

A research framework for building SPI proposals in small organizations: the COMPETISOFT experience

Francisco J. Pino¹ · Félix García² · Mario Piattini² · Hanna Oktaba³

Published online: 7 May 2015
© Springer Science+Business Media New York 2015

Abstract Establishing a research strategy that is suitable for undertaking research on software engineering is vital if we are to guarantee that research products are developed and validated following a systematic and coherent method. We took this into account as we carried out the COMPETISOFT research project, which investigated software process improvement (SPI) in the context of Latin American small companies. That experience has enabled us to develop a research strategy based on the integrated use of action research and case study methods. This paper introduces the proposed research strategy and provides extensive discussion of its application for: (1) developing the Methodological framework of COMPETISOFT for SPI, (2) putting this framework into practice in eight small software companies, and (3) refine the Methodological framework due to the practice feedback. The use of this research strategy allowed us to observe that it was suitable for developing, refining, improving, applying, and validating COMPETISOFT's Methodology framework. Furthermore, having seen it applied, we believe that this strategy offers a successful integration of action research and case study, which can be useful for

✉ Francisco J. Pino
fjpino@unicauca.edu.co

Félix García
Felix.Garcia@uclm.es

Mario Piattini
Mario.Piattini@uclm.es

Hanna Oktaba
Hanna.Oktaba@ciencias.unam.mx

¹ IDIS Research Group, Electronic and Telecommunications Engineering Faculty, University of Cauca, Calle 5 # 4 – 70, Popayán, Colombia

² ALARCOS Research Group, Information Systems and Technologies Department, University of Castilla–La Mancha, Paseo de la Universidad 4, 13071 Ciudad Real, Spain

³ KUALI-KAANS Research Group, Facultad de Ciencias, National Autonomous University of Mexico (UNAM), Ciudad Universitaria, 04510 Mexico, D.F., Mexico

conducting research in other software engineering areas which address needs of small software companies.

Keywords Qualitative methods · Action research · Case study · Software process improvement · Small organizations · SME · COMPETISOFT

1 Introduction

According to Basili (2000), the understanding of a discipline entails learning, that is to say: (1) observation, reflection, and encapsulation of knowledge; (2) model building (application domain, problem-solving processes); (3) experimentation; and (4) model evolution. Software engineering (SE), like any other discipline, has two fundamental goals (Philips 1998), which are to: (1) increase theoretical knowledge, so as to understand why things happen in a particular area of interest and (2) improve the practice of the particular area so that research results can be useful. But new theoretical knowledge (models, process, techniques, etc.) is often proposed without its practical application, a fact which generates a clear disconnection between theoretical research and its application (Moody 2000). To avoid this lack of connection, it is fundamental to analyze the available research methods for SE, if we are to find out which research strategy is most suitable for the particular needs and features of the research to be tackled.

On the other hand, the discipline of SE related to software process improvement (SPI) in the context of the small enterprises (SMEs) of the Latin America software industry was researched by the COMPETISOFT project (Oktaba et al. 2007) among others. In this project, a *Methodological framework* was created, whose goals were the software processes improvement seeking to increase the level of competitiveness of these firms. In developing the COMPETISOFT project, over 100 researchers were involved. They came from: a national body for standardization and certification, more than 10 SMEs, and 27 university research groups from 13 countries in Latin America. Given the large scope of this project, initially it was necessary to define an appropriate *research strategy* to guide and coordinate the work of all the researchers. This strategy was based on some research methods for SE and enabled us systematically: (1) to obtain in-depth knowledge of SPI in SMEs, (2) to generate theoretical knowledge in this area (*Methodological framework*), and (3) to apply the proposed framework in some companies. The successful experience gained with COMPETISOFT project encouraged us to present the generalization of the *research strategy* that we defined to carry out the project.

In this sense, the contribution of the paper is the description of a *research strategy* that: (1) contains an explicit research process, (2) is predominantly focused on action research (AR), but also (3) integrates with AR a positivist case study (CS) approach within the research process. The paper also presents how we used this strategy to develop the components that make up COMPETISOFT's *Methodological framework* and to apply them in eight SMEs. By means of that *research strategy*, we have developed, refined, and validated the components of this *framework* systematically and have observed that it is suitable for developing, refining, improving, applying, and validating the proposal in the context of SPI for SMEs. We believe it can also be useful for conducting research in other SE areas with similar characteristics, for example for developing and validating of standard proposals for software engineering.

The rest of the article is structured as follows: In Sect. 2, an overview of action research and case study is given, along with a discussion of related work. Section 3 describes the generalization of the *research strategy* developed for the COMPETISOFT project as an integration of AR and CS methods. The application of this strategy is shown in Sect. 4, and contribution of it to software standards is presented in Sect. 5. A discussion of this work is offered in Sect. 6, and at the end of the paper, we present conclusions.

2 Background

This section is based on studies found in fields: SE and information systems (IS), since they have certain elements in common. But findings are focused on SE field. Under this perspective, some empirical and qualitative research methods used in the area of SE include: action research (AR) and case study (CS). Next, some general and concise information on these methods as well as the related works is provided.

2.1 Overview of action research and case study

According to Baskerville (1999), AR does not refer to a specific research method, but rather to a set of methods of the same type which shares the following properties: (1) focus on action and change; (2) focus on a problem; (3) an “organic” process model which involves systematic and interactive phases; and (4) participants’ collaboration. In this respect, more than one definition of this research method may be found in the literature, such as those presented in McTaggart (1991), French and Bell (1999), Wadsworth (1998). However, based on studies presented in Gustavsen (1993), Levin (1993), Kock et al. (1997), Davison (1998), Coghlan (2001). Chiasson et al. (2009) define AR as “a form of applied research that develops a solution to a practical problem, which is of value to the people with whom the researchers are working, while at the same time developing theoretical knowledge of value to a research community.” Furthermore, according to Baskerville and Wood-Harper (1998), there are different AR forms: canonical AR, IS prototyping, soft systems methodology, action science, participant observation, action learning, multiview, ethics, clinical field work, and process consultation. We might also refer to McKay and Marshall (2001) and Dittrich et al. (2008), who propose other forms of AR, which are, respectively, dual imperatives of AR and cooperative method development. Regarding AR process, several proposals exist (Davison et al. 2004; Checkland 1991; McKay and Marshall 2001), which propose different activities. From a general perspective, AR is seen as an iterative process which (1) involves researchers and practitioners acting together on a particular cycle of problem diagnosis, action intervention, and reflective learning (Avison et al. 1999) and (2) combines research cycles, focused on scientific goals and problem-solving cycles focused on a problematic situation (McKay and Marshall 2001).

With regard to CS, as stated in Runeson and Höst (2009), the definitions presented in Robson (2002), Yin (2003), Benbasat et al. (1987) coincide in their conclusion that this research method sets out to “investigate contemporary phenomena in their context.” In Braa and Vidgen (1999), two types of case studies are distinguished: hard CS, such as the method presented by Yin (2003), and soft CS, which is described in Walsham (1993). Hard CS focuses on a positivist research philosophy, “in which an objective reality is described in detail, variables identified and analyzed, and as a result an objective reality is

uncovered” (Howard et al. 2004). Hard CS explores a phenomenon within its real context, especially when the boundaries between phenomenon and context are not clearly evident. It can be used in three modes: explanatory, descriptive, and exploratory (Yin 2003). CS process involves five main activities: CS design, preparation for collection of data, collection of evidence, analysis of evidence, and reporting of the CS. By contrast, soft CS has an interpretivist approach and might make use of ethnographic methods (Howard et al. 2004). It “depends, not on the representativeness of such cases in a statistical sense, but on the plausibility and cogency of the logical reasoning used in describing the results from the cases” (Walsham 1993). CS does not use experimental control, and it employs “multiple methods of data collection to gather information from one or a few entities (people, groups, or organizations)” (Benbasat et al. 1987).

Based on the studies presented in Davison (1998) and Runeson and Höst (2009), mentioned above, in Table 1 we have synthesized and extended a set of characteristics of AR and CS research methods.

2.2 Applications of AR and CS in software engineering

The application of qualitative research methods to test hypotheses and to validate theoretical proposals has increased in SE lately (Montesi and Lago 2008; Zekowitz 2009). A complete report of the usage of AR in SE can be observed in Medeiros and Travassos (2011), while it is worth considering the work by Runeson and Höst (2009) on conducting and reporting CS in SE. It must be said, however, that from the use of these methods in SE, some difficulties in implementing them have been observed, including:

Table 1 Characteristics of AR and CS

Item under analysis	Action research	Case study
Research philosophy	Usually interpretivist and critical theory	May be positivist or interpretivist
Primary objective	Improvement	Exploratory, confirmatory
Suitable for research	In real-world setting	In real-world setting
Degree of realism	High	High
Level of control	Low	Low–medium
Primary data	Primary qualitative	Primary qualitative
Research design	Flexible	Flexible
Role of the researcher	Active participant	Observer
Focus	How? Why? and How to?	How? and Why?
Research process	Iterative	Sequential
Process' general activities	Diagnosis, action, reflection	Design, preparation for data collection, collection evidence, analysis of evidence, reporting

- The AR method in SE is related to two different participants: the research customer and the SE researcher. Often the needs of these participants are quite different, in fact even completely opposite to each other. That being so, the major challenge of those researchers who use action research in SE is to try to satisfy both demands.
- According to Brereton et al. (2008), the CS method is very useful for SE, but it is quite difficult to apply rigorously. Besides, the work on the success of empirical studies in SE discussed in Zannier et al. (2006) points out that the term “case study” is often used incorrectly and that many case studies lack hypothesis and/or a real CS.

According to Wohlin (2005), there is a need for a greater number of scientific approaches that include measurement and empiricism in SE, but there are, surprisingly, very few studies in this field, which use research methods such as AR and CS (Glass et al. 2002). The empirical method that is most widely used nowadays in SE is that of the CS (Zelkowitz 2009); nevertheless, the great majority of its applications suffer from a lack of a systematic and rigorous structure in their execution (Zannier et al. 2006). Meanwhile, AR has been used for SE since 1985, when it was introduced in this field by Wood-Harper (1985). Its use has increased, but it is not always followed with rigor when defining, applying, and reporting SE research (Medeiros and Travassos 2011). As (Mathiassen 2002) says, one notable weakness: (1) of practice studies (such as CS) is that they separate research from practice and (2) of AR “is the limited support provided to structure the research process or findings.” As we see it, this brings about the problem a good part of the results obtained is not supported by information gathered systematically by the coherent application of the empirical research method, bringing about a consequent decrease in the scientific rigor of the SE research.

