 and 3: Improvement proposal, General improvement plan (made up the *assessment report* and the *preliminary improvement plan*), Improvement implementation plan and Improvement report (made up the Improvement iteration reports). It is important to highlight that PfemCOMPETISOFT and PmCOMPETISOFT provide a self-contained template for each work product, to make its construction and use easier. Furthermore, the data collected were stored by means of the use of these self-contained templates. At the end of the improvement cycle we interviewed the companies on their opinion of the results obtained from the improvement initiative.

A description of the execution of the *field procedure* and *data collection* is presented in the following sub-sections.

5.2.1. Initiating the cycle

The initiation of the improvement cycle involved several steps. The first one was the start-up of the improvement cycle. The COMPETISOFT advisor compiled all the necessary information about the companies. The top management of both companies and the COMPETISOFT advisor signed a document called *collaboration agreement for process improvement*, which guaranteed the agreement of all the parties in the execution of the improvement cycle.

After that, the tasks for the creation of the *improvement proposal* were performed. In their initial meeting, both companies recognized that their processes for developing software were chaotic. Given that fact, the *management improvement group* decided to establish only the *software development* – SD – and *specific project administration* – SPA – processes within the scope of the improvement cycle. The goal was to improve the practices of these processes themselves within the company by following the COMPETISOFT strategy. Another quantitative goal consisted of increasing the capability of the *software development and specific project administration processes* by one level. People were assigned to the roles established by the improvement process. The *management improvement group* of the organizations and the COMPETISOFT advisor selected a person *responsible for process improvement*. Finally, each company selected a pilot project through which to incorporate the process improvements.

Finally, the *improvement proposal* was socialized into the organization. The advisor and the person *responsible for process improvement* presented the project through a cycle start-up meeting with all the people involved in the pilot projects. The advisor shared with them the work which was to take place during the improvement cycle and obtained feedback about the employees' expectations of the improvement project.

5.2.2. Diagnosing the process

The process diagnosis involved different tasks. First of all, the tasks for the creation of the *assessment report* were performed. The scope for this first improvement cycle includes an assessment of the *software development and specific project administration processes* as opposed to the second capability level of the assessment method (Pino et al., 2010) of *METvalCOMPETISOFT assessment methodology* (which is conformance with ISO/IEC 15504-2). The COMPETISOFT advisor played the role of *evaluator*. The advisor assessed the processes by applying the technique of evidence gathering: interviews and surveys, using the information-gathering tools developed for this purpose.

When all the above had been done, the advisor had a meeting with another member, the person responsible for the *software development process* of each company, in order to obtain each company's activity diagram of the *software development and specific project administration processes* (see Fig. 5) and to apply the assessment tools. There was no meeting with the person responsible for the *specific project administration process*, because both companies expressed that they did not carry out any visible and formal activity related to this process.

The *software development and specific project administration processes* were assessed in the following process attributes: PA 1.1 Process performance, PA 2.1 Performance management, and PA 2.2 Work product management from the assessment method of METvalCOMPETISOFT (see Table 2). The assessment concluded that the processes had a zero level of capability (incomplete process). That is, the processes were not well implemented or did not attain their goals, there was little evidence of any systematic achievement of the goals of the process, and there were failures that limited or prevented the fulfillment of the goals of the process. In addition, few results or outputs were identified in the process.

The improvement opportunities of the processes evaluated were reported and prioritized by using as a base the information collected in this activity, together with the needs of the companies. Both companies decided to improve the *software development process* first and then improve the *specific project administration process*. For both companies the *preliminary improvement plan* was defined, in which the number of improvement iterations that make up the cycle of improvement was presented, together with the order they need to be carried out in and the overall schedule. This draft of the plan for the improvement cycle included three iterations, two related to the *software development process*, and another related to the *specific project administration process*. The most important risks for the performing of the cycle of improvement were determined and their corresponding management was registered. Moreover, the training on SPI of the employees involved in the improvement cycle was established. The advisor and the person *responsible for process improvement* created the *general improvement plan* with the *assessment report* and the *preliminary improvement plan*.

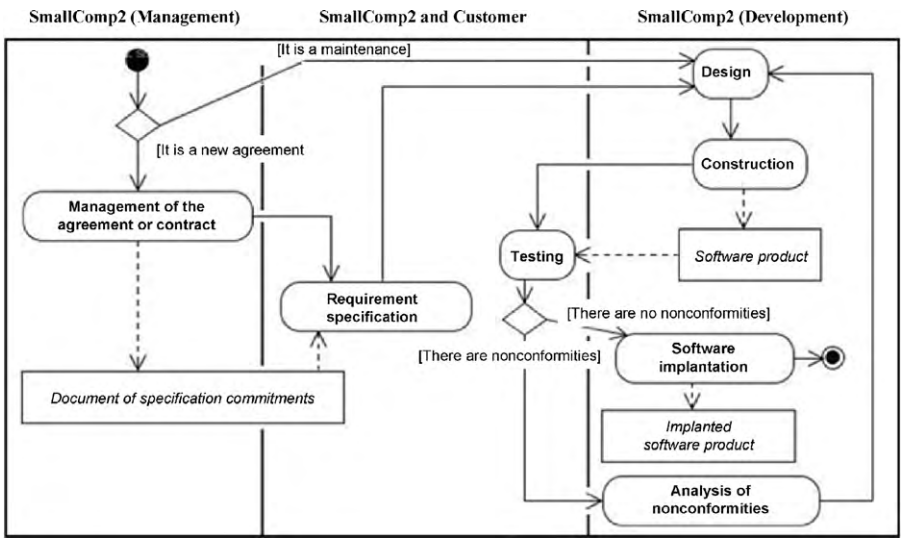

Fig. 5. Activity diagram for SmallComp2.

