

# Supporting the Management Process of Software Process Improvement Initiatives based on NMX-I-059/02-NYCE-2005

Garcia, I. and Cruz, D.

Postgraduate Department  
Technological University of the Mixtec Region, Mexico  
{ivan@mixteco.utm.mx, dago@mixteco.utm.mx}

**Abstract.** Nowadays there are models and standards which attempt to introduce quality in the enterprises' software development process with the objective to introduce high quality levels in the produced software. The NMX-I-059/02-NYCE-2005 standard (also known as MoProSoft) is focused on small and medium software enterprises, or small groups of software development within a larger organization, with the aim of promoting the standardization of an effective process in the software industry. Mexican enterprises now have a software standard that enables them to achieve a high level of quality in the software that they produce. However, the adoption of any standard is not an easy task. This paper aims to show that the development and implementation of a RIA-based tool that could support improvement initiatives, therefore strengthening the standard adoption.

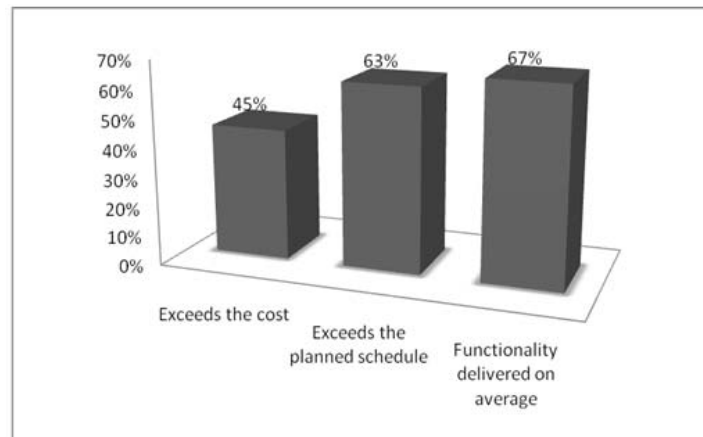
**Keywords:** Software process improvement, effort and adoption time, MoProSoft, small software enterprises, process assessment and improvement.

## 1 Introduction

Software has arisen as a fundamental pillar in the evolution of computational products and services. In the last two decades, software has changed from a "specific problem solution" to an autonomous industry [11]. However, this change keeps the old same problems since the "software crisis" in 1969. Nowadays, the quantity and quality demands dominate the market. According to the last report of Standish Group Inc. [38]:

- Software is (almost) always delivered out of the initial planning,
- Software is more expensive than the original cost,
- Software has a different functionality.

Pressman says that: "*the majority of software crisis causes have their origins in myths and theories that arose in the early years of Software Engineering*" This origin makes the myths more dangerous; but the truth is that they do not look like myths any more. Figure 1 shows that a recent study [39] establishes that software does not accomplish the original requirements because: 45% of software exceeds the cost, 63% of software exceeds the planned schedule, and software fulfills the 67% of the required functionality.



**Fig. 1.** Principal causes of failure in software projects

Software industry has tried to increase productivity and quality adopting alternative methodologies and technologies, but it has been recognized that the main problem is related to the incapacity to manage the software process [27] [17]. In this way, enterprises have been changed from technological-based solutions to process-based solutions. From the beginning of the 90's, industry and researchers interested in Software Engineering have been expressing special interest in Software Process Improvement (SPI) [21]. An indicator of this is the increasing number of international initiatives related to SPI, such as CMMI-DEV v1.2 [35], ISO/IEC 15504:2004 [14], SPICE [13], and ISO/IEC 12207:2004 [15].

In addition, many methods for evaluating improvements in organizations, such as SCAMPI [19], ISO/IEC 15504:2004 and CBA-IP1 [4], and improvement models such as IDEAL [18] have been developed. This interest in software improvement in large enterprises is now being extended to small enterprises. However, the problem is the high implementation cost, independent of the size of the company [5]. Because models have been developed for large enterprises, only a few small software Enterprises (SE) are aware of them. In Mexico, in April 2007 there were almost 1,500 SE that accounted for 99.87 % of all companies in this category. Due to this, the Software Industry Process Model (MoProSoft) [23] and its process-assessment method (EvalProSoft) were defined [24].

Since the approval of MoProSoft and EvalProSoft as the Mexican standard NMX-I-059-NYCE-2005, the interest of the SE on acquiring this certification has grown. Besides the certification, as a result of the correlation between the process quality and the obtained product quality, the Mexican SE have gained impulse to improve their software processes as a strategy to assure the quality of their software products [6]. However, besides the certification desire, the SE has the problem to adopt quickly and efficiently the standard NMX-I-059-NYCE-2005 without loss resources and time to develop new projects.

This paper presents the experiences on adopting a RIA-based tool as support to facilitate the certification process by four small software companies located in the states

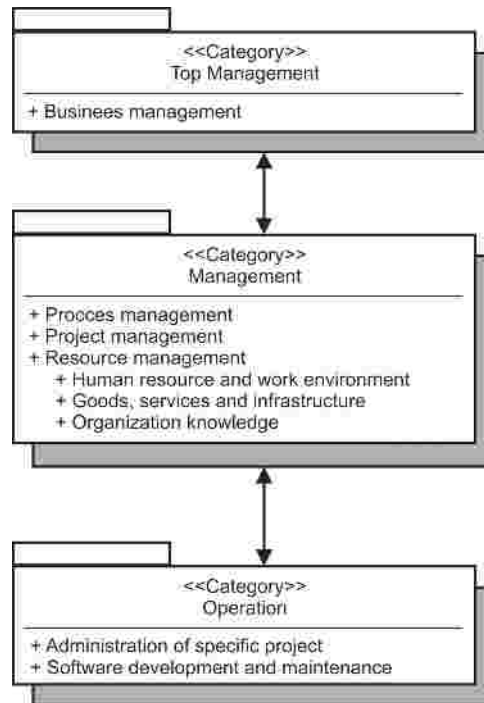
of Tlaxcala, Puebla and Oaxaca. The document is structured as follows: Section 2 is an introduction to the MoProSoft model and Section 3 discusses related work. Section 4 describes the architecture and development of RIA-based tool. Section 5 describes the methodology used and presents the characteristics of the case study. Section 6 describes the initial scenario prior to the tool support and presents results and lessons learned. Finally, conclusions are shown in Section 7.

