Rory V. O’Connor
Antanas Mitasiunas
Margaret Ross, Eds.

SPETP 2015

1st International Workshop on Software Process Education, Training and Professionalism

Gothenburg, Sweden

June 15, 2015

Preface

These Proceedings contain the papers accepted for publication and presentation at the first 1st International Workshop on Software Process Education, Training and Professionalism (SPETP 2015) held in conjunction with the 15th International Conference on Software Process Improvement and Capability dEtermination (SPICE 2015), Gothenburg, Sweden, during June 15-17, 2015.

During the 14th International Conference on Software Process Improvement and Capability dEtermination (SPICE 2014) held in Vilnius, Lithuania, at a post conference dinner, a group of key individuals from education and industry started to discuss the challenges faced for software process education, training and professionalism, especially with the background of the new modes of learning and teaching in higher education.

Further discussions held post conference with key players in the relevant professional and personal certification fields led to a consensus that it is time for the industry to rise to the new challenges and set out in a manifesto a common vision for educators and trainers together with a set of recommendations to address the challenges faced. It was therefore agreed co-located the 1st International Workshop on Software Process Education, Training and Professionalism with the 15th International Conference on Software Process Improvement and Capability dEtermination.

This workshop focused on the new challenges for and best practices in software process education, training and professionalism. The foundation for learning of software process should be part of a university or college education however software process is often treated as 'add one' module to the core curriculum. In a professional context, whilst there have been a number of initiatives focused on the certification related to the software process professional these have had little success for numerous reasons.

Cooperation in education between industry, academia and professional bodies is paramount, together with the recognition of how the education world is changing and how education is resourced, delivered (with online and open learning) and taken up. Over the next 10 years on-line learning is projected to grow fifteen fold, accounting for 30% of all education provision, according to the recent report to the European Commission on New modes of learning and teaching in higher education.

It is a great pleasure to see the varied contributions to this 1st International Workshop on Software Process Education, Training and Professionalism and we hope that our joint dedication, passion and innovation will lead to success for the profession through the publication of the manifesto as a key outcome from the workshop.

On behalf of the SPETP 2015 conference Organizing Committee, we would like to thank all participants. Firstly all the authors, whose quality work is the essence of the conference, and the members of the Program Committee, who helped us with their expertise and diligence in reviewing all of the submissions. As we all know, organizing a conference requires the effort of many individuals. We wish to thank also all the members of our Organizing Committee, whose work and commitment were invaluable.

June 2015

Rory V. O'Connor

Antanas Mitasiunas

Margaret Ross

Organization

General Chair
Alec Dorling, Impronova AB, Sweden

Workshop Chair
Rory V. O'Connor, Dublin City University, Ireland

Workshop Co-Chairs
Antanas Mitasiunas, Vilnius University, Lithuania
Margaret Ross, Southampton Solent University, UK

Workshop Moderator
Jørn Johansen, Whitebox, Denmark

Proceedings Chair
Rory V. O'Connor, Dublin City University, Ireland

Program Committee
Terry Rout, Griffith, Australia
Aileen Cater-Steel, Australia
Gerhard Chroust, Austria
François Coallier, Canada
Onur Demirors, Turkey
Dennis Goldenson, USA
Christiane Gresse von Wangenheim, Brazil
Ho-Won Jung, South Korea
Antonia Mas Pichaco, Spain
Fergal McCaffery, Ireland
Tom McBride, Australia
Clenio Salviano, Brazil
Fritz Stallinger, Austria
Saulius Ragaisos, Lithuania
Taz Daughtry, USA
Vincent Ribaud, France
Claude Y. Laporte, Canada
Rory V. O'Connor, Ireland
Patricia McQuaid, USA
Juan Garbajosa, Spain
Jan Pries-Heje, Denmark
Geoff Staples, UK
Miroslaw Staron, Sweden
Barbara Gallina, Sweden
Ricardo Colomo Palacios, Norway
Tomas Schweigert Germany
Alec Dorling, UK

Table of Contents

Process Improvement - Barriers and Opportunities for Teaching and Training

Margaret Ross
Southampton Solent University
Faculty of Maritime and Technology,
UK
margaret.ross@solent.ac.uk

Abstract

Barriers and opportunities associated with professionals are considered. Practical possibilities for increasing the understanding and implementation of process improvement are discussed, including the use of webinars, possibly with open badges to record and encourage participation. The use of MOOCs, potentially leading to on-line assessed qualifications, could increase the number of practitioners with the relevant knowledge, particularly in more remote regions. Changes to SFIAplus could enhance awareness of process improvement, and so encourage employers to authorise relevant training.