5.2.3. Formulation improvements

For the formulation of the improvements of the *software development and specific project administration processes*, the *management improvement group* and *process improvement group* (made up of the advisor and the *responsible for process improvement*) of each company carried out various activities. The *management improvement group* reviewed and validated the draft of the *preliminary improvement plan*. However, after taking the state of the processes, the company's needs, and the available resources into account, the *management improvement group* decided that for the *specific project administration process* it would be sufficient to implant initial basic practices in the process. Therefore, for each iteration the *process improvement group* used the improvement opportunities that had been found to make a mid-term plan and design the improvements incorporation for the *software development and specific project administration processes*, which were registered in the *improvements implementation plan*.

During the first improvement iteration, which was related to improving the *software development process*, the strategies for the improvement of this process were defined by the *process improvement group*. These strategies were also used in the next improvement iterations related to improving the *specific project administration process*. These strategies are: (i) the establishing of an effective communication channel between the advisor and the employees *responsible for process improvement* in the companies; (ii) continuous and rapid evaluation of the process following an assessment process; (iii) basic measurement of the process activities, through the measurement of the effort involved in the activities established in the improvement project, both at the level of the improvement process and at the level of the activities performed by the participants in the pilot projects; (iv) ongoing learning, which trains the people involved in the project in process improvement. In addition to this training, there needs to be joint work between the advisor and the *responsible for process improvement* so that the latter can be better equipped for the task of implanting an improvement process which will then carry on working in the company on a continuous and permanent basis; and (v) involvement of the company's employees (*responsible for process or participants*) in the definition of the *software development and specific project administration processes* and the techniques and document templates to be used in the activities specified for these processes. The *process improvement group* then established the mid-term planning of each improvement iteration, defining activities, resources, and responsibilities which support the strategies proposed previously.

As the capability level of the *software development process* is zero (incomplete process), it was necessary to create a definition of this process initially (the same thing happened with the *specific project administration process*). To support the achievement of this goal, a first improvement sprint, focusing on the definition of the process (under intervention by each iteration) took place. By means of this initial sprint, the *process improvement group* (team involved in the carrying out of this sprint) indentified and assigned the improvement tasks so as to create a draft design of the processes for each company. This version of these processes was defined using the activities and work products which satisfy the first level of the COMPETISOFT *process reference model*. The process pattern established by this *process reference model* was the basis for the process definition. Each company had to use this new process in its pilot improvement project.

5.2.4. Executing improvements

The processes defined in the previous activity were put into practice within the company by means of the execution of new

Table 2
Capability of the organization's processes.

Organization	Process	Process attributes			Capability level
		PA 1.1	PA 2.1	PA 2.2	
SmallComp1	Software development	0.18 (P)	0.04 (N)	0 (N)	0 (incomplete)
	Specific project administration	0 (N)	0 (N)	0 (N)	0 (incomplete)
SmallComp2	Software development	0.21 (P)	0.10 (N)	0.10 (N)	0 (incomplete)
	Specific project administration	0 (N)	0 (N)	0 (N)	0 (incomplete)

Values of process attributes: F, fully achieved; L, largely achieved; P, partially achieved; N, not achieved.

Table 3
Intermediate development process assessment.

Organization	Process	Increase of capability of PA's			Capability level
		PA 1.1	PA 2.1	PA 2.2	
SmallComp1	Software development	0.69 (from P to F)	0.37 (from N to P)	0.48 (from N to P)	1 (performed)
SmallComp2	Software development	0.66 (from P to F)	0.31 (from N to P)	0.38 (from N to P)	1 (performed)

N, not achieved (0–15% achievement); P, partially achieved (>15% to 50% achievement); L, largely achieved (>50% to 85% achievement); F, fully achieved (>85% to 100% achievement).

improvement sprints which pursued this goal. The work team responsible for carrying out these sprints was composed of the individual *responsible for process improvement* and the persons *responsible for processes* under intervention. This team identified the improvement tasks needed to put the defined process into practice within the company. The team self-organized the assignation of these tasks between members. As we have pointed out before, the improvement tasks were only applied in pilot projects, selected by the *management improvement group* of each organization. Thus, these tasks were carried out in a controlled environment, the complexity of their application was reduced and it affected only a small part of the company. Moreover, due to the close relationship between the two organizations, the *responsible for process improvement* was the same for both of them. Although this person had basic training in quality management and process-based organization, he was in constant contact with the COMPETISOFT advisor (through a weekly virtual meeting) in order to report on the evolution of these improvement tasks and to solve any possible doubts.