## **2 The MoProSoft Model**

In 2006 the Mexican government, through the National Plan for Development, established the objectives to increase and extend the country's competitiveness using the information and technology. One of the crucial issues to address was to "promote the IT industry development" Thus, the Economics Secretary and enterprise organisms designed the Software Industry Development Program (PROSOFT) to promote the software industry and extend the IT market in Mexico. According to the government and industry requirements, the selection process of one model that would be adopted by the Mexican software industry it would be based on five criteria [25]. The model should be:

- Adequate for SE with a low maturity level,
- Affordable cost of adoption and evaluation,
- Acceptable as national norm,
- Specific for software enterprises,
- Defined as a set of processes based on international practices.

According to Oktaba [25] no one of the analyzed models (ISO 9000:2000 [32], CMM/CMMI [34], ISO/IEC 12207 [15], and ISO/IEC 15504 [14]) completely fulfilled the established criteria. The new proposed model, MoProSoft, is developed taking into account the better practices of models as CMMI-SW [34], ISO 9000:2000, PMBoK [29], among others. This model provides a new process structure, new elements to document the process, a more precise relation among processes, and an explicit mechanism for SPI [25] [26]. MoProSoft is conformed by the Software Industry Process-Assessment Method (EvalProSoft) which is based on ISO/IEC 15504 [Part 2] recommendations. Figure 2 shows the MoProSoft' categories emphasizing that, unlike other reference models, has the Top Management category, whose main objective is to organize the activities of the SE executives by means of introducing modern management and Software Engineering practices.

**Fig. 2.** MoProSoft structure

To carry out an initiative of SPI within an organization, it is necessary to involve: a model that guides the improvement, a method for assessing a process, and a reference model to follow. The capability levels achieved and their process attributes are located over a scale of six levels, where the level 0 is associated with the lowest capability level, indicating that the purpose of the process is not reached. Table 1 shows the levels which are used to determine whether a process has reached a capability level.

**Table 1.** MoProSoft capability levels

Process attributes	
<b>Level 0</b> Incomplete	It means that the process has not reached the expected objectives.
<b>Level 1</b> Realized	The implemented process achieves its purpose and obtains defined results.
<b>Level 2</b> Managed	The realized process is implemented in an administrative way, and the work products are properly established, controlled and maintained.
<b>Level 3</b> Established	The process is managed and implemented through defined objectives, which are capable of achieving the desired outcome.
<b>Level 4</b> Predictable	The established process operates within certain limits to achieve their results.
<b>Level 5</b> Optimized	The predictable process is continuously improved in order to achieve current and future applicable business goals.

At the end of 2009, 171 organizations have been assessed under the standard NMX-I-059/02-NYCE-2005. From these, 93% has level 1 and 5% level 2 [40]. But, Mexican industry should accelerate the standard adoption, and according to Young et. al [43] a way to do it is introducing software tools to process improvement.

### 3 Related Work

Rapid assessments are a success factor for the SPI efforts of any organization because they can be frequently applied, with minimal effort-time and resources. These assessments provide information to enterprises about the impact of improvement actions, introduced through a SPI effort, in processes. In literature there is some research related to rapid assessments: SEAL OQ [12], PIASS [22], SPIS [8], KMT [1], SPQA.Web [28], and some commercial tools created by specialized companies: CMMI v1.2 Browser [42], CMM-Quest [33], Appraisal Wizard [2], SPICE 1-2-1 [36], IME Toolkit [16], MKS Integrity Suite [20], and Stages for CMMI [37].

The standard NMX-I-059/02-NYCE is relatively new since its creation at 2005. There are some researches which try to accelerate and support the SPI initiatives focused on MoProSoft; however, none of them could be considered a SPI tool.

Research by [41], for example, provides a tool for accelerating the adoption of MoProSoft and is supported by the AceleraProSoft<sup>1</sup> project. The main functionality of Kuali tool is that it provides a mechanism to manage and control all documents that result from implementing the MoProSoft processes. But, it does not provide SPI support to assess or adopt efficiently the model.

Caballero [3] provides a document management tool, named MDM, that supports enterprises to document the MoProSoft processes; specifically business management, process management, and administration of specific projects. MDM was developed to provide support via Web and enables the creation of templates for each process. However, MDM is not an SPI tool.

The Guiding and Monitoring Tool for Automation of MoProSoft (Assistant HIM) [44] provides the support to adapt and monitor the model through a Web environment. Assistant HIM is an electronic guide that shows processes, activities, product, and roles according to MoProSoft. The Web tool uses a knowledge base (developed with the Resource Description Framework) that generates the needed information for model in XML files. Nevertheless, Assistant HIM assumes that the SPI effort was already conducted and provides only a guide to “easily” adopt the model, but it is not a SPI tool too.