2.3 Combining AR and CS: related works

The issue of mixing AR and CS has been tackled by some prior studies (Braa and Vidgen 1999; Mathiassen 2002). Braa and Vidgen propose a research framework for IS contexts, and they identified a hybrid research method (so-called action case) which is a combination of AR and soft CS. Mathiassen proposes a particular form of AR, known as collaborative practice research (CPR). A characteristic of CPR is that its process is guided by a pluralist approach (focusing on using different research models), in which AR is used from the outset and it employs other research methods as supplementary approaches (dominant approach of AR). For their part Chiasson et al. (2009) discusses at length the role of pluralist approaches in AR by carrying out a systematic review on studies related to AR within the IS discipline. This review presents how these studies blend: (1) AR with other research methods and (2) research cycles with problem-solving cycles. Other work that addresses a dominant approach to AR is presented in Mathiassen et al. (2002), which studies SPI in four software organizations over a period of 3 years by using dominant AR sequentially with methods such as CS, field experiments, and a literature survey. In the context of this last study, some particular issues of SPI have been discussed, for example: Iversen and Mathiassen (2003) present a CS of an organization that implements a software metrics program to measure the effects of its improvement efforts; Iversen et al. (2004) report a proposal to manage the risk of SPI projects following the CRP approach; and Aaen et al. (2001) offer a conceptual framework obtained from an literature review, which describes the features, risks, and benefits of SPI initiatives, as well as complementary approaches for a professionalization of the industry. Analyzing the studies described above, we have found that although they address the issue of blending AR with other

research methods, none of them describes how to carry out integration at the level of their research processes. Neither do these studies give any explicit description of a research process that integrates cycles and activities of AR with activities of hard CS.

On the other hand, some studies such as Wouters (2009, Kauppinen et al. (2004), Mejía et al. (2007) tackle research topics in the process improvement area of SE, and they mention the use of AR and CS. In the main, these works have conducted their research using AR, with which practical cases in real contexts have been carried out. These studies do not present a pluralist approach, however, since the cases were not guided by the CS method. Furthermore, in the relevant literature there are studies presenting results on the application of process improvement proposals within SMEs, and from some of these, we have analyzed the research method used in the application of the proposal presented:

- In Casey and Richardson (2004), Richardson (2001), a use of the AR method is discussed, but these studies do not deal with CS.
- In Pino et al. (2008), forty studies are presented, of which a high percentage used the term “case study.” The vast majority of these did not describe a systematic and rigorous structure for their application. It is rather the case that this group is made up of observational studies or experience reports, which focus on describing lessons learned, industry experiences, and connection between application and practice. Nevertheless, the low level of control over the process followed in applying the theoretical proposal is apparent in these studies, and none of them addresses the use of AR with CS.

As a result of this literature review, it is concluded that: (1) the great majority of applications of AR and CS in SE are not always followed rigorously and systematically, and (2) there are few studies on how different research methods have been adopted and blended in the dominant approach of AR (Chiasson et al. 2009). In this respect, the main contribution of this paper is thus to establish a *research strategy* that includes a research process that will guide a dominant approach to AR, while integrating CS, such as described in the following sections.

3 COMPETISOFT’s research strategy

With the requirements and characteristics of the COMPETISOFT project in mind, the first step in developing the COMPETISOFT project was to search for the most suitable research method to define the *Methodological framework* and put it into practice in SMEs for SPI purpose. We initially considered only the AR method as a very good candidate for the definition, refinement, and application of the COMPETISOFT *Methodological framework* components, since:

- According to Baskerville (1999), (1) development methodologies are an important area in the ideal domain of AR; (2) it is a valid research method for studying the effects of specific alterations in development methodologies in human organizations; and (3) AR can be used to generate new theory and to reinforce existing theory.
- This method is a dynamic approach to IS research driven by contributions to both theory and practice (Chiasson et al. 2009).
- A fundamental premise for the AR method is that complex social processes (the use of software process technologies in SMEs is an example of such a process) can be studied better by introducing changes in these processes, observing the effects of those changes (Baskerville 1999).

- Furthermore, there are several papers from the literature that deal with the suitability of AR for IS and SE, namely French and Bell (1999), Avison et al. (1999), Seaman (1999), Polo et al. (2002), among others.

The first research cycles of AR would be used to define an initial version of the *Methodological framework* (generation of our proposal), which should be applied in the organizations participating in the project during the following problem-solving cycles. However, due to the fact that these organizations were geographically dispersed, we needed a guideline for the systematic application of this proposal and considered that the use of the CS method within the problem-solving cycle was an appropriate strategy for reaching a systematic implementation of the *Methodological framework*. We came to that conclusion because of the viability of this method when individuals, groups, organizations, and social events are investigated (Yin 2003); it is often appropriate for research in SE (Höst and Runeson 2007).

As a result of this project context analysis, the development and application of the COMPETISOFT *Methodological framework* components were carried out by means of our own *research strategy*, which was defined taking into account the integrated use of: AR and CS. Figure 1 shows a high-level general view of the strategy followed for the integrated application of these two research methods in the COMPETISOFT project. In this figure, the left part shows the general structure of the proposed research process and the right part presents the main outcomes, along with the knowledge produced and exchanged from both the research cycle and the problem-solving cycle.

To carry out the *research cycles*, we followed the task proposals by (McKay and Marshall 2001), which are: (1) identify the research theme, (2) carry out a reconnaissance of relevant literature, (3) plan and design the research project, (4) take action steps, (5) implement, (6) monitor research, (7) evaluate in terms of research questions, and (8) amend plan and design. We have grouped these tasks together within the three generic activities of AR shown in Fig. 1. The *diagnosing* activity groups tasks 1, 2, and 3 (from McKay and Marshall), since, according to (Baskerville 1997), in this activity there should be a distinction made between diagnosis (to identify initial problems) and the planning in itself (to specify actions to solve these problems). Moreover, the *action* activity groups together

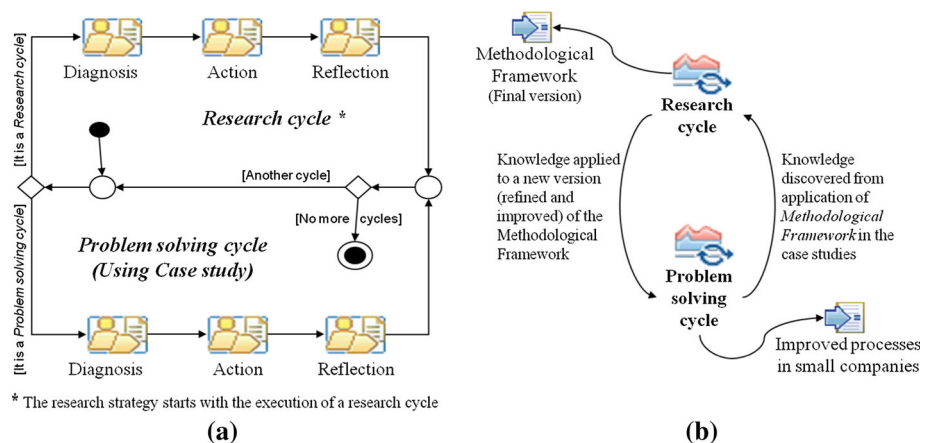


Fig. 1 Strategy for the integrated use of AR and CS in COMPETISOFT

tasks 4 and 5, and the *reflection* activity groups tasks 6, 7 and 8. As far as research roles participating in the project are concerned, the *researchers group* was made up of academics from different universities and *IT professionals* were information technology practitioners from the SMEs and the standardization and certification national body. Figure 2 shows an activity diagram of the *research cycle* in terms of tasks, roles, and the main research products.

On the other hand, as seen from Fig. 1, a *problem-solving cycle* entails the three generic activities of AR (diagnosis, action, and reflection). However, activities such as those described in Runeson and Höst (2009) and (Brereton et al. 2008) should be taken into account when conducting a CS. We present Table 2 to show how the activities of the *problem-solving cycle* can be met from those described for CS. Furthermore, this table shows how the activities of CS are integrated into the three generic activities of AR.

By analyzing the description of the activities of AR and CS, along with the relationship between them, the *researchers group* responsible for the application of the *Methodological framework* established the way to guide the *problem-solving cycles*. In this sense, to apply the COMPETISOFT’s *Methodological framework* in the real context of eight SMEs, the protocol template for planning CS presented in Brereton et al. (2008) and the guidelines for conducting CS presented in Runeson and Höst (2009) were followed. We also included the intervention activity to describe the action steps carried out within these companies during the execution of the SPI project. It is very important to highlight that *researchers* were actively involved (as coaches) in the actions leading to organizational change during the intervention activity. Based on this discussion, an activity diagram of the *problem-solving cycle* (which integrates the CS activities) in terms of tasks, roles, and the main outcome expected, is described in Fig. 3.

4 Application of the research strategy

Figure 4 shows an overview of the application of the *research strategy* through the timeline of the execution of the different cycles in COMPETISOFT. For the definitive version of the *Methodological framework* of COMPETISOFT, we carried out *research cycles* (RC labeled from 1 to n) and *problem-solving cycles* (PsC labeled from E1 to E8).

