

OWPL: A Gradual Approach for Software Process Improvement In SMEs

Simon Alexandre

CETIC
8 rue Clement Ader
6041 Gosselies, Belgium
+32(0)71-91 98 00
Simon.alexandre@cetic.be

Alain Renault

CRP Henri Tudor-CITI
29, avenue John F.
Kennedy
L-1855 Luxembourg-
Kirchberg
Alain.renault@tudor.lu

Naji Habra

University of Namur
21 rue Grandgagnage
5000 Namur, Belgium
+32(0)81-72 49 95
nha@info.fundp.ac.be

Abstract

This paper describes an experience with a Software Process Improvement (SPI) approach particularly adapted to small structures with low software maturity level (e.g. small and medium enterprises in software businesses, small software teams in bigger businesses, small software teams in public organisations). A characterisation of Software Process in-the-small is first made on basis of a deep analysis of software processes actually followed by a number of small and medium enterprises (SMEs). The approach proposes a gradual Software Process Improvement framework based on three nested models. It allows SMEs to start SPI in a very targeted manner, to quickly progress within a limited budget and, eventually, to reach an acceptable level according to SPI standard models such as CMM and SPICE.

1. Introduction

Most of big companies are now aware of quality issues and invest time and money in improving their products and services. Some of them have created their own reference models but most are using standard models like SPICE¹ [1], CMM [2] and CMMI² [3] developed by international organisations. However, those available models are not directly usable for very small businesses as they are much too complicated and too expensive to implement. The same statement has been made by several ISO national bodies that have recently decided to invest in the creation of ISO standards for VSE. It is worth mentioning that the *OWPL Gradual*

Framework represents one of the inputs currently considered by the new ISO/IEC SC 7 Working Group 24 [16]. Briefly, the working group is mandated to facilitate access to, and utilization of, ISO (the International Organization for Standardization) Software engineering standards in very small enterprises (VSEs), a term which includes small software development departments and small projects within larger organizations. VSEs are typically organizations (or projects) which have 25 or fewer employees.

Meanwhile, there was a need to implement a software improvement approach [4] that can be used by small development structures (from 1 to 50 people in charge of software development) or even medium enterprises with a very low software maturity level. This approach has been tested on several typical cases (companies of different sizes and activities) so that concrete conclusions from this experience can now be drawn.

The approach involves a series of gradual assessments: a *micro-evaluation*, an *OWPL-evaluation* and a SPICE or CMM assessment. Experience with these evaluation actions [5], as well as with other supporting actions (thematic awareness conferences, training modules and thematic groups about different software quality aspects) has allowed us to gather a rich knowledge base on software process in SMEs.

Similar initiatives of standard model adaptation or utilisation can be found in [6][7][8][9]. The particularity of our approach lies on a few essential points:

- The OWPL model has more been influenced by SPICE than by CMM ;
- It focuses on the lowest maturity levels ;
- The model emphasises more on training than on documentation and formalisation aspects ;

¹ ISO/IEC15504

² Software Engineering Institute, Carnegie Mellon University

- The selection method used to constitute the sample of people taking part to the evaluation itself guaranties the relevance of the information collected;
- *The scoring method used to evaluate each practice and process guaranties the quality and relevance of final capacity profile;*
- The OWPL gradual approach permits the use of the appropriate method and helps smooth implementation of new practices.

Section 2 of this paper gives a brief description of the basic characteristics of software process in SMEs. Section 3 presents the gradual approach and the OWPL model. Section 4 gives feed-back on practical OWPL experience by means of two case studies and section 5 contains concluding remarks.

2. Software Process in the Small

The *micro-evaluation* has been used since 1998 till now in Belgium and Canada in more than 60 companies like small IT companies, IT services in other businesses, public administrations. Results collected from these assessments have been used to profile SME and VSE [14].

One of the major characteristics of small software structures is the lack of resources assigned to the software process in general and to quality tasks in particular. Indeed, small software structures have by definition small teams, and people are absorbed by pressing matters related to confined deadlines which generally correspond to production tasks.