An improvement sprint focusing on implementing the defined process started in both organizations with an informative meeting directed towards the employees participating in the pilot projects (*responsible for process or participants*). In this meeting, the *responsible for process improvement* presented the assistants with a general view of the COMPETISOFT *process reference model* and a more detailed view of the process under intervention in the sprint (*software development or specific project administration process*). The *responsible for process improvement*, along with the assistants, also reviewed the goals of the improvement sprint, the activities completed up until that point, project milestones, way of working, etc. From this time on, the *responsible for process improvement* had at least one weekly meeting with each team, to work on the improvement tasks. In the first meeting associated with this type of improvement sprint, the definition of the corresponding draft (or latest version) process was presented to the participants, and the process was enriched and refined by following the suggestions of the employees involved in the improvements, thereby obtaining the approved final version of the process. The employees involved in the pilot improvement project used this new process in their daily activities. During the following meetings (to manage problems and conduct sprint) the participants discussed and clarified how to carry out each process activity and doubts were resolved. Furthermore, they worked together to create the templates for the documents produced in each activity of these processes (for example for the *software development process*, along with the project plan, requirements specification, system specification, user manual, operation manual, etc.). As well as all that, the participants updated the plan for the work of the improvement sprint that remained unresolved. In addition the *responsible for process improvement* updated the state and priority of the improvement opportunities in the *general improvement plan*.

The joint work of the person *responsible for process improvement* and the members of each team, their active participation, and the fact that they could make decisions with the advice of this person were key points in the execution and success of the improvement sprints and iterations. Upon the completion of each improvement sprint, a meeting took place with the participating employees, to discover their opinions of, and suggestions about, the good and bad aspects of the improvement sprint and iteration current and how they viewed the implementation and use of the new processes in the pilot project. Moreover, the institutionalization strategy for the new processes was that these should be used in all of the organization's new projects from that moment onwards.

5.2.5. Cycle revision

Twelve weeks after the commencement of the improvement cycle, the advisor carried out a new assessment of the development process (only in the pilot projects) in both organizations. As we can see in the results shown in Table 3, this process increased its capability to level one. Furthermore, the *improvement reports* generated for each organization show that base practices of the *specific project administration* process were implanted in both organizations. The implantation of these practices allowed the value of process attributes of this process to be increased.

A specific self-contained template was developed for each work product, to make its construction easier. An important item in these templates is to register the effort. By doing this, the effort of carrying out the tasks associated with each activity related to say products was registered in each of the work products. Table 4 shows the effort devoted to each activity of the improvement cycle.

The table shows that the time devoted by the COMPETISOFT advisor to SmallComp1 is zero in the activity of improvement execution. This is because the time the COMPETISOFT advisor devoted during the execution phase corresponds with the weekly meetings with the *responsible for process improvement*, which were the same for both organizations.

On the other hand, we asked companies about the improvement initiative, by means of an unstructured interview carried out with the *responsible for process improvement* and with one *responsible for process* which was improved. From the point of view of the companies, we can highlight several important conclusions reached after the completion of the first improvement cycle:

Table 4
Effort for the improvement cycle.

	COMPETISOFT	Company
Initiating		
SmallComp1	1 Ps. × 4 h	1 Ps. × 4 h
SmallComp2	1 Ps. × 9 h	1 Ps. × 4 h
Diagnosing		
SmallComp1	1 Ps. × 6 h	1 Ps. × 4.5 h
SmallComp2	1 Ps. × 19 h	1 Ps. × 4 h
Formulating		
SmallComp1	1 Ps. × 1 h	1 Ps. × 0.5 h
SmallComp2	1 Ps. × 3 h	1 Ps. × 1 h
Executing		
SmallComp1	1 Ps. × 0 h	1 Ps. × 24 h
SmallComp2	1 Ps. × 5 h	1 Ps. × 30 h
Revising		
SmallComp1	1 Ps. × 4 h	1 Ps. × 6 h
SmallComp2	1 Ps. × 5 h	1 Ps. × 8 h

Ps. = person; h = hours.

- The application of COMPETISOFT and specifically the approach offered by PmCOMPETISOFT and the *Lightweight process to incorporate improvements* in driving the process improvement allowed both organizations to obtain an important and rapid improvement in two of the key processes for them: *software development* and *specific project administration*.
- Obtaining good results in relatively short periods was an important aspect for the motivation and involvement of the participants in a project like this. Seeing such rapid results and their direct participation caused the employees to realize the possibilities of process improvement in general, and COMPETISOFT in particular, despite the initial reticence that these projects might have caused.
- Applying the improvements in pilot projects significantly reduced the resources needed and the risk associated with the implementation of improvements in key processes of the companies.
- At the end of the first cycle of improvement, both organizations had moved from a chaotic and unpredictable software process to a tangible one, which is currently being used on development projects. Both the management and the employees of the companies have seen the benefits of this result and, most importantly, they have realized the need to maintain continuous and ongoing improvement, following the same approach as in this first cycle.