The lack of relevant knowledge and experience of teachers and lecturers is considered, together with the problems of pressures by other topics on academic courses. The actions that could be undertaken to promote and assist the teaching of process improvement in colleges and universities range from provision of suitable case studies to the inclusion of process improvement within the accreditation of courses. The opportunities associated with the new higher apprenticeships could provide potential practitioners with process improvement skills for the future. Changes in the requirements by professional bodies for syllabus content for accredited courses, aligned with enhanced SFIAplus, could increase awareness of process improvement. The syllabus could be aligned with professional courses, such as by the ECQA course, so students could obtain the professional qualification as well as their degree. Taking the long term approach of ten years plus, the relevance of quality and process improvement could be introduced at the appropriate stages into the new schools computing curriculum, started in 2014 in the UK, to raise awareness of the need for process improvement to the future workforce.

1. Introduction

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

The EC report on "New Modes of Learning and Teaching in Higher Education" identified that changes in technology provide an "enormous potential for widening access to Higher Education and increasing the diversity of the student population" (EC report, 2014, p.10, paragraph 3). It reported, from the UNESCO investigation, that the current estimated number of 100 million students worldwide, in Higher Education, is expected to increase to over 250 million by 2025 (EC report, 2014, p.14, paragraph 3). Many of these could be from outside the EU. They might be physically attending courses, or virtually, using e-learning techniques. These students might be already in full-time employment, returners after a career break, or undertaking lifelong learning following retirement.

It is recognised that quality and process improvement is desirable to produce and maintain high quality systems. Unfortunately there are barriers to those wishing to acquire the necessary skills, particularly for process improvement, whether student or professional. However, with the increased use of new ideas including developments in technology, there are now opportunities that could be made available to professionals, to students and their lecturers

2 Barriers to Students

The benefit of appropriate processes to achieve quality and process improvement is more visible with projects involving teams. The major projects for postgraduates and final year undergraduate courses in the UK are normally individual projects. The group projects, usually involving a maximum of six students, often achieve small outputs and are possibly in the first or second year of the undergraduate course. The students, with this limited experience, do not realise the practical benefits of quality, process improvement or even of documentation.

Most syllabuses are already very full on these courses, with constant pressure to introduce additional topics. Dedicated units on quality and process improvement are not usual. These topics are more likely to be included as part of a series of other units, including design and programming. Some students are more interested in the latter, and feel issues such as process improvement are not relevant or of interest to them.

The lecturers and teachers in universities and colleges often have had no practical experience, or even adequate training, in process improvement. The cost of relevant courses is often seen to be too high for academic budgets. There is limited access to material, especially case studies and case histories, which can often make a topic more interesting and understandable for the students. The cost of standards, even with academic discounts, is viewed by academic libraries, with limited resources, as too high. The websites of the standards organisations, are designed for practitioners rather than being suitable for academics or their students, making the concept of process improvement less attractive to them.

There is pressure in many universities on the type of research that would result in obtaining a high research rating leading to larger allocations of money from the Government to those universities. Many lecturers are on short-term contracts, depending on producing these highly rated publications and obtaining research contracts. These are not often related to promoting quality and process improvement, particularly in SMEs. Similarly, a lecturer, on a full-time contract, often has little time for spending with SMEs to encourage process improvement. This can be addressed by Government funding to "buy" some of the lecturer's teaching hours to work with SMEs. Publications relating to improving process improvement of SMEs are not usually highly rated in the competitive academic research community. Similarly the development of case studies, based on process improvement in SMEs, useful to assist in the teaching and learning of students, would not often be viewed as a high research priority. This attitude could discourage young lecturers, needing to consider their future careers, from following these extremely useful areas of investigation.

These lecturers or students, particularly PhD students, that investigate the effectiveness of process improvement, are encouraged to publish in journals if possible or at conferences that are mainly attended by academics. There is often active pressure not to write articles for trade papers or magazines that would be read by SMEs. This attitude results in very poor dissemination of the outcomes of the research into the wider community, and in particular, to SMEs. (Georgiadou et al, 2014)

The trend in recent years away from the conventional day release to full-time study has reduced the contact between university lecturers and local companies. To ensure that the syllabuses for these university courses address issues of concern to employers, and conversely, to promote their current or future employees with a relevant understanding and knowledge of topics, such as quality and process improvement, there needs to be an effective dialogue between the universities and the employers.