Reyes et. al [30] provides an Instrument of Self-Assessment for Diagnosing the Software Process using MoProSoft. This tool is focused on the process management process and assesses the organizations to identify improvement areas. The collected information is quantitative and qualitative and correspond with a Likert scale [31] that

---

<sup>1</sup> AceleraProSoft is an initiative developed by the Mexican Economics Secretary, together with Microsoft, Visionaria and Amity. This initiative attempts to guide enterprises to increase their sales from short and medium term. Besides, it focused its efforts to strengthen the planning, operation, sales management, and marketing research.

represents the MoProSoft' capability levels showed in Table 1. This tool could be considered as a SPI tool because it assesses the software process and identifies an organization's strengths and weaknesses; however, it only assess and does not provide an improvement plan nor monitoring the improvement activities.

Summarizing, Table 2 shows a comparison among the analyzed tools according to their function and operation perspectives with the aim of highlighting some gaps and obtain an initial benchmark.

As we can see, the available tools could be used to rapidly adopt the standard NMX-I-059/02-NYCE-2005, however there is not information about their success in real small environments. That is, there is a limit to being able to provide more helpful and diverse support for establishing SPI initiatives using MoProSoft and managing the whole process of SPI.

We provided an alternative way to reduce the effort of adoption using a Rich Internet Application (RIA) tool. In addition, the proposed tool provides not only a fundamental process assessment and improvement features, but also an improvement plan generation and remote online assessment and self-evaluation respectively depending on organizations.

**Table 2.** MoProSoft' tools features

Criteria	MoProSoft tools			
	Kuali	MD M	Assistant HIM	Self- Assessment
Supports SPI initiatives	No	No	No	No
Assesses and generates improvement plans	No	No	No	Only assess
Obtains a snapshot of current process	No	No	No	Yes
Covers all model processes	No	No	Yes	No
Developed for Web environment	No	Yes	Yes	Yes
Measures and monitoring the SPI initiative	No	No	No	No
Applied on SE	No	No	No	No

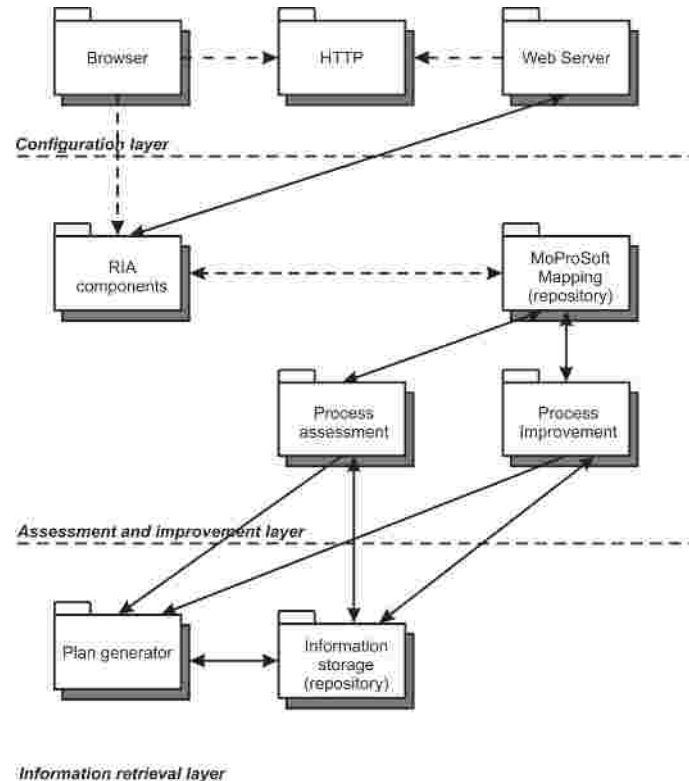
#### 4 A RIA-based tool for SPI

The SelfVation (SELF-eValuATION) tool provides support to easily adopt the MoProSoft model. SelfVation is a RIA-based tool. RIAs are web applications that have most of the characteristics of desktop applications, typically delivered either by way of a standard based web browser, via a browser plug-in, or independently via sandboxes or virtual machines. Examples of RIA frameworks include Ajax, Curl, GWT, Adobe Flash/Adobe Flex/AIR, Java/JavaFX, Mozilla's XUL and Microsoft Silverlight.

We are trying to provide a RIA-based tool that establishes an iterative approach to process improvement using the standard NMX-I-059/02-NYCE-2005 and which a small organization could adopt. Following this approach, the focus of the first step

would be to understand what exists in the organization and determine what causes significant problems. Then solutions could be devised in the action plan and evaluated in pilot studies or even controlled experiments.

SelfVation' architecture is based on three layers separating the configuration settings, the assessment and improvement mechanisms, and information retrieval (see Figure 3).

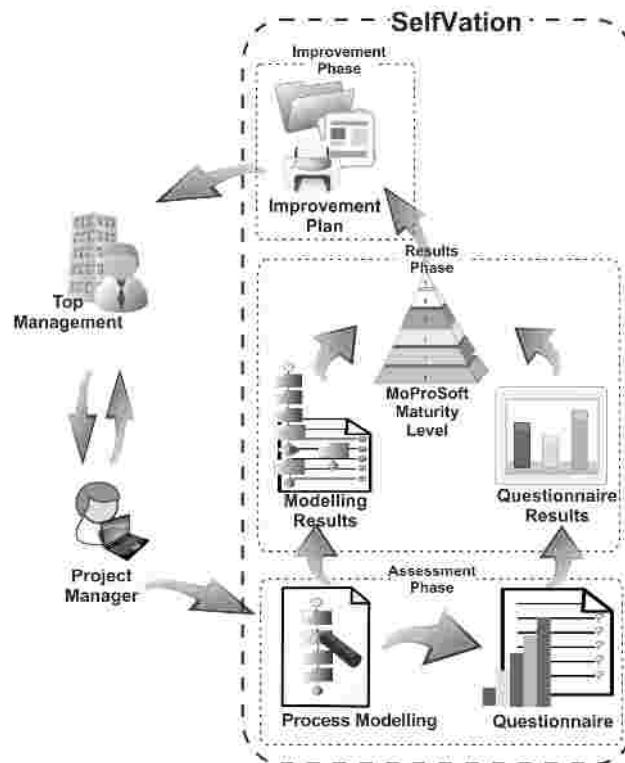


**Fig. 3.** The architecture of SelfVation

This kind of architecture makes possible that through the configuration layer, all SEs take control over the complete SPI initiative and obtain, at the same time, customized results; all the business characteristics are evaluated with the assessment and improvement layer, establishing a real mapping of current software processes. The information retrieval layer manages all the generated information and uses the plan generator to send the complete information to clients. This architecture provides a light application for any client, and offers advantages related with implementation and management capabilities in a flexible computational program.