The COMPETISOFT project started with the *initial research cycle* (RC_1), during which project managers analyzed the research problem and identified the relevant literature

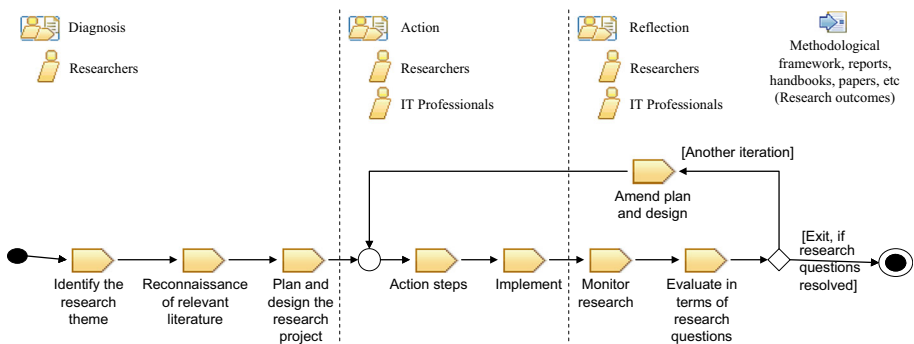


Fig. 2 Activity diagram of the research cycle

Table 2 Relationship between activities of AR and CS

Action research	Case study	
Activities of problem-solving cycle	Activities from Brereton et al. (2008)	Activities from Runeson and Höst (2009)
Diagnosis	Background Design Case selection Case study procedures and roles Data collection	Case study design Preparation for data collection
Action		Collection evidence
Reflection	Analysis Plan validity Study limitations Reporting	Analysis of collected data Reporting

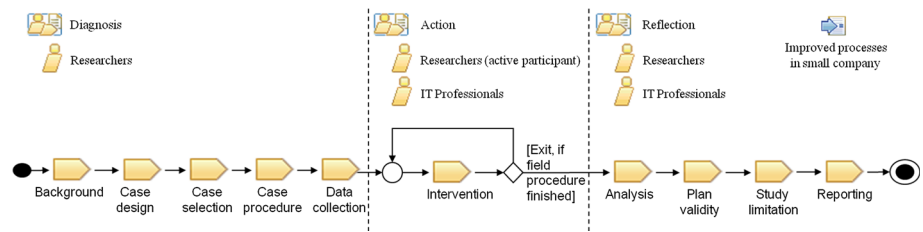


Fig. 3 Activity diagram of the problem-solving cycle

of the research theme. The planning and design for the execution of this project (aiming to develop the components of the *Methodological framework* of COMPETISOFT) were also established. Regarding planning, the months registered in the timeline of Fig. 4 are those in which a general meeting with the representatives of the organizations (from university research groups, SMEs, and the national standardization body) involved in the project was performed. In this respect, five project coordination meetings were held. These meetings lasted between 3 and 5 days, and they were held in: (1) March 2006 in Ciudad Real, Spain; (2) December 2006 in Buenos Aires, Argentina; (3) June 2007 in Ciudad Real, Spain; (4) September 2007 in Popayán, Colombia; and (5) May 2008 in Lima, Peru. These meetings offered a space for reflection on the execution of the project for the work groups established. In the first coordination meeting (March 2006), the planned tasks and responsibilities were distributed to the different work groups of the project. Each work group initially carried out its *research cycle* (RC_2 to RC_i), constructing the first version of different components assigned (models, processes, techniques, tools, templates) that make up COMPETISOFT’s *Methodological framework*. Each component developed by the different work groups was sent to the project administrator, who was responsible for integrating them to create a new version of the *Methodological framework*. The first stable version was released in December of 2006 during the Argentina coordination meeting. Then the released version of the *Methodological framework* was applied in the SMEs by

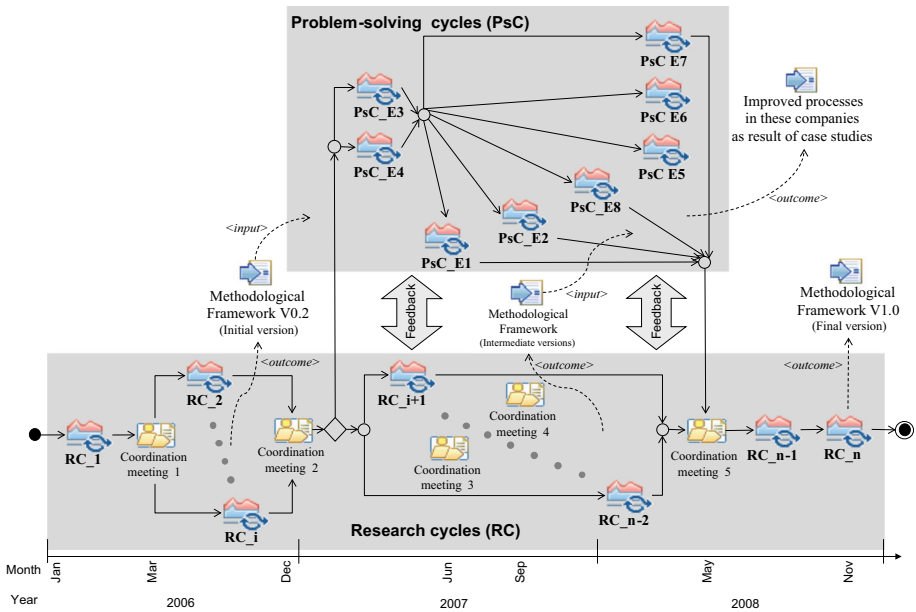


Fig. 4 View of the research and problem-solving cycles carried out in COMPETISOFT

means of the CS method during *problem-solving cycles* (PsC_{E1} to PsC_{E8}). The information and knowledge acquired from each *problem-solving cycle* were registered in the respective CS reports. Each report was also sent to the project administrator, who was in charge of its distribution among the work groups responsible for developing the components. This knowledge was used by the following *research cycles* (RC_i + 1 to RC_n - 1) to refine and improve the components of the framework, thereby creating a new version of this. In every case, the latest version available was used as input for the execution of the following *problem-solving cycle*. That is to say, during the period between December 2006 and May 2008 a continuous feedback between *problem-solving cycles* and *research cycles* (and vice versa) took place. The coordination meetings (of June 2007, September 2007, and May 2008) allowed a general reflection on the part of everyone involved in the project about the available version of the *Methodological framework* and its application in the SMEs. During the *final research cycle* RC_n, the project managers created and released version 1.0 of the *Methodological framework* of COMPETISOFT. Since we used AR, the members of the work group *IT professionals* served as researchers during the execution of a *research cycle*. The *researchers*, in turn, played a nonintrusive active role in practice (as coaches and advisers of the framework to *IT professionals*) during the execution of a *problem-solving cycle*.

4.1 Research cycles

By means of the defined *research cycles*, each work group constructed the component of the *Methodological framework* that had been assigned. The activity diagram presented in Fig. 2 was distributed among work groups, for them to have a guideline when carrying out their *research cycle*. It is important to highlight that in several work groups, *IT*

professionals of SMEs were involved during a research cycle. Making use of their experience, they contributed in the construction of components and in the initial validation on whether these were suitable for companies or not.

In the following paragraphs, a general description of a *research cycle* is presented. This description focuses on the *initial research cycle* (RC_1 from Fig. 4), since our *research strategy* is dominant and iterative, and this started with a *research cycle*. The cycle was used to identify and organize the whole work involved in developing the COMPETISOFT project by project managers. In this cycle, we analyzed how to develop and apply the *Methodological framework* through both the *research cycles* and the *problem-solving cycles*. It is important to highlight that the other *research cycles* (RC_2 to RC_n – 1) were carried out by the different work groups of the project, but due to space restrictions these *research cycles* are not described in this section.

4.1.1 Identify research theme and relevant literature

To identify these and other elements, we followed the AR approach proposed in Checkland (1991) and expanded in McKay and Marshall (2001). This establishes that any piece of research will consist of: the area of interest under investigation (A), a conceptual framing of the investigation (F), the research methods (M_R and M_{PS} for each of AR cycles), and a real-world problem (P). In the COMPETISOFT project, these were:

- Research theme (area) (A): SPI and, in particular, how the SMEs can carry out SPI appropriately, since models from SEI or ISO are difficult to be applied by these companies (Saedian and Carr 1997; Johnson and Brodman 1999; Hareton and Terence 2001; Staples et al. 2007; Laporte et al. 2008a).
- Research Framework (F): Reconnaissance of relevant literature about proposals that address SPI in SMEs, including the proposals developed in the context of Latin

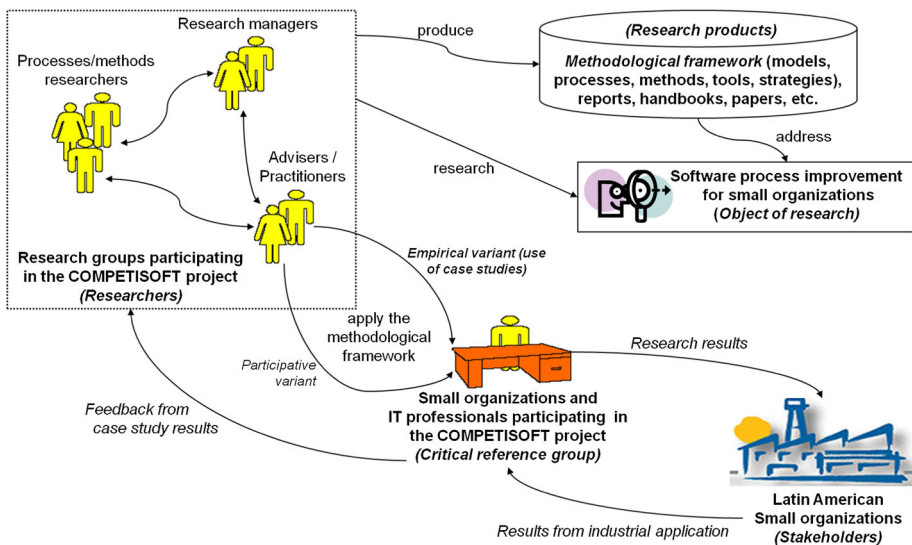


Fig. 5 Application of AR and CS in COMPETISOFT

America, was done by carrying out a systematic literature review, which is presented in Pino et al. (2008).

- Research method for the research cycle (M_R): The AR method guided the development of the *Methodological framework* of COMPETISOFT.
- Research method for the problem-solving cycle (M_{PS}): The CS methods guided the application of the *Methodological framework* of COMPETISOFT.
- Problem in the real world (P): The research addressed problems in applying SPI in SMEs.

4.1.2 Plan and design the research project

During the *initial research cycle*, the activities leading to the development of the *Methodological framework* were planned and designed. The elements to consider in AR, presented in Wadsworth (1998), were born in mind in making a general scheme of the research project and in planning the research. These elements are (1) the researched object, (2) researchers, (3) critical reference group, and (4) stakeholders of the research. The identification of these elements in the COMPETISOFT project is shown in Fig. 5.

In COMPETISOFT, the *researched object* or problem to be tackled is SPI in the context of SMEs. *Researchers* constructed and delivered the different components (*research products*) of the *Methodological framework* of COMPETISOFT to address the researched problem. The *researchers group* was divided into: (1) *research managers*, who were responsible for the management of the COMPETISOFT project, (2) *processes/method researchers*, in charge of developing the components of COMPETISOFT's *Methodological framework*, and (3) *advisers*, field researchers who carried out the application of the *Methodological framework* in the organizations of the *critical reference group* (which involves the SMEs and *IT professionals* taking part in the project). The research proposals developed were applied in this group, aiming to have initial feedback to refine, improve, and validate these proposals (this issue is discussed in Sect. 4.2). The *stakeholders of the research* are the small software development organizations of Latin America. Finally, under this point, it is important to highlight that the authors of this paper played different roles: All authors were *process/method researchers*, but apart from these roles, the first author was *adviser* and the last two authors were *research managers*.