5.3. Analysis and discussion

In this section, we highlight the most relevant aspects of this first application of the proposed PfemCOMPETISOFT process in the two case studies by means of a qualitative data analysis. By means of this analysis we try to derive results and conclusions from the evidence given by the data collected.

In both companies the management group is committed to the improvement project, since they have realized that, in order to grow and to do better, the company needs to be managed through the strategy of process improvement, from the execution of the software development process to the software process management. At the commencement of the improvement cycle, the software development process was chaotic, as can be deduced from Table 2 in which the capability level of the processes assessed in both companies is zero. This makes it clear that there is no visibility in the processes of these companies, which signifies that the software development has a far greater dependence upon people rather than upon processes. For example, as we can see in Fig. 5, the only visible product of the SmallComp2 is the software product. In addition, there is no possibility of traceability of this software product, with the problems that this implies.

To implant the improved processes, and in an effort to increase their capability, strategies with which to tackle the skepticism and resistance to change on the part of the development group had to be considered. Those attitudes are innate to this type of improvement project, but it is plain that developers must feel confident about the way in which they are working. One strategy for addressing this risk is the formation of group development in process improvement, emphasizing the advantages that can lead to conducting the daily work with processes which incorporate better practices. Another is to foster the group's implication in the improvement project in such a way as to support and contribute to the definition and implementation of the processes to be improved, thus causing the improvement to be bottom-up.

We have used the Scrum agile method satisfactorily to establish improvement strategies through the definition and application of the *Lightweight process to incorporate improvements*. In the case studies carried out, the Scrum agile method was used to define and implant processes in the two small organizations, by means of the proposed lightweight process. Through this new lightweight process that had been developed, the improvement iterations and sprints were used to strengthen the work and collaboration of the employees taking part in these iterations, in the quest to generate early and ongoing achievement of improvements in both processes. Furthermore, in both organizations PfemCOMPETISOFT allowed us to include all those who were involved in an improvement iteration in the work carried out to incorporate the improvement opportunities within the processes under intervention and with which they have some relationship. To be specific, we have used Scrum in our lightweight process to support the definition and putting into practice of processes within small organizations. After carrying out PfemCOMPETISOFT, the desired product is a process with a higher capability level and this is put into practice. This capability increase is given by incorporating best practices (from a process reference model) on the definition of the process and its implantation.

On the other hand, PmCOMPETISOFT describes the activities of formulating and executing improvement prescriptively. However, from the application of PfemCOMPETISOFT we observed that this process has allowed us to give more flexibility to the team involved in the improvement iteration during the carrying out of these activities. This flexibility is supported by all elements of Scrum that have been adapted and included in lightweight process, but especially by the improvement sprint, since some sprints should seek to define processes and others to implement these processes. The first sprint must have as its goal to define an initial process version in order to create a work product upon which the improvement team can work in the following sprints. In PfemCOMPETISOFT the boundary between formulating and executing is not clearly traced; it is more that an improvement sprint can identify activities to refine and extend the definition of the process and also put it in practice. The scope of an improvement sprint depends on the complexity of the improvement opportunities and cases to be tackled in the improvement iteration. In this sense, from case studies we observed that the goal of the activity for formulating improvement (to define the improvement) was satisfied by means of the execution of an early improvement sprint with this aim. Similarly, the goal of the executing improvement activity (to put the defined process into practice within the company) was achieved through the improvement sprints that followed, which focused on this goal.

We could add that, according to Komo-Sirviö (2004) one of the activities which is critical to the success of an improvement process is that of carrying out (implementing and monitoring) a specific improvement plan. Basing our opinion upon the application of PfemCOMPETISOFT, as well as from results obtained from the case studies, we consider that using Scrum to manage the implementation of improvements in small businesses is a suitable way to support this success factor. It was also appropriate for the monitoring and supervision of the SPI project.

Although the whole idea of Scrum (and not of just a few elements) pursues self-management, we have tried to analyze how some of Scrum's elements are related both to some of the early principles defined to PmCOMPETISOFT and to some success factors involved in process improvement in small businesses (see Table 5). The relationship shown in this table is based on our subjective analysis of the description of Scrum's elements and their stronger connection with some of these principles and success factors. For example, even though the whole of Scrum supports a rapid investment return, we believe that the sprint concept, which allows the creation of increments of the improved process in the short-term, is more directly related to this success factor.

As seen from the table, Scrum has a relationship with the vast majority of principles and factors considered. Factors and principles related to: (i) early and ongoing achievement of improvement and learning, and (ii) effective participation, collaboration and communication of employees in the improvement project are strongly supported. We have not related any element of Scrum with the

Table 5
Relationship between practices of Scrum and SPI' success factors.