Figure 4 illustrates that the SelfVation strategy begins with an evaluation that is performed in two ways: *process modeling* and *questionnaires*. Both results are mapped against the standard NMX-I-059/02-NYCE-2005 and results are analyzed by an internal manager who generates and improvement plan. SelfVation covers the Top

Management category managing the top management level information, and the Management and Operation categories managing the practices conducted by the project managers.



**Fig. 4.** SelfVation modules

SelfVation provides three types of user interface modules: (1) the assessment phase, (2) the results phase, and (3) the improvement phase.

#### 4.1 The assessment phase.

This phase guides the project managers to obtain all knowledge from their daily labor. Previously, top management had selected for MoProSoft' processes to evaluate. This interface uses a refined version of the Liker scale and establishes 'levels of performance' from the two-phase questionnaire proposed in [10]. Table 3 shows that each answer of Liker scale corresponds with an established level of performance to determine the percentage in which each practice is performed.



**Table 3.** SelfVation' perform levels

Possible answer	Perform level	Description
<i>Always</i>	4	The activity is documented and established in the organization (between 75% and 100% of the time).
<i>Usually</i>	3	The activity is established in the organization but rarely documented (between 50% and 74% of the time).
<i>Sometimes</i>	2	The activity is weakly established in the organization (between 25% and 49% of the time).
<i>Rarely</i>	1	The activity is rarely performed in the organization (between 1% and 24% of the time).
<i>Never</i>	0	The activity is not performed in the organization.

Giving a specific weight to each response will enable us to easily analyze the results of the evaluation and to identify which practices are common within the whole organization and which ones are not performed at all. The second part of assessment is related to process modeling. SelfVation uses the Process Change Methodology [7] to describe the current process based on MoProSoft and mapping it against an ideal process according to the model. None of existing tools had ever been used this kind of assessment. SelfVation introduces the use of graphical notation to assess or modify current and pilot processes.

#### 4.2 The result phase.

This phase, which allows members of top management to obtain any information on the SPI cycle at any time; they can control the performance of its project managers through the assessment phase and obtain the final results and graphics derived from the entire process. The mapping process presents a categorized level of performance, in accordance with the assessed process. The project manager can meet its own level of performance. This phase just provides performance level results to project managers; the entire results of the organization can only be reviewed by top management through a reports generator.

#### 4.3 The improvement phase.

Improvement phase takes the results obtained in previous phase and provides the mapping with the knowledge base that contains the MoProSoft activities. The improvement guide is offered in a RIA improvement performance. This means, that the improvement mechanism of SelfVation can avoid the latency of round-trips to the server by processing locally on the client and are often a lot faster. Offloading work to the clients can also improve server performance. Conversely, the resource requirements can be prohibitive for small, embedded and mobile devices.

## 5 Methodology

On the graduate program of Masters in Computer Science of the Technological University of the Mixtec Region (UTM), a research project was conducted to obtain data that contribute to a successful implementation of the SelfVation tool. The main tasks of the methodology for this project were:

1. Identification of small software organizations which want to assume an improvement commitment,
2. Using SelfVation to assess the organizations' actual situation,
3. Using SelfVation to improve the weakness detected in step 2.
4. Implementing the mandatory processes awareness through the tool.
5. Performing and official assessment of compliance with MoProSoft capability level.

In order to choose the right SEs, we decided to experiment with three kinds of enterprises: ones which know and uses the standard, ones which only know the standard, and those which do not know the standard. Since our main objective is to reduce the effort-time to adopt the standard NMX-I-059/02-NYCE-2005, we believed that one of the crucial factors is that SelfVation provides the same guide to those enterprises which really implement the model and the others who do not have previous knowledge. Table 4 shows the characteristics of participating small enterprises.

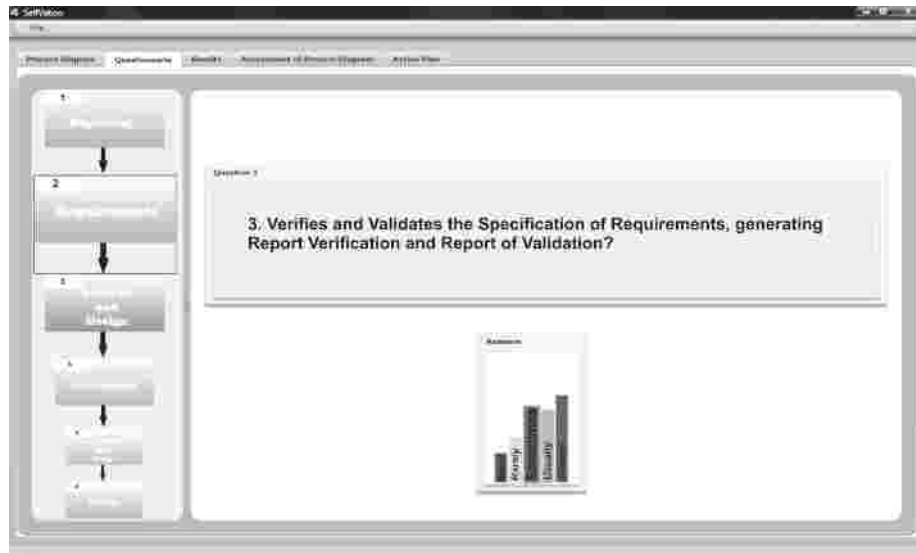
**Table 4.** Profiles of the participating SEs