As a consequence of this lack of resources, the software process of SMEs usually ranks at a very low level according to standard software process maturity scales. However, a great number of small structures are remarkably successful and show obvious technical competence in their specific technical domain. In fact, the global weakness of their software process is to be considered under a more refined viewpoint. On the one hand, they are indeed highly dependent on projects, actors and/or technical capabilities in a very specific domain. On the other hand, their software process presents a great variance in quality: some practices can be very mature while others are extremely poor or even nonexistent. This allows us to claim that their software process can quickly and substantially be improved by means of a well-disciplined and well-targeted approach that does not necessarily require disproportional resources investment. This statement underlies the gradual SPI approach presented in Section 3.

Despite the great diversity in the quality of the different software processes within the same company

or in different SMEs, some characteristics can be observed recurrently:

- The software lifecycle is often very simplified. Generally a definition of a few main phases (e.g., analysis – implementation – maintenance) hardly exists and there can even be just one big phase. Development and testing are usually the two most important ones. Whilst testing is usually considered as the most important phase beside development, it is often shortened to meet deadlines or face lack of resources.
- Processes within the same organisation can present very unequal quality levels and very good processes can be combined with very low-level ones. In this case, the high quality level practices are often customer-supplier relationship practices, legally or contractually imposed, and that are most of the time considered as a burden rather than an asset by the organisation itself.
- In general, control procedures are too poorly formalised. In view of the lack of control, motivation to respect new procedures can quickly disappear. This is mainly a cultural issue but its impact is rather important at SMEs level. Lack of control can be considered as the basic cause behind the uneven software process quality.
- Concerning project management and planning practices, no specific features could be observed in most SMEs. Some of them have a very good project management process whilst others don't. But practices can be very uneven within the same organisation depending on the project, the client, the project manager or the development team concerned.
- With regard to training and human resources practices, one can notice that resources devoted to such practices are usually very limited due to the severe budget constrains in general. When training activities are carried out, this is often done to respond to punctual technical deficiencies. It is considered as an “emergency exit” in case the team members cannot perform effective self-training. There is no evidence of prospective training.
- Customers are offered very unequal level of service depending on their origin. Internal customers usually receive a much poorer service

than external ones for whom more rigorous practices are carried out.

- Small structures are project-minded: their processes are rarely driven by a long-term strategy. Consequently, learning and knowledge management practices can rarely be observed. Notice that this is frequent in larger structures as well. However, there is more evidence of this characteristic in SMEs.
- Because of their comparative size, SMEs are not able to impose their methodological approach, even if it is a very good one. They have to follow the methodological guidelines of their (larger) customers. As a consequence, flexibility becomes a vital concern.
- Internal communication is usually very informal. This characteristic can be viewed as an advantage in comparison with bigger structures where bureaucratic communication can be slow and inefficient. However, a minimum of formalisation is necessary to ensure a good level of quality of the software process.

Usually there is no risk management. This is related to the short-term view reported above. In practice the risk taken can be extremely high, though uncalculated. This can explain why some small structures show the most spectacular success.

3. The OWPL Approach

3.1. The Gradual Approach

The overall objective of our approach was to draw 'small structures' attention to SPI matters. Then we try to help them bring their software practices to a more homogenous level in order to start a continuous improvement process. This objective brings us to propose a gradual approach (see Figure 1 below) based on a three-stage software process improvement framework.

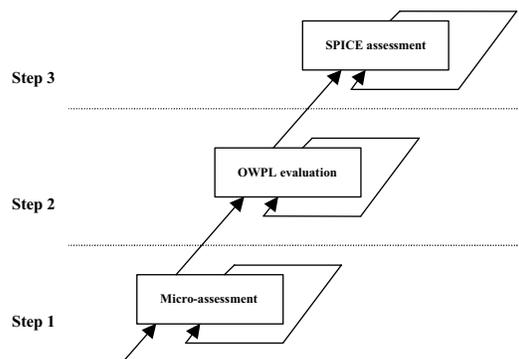


Figure 1 The Gradual Approach

At the first stage, a very simplified questionnaire called the *micro-evaluation* is used to collect information about the current software practices in small structures and to make people sensitive to the importance of software quality aspects. This questionnaire covers six key axes selected on basis of former experience with SME evaluation as the most pertinent and the most prior to the targeted organisations. These axes are Quality assurance, Customers management, Subcontractors management, Project management, Product management, and Training and Human Resources management. The *Micro-evaluation* involves one person in the evaluated organisation. This person must either have sufficient knowledge of software quality matters or already be in charge of software quality.