	Scrum's elements							
	Product backlog	Release planning meeting	Sprint	Sprint planning meeting	Sprint backlog	Scrum meeting	Sprint review	Sprint retrospective
PmCOMPETISOFT's principles								
Early and ongoing achievement of improvements	X	X	X	X	X	X	X	X
Continuous and rapid process diagnosis								
Elemental process measurement		X	X					
Effective group collaboration and communication		X	X	X		X	X	X
Continuous learning		X	X	X		X	X	X
Success factors of SPI								
Attainment of a rapid investment return			X					
Definition of feasible SPI objectives with the available resources	X	X		X	X			
Participation of employees in SPI (button-up)		X	X	X	X	X	X	X
Suitable communication between SPI participants		X	X	X		X	X	X
SPI should be based upon learning, not upon control		X	X	X		X	X	X

principle of process diagnosis, because this principle involves carrying out a rapid software process assessment by following a focus on reviewing the process capability. The component of our *Improvement framework* that gives support to this principle is MEValCOMPETISOFT assessment methodology.

In addition, as can be seen in Table 4, the execution of improvements is the direct responsibility of the organization's employees, and the advisor's effort in this is minimal. The fact of having used Scrum through the proposed lightweight process to guide the activities of formulation and execution of the improvements has been a determining factor in the success of the improvement cycle, since its management practices allowed the person *responsible for process improvement* to involve the employees who participated, giving them an active part in the process improvement, thus generating improvements (define the process and its implementation) in a short amount of time. Communication (usually informal) took place between the *responsible for process improvement* and the participating employees in the improvement iterations, through which problems were quickly resolved and decisions were made rapidly, thus allowing to continue the implementation of improvement cases. On top of all this, it offered the employees who took part flexibility in the incorporation into the processes (under intervention) of the improvement opportunities through the iterations. The flexibility which the *Lightweight process to incorporate improvements* provided for the *responsible for process improvement* and the employees participating in the improvement project is seen in two aspects: first of all in who built the improvements and secondly in what they chose to work on and in what order.

On the other hand, Table 4 shows the effort involved in the improvement cycle following PmCOMPETISOFT and the *Lightweight process to incorporate improvements* by both organizations. The total effort for SmallComp1 was 54 h (h), of which 28% (15 h) corresponds to the advisor and 72% (39 h) corresponds to the company. For SmallComp2 the total effort was 88 h, 46% (41 h) of which corresponds to the advisor and 54% (47 h) corresponds to the company. Note that the effort involved on the part of the companies in the performance of the activities of the improvement cycle is similar (39 and 47 h, respectively). However, the effort of the COMPETISOFT advisor is greater (almost triple) in the case of SmallComp2. This is because the gathering and analysis of information relating to the activities of initiating, diagnosing and formulating were performed first for this organization, and then for SmallComp1. That is, this effort is related to the learning and experience acquired by the advisor in the tasks and products that had to be carried out for the carrying out of the improvement process activities.

It could be said that the effort related to SmallComp2 corresponds to a company which was inexperienced in improvement (it was carrying out an improvement cycle for the first time), and that the effort related to SmallComp1 corresponds to the execution of an improvement cycle in a company which was already familiar with PmCOMPETISOFT and the *Lightweight process to incorporate improvements*. Fig. 6 shows the distribution and tendency of the effort involved on the part of the advisor and of the companies in each of the activities carried out in the two case studies. This figure shows that the greatest effort in the improvement cycle was in the final activities of the process (executing and reviewing), and that this effort falls back upon the organizations. This demonstrates the importance of offering agile strategies for the management and realization of this work. We can also note that the advisor's participation in the initial activities was of a high level (initiating, diagnosing and formulating).

If we bear in mind that the first improvement cycle lasted 12 weeks, then the average effort for each of the companies was approximately 6 h per week (taking into account the advisor's and the company's time). We propose that one person should be assigned to the *responsibility for process improvement* for a quarter

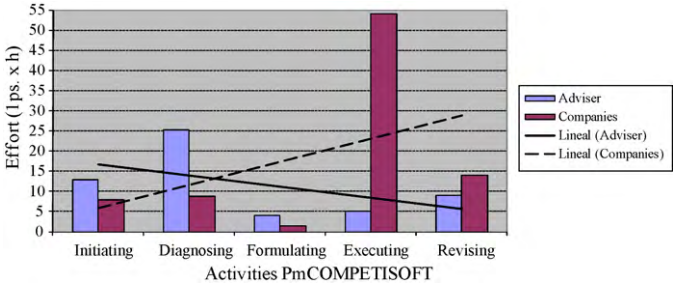

Fig. 6. Distribution and tendency of effort in the case studies.

of their time, which is a load that we consider can be borne sufficiently by small businesses. If we bear this aspect in mind then the processes proposed and described in this paper are better fitted to the characteristics of small organizations. This is because it is not necessary for them to employ a person who carries out the improvement initiatives on a full-time basis, as is the case in Casey and Richardson (2004) and Kautz et al. (2000) in which the IDEAL model is applied to small and medium organizations. Furthermore, these studies highlight the importance of the Management steering group in the IDEAL model as a success factor in process improvement. In our proposed processes this aspect corresponds to the *management improvement group*.