4.1.3 Action and reflection

The *research managers* formed a work group which was in charge of the execution of the *initial research cycle*; this group was made up of the *research managers* themselves, along with *process/method researchers*, and it conducted a systematic literature review on the SPI for SMEs (*object of research*). Identifying and analyzing the existing approaches on this issue allowed us to discover that the *object of research* has a high level of complexity and that our proposal would have to take into account different components (models, processes, techniques, tools, templates, among others). Based on this knowledge, the *researchers* designed an initial architecture of the *Methodological framework* of COMPETISOFT, which defined three components: process reference model, improvement model, and process evaluation model. This group also analyzed the proposals which were related to the *object of research* and which had previously been developed in the Latin American context, and some of them were chosen to become the basis on which these models were constructed. Some researchers and *IT professionals* of COMPETISOFT had already taken

part in the development and application of those proposals, which enabled us to avail ourselves of their experience in putting the project into execution.

The *research managers* then established a plan for the execution of the project, so that the components of the *Methodological framework* could be developed. They worked out what tasks and resources would be needed to develop these components. They also planned out the distribution of responsibilities between the work groups that were to be created for the project to be implemented properly. In the first coordination meeting, the architecture, the proposals to be taken into account, and the plan were presented to all the *researchers* and *IT professionals* of the research groups, SMEs and the national standardization body. In the quest to refine the architecture and the plan, a discussion on the issues mentioned took place, the goal being to obtain feedback on them from these key people. The other work groups responsible for the construction of the different components of the framework were set up, and the appropriate component, according to the particular experience of each work group, was assigned to each respective team. These groups were composed of *researchers* and *IT professionals* from the different organizations involved in the project.

From the viewpoint of the *initial research cycle*, all other *research cycles* (RC_2 to RC_n) can be seen as the action activity of this cycle, given that the tasks needed to

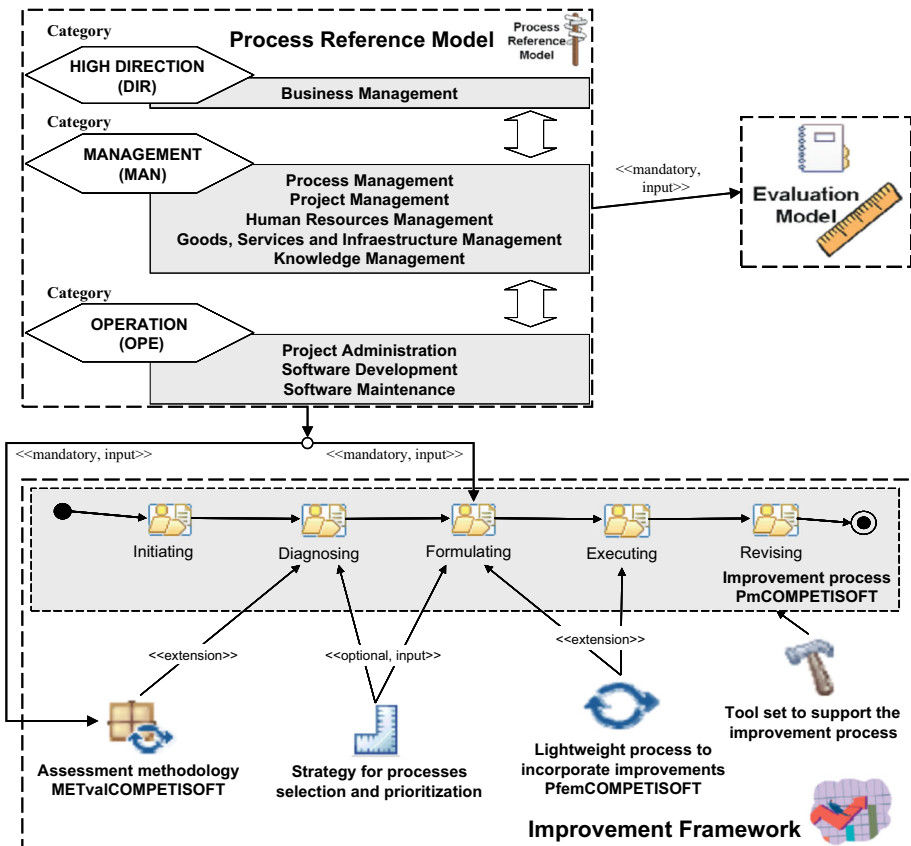


Fig. 6 General diagram of the methodological framework of COMPETISOFT

develop the respective component were performed within them. Nevertheless, each work group planned and carried out all the activities (diagnosis, action, and reflection) of its own *research cycle* to achieve the tasks assigned and to construct the component it was specifically responsible for. The different iterations that involve the action and reflection activities of these *research cycles* pursued two well-defined purposes: (1) The initial goal was to develop a first, stable version of each component that could be applied to the companies of the *critical reference group* (RC_2 to RC_i from Fig. 4), and (2) the later goal was to refine and improve each component, thus creating a final version of them (RC_i + 1 to RC_n), based on the knowledge acquired from having applied them.

We should add here that coordination meetings can be seen as the reflection activity of the *initial research cycle*; it was in these meetings that the research project was evaluated, and it was here that the plan and the architecture were adjusted. In this respect, the monitoring of the work done by each work group was performed by the *research managers*, who did so by means of a check on the deliverables established. In addition, there was continuous reflection throughout the whole course of the project as the research results were shared and analyzed with those involved in the work groups. This activity allowed participants to ensure greater depth of insight into the functioning of the models, processes, practices, techniques, tools, and templates proposed for SPI on SMEs. It is a fact that the feedback obtained from people involved in the work groups of the *research cycles* and the *problem-solving cycle* made it possible to refine, improve, stabilize, and release the different components of the *Methodological framework*.

4.1.4 Methodological framework of COMPETISOFT

The main outcome of the execution of the research cycles is the *Methodological framework* which is composed of (1) a *process reference model*, (2) an *improvement framework*, and (3) a *process evaluation model*. A broad view of the components of this *Methodological framework* is shown in Fig. 6, and the complete description of this framework on the Web is presented in CYTED (2015). In the following lines, we present a brief description of three models in which the different components of the *Methodological framework* are grouped.

The *process reference model* (based on the MoProSoft model (Okta 2006)) groups the processes into three categories (see Fig. 6). The purpose of these categories is to provide specialized processes for each functional group of the software development enterprise. These categories are:

- The high direction category (DIR) is concerned with practices related to business management. It provides the directions for the processes of the management category and receives reports from these. This category includes the Business Management process.
- The management category (MAN) deals with process, project, and resource management practices, which are in line with the business goals of the Top management category. It provides the elements for the performance of the Operations category processes, receives and evaluates the information generated by those processes, and informs the Top management category of the results. Process Management, Project Management, Human Resources Management, Goods, Services and Infrastructure Management, and Knowledge Management are the five processes that comprise the MAN category.

- The Operation category (OPE) addresses the practices of software development and maintenance projects. This category performs the activities using the elements provided by the management category and delivers reports, as well as the software products generated. It includes the project administration process, the software development process, and the Software Maintenance process.

In the reference model, the activities related to process improvement are described in a general way in the process of *Process management*. Nevertheless, to guide explicitly and in detail the implementation of these practices in the context of small organizations, the *improvement framework* (Pino et al. 2009) was developed. Its aim is to provide improvement infrastructure, with the techniques and tools to support improvement initiatives in small companies. This framework defines five components (see Fig. 6):

- A process called PmCOMPETISOFT, whose focus is on small enterprises, providing them with a guide with which to manage and lead the software improvement process step by step. This process includes five activities: initiating the cycle of improvement, diagnosing the process, formulating improvements, executing improvements, and revising the cycle of improvement.
- A lightweight process to incorporate improvements, which uses the SCRUM agile method to support the managing and carrying out of the activities of the formulation and execution of improvement. Our objective is to offer all those who are involved in the improvement cycle of small organizations a lightweight sub-process which will allow them to put into operation any improvement opportunities encountered.
- A strategy for process selection and prioritization, which presents the selection of a set of processes that are considered critical to the implementation of a process improvement project in small companies. Our objective is to present this type of companies with a strategy for dealing with the first processes that must be considered when they undertake an improvement project.
- A methodology for software process assessment, called METvalCOMPETISOFT, which supports the activity of diagnosing the software processes in small organizations. This methodology describes: (1) a process for software process assessment, called PvalCOMPETISOFT, which offers a step-by-step guide to the activity of software process diagnosis; (2) a light assessment method to determine the capability of software processes of a small organization (conformance to ISO/IEC 15504 but considering only up to level two of the capability model); and (3) a tool to support the process assessment called EvalTOOL (Martínez-Ruiz et al. 2009).
- A tools set to support the improvement process, such as: (1) GENESIS (Hernández et al. 2008), which is used to support the management and implementation of the activities of PmCOMPETISOFT and (2) HEPALE! (Cruz Mendoza et al. 2009), which is an educational tool for supporting acquaintance with and understanding of the Methodological framework.

Regarding *Process Evaluation Model*, we suggest that each country should define its own evaluation model which must be in accordance with International Standard ISO/IEC 15504-2 (see Fig. 6) in order to allow mutual recognition of formal evaluations of COMPETISOFT throughout Latin American countries.

4.2 Problem-solving cycle

Using the *problem-solving cycles*, several work groups applied the *Methodological framework's* components in the companies of the *critical reference group*. As far as the application of the proposal sketched in Fig. 5 is concerned, there are different ways to apply research products in the *critical reference group*. These forms depend mainly on the characteristics of the research project and can be diagnostic, participative, empirical, and experimental (French and Bell 1999). In the *research strategy* of COMPETISOFT, we have used the participative and empirical variants during the *problem-solving cycle*:

- It was participative, because the *critical reference group* put into practice the *Methodological framework* (*research product* developed by the *processes/methods researchers*) with support from the model's *advisers*; this group shared its experiences, effects, and results of the application with the *advisers*. A couple of studies in the COMPETISOFT context where this variant has been used can be seen in Luzuriaga et al. (2008) and Pino et al. (2012).
- It was empirical because, with the involvement of the *adviser* and by means of the integration of the CS method within the *problem-solving cycle*, the *critical reference group* has conducted a comprehensive and systematic record of its actions using the components of the *Methodological framework*. As regards researchers' involvement, the COMPETISOFT *advisers* were directly involved during the intervention with the *Methodological framework* in the SMEs of the *critical reference group* during the execution of *problem-solving cycles*.