SE	Activity	Size	Experience
1	Software development and maintenance	10	The standard is implemented
2	Software development and maintenance	20	The standard is known, but not implemented
3	Development of software solutions for small enterprises	15	The standard is not known
4	Hardware & software services and integral solutions	8	The standard is known, but not implemented

A case study is commonly employed as an empirical research strategy in information systems field, often used for describing relationships within organizational settings [9]. In the context of our research, we employ a case study as the method to provide an organizational context for the application of SelfVation for identifying SPI factors and their implementation and deployment to four particular small software companies.

## 6 A case study: adopting SPI initiatives in four Mexican small software enterprises

We designed a controlled experiment with four small enterprises in which we focus our effort in a semiformal version of the EvalProSoft approach. An assessment team and four project managers were chosen, besides we separated the four organizations into three categories according to their experience with the standard. The selected project managers are professional who know the enterprise's culture and the way that the development projects are conducted. The project managers had received the standard NMX-I-059/02-NYCE-2005 basic training. In a similar manner the project managers had answered a structured questionnaire related to standard' categories. Figure 5 illustrates the assessment phase using questionnaires; the answers are agreed with the way that the enterprise works by each phase of the development lifecycle that the standard establishes (left side of the Figure 5).



**Fig. 5.** Assessment phase

Once the questionnaires were completed we proceed to assess enterprises using the modeling process evaluation. This RIA component provides an easy interface to capture the essence of SE' development process (see Figure 6).

These two assessment components enable us to identify the strong and weak points in order to show the current situation of the four enterprises according to the standard NMX-I-059/02-NYCE-2005, so that it is indicated where they focus their efforts to raise the quality of the software that they develop and maintain. Figure 7 shows the coverage of the standard for enterprise SE1.

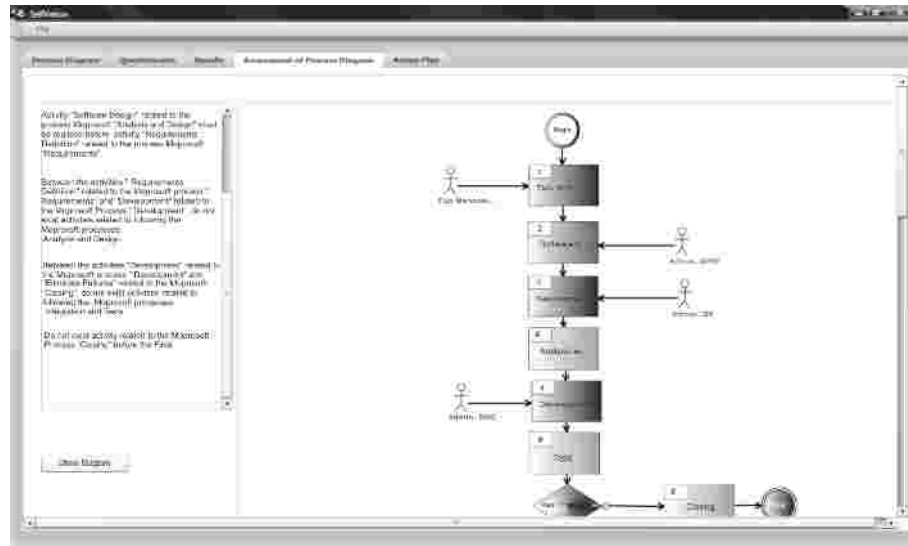


Fig. 6. Process modeling assessment

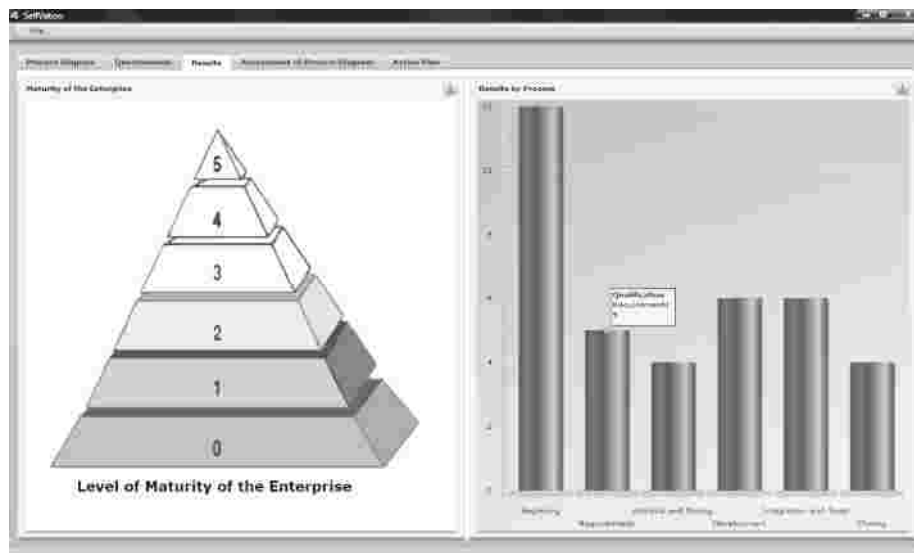


Fig. 7. Results phase

These mapped results are used by the reports generator module and create an improvement plan according the current situation for every organization. The improvement plan of Figure 8 shows different domains (practices, products, inputs, outputs, measurement criteria, entry criteria, exit criteria) which enable small companies to establish the corrective actions indicated in the final report.