The results of the improvement cycles in both companies were evaluated through unstructured interviews to the main stakeholders. These interviews revealed that *SmallComp1* was satisfied with the result of the improvement cycle, since having a visible and explicitly defined process for software development and project management permitted the company to attend the demands of more customers, and to extend the staff. *SmallComp1* particularly appreciated the approach of working with small improvement steps with a tangible short-term reward. The program established the base for process improvement, and Company1 is currently enhancing its processes to be CMMI level 2 compliant. Regarding *SmallComp2*, the improvement cycle permitted to this organization to have a better insight and control on the software development and project management processes. As in the case of *SmallComp1*, the organization particularly appreciated the approach of working with small improvement steps with short-term reward.

It is important to highlight that in the first improvement cycles, the COMPETISOFT advisor worked closely with the person *responsible for process improvement* in each company, and they were thus able to acquire training and experience in process improvement. The first improvement cycles and the improved processes established the basic infrastructure with which to support the process for incorporating improvements, along with software processes management, paving the way towards the objective of becoming ongoing and continuous processes in the small companies. In this sense is important to underline the work that SmallComp1 has started in order to achieve a formal assessment at CMMI level 2.

The case studies carried out and presented in this paper have some limits: (i) the observations and conclusions presented are based on two case studies, which can limit the power of generalization, and (ii) the bias of the case studies, because the development of daily activities by employees may proceed differently precisely because they are being observed.

6. Conclusions and future work

In this work, an approach that focuses on agile management such as Scrum has been integrated into a process which sets out to guide the process improvement in small companies. The result has been the *Lightweight process to incorporate improvements* (*PfemCOMPETISOFT*), which defines in detail the elements needed to guide the activities of formulating and executing improvements (improvement iteration) in a small organization. PfemCOMPETISOFT extends the improvement iteration of our primary PmCOMPETISOFT process with ideas drawn from the Scrum method. This proposal is integrated into PmCOMPETISOFT process, the goal being to form a comprehensive package for the teams in charge of improvement formulation and execution in a small organization. That is possible thanks to the fact that the Scrum agile method provides a project management framework and that the proposed processes supply suitable improvement practices. Our proposal also attempts to make it possible to establish process improvement with a minimum of expense, that is, with few resources and in a short time. We have also presented how this process has been set up in two small organizations by means of an improvement project.

The description of the case studies undertaken and the analysis carried out gives evidence that the *Lightweight process to incorporate improvements* is useful and practical for driving the formulation and execution of improvements in small software enterprises. This evidence is related to the increase of the capability of the processes to be improved, the effort of applying the proposed process, the joint work done between the person *responsible for improvement process* and the person *responsible for process or participant* in implementing improvement opportunities, and the benefits described by the companies. Thus the results in terms of: the final process capability obtained, the effort, the benefits and the internal organization of improvement work, are an indicator that the *Lightweight process to incorporate improvements* enables small companies to involve their employees in the execution of the improvement activities for incorporating the improvement opportunities in their processes, investing a effort that is suitable for this type of company.

The application of Scrum as support to the carrying out of formulation and execution activities was significant for the success of the improvement iterations and consequently of the improvement cycle in the organizations. Notice that none of the companies had staffs who were experts in process improvement. A first advantage of this approach is that the effort needed to manage and carry out the improvement tasks was appropriate to the resources of both companies. Perhaps the most important consequence of applying Scrum is that the improvement opportunities were incorporated in a more comfortable way than by following a rigid and prescriptive process. Following this approach, the improvement happened in a more "natural" way for all employees involved (from programmers to managers), making any initial reticence disappear.

Although this process has been designed and developed to support the activities of formulating and executing improvements of PmCOMPETISOFT, this can also be used independently, provided that there is an input work product, which describes a set of process improvement opportunities, along with their prioritization. Furthermore, the process ensures that the improvement process keeps on working within the company, so promoting continuous process improvement.

Given that the results of the case studies are encouraging, new improvement cycles are planned for the two organizations, taking into account the aspects discovered in the first cycles. We shall also monitor the setting up of the *Improvement framework* in the software process improvement projects that are being carried out by Latin American companies involved in COMPETISOFT. Our goal is to obtain feedback from a representative set of case studies that can be used to refine and improve the *Lightweight process to incorporate improvements* and the rest of the components of the *Improvement framework* of COMPETISOFT.

Acknowledgments

We would like to thank anonymous reviewers for their useful comments in order to improve this paper, and we acknowledge the assistance of the following projects: INGENIO (PAC08-0154-9262, JCCM of Spain), ARMONIAS (PII2109-0223-7948, JCCM of Spain) and PEGASO-MAGO (TIN2009-13718-C02-01, FEDER and MEC of Spain). Acknowledgements by the first author to the research fellowships granted by JCCM and funded by European Regional Development Fund (ERDF).

References

Abrahamsson, P., Salo, O., Rankainen, J., Warsta, J., 2002. Agil Software Development Methods: Review and Analysis. VTT Publications 478, Finland.