The information collected is then used as a starting point to determine the goals of a more accurate evaluation according to the OWPL model which has been developed on the same bases [10][15].

Bigger companies with a medium/high quality level are eventually invited to undertake an ISO/IEC15504 or a CMMI evaluation if this appears appropriate.

3.2. The OWPL Model

The OWPL model has been designed with respect to the particular context of small businesses so that it could help them to improve their software practices accordingly.

This model is based on the hypothesis that a key issue of success in any company lies on well-defined goals, which should be structured on a hierarchical way depending on their level of operability (strategic vs. operational). This means that the company is the one

that really defines its processes and the related goals according to the business strategy.

The structure of OWPL model involves processes, practices and success factors (see Figure 2 below). It defines 10 processes (requirements management, project planning, project tracking and oversight, development, documentation, testing, configuration management, subcontractors management, quality management, and experience capitalisation process), each decomposed into a number of practices (from 3 to 12). It is also supported by success factors. Each of the above processes is assigned a general goal in accordance with the organisation's defined objectives. It involves a number of practices and is supported by a number of success factors. Each practice is defined by its goal, its inputs and outputs, the resources assigned to support it and its weight. This last attribute is an indicator of the importance of the practice for the whole process improvement.

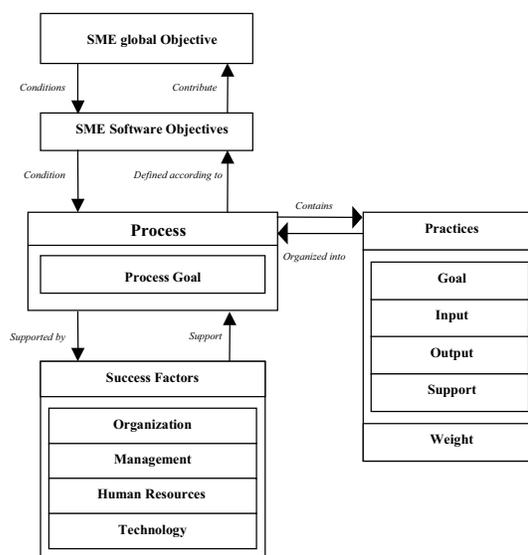


Figure 2 Structure of OWPL

Success factors are general requirements related to the environment of the process. Those factors determine the processes effective success. They correspond in fact to CMM Common Features, or to SPICE Attributes and include organisational, management, technical and human resources factors.

A detailed description of the OWPL model can be found in [10][15].

4. Case Studies

4.1 Context

The *Micro-evaluation* model has been tested on a significant number of medium, small and very small structures (from 4 to 50 software developers). It has been used for two rounds of evaluations with at least a six month delay between each round. The OWPL model has also been experimented on a number of cases where a *micro-evaluation* had been previously implemented.

This significant experience allows us to draw some conclusions on the relevance of our approach. In order to illustrate those conclusions, we will make reference to two representative cases.

4.1.1 Case 1

Case_1 is a software services company. Founded in the nineties *Case_1* grew quickly and it employs now about thirty people for the design of database applications. The quality approach in this company began in 1998 with a CMM-based software process improvement project. The main consequence of this has been an awareness of quality concerns. But in order to assess the quality level of the software practices and to get quick results, the company CEO decided in 1999 to do an OWPL *micro-evaluation*. This first step highlighted the main strengths and weaknesses (see figure 3). This has been helpful to concentrate subsequent efforts on some critical processes, namely the requirements management, the project planning, the configuration management and the project tracking and oversight process. Conscious that the *micro-evaluation* approach is subjective as only one person is interviewed and eager to go further in practices improvement, *Case_1* leader asked for an OWPL-assessment. This evaluation, based on the *micro-evaluation* conclusions, goes further in details and offers a more complete analysis. Strengths and weaknesses of every evaluated process are highlighted, short and medium term recommendations are proposed taking the specific level of the different success factors into account. This helped identify deep deficiencies at management level of *Case_1*.