In short, in the quest to implement the empirical variant of the AR method, we have integrated the CS method in the *problem-solving cycle*. In this section, we discuss our work on the implementation of this type of AR's variant by following the activities shown in Fig. 3. Figure 7 presents an overview of the execution of *problem-solving cycles* in terms of the *case studies* carried out. In this figure, the upper part shows the CS activities and

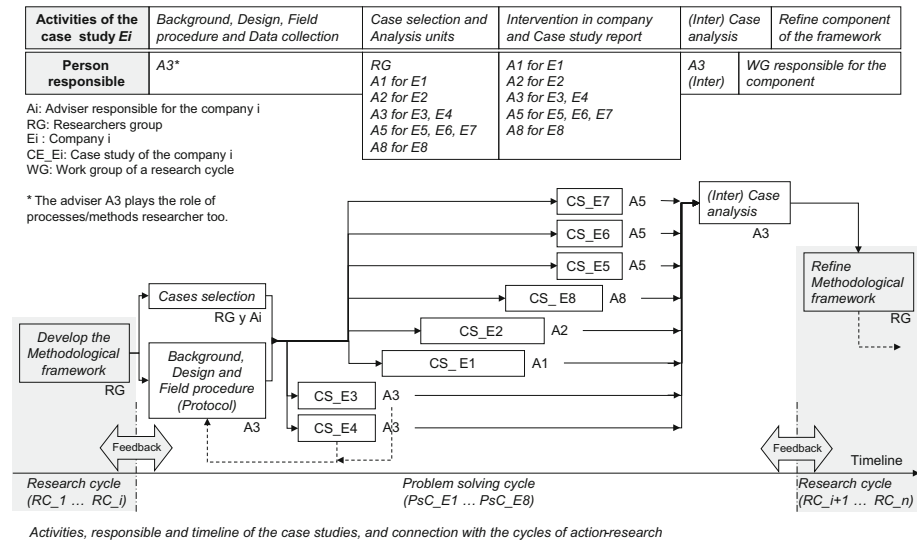


Fig. 7 Overview of case studies carried out in COMPETISOFT

person responsible of them, and the down part shows the timeline of research and solving cycles.

A *process/methods researcher*, who also plays the role of *adviser* (A3), was in charge of preparing the protocol of the case studies, by including the field procedure. This field procedure was distributed to the other *advisers* (A1, A2, A5, and A8) so they had the guideline with which to carry out the intervention in their companies. During the intervention in the company, the *adviser* shared the data collection templates of the different activities of the field procedure with the *researchers*. At the end of the intervention in the company, each *adviser* created the respective CS consolidated report. This evidence was used by the work groups of the *research cycles* to refine and improve the components of the *Methodological framework* of COMPETISOFT (by means of the RC_i + 1 to RC_n - 1 from Fig. 2). Based on these reports, the *process/methods researcher* and *adviser* (A3) carried out an analysis of all the case studies performed. A description of the *problem-solving cycles* in terms of diagnosis, action, and reflection is presented below.

4.2.1 Diagnosis

The background on previous research related to the application of SPI in SMEs was identified during the *research cycles* prior to the *problem-solving cycles*. Based on this knowledge, we have defined the research questions (main and additional) being addressed by the case studies (see Table 3). By asking these questions, we sought to know whether the components of the *Methodological framework* have a useful function, if they are of practical use and whether they conform to the reality of SMEs.

Taking into account the focus presented by Yin (2003), the design type of the CS in the COMPETISOFT project was multiple cases—embedded, since the proposed *Methodological framework* was applied in the context of eight SMEs to improve one or several of their processes (different analysis units). The *object of study* is the *Methodological framework* (specifically the research products *process reference model* and the *Improvement model*). The *measures* used to investigate the research question were: (1) the effort invested by SMEs in the *Improvement model*'s activities to improve selected processes (related to the practical use of the framework and the reality of companies), (2) the process capability level of the processes selected for improvement (related to useful functioning of the framework), and (3) we also took into account the benefits described by the companies involved in the case studies.

The criterion for the case study selection was SMEs who were part of the *critical reference group* of COMPETISOFT. The eight companies selected (from Argentina, Chile, Spain, and Colombia) had to be committed to the accomplishment of a SPI cycle of at least

Table 3 Research questions of the case studies

Research questions (main and additional)

MRQ	Is COMPETISOFT's Methodological framework suitable (useful and practical) for carrying out process improvement initiatives in small software companies?
ARQ1	Is the effort of applying process improvement by following COMPETISOFT's Methodological framework appropriate for small software companies?
ARQ2	Does the use of the COMPETISOFT's Methodological framework make it possible to increase the process capability of small software organizations?

three months. Table 4 describes the properties of the companies that took part in the case studies (hereafter called E1, E2, E3, E4, E5, E6, E7, and E8, to guarantee their anonymity). In each company, the analysis units were: (1) the processes to be improved (processes under intervention) which were selected from the *process reference model*, and (2) the activities and strategies to guide the SPI (described by the *Improvement model*).

Throughout the reflection activity on work performed (which was carried out during the coordination meeting held in Argentina at December 2006), the *researchers* reached a consensus on the guidelines that the companies should take into account when carrying out the first cycle of process improvement:

- Select any process to be improved from the *operation category* of the *process reference model*, with which *analysis units* could be the processes: Project Administration (PA), Software Development (DS), and Software Maintenance (MS).
- Use the PmCOMPETISOFT process (another analysis unit) and the remaining components of the *Improvement model* to guide the improvement implementation within the processes chosen by companies.

The procedure governing field procedure of the case studies (see Fig. 8) is closely related to the activities, sub-processes (which give more details to execute these activities), and roles described in the PmCOMPETISOFT process of the *Improvement model*. The data collection plan for the case studies was related directly to the activities flow of such process, and the data collected were directly related to the items defined in each of the work products described by PmCOMPETISOFT. The data collected were stored using the self-contained template of these work products.

4.2.2 Intervention in each organization (action)

Each one of the organizations started its improvement initiative with the support and involvement of an *adviser* in process improvement who came from COMPETISOFT's

Table 4 Features of the organizations involved in the case studies

Company	Country	Employees	Path	Main areas of professional activity
E1	Argentina	8 (7)	16 years/N&I	Development of new tailored information systems with ongoing integration of new technology
E2	Chile	18 (12)	10 years/N&I	Computer Engineering projects for the agricultural industry (wine and food)
E3	Spain	7 (6)	5 years/N	Development of WEB software applications.
E4	Spain	21 (15)	13 years/N	Software development through contracts and agreements with public organizations
E5	Colombia	4 (4)	3 years/N	Software to manage and control the ISO 9001-2000 quality management system
E6	Colombia	6 (6)	3 years/N	Web application development-oriented agricultural services
E7	Colombia	4 (4)	3 years/N	Software to mobile telephony and devices
E8	Argentina	12 (5)	4 years/N&I	Custom software development on the area of business management

Employees Number of employees in the enterprise (people in software development and maintenance)

Path Number of years of existence of the company/scope of the market for its products (National-N/International-I)

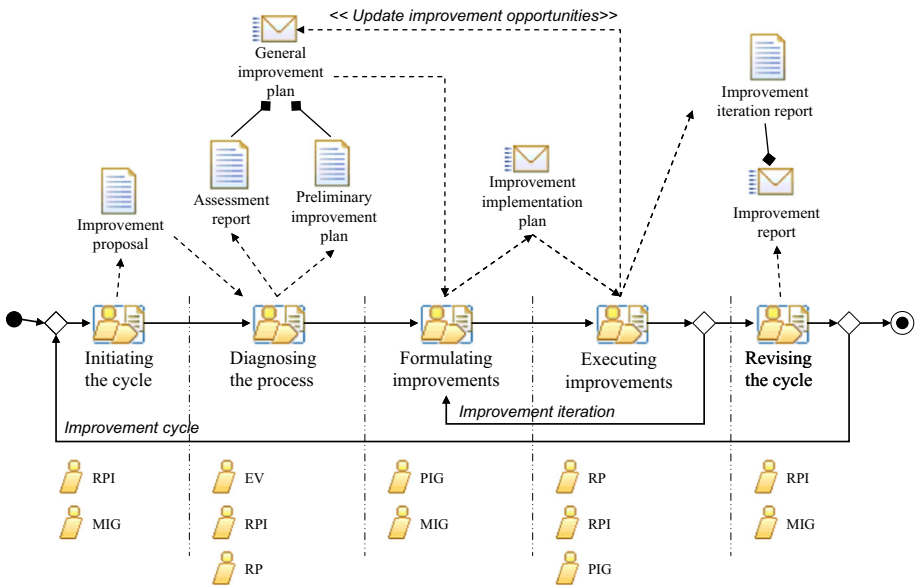


Fig. 8 Procedure governing field procedure of case studies

researchers group. The *adviser* and the *IT professionals* of each organization were responsible for seeing that the field procedure and data collection were carried out properly in the SMEs during the intervention with the COMPETISOFT methodological framework. Each adviser generates a detailed report of the work performed to improve the processes of its company. It is important to highlight that the *adviser* was an active participant in the organizational improvement, since (1) he/she trained and supported the *Management Improvement Group (MIG)* in the *Initiating the cycle* activity, (2) he/she was the *Evaluator (EV)* in the *Diagnosing the process* activity, and (3) he/she formed part of the *Process Improvement Group (PIG)* in the improvement iteration. The active participation of the adviser inside of these groups was to train, advice, or counsel to IT professionals on how they could implement process improvements in the context of the VSE (nonintrusive active participation). Considering that the advisor is a researcher and that he/she participated actively in carrying out the intervention in practice, this activity entails an approach which is more like action research than case study.

During the intervention, measures used to investigate the research question were collected. In this respect, at the beginning and at the end of the SPI project in each company, an internal assessment was performed, to determine the process capability level (see Table 5). Furthermore, the amount of effort used to carry out the organizational improvement following the *Methodological framework* of COMPETISOFT was also established (see Table 6). This information was obtained by synthesizing the data presented in each CS report prepared by advisers.