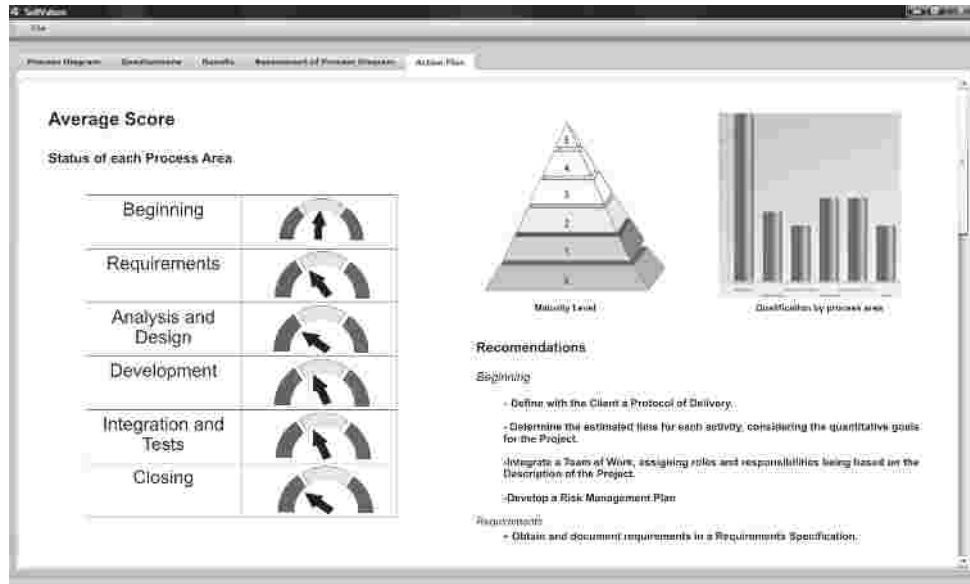


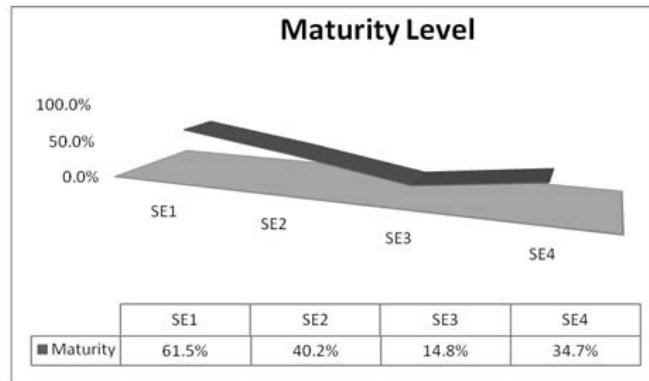
Fig. 8. Improvement phase

### 6.1 Experimental results and learned lessons.

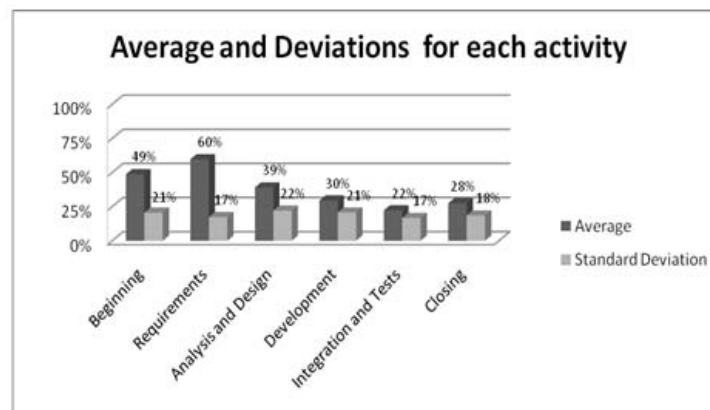
After implementing the assessment and improvement phase we obtain information about the practices and effort performed by the four SE. As we said before, SE1 has an official implementation of standard NMX-I-059/02-NYCE-2005 and provides us a measure of contrast. All enterprises were assessed in Level 1 of MoProSoft (a realized level) and Figure 9 illustrates the obtained results.

We believe there exists a relation between the enterprise size and the effectiveness for standard adoption. We do not try to affirm that there exists a proportional relation between both factors, but it is possible that small teams are more organized and make a rapid adoption of standard. Figure 9 shows that SE1 obtained a 61.5% of coverage in Level 1 that corresponds with its real conformance level. SE2 and SE4 obtained similar coverage results, approximately a 37%, which are insufficient to achieve Level 1 (according to the assessment mechanism over the 60%). SE3 obtain a lower coverage ratio of 14.8% that locate it in an *ad-hoc* or chaotic situation. These results were obtained from a launch of an improvement initiative using SelfVation.

On the other hand, the coverage by each activity of standard NMX-I-059/02-NYCE-2005 was obtained. Figure 10 shows that the best coverage is obtained for the Requirements activity (60% of average and 17% of standard deviation) and indicates that the requirements are used as a basis to develop a plan and construct the software product. The worst coverage is obtained for Integration and Test activity (22% of an average and 17% of standard deviation) because small companies do not use formal methods of product validation/verification.



**Fig. 9.** Maturity level per enterprise



**Fig. 10.** Coverage per activity

The purpose of this experiment was to demonstrate the ease and usefulness of SelfVation as an SPI tool for SE, which also reduces the costs of undertaking an SPI initiative, considering the particular conditions of these organizations. The initial assessment with SelfVation was conducted to establish the base line capabilities of the enterprises processes according to the standard NMX-I-059/02-NYCE-2005. The result was “poor” –between 0 and 1 (according to MoProSoft levels). During the following 3 months SelfVation coached the enterprises on SPI tailoring and adoption through improvement plans.