An improvement plan has been developed on this basis. It mainly addresses the different success factors (on the short term) and then some software practices (on the mid term). The company's organisation has been brought in question and all the roles and responsibilities formally redefined. Then a *technology cell* has been created to analyse technological risks of every new project before handing-over a commercial offer.

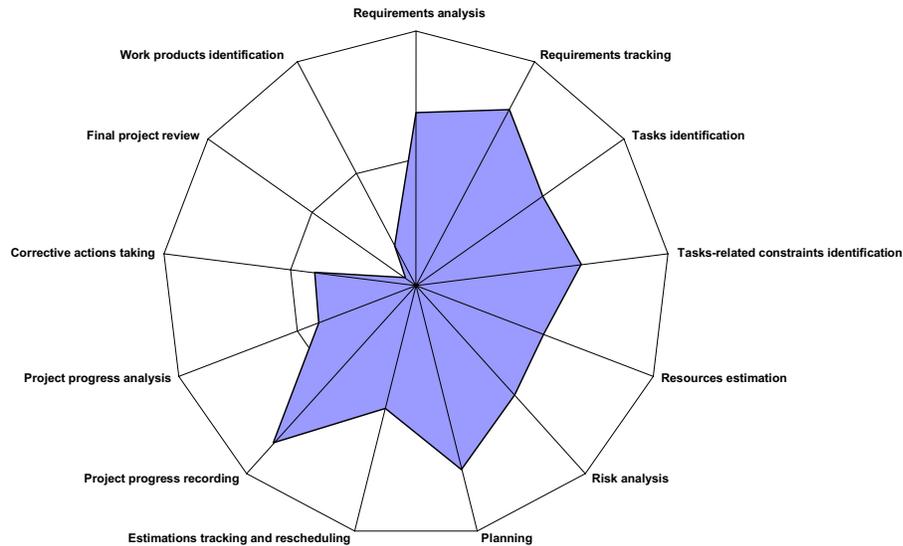


Figure 3 Case_1 - Practices analysis

Twelve month later, 25% reduction of the projects losses could be noticed. However, market pressure and inadequate commercial policy have progressively pushed *Case_1* back to its previous less effective practices, and project losses began increasing again. ... and they decided to resume the software quality approach..

4.1.2 Case 2

Case_2 is a small company in software services which employs about twenty persons. Aware of quality matters due to the Y2K issue, the company asked for a *micro-evaluation* in order to know its main processes strengths and weaknesses. Three months later they decided to go further and asked for an OWPL evaluation of the critical processes raised by the *micro-evaluation*. This OWPL evaluation focused on three low level processes: requirements management, project planning, and project tracking and oversight.

The combined analysis of the success factors on the one hand, and of the software practices on the other hand has lead the evaluation team to draw the appropriate conclusions:

- The software development practices were generally properly implemented though some improvements of those practices could still be implemented ;
- The project management practices were suffering from lack of formalisation and standardisation so

that customers could no expect even quality of service ;

- The members of the development teams on the one hand and of the direction on the other hand, had to cope with dramatic communication problems between those two groups.

As some of these aspects had already been highlighted thanks to the *micro-evaluation*, it has been possible to take them into account and prevent those communication problems to hinder the evaluation process. A key element for the success of this evaluation has been the organisation of an information session prior to the evaluation, during which every member of the company had the opportunity to query freely (CEO had been expressly asked not to attend this meeting) the evaluation team on any aspect of the improvement process.

As a result team members motivation was perceptible and they spontaneously decided (with CEO blessing) to set up a SEPG to start an improvement action and implement the recommendations of the OWPL evaluation.