Ambler, S., 2002. Agile Modeling: Effective Practices for Extreme Programming and the Unified Process. John Wiley & Sons, Inc., New York.
Avison, D., Lan, F., Myers, M., Nielsen, A., 1999. Action research. Communications of the ACM 42 (1), 94–97.
Beck, K., 1999. Embracing change with extreme programming. IEEE Computer 32, 70–77.
Brereton, P., Kitchenham, B., Budgen, D., Li, Z., 2008. Using a Protocol Template for Case Study Planning. in Evaluation and Assessment in Software Engineering. BCS-eWIC, Bari, Italia, p. 1–8.
Calvo-Manzano, J.A., Cuevas, G., San Feliu, T., De Amescua, A., Pérez, M., 2002. Experiences in the application of software process improvement in SMES. Software Quality Journal 10 (3), 261–273.
Casey, V., Richardson, I., 2004. A practical application of the IDEAL model. Software Process: Improvement and Practice 9 (3), 123–132.
Chiasson, M., Germonprez, M., Mathiassen, L., 2009. Pluralist action research: a review of the information systems literature. Information Systems Journal 19 (1), 31–54.
Coleman, G., O'Connor, R., 2008. Investigating software process in practice: a grounded theory perspective. Journal of Systems and Software 81 (5), 772–784.
Cruz, R., Morales, M., Morgado, M., Pino, F., Oktaba, H., Ibargüengoitia, G., Piattini, M., 2009. Supporting the software process improvement in very small entities through e-learning: the HEPALE! Project. In: Mexican International Conference on Computer Science (ENC 2009). Mexico City: IEEE Computer Society, pp. 221–231.
CYTED, 2008. COMPETISOFT Methodological Framework on EPF Composer (in Spanish). Available from: http://alarcos.inf-cr.uclm.es/Competisoft/framework/.
Eclipse, 2007. Eclipse Process Framework Project (EPF). Available from: http://www.eclipse.org/epf/.
French, W.L., Bell, C.H., 1999. Organization Development: Behavioral Science Interventions for Organization Improvement. Prentice-Hall, London.
Hernández, M., Florez, A., Pino, F., Garcia, F., Piattini, M., Ibargüengoitia, G., Oktaba, H., 2008. Supporting the improvement process for small software enterprises through a software tool. In: Software Enginering Symposium during Ninth Mexican International Conference on Computer Science (ENC'08), Mexicali, México: SES Proceedings, pp. 1–8.
Hofer, C., 2002. Software development in austria: results of an empirical study among small and very small enterprises. In: Proceedings of the 28th Euromicro Conference (EUROMICRO'02), IEEE Computer Society, pp. 361–366.
Horvat, R.V., Rozman, I., Györkös, J., 2000. Managing the complexity of SPI in small companies. Software Process: Improvement and Practice 5 (1), 45–54.
Hunt, A., Thomas, D., 2000. The Pragmatic Programmer. Addison Wesley.
Hurtado, J., Bastarrica, C., 2006. Implementing CMMI using a combination of agile methods. CLEI Electronic Journal 9, 1–15.
Hurtado, J., Pino, F., Vidal, J., Pardo, C., Fernandez, L., Agile, S.P.I., 2008. Software process agile improvement, a colombia approach to software process improvement in small software organizations. In: Oktaba, H., Piattini, M. (Eds.), Software Process Improvement for Small and Medium Enterprises: Techniques and Case Studies. Idea Group Inc., USA, pp. 177–192.
ISO, 2004a. ISO/IEC 15504-2:2003/Cor.1:2004(E). Information technology – Process assessment – Part 2. Performing an assessment. International Organization for Standardization. Available from www.iso.org.
ISO, 2004b. ISO/IEC 15504-4:2004 Information technology – Process assessment – Part 4. Guidance on use for process improvement and process capability determination. International Organization for Standardization. Available from www.iso.org.
ISO, 2008. ISO/IEC TR 15504-7:2008. Information technology – Process assessment – Part 7. Assessment of organizational maturity. International Organization for Standardization.
Kähkönen, T., 2005. Life cycle model for software process improvement project deploying an agile method. In: Proceedings of the International Conference on Agility (ICAM 2005), Helsinki, Finland, pp. 1–7.
Kautz, K., Hansen, H.W., Thaysen, K., 2000. Applying and adjusting a software process improvement model in practice: the use of the IDEAL model in a small software enterprise. In: Proceedings of the 22nd International Conference on Software engineering (ICSE 2000), Limerick, Ireland, pp. 626–633.
Komo-Sirviö, S., 2004. Development and Evaluation of Software Process Improvement Methods. Espo 2004, VTT Publications 535.
Krasner, H., 2001. Accumulating the body of evidence for the payoff of software process improvement. In: Hunter, R.B., Thayer, R.H. (Eds.), Software Process Improvement. Wiley-IEEE Computer Society, pp. 519–540.
Laporte, C., Alexandre, S., O'Connor, R., 2008. A software engineering lifecycle standard for very small enterprises. In: EuroSPI 2008. Springer CCIS, Dublin, Ireland, pp. 129–141.
MBI, 2004. *Panorama de la Industria del Software en Latinoamérica*. Mayer & Bunge Informática LTDA, Available from http://www.mbi.com.br/MBI/biblioteca/relatorios/200409panswlatam/200409_panorama_industria_software_america_latina.pdf, pp. 97.
McKay, J., Marshall, P., 2001. The dual imperatives of action research. Information Technology & People (special issue on Action Research in Information Systems) 14 (1), 46–59.
Oktaba, H., Garcia, F., Piattini, M., Pino, F., Alquicira, C., Ruiz, F., 2007. Software process improvement: the COMPETISOFT Project. IEEE Computer 40 (10), 21–28.
Pino, F., Pardo, C., García, F., Piattini, M., 2010. Assessment methodology for software process improvement in small organizations. Information and Software Technology, doi:10.1016/j.infsof.2010.04.004.
Pino, F., Garcia, F., Piattini, M., 2008. Software process improvement in small and medium software enterprises: a systematic review. Software Quality Journal 16 (2), 237–261.
Pino, F., Hurtado, J., Vidal, J., García, F., Piattini, M., 2009a. A process for driving process improvement in VSEs. In: International Conference on Software Process (ICSP 2009), Vancouver, Canada: LNCS 5543, pp. 342–353.
Pino, F., García, F., Piattini, M., 2009b. An integrated framework to guide software process improvement in small organizations. In: European Systems & Software Process Improvement and Innovation (EuroSPI 2009), Madrid, Spain: CCIS 42, Springer, pp. 213–224.
Pino, F., Garcia, F., Piattini, M., 2009c. Key processes to start software process improvement in small companies. In: Proceedings of the 24th Annual ACM Symposium on Applied Computing (SAC'09), Honolulu, Hawaii, USA, pp. 509–516.
Richardson, I., Wangenheim, C.G.v., 2007. Why are small software organizations different? IEEE Software 24 (1), 18–22.
Salo, O., Abrahamsson, P., 2007. An iterative improvement process for agile software development. Software Process: Improvement and Practice 12 (1), 81–100.
Schwaber, K., 1995. The scrum development process. In: OOPSLA'95 Workshop on Business Object Design and Implementation. ACM Press, Austin, Texas, USA, pp. 1–23.
Schwaber, K., 2009. *Scrum Guide*. ScrumAllienace. Available from www.scrumalliance.org/resource_download/598.
Scott, L., Jeffery, R., Carvalho, L., D'Ambra, J., Rutherford, P., 2001. Practical software process improvement—the IMPACT Project. In: Proceedings of the Australian Software Engineering Conference, pp. 182–189.
Wangenheim, C.G.v., Weber, S., Rossa Hauck, J.C., Trentin, G., 2006. Experiences on establishing software processes in small companies. Information and Software Technology 48 (9), 890–900.
Yin, R.K., 2003. Case Study Research: Design and Methods. Sage Publications, Newbury Park.