Finally, we applied the second assessment with SelfVation to each enterprise; all four enterprises achieved a 1.00 average increase in the Maturity Level of two areas: Administration of Specific Project and Software Development and Maintenance. This increased level is obtained from the initial process coverage over 100%. For example, SE1 has an initial coverage of 61.5% when the first assessment was conducted. A reassessment after using SelfVation showed an increase in processes of 0.28 (in first iteration). Therefore the process has an improved level of  $61.5\% + 61.5\% \times 0.28$ .

**Table 5.** Improvement and effort data by enterprise

	Enterprises			
	SE1	SE2	SE3	SE4
Total effort (hours)	287	484	554	220
Total effort SelfVation	243	400	495	185
Effort per person (hours)	28.7	24.2	36.9	27.5
Effort with SelfVation	14.2	11.3	17.9	20.2
Improvement average	0.28	0.86	1.43	1.00

An official assessment in SE1 demonstrates that the SelfVation' results coincide with obtained data; but we obtained this information with less effort, it means we are capable to reduce the time of adoption.

These results show the applicability of our tool; however we need to experiment with more than four SEs. We are designing a new experiment that involves 15 small companies from different cities in the Mexican Republic. The lessons learned will be discussed in future research.

## 7 Conclusions

The implementation success when enterprises try to adopt an improvement initiative depends on the organization's top management commitment. Small companies are not the exception, but this research enables to identify that SE does not understand the benefits that this improvement process would have in the company. To initiate the improvement program, involvement and experience of the key personnel could be the key factors that contribute to strengthen the improvement initiative. But, the standard NMX-I-059/02-NYCE-2005 is relatively new in Mexico and small companies have no experience in SPI.

This research, therefore, has developed an instrument to define and implement SPI initiatives to improve the current status of SE practices using the standard NMX-I-059/02-NYCE-2005 as reference model. Its purpose was to investigate its feasibility in SE and to influence the direction of future research. One limitation of this study is the generalization of its findings based on the limited amount of data collected and analyzed relative to the number of small organizations. This suggests that this qualitative study will be augmented by quantitative studies to strengthen the data supporting the need and applicability of SelfVation to the Mexican small organization community.

At the moment of this report, the selected SE1s have decided to improve their processes in order to achieve a level 2 certification. We are collaborating through the implementation of an SPI iterative cycle using SelfVation.

## References

1. Alagarsamy, K., Justus, S. and Iyakutti, K. "On the Implementation of a Knowledge Management Tool for SPI" Proc. of the International Conference on Computational Intelligence and Multimedia Applications, ICCIMA 07. IEEE Computer Society, pp. 48-55. 2007.
2. Appraisal Wizard, Formal or Informal Appraisal Tool, 2009. Available at: <http://www.isd-inc.com>
3. Caballero-De la Villa, D. "Manejador de Documentos de MoProSoft". Universidad de las Americas Puebla. Mayo, 2005 (in Spanish).
4. Dunaway, D.K., Masters, S. CMM-Based Appraisal for Internal Process Improvement (CBA IPI). Method Description. Technical Report CMU/SEI-96-TR-007. Carnegie Mellon University, Software Engineering Institute, Pittsburgh. 1996.
5. Dybå, T. "Factors of software process improvement success in small and large organizations: an empirical study in the Scandinavian context" ACM SIGSOFT Software Engineering Notes, 28(5): 148-157. September 2003.
6. Flores, B., Astorga, M., Olguín, J. and Andrade, M. C. "Experiences on the Implementation of MoProSoft and Assessment of Processes under the NMX-I-059/02-NYCE- 2005 Standard in a Small Software Development Enterprise" Proc. of the 2008 Mexican International Conference on Computer Science, IEEE Computer Society, pp. 323-328, 2008.
7. Fowler, P.; Middlecoat, B.; & Yo, S. Lessons Learned Collaborating on a Process for SPI at Xerox (CMU/SEI-99-TR-006, ADA373332). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1999.
8. Fukuyama, S., Miyamura, S., Takagi, H. and Tanaka, R. "Validating the ISO/IEC 15504 Measure of Software Requirements Analysis Process Capability", IEICE Transactions on Information and Systems, E83-D(4): 747-756, 2000.
9. Galliers, R. Information Systems Research. Issues, Methods and Practical Guideline. Alfred Waller Ltd. Chippenham, Wiltshire. England. 1992.
10. Garcia, I., Calvo-Manzano, J., Cuevas, G. and San Feliu, T. "Determining Practice Achievement in Project Management Using a Two-Phase Questionnaire on Small and Medium Enterprises" Proc. of the European Systems and Software Process Improvement and Innovation Conference, EUROSPI 2007, LNCS 4764, pp. 46-58, Springer-Verlag Berlin Heidelberg. 2007.
11. Gırba, T. and Ducasse, S. "Modeling history to analyze software evolution," Journal of Software Maintenance: Research and Practice (JSME), vol. 18, pp. 207-236, 2006.
12. Him Lok, R. and Walker, A. J. "Automated Tool Support for an Emerging International Software Process Assessment Standard" Proc. of the Third International Software Engineering Standards Symposium; Emerging International Standard, ISESS 1997. IEEE Computer Society, pp. 25-35. 1997.
13. ISO/IEC TR 15504:1998(E): Information Technology – Software Process Assessments. Parts 1-9. International Organization for Standardization: Geneva. 1998.
14. ISO/IEC 15504-2:2003/Cor.1:2004(E): Information Technology – Process Assessment – Part 2: Performing an Assessment. International Organization for Standardization: Geneva. 2004.