Everyone was really surprised of this enthusiasm and it appeared afterwards that two previous audits had failed coming into actions.

4.2 Case Analysis

Case_1 shows the major difficulties of SPI in SMEs. Experience in SMEs shows that a punctual software process improvement is not appropriate. In fact, any improvement approach inevitably introduces new practices and, at least, two elements are able to act as potential brake. The first one is the resistance to change which should be taken into account as “*processes are like habits: hard to establish and even harder to break*” [11]. The second one is the short-term management which prevails in SMEs whereas improvement approach success necessitates long-term support.

One-shot improvement actions are usually bound to fail. SPI may be compared to an old steam engine: it needs progressive but continuous contribution in order to work properly.

Case_2 highlights the necessity to have a global approach when working with small structures. Restricting the evaluation scope to the development practices would probably have permitted to identify improvement opportunities to be implemented by the team members, but the concern was mainly at management level and the communication problem had to be tackled first.

Small structures need TQM-like approach as every single aspect of the organization is closely connected to the others.

4.3. Adequacy of the Gradual Approach

We have successfully implemented our approach on several cases of different sizes and types of activity and we can state that it is indeed particularly adapted to the Software Process in-the-small as it gives an adequate answer to the particularities of software process in small structures:

Simplified and low cost evaluation approach: The first obvious merit of a simplified approach lies in its low cost which makes it affordable for small structures with few resources. In practice, this appears to be the only way to counter the natural trend that gives absolute priority to development tasks and indefinitely postpones process improvement or quality tasks in general.

Accurate insight into assessed processes: The small size of the concerned organisations allows in-depth analysis of the practices of all the teams concerned with the selected processes. Two people from each team - a manager and an operational - take part to the evaluation. We strongly recommend not to perform too

many processes at a time in order to implement moderate but continuous improvement.

Gradual approach based on continual assessment-improvement cycle: The fact that the approach is gradual allows different tunings to be done. For example, evaluation can be tuned with respect to the available resources, so that very small structures can use only the micro assessment framework by iterating the evaluations in a cyclic way. Other structures can start their SPI cycle by a *micro-evaluation* to identify weak processes, then go on with a first OWPL assessment, and eventually continue other cycles with the OWPL model in order to improve other processes.

Models using simplified vocabulary: The simplified vocabulary of the models renders them usable by different actors who are not always software engineers or quality specialists. The number of actors actually involved in software process improvement is increasing. The improvement process concerns management people as well as technicians at different levels. The improvement will be more likely effective if actors are already sensitised and actively participate in the improvement process.

- Improvement based targeted recommendations: The OWPL evaluation report containing the summary of all the collected information, the data analysis and recommendations for an appropriate improvement plan follows these important rules :

Though it is recommended to define and keep in mind long-term goals, it is important to take into account the fact that most SMEs have a day-to-day management policy (or at least a very short-term one). It allows them to maintain a certain flexibility that is essential to face technology and market changes as they can not influence them.

- The recommendations made must help them reinforce their flexibility thanks to a short-term, cheap and easy-to-implement action plan. Tools and methods recommended should not be a burden.
- The defined processes and procedures must prevent excessive bureaucratic tasks while insuring adequate formalisation level.
- Improvement actions are prioritised in connection with the SMEs' identified goals.
- The software lifecycle must be defined and early phases should not be neglected as they are the foundations of the projects.

5. Conclusions

The different experiences show that our first target selection criterion (i.e. the organization size) was not the most relevant one. In fact, the software maturity level appears to be even more significant and medium size companies with low maturity level have "software-in-the-small" practices. This strengthens the pertinence of this gradual approach as a means to tackle the problem.

Software process in small software structures can be characterized by several weaknesses that would rank it at a very low maturity level on standard software process models like CMM or SPICE. However, a deeper view of these weaknesses shows several strengths as well. Deep knowledge of software processes in SMEs and experience in evaluating these processes have allowed us to elaborate a refined characterization of the software process in-the-small in general. Based of this characterization, a software process improvement framework particularly adapted to small structures has been developed. The framework of SPI in-the-small is simple and cheap; it is gradual and uses a simplified vocabulary. The framework's main objective is to provide well-targeted recommendations for software process improvement allowing SMEs to enhance their process in a realistic and affordable way.