4.2.3 Reflection

CS reports recorded the knowledge discovered about the application of the *Methodological framework* in the SMEs. This knowledge was used by the *research cycles* $RC_i + 1$ to

Table 5 Initial and final capability levels of the processes under intervention

Organization	Assessment	Capability level of processes								
		OPE			DIR			MAN		
		SD	PA	SM	BM	PM	PjM	HRM	KM	IM
E1	Initial	–	2	–	–	–	–	–	–	–
	Final	1	2	*	1	1	1	1	1	1
E2	Initial	0	1	0	–	–	–	–	–	–
	Final	1	2	*	*	–	–	–	–	–
E3	Initial	0	0	–	–	–	–	–	–	–
	Final	1	*	–	–	–	–	–	–	–
E4	Initial	0	0	–	–	–	–	–	–	–
	Final	1	*	–	–	–	–	–	–	–
E5	Initial	1	0	–	–	–	–	–	–	–
	Final	1*	1	–	–	–	–	–	–	–
E6	Initial	1	1	–	–	–	–	–	–	–
	Final	1	1*	–	–	–	–	–	–	–
E7	Initial	0	0	–	–	–	–	–	–	–
	Final	1	1	–	–	–	–	–	–	–
E8	Initial	0	0	–	–	–	–	–	–	–
	Final	0*	1	–	–	–	–	–	–	–

Processes: *SD* software development, *PA* project administration, *SM* software maintenance, *BM* business management, *PM* process management, *PjM* project management, *HRM* human resources management, *IM* goods, services, and infrastructure management, and *KM* knowledge management

* Best practices of this process have been put into operation, – Process not assessed

Table 6 Effort involved in the improvement initiative of each company

	Organization								
	E1	E2	E3	E4	E5	E6	E7	E8	
Effort (h × 1 person)									
Adviser	40	89	15	41	42	38	65	71	
Company	264	255	39	47	27	11	23	16	
Total	304	344	54	88	69	49	88	87	
Cycle length (weeks)	24	20	12	12	10	10	10	16	
Effort per week (h/person)	12,7	17,2	4,5	7,3	6,9	4,9	8,8	5,4	

RC_n (see Fig. 4) to refine, improve, and create the definitive version of the components of the *Methodological framework* of COMPETISOFT. These components therefore respond to needs detected in early interactions with the companies of the *critical reference group*. For instance, the work group responsible for the construction of the *Improvement model* performed several *research cycles* to develop the components of this model. Initially, a *research cycle* RC_i was carried out to define PmCOMPETISOFT (which describes the activities needed to manage and lead the process improvement initiatives in SMEs).

However, from the early reflection activities (coordination meetings and feedback from case studies involved in PsC_E3 and PsC_E4), it was observed that:

- The level of detail to be found in these activities was not high enough for organizations that wanted either a greater breakdown of these activities or greater autonomy with respect to the adviser. To deal with that issue, the components MET-valCOMPETISOFT and PfemCOMPETISOFT were developed, through two *research cycles* ($RC_i + 1$ and $RC_i + 2$). These new components aim to offer more explicit guidelines for carrying out some PmCOMPETISOFT activities.
- Companies need to be provided with guidelines that address the issue of what the first processes that must be considered are as they undertake a process improvement project. In this respect, the strategy for selection and prioritization was produced during the course of another *research cycle* ($RC_i + 3$).
- To help the dissemination of the knowledge, as well as the management and implementation of the components of the *Improvement model*, it was very important to have software tools. With this need in mind, other *research cycles* (related to each tool) were carried out to create the tool set that would support the improvement process.

On analyzing the information obtained on measures and CS reports, we have noticed that:

- *Advisers* observed and registered that the group of employees involved in the improvement initiative of each organization did not spend much time on that organization's effort to carry out the improvement responsibilities they had been given (see Table 6). This ability on the part of companies to take on such responsibilities properly is evidence that it is feasible for SMEs to use the methodological framework (*ARQ1 research question*).
- Through the application of COMPETISOFT's *Methodological framework*, the SMEs introduced new best practices to their processes, allowing them to increase the capability of some of those processes. The *process reference model* gave companies the best practices recommended for incorporation in their processes, while the *Improvement model* offered a way to guide the implantation of these practices within processes under intervention, thus improving them. It is evident that this *framework* responds favorably to the *ARQ2 research question*.

Based on the experiences obtained from the case studies carried out within problem-solving cycles, the increase in the capability of the processes to be improved, the effort of applying the proposed framework, and the benefits described by SMEs, we consider that the *Methodological framework* of COMPETISOFT is suitable for leading process improvement initiatives in this type of organizations (responding to the *main research question*). The results, in terms of effort, increase in capability and benefits, are an indicator that the proposed framework can be a practical and useful strategy when facing the difficulty of carrying out SPI in SMEs. Furthermore, from the problem-solving cycles, we have been able to confirm that the proposed *Methodological framework* of COMPETISOFT was executed properly by the SMEs involved in the improvement initiatives.

To address threats to the validity of the case studies, different issues were considered, which are described below:

- The *design of case study* and the *data collection plan* were checked against the checklists for case studies on software engineering proposed in Runeson and Höst

(2009), and it was noted that they achieved a high percentage of the items proposed by this checklist.

- Regarding *construct validity*, we have used multiple sources of evidence for the data collection, including documentation archive, surveys, interviews, and participative observation. These sources of evidence have been obtained from different roles which have participated in the improvement cycle. Furthermore, the use of templates related to each activities of the field procedure have allowed us to maintain a chain of evidence which establishes a traceability between research questions, recorded data, evidences, and analysis.
- As regards *internal validity*, from the analysis presented in the previous sub-section, it can be determined that the decision to use the COMPETISOFT's *Methodological framework* to guide the process improvement in the small organizations has led them to enhance the capability of their processes under intervention.
- On the topic of the *external validity*, we initially applied the Methodological framework of COMPETISOFT on the E3 and E4 organizations (see Fig. 7). This application was carried out with the support of a single adviser (he/she is also part of the processes/methods researchers group responsible for the problem-solving cycle). The field procedure was always performed in the E4 organization first and then in the E3 organization. This first application allowed us to review, validate, and refine the protocol and the field procedure of the case study. Finally, the replication material of the case study was distributed to the various advisers, the purpose being to carry out the replication of the case study in the remaining six companies.
- Regarding *reliability*, the processes/methods researchers developed the replication material of the case study and this material was distributed to the advisers. It was observed that by following this material for conducting case studies in enterprises E1, E2, E5, E6, E7, and E8, similar findings and conclusions to those obtained in the first two case studies (E3 and E4 organizations) were found.

The case studies carried out to use the Methodological framework of COMPETISOFT in small companies presented in this paper have some limitations:

- The observations and conclusions presented are based on eight case studies, which can limit the power of generalization. Although these companies are representative of the software industry in Latin America, the number of companies taking part in the case studies is a low percentage of the overall population. In this respect, since this study has been performed in the Latin American context, this methodological framework may not apply to other settings (e.g., Europe, North American, Asian, etc.).
- The bias of the case studies, because the performance of daily activities by employees may proceed differently precisely because they are being observed, or due to some particular kind of handling of events and data by the advisers.

5 Research strategy and software engineering standards

With regard to the contribution of the research strategy to software engineering standards, it is worth to stress that this strategy has been:

- Related to the project that carried out the Working Group 24 of ISO/IEC JTC1/SC7 to develop the technical report ISO/IEC 29110 Software Engineering—Lifecycle Profiles for Very Small Entities (ISO 2011). Part five of this report describes the Basic Profile in

order to define a software implementation and project management guide for a subset of processes of ISO/IEC 12207. This profile is composed of Project Management and Software Implementation processes, which (1) are based on the MoProSoft model and (2) were improved and refined for the COMPETISOFT's process reference model by means of research cycles. Furthermore, the improvement projects carried out in SMEs to enhance Software Development (DS) and project Administration (PA) processes (described in the Sect. 4.2: "Problem-solving cycle") have also contributed as a pilot project in the introduction of the basic profile of ISO/IEC 29110 in small organizations (Laporte et al. 2008b). These trials have helped to validate the approach and obtain feedback in order to improve the documents before they go to ISO/IEC publications (Laporte et al. 2008b).

- Used for building a software engineering maturity model (Garzás et al. 2013), which allows to assess and to certify the Ibero-American software industry (thinking especially of those small firms with fewer than 50 employees) by means of organizational maturity levels (in the same way as the CMMI model does). To create this new scheme for the certification, different actors, such as the government, the industry, and the scientific-academic community of Spain, were involved. Following the strategy proposed, a first research cycle in which the research manager planned and designed the execution of the project was performed. The manager analyzed the goals and deliverables for the project, together with the experience, strengths, and interests from the researchers and practitioners involved in that project. Taking this analysis into account, the manager made a general plan and distributed the planned tasks to the different work groups of the project, aiming to fulfill the commitments acquired. Subsequent research cycles were carried out with the objective of developing the software engineering maturity model. This model was applied in 16 small organizations participating in the pilot project by means of problem-solving cycles. At this moment, there are already 50 enterprises which have level 2 or 3 certifications of this model.

6 Discussion

As Baskerville and Wood-Harper (1998) say, AR needs to be adapted if it is to be made more suitable for research needs. In this sense, our *research strategy* for applying AR is based on, and conforms to, several previous approaches using this method:

- Firstly, the general model of AR proposed by McKay and Marshall (2001) and the general outline of AR presented in Chiasson et al. (2009). This approach describes how AR 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.
- Secondly, the generic activities (problem diagnosis, action intervention, and reflective learning) of a cycle of AR indicated in Avison et al. (1999). We believe that it is possible to group the activity proposals according to other AR approaches, like those of Davison et al. (2004), Checkland (1991), McKay and Marshall (2001), within these three generic activities.
- Finally, the CS research method, described in Yin (2003), which we have used and integrated to support and guide the *problem-solving cycle*.