15. ISO/IEC 12207:2002/FDAM 2: Information Technology – Software Life Cycle Processes. International Organization for Standardization: Geneva. 2004.
16. IME Toolkit, 2009. Available at: [http://www.man-info-systems.com/index\\_files/FreeTools.htm](http://www.man-info-systems.com/index_files/FreeTools.htm)
17. Johnson, J. *My Life Is Failure: 100 Things You Should Know to Be a Better Project Leader*. Standish Group International Publisher. 2006.
18. McFeeley, B. “IDEAL: A User’s Guide for Software Process Improvement” CMU/SEI-96-HB-001, Software Engineering Institute, Carnegie Mellon University. 1996.
19. Members of the Assessment Method Integrated Team. Standard CMMI® Appraisal Method for Process Improvement (SCAMPI), Version 1.1, CMU/SEI-2001-HB-001. Software Engineering Institute, Carnegie Mellon University. Pittsburgh, PA. 2006.
20. MKS Tool, 2009. Available at: [http://www.mks.com/process\\_improvement](http://www.mks.com/process_improvement)
21. Moe, N. B. and Dybå, T. “Improving by Involving: A Case Study in a Small Software Company” Proc. of the European Software Process Improvement Conference 2006 (EUROSPI 2006), LNCS 4257, pp. 159-170, 2006.
22. Nakakoji, K. “PIASS: Process-Improvement Activity Support System” Technical Report SRA-SEL-97081, Software Engineering Lab., SRA Inc., Tokyo, Japan, 1997.
23. NMX-I-059/01-NYCE-2005. Information Technology – Software – Process and Assessment Model to Software Development and Maintain – Part 02: Processes requirements (MoProSoft). NMX-NYCE. 2007.
24. NMX-I-059/04-NYCE-2005. Information Technology – Software – Process and Assessment Model to Software Development and Maintain – Part 04: Guidelines for processes assessment (EvalProSoft). NMX-NYCE. 2007.
25. Oktaba, H. “MoProSoft: A Software Process Model for Small Enterprises”. Proc. of the First International Research Workshop for Process Improvement in Small Settings, Software Engineering Institute, Carnegie Mellon University. Special Report CMU/SEI-2006-SR-001, pp. 93-101, 2006.
26. Oktaba, H., Garcia, F., Piattini, M., Ruiz, F., Pino, F. and Alquicira, C. “Software Process Improvement: The Competisoft Project” *Computer*, 40(10): 21-28. 2007.
27. Oktaba, H. and Piattini, M. *Software Process Improvement for Small and Medium Enterprises: Techniques and Case Studies*. Information Science Reference Publisher. 2008.
28. Pino, Francisco J., García, F. & Piattini, M. “Herramienta de Soporte a la Valoración Rápida de Procesos Software”. *IEEE Latin America Transactions*, 5(4). July, 2007 (in Spanish).
29. Project Management Institute. *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*. Project Management Institute. 2004.
30. Reyes, P. Y., Margarin, M. L., Álvarez, F. and Muñoz, J. “Diseño de un Instrumento de Auto-evaluación para Diagnosticar el Estatus de las Organizaciones en México con Respecto al Modelo ProSoft: Proceso de Gestión de Procesos de la Categoría Gestión”. Universidad Politécnica de Aguascalientes. 2008 (in Spanish).
31. Russell, C. and Bobko, P. “Moderated Regression Analysis and Likert Scales too Coarse Comfort” *Journal of Applied Psychology*, 77(3): 336-342. 1992.

32. Russell, S. "ISO 9000:2000 and the EFQM Excellence Model: competition or co-operation?" *Total Quality Management*, vol. 11, no. 4/5&6, pp. 657-665, 2000.
33. Self Assessment Tool CMM-Quest, 2009. Available at: <http://www.cmm-quest.com/>
34. Software Engineering Institute. CMMI for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing (CMMI-SE/SW/IPPD/SS, V1.1). Continuous Representation. CMU/SEI-2002-TR-011, Software Engineering Institute, Carnegie Mellon University. 2002.
35. Software Engineering Institute. CMMI for Development (CMMI-DEV, V1.2) CMU/SEI-2006 TR-008, Software Engineering Institute, Carnegie Mellon University. 2006.
36. SPICE 1-2-1, 2009. Available at: <http://www.synspace.com/tools.html>
37. Stages for CMMI, 2009. Available at: <http://www.methodpark.com/en/products/stages-special-editions/stages-for-cmmi-process-management-process-asset-library/>
38. Standish Group International, Inc. CHAOS Report 2007: The Laws of CHAOS. 2007.
39. Standish Group International, Inc. CHAOS Summary 2009. 2009.
40. TI-NYCE Verification unity. Reports list. Available at: <http://www.nyce.org.mx/dictamenes.htm> [Online] November, 2009.
41. Strevel, C. "Kuali: Herramienta Auxiliar para implementación de MoProSoft". DevDays, Intellect. 2005 (in Spanish).
42. Wibas CMMI Browser, 2009. Available at: [http://www.cmmi.de/cmmi\\_v1.2/browser.html#hs:null](http://www.cmmi.de/cmmi_v1.2/browser.html#hs:null).
43. Young, H., Fang, T. and Hu, C. "A Successful Practice of Applying Software Tools to CMMI Process Improvement" *Journal of Software Engineering Studies*, 1(2): 78-95, December 2006.
44. Zurita-Rendón, H. "Arquitectura de la Herramienta Integral para MoProSoft". Universidad Nacional Autónoma de México. 2005 (in Spanish).