Our SPI experience with a number of regional SMEs implementing the gradual framework has shown the feasibility of such an approach. The *micro-evaluation* has proved its effectiveness in producing a pertinent diagnostic at a very low cost and the feedback we have received from the assessed companies is very satisfactory.

The OWPL model has convinced us of the absolute necessity of using the good methodology in the right place as each case is particular, with its own context. Problems and their solutions are particular to each SME, so that tailored support is needed.

6. References

- [1] El Emam K., Drouin J.-N., Melo W., "SPICE The Theory and Practice of Software Process Improvement and Capability Determination", IEEE Computer Society, USA, 1998.
- [2] Paulk M.C. et al., "The Capability Maturity Model – Guidelines for Improving the Software Process", Addison-Wesley, England, 1995.
- [3] "CMMIsm for Systems Engineering/Software Engineering/Integrated Product and Process Development", Version 1.02, CMMIsm - SE/SW/IPPD, V1, 02CMU/SEI-2000-TR-031.
- [4] Habra N., Niyitugabira E., Lamblin A.C. and Renault A., "Software Process Improvement in Small Organizations Using Gradual Evaluation Schema", in *Proceedings of the International Conference on Product Focused Software Process Improvement*, VTT, Oulu, Finland, June 1999.
- [5] Renault A., "Micro-évaluation des Pratiques Logicielles - Etude de cas", Institut d'Informatique, University of Namur, Belgium, 1999, in French.
- [6] Laryd A. and Orci T., "Dynamic CMM for Small Organizations" To appear in the *Proceedings of ASSE2000*, 2000.
- [7] Johnson D.L. and Brodman J.G., "Tailoring the CMM for Small Businesses, Small Organizations, and Small Projets", IEEE Software Process Newsletter N° 8, Winter 1997.
- [8] Olofsson S., "Tailoring P-CMM for Small Companies", Technical Report. UMINF-00.1, Department of Computing Science, Umeå University, 2000.
- [9] O Hara F., "Irish Experiences with Software Process Improvement", in *Proceedings of the European Software Process Improvement Conference 2001*, Limerick, Ireland, October, 2001.
- [10] Habra N., Niyitugabira E. and Renault A., "Modèle OWPL : Evaluation et Amélioration des Pratiques Logicielles dans les PME Wallonnes", Technical Report 1/99, Institut d'Informatique, University of Namur, Belgium, 1999, in French. (<http://www.software-quality.be/>)
- [11] Humphrey W., "A Discipline for Software Engineering", Addison-Wesley, England, 1995.
- [12] Demirors O. and Demirors E., "Software Process Improvement in a Small Organization : Difficulties and Suggestions", *Proceedings of the EWSPT06, Sixth European Workshop on Software Process Technology*, Surrey, UK, September 1998.
- [13] Zahran S., "Software Process Improvement - Practical Guidelines for Business Success", Addison-Wesley, England 1998.

- [14] Cl. Y. Laporte, J-M. Desharnais, A. Renault, N. Habra, M-M. Abouelfattah, J-Cl. Bamba, "Initiating Software Process Improvement in Small Enterprises: Experiments with CETIC's Micro-Evaluation Framework", International Conference on Software Development (SWDC-REK 2005) Reykjavik, Iceland, 2005.
- [15] Renault A., Habra N., Niyitugabira E., Alexandre S. and Deprez J.-C "OWPL Model: Software Process Improvement for VSE & SME", Technical Report, CETIC-FUNDP,, Belgium, 2006, (http://www.cetic.be/IMG/pdf/OWPL-Model_V1-2-2-EN-V22.pdf)
- [16] Laporte, C.Y., April, A. and Renault, A., Applying ISO/IEC Software Engineering Standards in Small Settings: Historical Perspectives and Initial Achievements. To be published in Proceedings of SPICE2006, Luxemburg, 2006.