According to Chiasson et al. (2009), there are two approaches for blending AR with other research methods: *dominant* and *sequential*. This study also describes three ways in which research and problem activities are mixed: the *research dominant*, the *problem-solving dominant*, and the *iterative* approaches. As seen from Sect. 3, our *research strategy* follows the approaches of AR: *dominant* (to blend research methods) and *iterative* (to blend activities). It followed the *dominant* approach, since the researchers used a method of AR from the outset; they also used the CS method to examine and explain research questions. Furthermore, our strategy followed the *iterative* approach, seen in the fact that the *Methodological framework* of COMPETISOFT (theoretical knowledge) was produced by various *research cycles* (see Fig. 4). This framework was applied in several SMEs by carrying out *problem-solving cycles*, and from each of these, we discovered practical knowledge, which was registered in the respective CS reports. This knowledge influenced the activities of the next *research cycle* in the discovery of new theoretical knowledge, which was included in the *Methodological framework* in creating new, refined, and improved versions of this (see Figs. 1b, 4).

Table 7 illustrates an analysis of the *research strategy* proposed in this paper. This analysis takes into account the characteristics of AR forms (first and second columns), the criteria used to evaluate these characteristics, and the scheme for analysis that have been presented in Baskerville and Wood-Harper (1998). The marks in the third column point out

Table 7 Research strategy analysis taking into account the characteristics of AR forms

Characteristic of action research	Our research strategy
Process model	
Iterative	<ul style="list-style-type: none"> The process models are reflective because our strategy: (1) is iterative, involving cyclical activities for researching and solving the problem (see Fig. 1); and (2) initially it focused more on the analysis of the theory than on the problem diagnosis (see Fig. 4)
Reflective	
Linear	
Structure	
Rigorous	<ul style="list-style-type: none"> The structure is rigorous because our strategy defines a set of clearly identified activities that must be followed in the execution of both the research cycle (see Fig. 2) and the problem-solving cycle (see Fig. 2)
Fluid	
Typical involvement	
Collaborative	<ul style="list-style-type: none"> The typical involvement of the researchers with the research subjects in this study was: (1) facilitative, because the researchers participated as experts to help the companies during the intervention activity; and (2) collaborative, since the researchers (advisers) also worked as equals with the people in the company who were responsible for the intervention activity. See Sect. 4.2.2
Facilitative	
Experiment	
Primary goals	
Organizational development	<ul style="list-style-type: none"> The primary goals of this study were: (1) scientific knowledge, because initially the researchers sought to understand the research object, to contribute to the literature in this field (e.g., see study Pino et al. 2008), and then they built a new study (the Methodological framework) based on this knowledge, which can be useful and practical for the practitioners of the research stakeholders; and (2) system design, since organizational improvement was incorporated by means of the problem-solving cycle
System design	
Scientific knowledge	
Training	

Key: • signifies a dominant characteristic, × signifies characteristics that occur together in this study

where this characteristic is present in our *research strategy*. In the right-hand column, a brief explication of this presence is given.

We have contrasted the characteristics of AR that our *research strategy* has with the characteristics of the various forms of AR discussed in Baskerville and Wood-Harper (1998). From this analysis, we observed that the distribution of characteristics that are satisfied by our proposal (presented in Table 7) differs from the distribution of characteristics of each AR form that are described in Baskerville and Wood-Harper (1998). Based on this aspect, we consider that the *research strategy* presented in this paper can be a new form of applying AR. In this respect, this form of AR can be used to guide research in SE and it needs or involves: (1) a reflective process model, (2) a rigorous structure, (3) a researcher involvement as facilitator and/or collaborator, and (4) a primary goal that is system design and/or scientific knowledge.

On the other hand, as regards classification of studies in SE presented in Travassos et al. (2008), the work presented in this paper is a primary study: It is observational and in vivo, since it involved research methods such as CS and AR. Furthermore, this study involved getting information on the research issue from the participants and projects in their own environments.

Some lessons from application of the proposed *research strategy* are described and are as follows:

- The application of the infrastructure suggested by the action research allowed the *research managers* to have a centralized administration of COMPETISOFT for the construction and application of its main research product: the *Methodological framework*. The applications of the *Methodological framework* in the SMEs of the *critical reference group*, which are geographically dispersed, have been carried out according to the general guidelines established by the *researchers group* responsible for the *problem-solving cycle*. This permitted us to have a homogeneous application and to obtain reliable information about the application of the proposed framework.
- The AR method is strengthened by the CS method, because this allows more control in carrying out the proposals developed (see *problem-solving cycle* in Sect. 4.2). That is, the *researchers* conducted the intervention with the new proposal in the *critical reference group* by following the activities of CS method systematically. This has meant an increase in the reliability of the results of the application of the *Methodological framework* of COMPETISOFT. By means of the integration of these two methods, a well-defined structure has been obtained, both for the development of the components of the *Methodological framework* of COMPETISOFT and the application of this new proposal in certain SMEs.
- The use of the protocol for CS presented in Brereton et al. (2008) and the checklist presented in Höst and Runeson (2007) has allowed case studies (*problem-solving cycles*) to be carried out systematically and rigorously. Using a protocol enables *researchers* to: (1) specify in detail how they try to answer the research questions posed and (2) support the results obtained. The checklist for CS presented in Höst and Runeson (2007) is a suitable guide for determining whether all the elements that must be taken into account when performing a CS have been considered. At the end of each of the main activities involved in the case studies within problem-solving cycles (design, preparation for data collection, collecting evidence, analysis of collected data and reporting), each group of questions on the checklist must be checked, which lets us validate and improve each activity planned in the protocol, thereby guaranteeing the rigor of the CS.

7 Conclusions

A *research strategy* that integrates action research and case study research methods has been described in detail in this paper. This strategy starts with a *research cycle* and proposes that activities from hard CS are integrated into the *problem-solving cycle*, thereby increasing the control and rigor when it is put into operation. We propose that the researcher should play a role more of “active participant” than of observer during the intervention activity of this cycle. Our proposed strategy establishes and supports an integrated use of AR and hard CS, beyond a sequential use of these (as has been proposed by most authors).

We must add that the *research strategy* enabled us to have a research infrastructure which was suitable for the needs of COMPETISOFT. In the context of this project, the *research strategy* addressed: (1) a dominant approach of AR for the development and application of the COMPETISOFT *Methodological framework*, and (2) the integration of AR and hard CS, to allow for increased rigor in the application of this framework within some SMEs. To be precise, by means of the *research cycles* the components of the *Methodological framework* were developed, and through the *problem-solving cycles*, the *Methodological framework* was applied. It is important to highlight that the *problem-solving cycle* was implemented by means of case studies, with the nonintrusive active participation of the researcher. That is to say, the intervention in the companies, using the proposed framework, was carried out by *researchers* who followed the CS method to record the information, as well to register data and document experiences. Based on these data and experiences, we may conclude that evidence points to the suitability of the *Methodological framework* for carrying out process improvement initiatives in SMEs. On the other hand, the difficulties in the implementation of these research methods in SE (discussed in Sect. 2.2) are addressed properly by the strategy proposed. That is to say, despite the difficulties outlined in this section, the systematic use of the AR and CS research methods in the context of SPI in SMEs was indeed feasible.

We believe that the *research strategy* presented can be used to validate in practice the theoretical proposals for standardization projects in the context of software engineering. Furthermore, we consider that this strategy can be useful for conducting other research projects in other SE areas, which have similar characteristics to those presented in this paper. That is to say, research projects that fulfill the following conditions: (1) they follow the AR qualitative method, (2) they have to start with a *research cycle* to analyze and understand the theoretical knowledge of a specific SE area when creating new research products (models, methods, process, techniques, etc.) that address a problem of the area under research, and (3) they must apply the research products by means of CS (*problem-solving cycle*) in order to obtain knowledge of their practical application, in an effort to refine, improve, and validate these products.

Acknowledgments This work has been funded by: (1) the GEODAS-BC project (Ministerio de Economía y Competitividad and Fondo Europeo de Desarrollo Regional FEDER, TIN2012-37493-C03-01), (2) the GLOBALIA project (Consejería de Educación, Ciencia y Cultura y Fondo Europeo de Desarrollo Regional FEDER, PEIII11-0291-5274), and (3) LATIn Project (Universidad del Cauca, VRI3930). Francisco J. Pino acknowledges the contribution of the University of Cauca, where he works as a Full Professor.