Francisco J. Pino has an European PhD in Computer Science from the University of Castilla-La Mancha (UCLM), Spain. He is currently an associate professor at the Electronic and Telecommunications Engineering Faculty at the University of Cauca, in Popayán (Colombia). He is member of the IDIS Research Group and his research interest is Software process improvement in small companies and Harmonization of multiple improvement technologies.

Oscar Pedreira holds MSc and PhD in Computer Science from University of A Coruña, Spain. He is assistant professor at the same university, and a member of the Databases Laboratory. He has been involved in several research projects on Database Technology and Information Systems, Software Engineering, and Geographic Information Systems.

Félix García received his MSc (2001) and PhD (2004) degrees in Computer Science from the University of Castilla-La Mancha (UCLM). He is currently an associate professor in the Department of Information Technologies and Systems at the UCLM. He is member of the Alarcos Research Group and his research interests include business process management, software processes, software measurement, and agile methods.

Miguel R. Luaces received his MSc degree in Computer Science from the University of A Coruña (Spain) in 1998 and an European PhD in Computer Science from the University of A Coruña (Spain) in 2004. He undertook research in the area of spatial, temporal and spatio-temporal databases at the FernUniversität Hagen (Germany) under the ChoroChronos project funded by the European Union. Today, he is an Assistant Professor at the University of A Coruña, and he is currently a member of the Databases Laboratory of the University of A Coruña where he has been involved successfully in a number of research and development projects. His research interests include Geographic Information Systems, Spatial and Spatio-temporal Databases, Software Engineering, and Web-based Information Systems.

Mario Piattini has MSc and PhD in Computer Science from the Technical University of Madrid and is a Certified Information System Auditor and Certified Information Security Manager by ISACA (Information System Audit and Control Association). He is a professor in the Department of Computer Science at the University of Castilla-La Mancha, in Ciudad Real, Spain. Author of several books and papers on software engineering, databases and information systems, he leads the ALARCOS research group of the Department of Information Systems and Technologies at the University of Castilla-La Mancha, in Ciudad Real, Spain. His research interests are: software process improvement, database quality, software metrics, software maintenance and security in information systems.