References

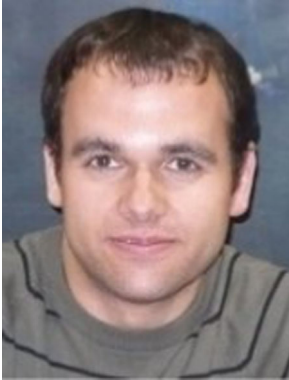
- Aaen, I., Arent, J., Mathiassen, L., & Ngwenyama, O. (2001). A conceptual MAP of software process improvement. *Scandinavian Journal of Information Systems*, *13*, 81–101.
- Avison, D., Lan, F., Myers, M., & Nielsen, A. (1999). Action research. *Communications of the ACM*, *42*(1), 94–97.
- Basili, V. (2000). Keynote on “Experimental software engineering”. In *Proceedings 7th European workshop on software process technology (EWSPT 2000), Kaprun (Austria)*. Springer LNCS 1780, pp. 150.
- Baskerville, R. (1997). Distinguishing action research from participative case studies. *Journal of Systems and Information Technology*, *1*(1), 25–45.
- Baskerville, R. (1999). Investigating information systems with action research. *Communications of the Association for Information Systems*, *2*(19), 1–32.
- Baskerville, R., & Wood-Harper, A. T. (1998). Diversity in information systems action research methods. *European Journal of Information Systems*, *7*(2), 90–107.
- Benbasat, I., Goldstein, D., & Mead, M. (1987). The case research strategy in studies of information systems. *MIS Quarterly*, *11*(3), 369–386.
- Braa, K., & Vidgen, R. (1999). Interpretation, intervention, and reduction in the organizational laboratory: A framework for in-context information system research. *Accounting Management and Information Technology*, *9*, 25–47.
- Brereton, P., Kitchenham, B., Budgen, D., & Li, Z. (2008). Using a protocol template for case study planning. In G. Visaggio, M. T. Baldassarre, S. Linkman, & M. Turner (Eds.), *Evaluation and assessment in software engineering* (pp. 1–8). Bari, Italy: British Computer Society.
- Casey, V., & Richardson, I. (2004). A practical application of the IDEAL model. *Software Process: Improvement and Practice*, *9*(3), 123–132.
- Checkland, P. (1991). From framework through experience to learning: The essential nature of action research. In H. Nissen, H. Klein, & R. Hirscheim (Eds.), *Information systems research: Contemporary approaches and emergent traditions* (pp. 397–403). The Netherlands: North Holland.
- Chiasson, M., Germonprez, M., & Mathiassen, L. (2009). Pluralist action research: A review of the information systems literature. *Information Systems Journal*, *19*(1), 31–54.
- Coghlan, D. (2001). Insider action research projects: Implications for practicing managers. *Management Learning*, *32*, 49–60.
- Cruz Mendoza, R., Morales Trujillo, M., Morgado, C. M., Oktaba, H., Ibarguengoitia, G. E., Pino, F. J., et al. (2009). Supporting the software process improvement in very small entities through e-learning: The HEPALE! Project. In *2009 Mexican international conference on computer science (ENC)*, September 21–25, 2009, pp. 221–231. doi:10.1109/enc.2009.33
- CYTED (2015). COMPETISOFT methodological framework on EPF composer (in Spanish). <http://alarcos.esi.uclm.es/competisoft/web/completo/index.htm>
- Davison, R. (1998). *An action research perspective of group support systems: How to improve meetings in Hong Kong*. Ph.D. Thesis, Hong Kong: City University of Hong Kong.
- Davison, R., Martinsons, M. G., & Kock, N. (2004). Principles of canonical action research. *Information Systems Journal*, *14*(1), 65–86.
- Dittrich, Y., Rönkkö, K., Eriksson, J., Hansson, C., & Lindeberg, O. (2008). Cooperative method development. Combining qualitative empirical research with method, technique and process improvement. *Empirical Software Engineering*, *13*(3), 231–260.
- French, W. L., & Bell, C. H. (1999). *Organization development: Behavioral science interventions for organization improvement*. London: Prentice-Hall.
- Garzàs, J., Pino, F., Piattini, M., & Fernández, C. (2013). A maturity model for the Spanish software industry based on ISO standards. *Computer Standards & Interfaces*, *35*(6), 616–628.
- Glass, R. L., Vessey, I., & Ramesh, V. (2002). Research in software engineering: An analysis of the literature. *Information and Software Technology*, *44*(8), 491–506.
- Gustavsen, B. (1993). Action research and the generation of knowledge. *Human Relations*, *46*, 1361–1365.
- Hareton, L., & Terence, Y. (2001). A process framework for small projects. *Software Process: Improvement and Practice*, *6*(2), 67–83.
- Hernández, M., Florez, A., Pino, F., Garcia, F., Piattini, M., & Ibarguengoitia, G., et al. (2008). Supporting the improvement process for small software enterprises through a software tool. In *Software engineering symposium during ninth Mexican international conference on computer science (ENC'08)*, October, 2008. SES Proceedings, Mexicali, México, pp. 1–8.
- Höst, M., & Runeson, P. (2007). Checklists for software engineering case study research. In *Proceedings of 1st international symposium on empirical software engineering & measurement (ESEM)*. IEEE Computer Society Press, pp. 479–482.

- Howard, M., Vidgen, R., & Powell, P. (2004). Exploring industry dynamics in e-procurement: Sense making by collaborative investigation. In *13th European conference on information systems*, Turku, Finland, pp. 1–12.
- ISO (2011). ISO/IEC 29110. *Software engineering—lifecycle profiles for very small entities (VSEs)*. Geneva: International Organization for Standardization.
- Iversen, J., & Mathiassen, L. (2003). Cultivation and engineering of a software metrics program. *Information Systems Journal*, *13*(1), 3–20.
- Iversen, J., Mathiassen, L., & Nielsen, P. (2004). Managing risk in software process improvement: An action research approach. *MIS Quarterly*, *28*(3), 395–433.
- Johnson, D. L., & Brodman, J. G. (1999). Tailoring the CMM for small businesses, small organizations, and small projects. In K. El Emam & N. H. Madhavji (Eds.), *Elements of software process assessment and improvement* (pp. 239–259). Silver Spring, MD: IEEE CS Press.
- Kauppinen, M., Vartiainen, M., Kontio, J., Kujala, S., & Sulonen, R. (2004). Implementing requirements engineering processes throughout organizations: Success factors and challenges. *Information and Software Technology*, *46*(14), 937–953.
- Kock, N., McQueen, R., & Scott, J. (1997). Can action research be made more rigorous in a positivistic sense? The contribution of an interpretive approach. *Journal of Systems and Information Technology*, *1*, 1–24.
- Laporte, C., Alexandre, S., & O'Connor, R. (2008a). A software engineering lifecycle standard for very small enterprises. In *EuroSPI 2008, Dublin, Ireland*. Springer CCIS, pp. 129–141.
- Laporte, C., Alexandre, S., Renault, A., & Crowder, K. V. (2008b). The development of international standards for very small enterprises. In *18th annual international symposium of INCOSE (INCOSE 2008)*, Amsterdam, The Netherlands, pp. 1–12.
- Levin, M. (1993). Creating networks for rural economic development in Norway. *Human Relations*, *46*, 193–218.
- Luzuriaga, J., Martínez, R., & Cechich, A. (2008). Setting SPI practices in Latin America: An exploratory case study in the justice area. In *2nd international conference on theory and practice of electronic governance (ICEGOV 2008)*, pp. 172–177.
- Martínez-Ruiz, T., Pino, F., León-Pavón, E., García, F., & Piattini, M. (2009). Supporting the process assessment through a flexible software environment. In *Software and data technologies (third international conference, ICSoft 2008)*. Springer CCIS 47, pp. 187–199.
- Mathiassen, L. (2002). Collaborative practice research. *Information Technology & People*, *15*(4), 321–345.
- Mathiassen, L., Pries-Heje, J., & Ngwenyama, O. (2002). *Improving software organizations: From principles to practice*. Upper Saddle River, NJ: Addison-Wesley.
- 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.
- McTaggart, R. (1991). Principles for participatory action research. *Adult Education Quarterly*, *41*(3), 168–187.
- Medeiros, P. S., & Travassos, G. (2011). Action research can swing the balance in experimental software engineering. In M. Zelkowitz (Ed.), *Advances in computers* (Vol. 86, pp. 205–276). Burlington: Academia Press.
- Mejía, R., López, A., & Molina, A. (2007). Experiences in developing collaborative engineering environments: An action research approach. *Computers in Industry*, *58*(4), 329–346.
- Montesi, M., & Lago, P. (2008). Software engineering article types: An analysis of the literature. *Journal of Systems and Software*, *81*(10), 1694–1714.
- Moody, D. (2000). Building links between IS research and professional practice: Improving the relevance and impact of IS research. In *Proceedings of the 21st international conference on information systems*, Brisbane (Australia), pp. 351–360.
- Oktaba, H. (2006). MoProSoft[®]: A software process model for small enterprises. In S. Garcia (Ed.), *Proceedings of the first international research workshop for process improvement in small settings* (pp. 93–101). Pittsburgh: Carnegie Mellon University.
- 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.
- Philips, P. A. (1998). Disseminating and applying the best evidence. *Medical Journal of Australia (MJA)*, *168*, 260–261.
- 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., García, F., & Piattini, M. (2009). 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., Ruiz, F., García, F., & Piattini, M. (2012). A software maintenance methodology for small organizations: Agile_MANTEMA. *Journal of Software Maintenance and Evolution: Research and Practice*, 24(8), 851–876.
- Polo, M., Piattini, M., & Ruiz, F. (2002). Using a qualitative research method for building a software maintenance methodology. *Software Practice and Experience*, 32(13), 1239–1260.
- Richardson, I. (2001). Software process matrix: A small company SPI model. *Software Process: Improvement and Practice*, 6(3), 157–165.
- Robson, C. (2002). *Real world research* (2nd ed.). Oxford: Blackwell.
- Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 14(2), 131–164.
- Saiedian, H., & Carr, N. (1997). Characterizing a software process maturity model for small organizations. *ACM SIGICE Bulletin*, 23(1), 2–11.
- Seaman, C. B. (1999). Qualitative methods in empirical studies of software engineering. *IEEE Transaction on Software Engineering*, 25(4), 557–572.
- Staples, M., Niazi, M., Jeffery, R., Abrahams, A., Byatt, P., & Murphy, R. (2007). An exploratory study of why organizations do not adopt CMMI. *Journal of Systems and Software*, 80(6), 883–895.
- Travassos, G., Medeiros, P. S., Gomes, P., Dias, A. C., & Biolchini, J. (2008). An environment to support large scale experimentation in software engineering. In *Proceedings of the 13th IEEE international conference on engineering of complex computer systems (ICECCS 2008)*, Belfast, Northern Ireland. CS Digital Library, pp. 193–202.
- Wadsworth, Y. (1998). What is participatory Action Research? *Action Research International (Paper 2)*.
- Walsham, G. (1993). *Interpreting information systems in organizations*. Chichester, UK: Wiley.
- Wohlin, C. (2005). An analysis of the most cited articles in software engineering journals. *Information and Software Technology*, 47(15), 957–964.
- Wood-Harper, T. (1985). Research methods in information systems: Using action research. In E. Mumford, R. Hirschheim, G. Fitzgerald, & A. T. Wood-Harper (Eds.), *Research methods in information systems* (pp. 169–191). Amsterdam, North-Holland: Elsevier Science Publishers B.V.
- Wouters, M. (2009). A developmental approach to performance measures—results from a longitudinal case study. *European Management Journal*, 27(1), 64–78.
- Yin, R. K. (2003). *Case study research: Design and methods*. Newbury Park: Sage Publications.
- Zannier, C., Melnik, G., & Maurer, F. (2006). On the success of empirical studies in the international conference on software engineering. In *International conference on software engineering (ICSE 2006)*, Shanghai, China, pp. 341–350.
- Zelkowitz, M. (2009). An update to experimental models for validating computer technology. *Journal of Systems and Software*, 82(3), 373–376.



Francisco J. Pino has a European Ph.D. in Computer Science from the University of Castilla–La Mancha (UCLM), Spain. He is currently a full professor at the Electronic and Telecommunications Engineering Faculty at the University of Cauca, in Popayán (Colombia). He is a member of the IDIS Research Group and his research interests are: software process improvement in small companies and qualitative research methods for software engineering.



Félix García received his M.S. (2001) and Ph.D. (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.



Mario Piattini has an M.Sc. and Ph.D. 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.



Hanna Oktaba has an M.Sc. and Ph.D. in Computer Science from the University of Varsovia, Polonia. He is currently a full professor at the Facultad de Ciencias of the National Autonomous University of Mexico (UNAM), in Ciudad de Mexico, Mexico. She leads the KUALI-KAANS Research Group and her research interests are: software engineering, software industry standard, and software process improvement in small companies.