Some universities are designing very short, self-contained units, within their degree courses. These can be undertaken as a form of "short course" for industry with the university credits allocated to these "short courses ". These could assist in increasing the links between academia and industry, and address industry's need to provide continuous professional development and lifelong learning (EC report, page 10, paragraph 5).

3 Barriers to the Professionals

There are limited numbers of professionals with adequate experience and knowledge of process improvement, to be able to influence the majority of organisations. In many cases, there is little opportunity of gaining practical experience, especially if they are employed by SMEs (Small to Medium-sized Enterprises). As the number of people in SMEs that are employed directly in quality and process improvement roles is very limited, there is little opportunity to gain initial experience for those wishing to move into these areas and limited opportunities to "learn on the job".

Individuals could address these problems by attending courses, but this requires both the available time and money, which would not always be supported by their employer. The cost of access to standards can also seem be a problem to them. The timing and location of these courses are not always compatible with their working schedule. This is particularly a problem for those working for organisations, including outsourcing companies, situated outside Europe.

SMEs and individual practitioners, particularly those involved with small web applications and the production of Apps, are often concerned more with the fast development rather than process improvement. For the development of Apps, there are no internationally accepted standards. The concept of global enforcement of such standards might be resented by some members of the Open Source community. Some organisations, such as Apple and Google, have quality guidelines, but those producing Apps have many other alternative potential "marketing locations" for their Apps. Although there are standards for quality and process improvement in this fast changing global world of software and hardware, these standards need to be constantly "refreshed" and cannot be enforced, especially in the communities of individuals and SMEs.

4 Opportunities for Professionals

The use of courses on aspects of quality and process improvement that can be taken by blended learning approach, where there is a combination of a remote study, video-conferencing and attendance, can provide an opportunity to gain required knowledge and skills. Possibility the attendance is arranged on occasional weekends or in blocks, so allowing attendees to travel long distances.

The author has been involved with such a Master's course for Six Sigma, where students flew in every two months from other parts of the UK and Europe (Protheroe et al, 2008). At the start of each of the Six Sigma units, the students were given the full learning material and the individual assignment, which in all cases was work related. The students, having studied the material for the first six weeks, then attended a weekend session, situated halfway through the unit. This involved a mixture of group workshops and individual discussions, similar to that of a modified "flipped classroom", allowing each student to progress at their own speed and direction (Almpanis et al, 2010). The students then continued to work alone, but with video and audio conferencing support, for the further six weeks, when they submitted their assignment (Almpanis et al, 2011). The students were also able to achieve the Black Belt for Six Sigma.

The author has also been responsible for the one, two and three day courses run by Tom Gilb Hon FBCS on different aspects of quality, held in London and other locations, organised through the BCS Quality Specialist Group which provided free training for BCS members, especially those that were currently under employed, as consultants or professionals. Training courses, leading to qualifications such as the ECQA SPI (Software, Systems and Service improvement), could increase the skills in the area of process improvement (ECQA, ND).

The identification of a suitable MOOCs (Massive Open Online Courses) covering part of the relevant skills, could be used to increase the knowledge of the professionals. Many of these MOOCs are free, or at a low cost (Dewar et al, 2014). Details of these MOOCs could be made available, say on relevant websites, such as those of SPICE and the professional bodies such as BCS Quality Specialist Group. Assessment of the relevant skills could be made at the Foundation Level, by online multiple choice questions, which could be organised by training organisations or professional bodies. These could be similar to those of Prince Foundation or the BCS Agile Foundation and other certificates (Agile, ND). It might be possible to link to the online assessment qualifications to the European Credit Transfer Scheme (ECTS) system.

The use of webinars, which are often free, could be used to increase the awareness of process improvement and impart some of the relevant skills. Examples of these are run at no cost by the BCS GreenIT Specialist Group. The presenters of the webinar can be located in different countries, as can the participants and who are also able to access the webinar after the event. Presenters or active participation at these webinars could be recognised for the individual by the collection of Open Badges. These normally involve no or little cost to produce, and can be available for easy, quick and free distribution regardless of country via the Internet.

The professional bodies could be encouraged to organise webinars and also physical meetings, by providing speakers with process improvement experience. These could be recorded and made freely available possibly by YouTube, to provide a useful resource for practitioners to update their skills and for opportunities for trainers and teachers to assist with the up-skilling of those currently in process improvement roles or aspiring to those roles, by utilising these online approaches.

5. Opportunities for Students and their Teachers

The lack of relevant experience of teachers and lecturers, which would enable them to inspire their students about quality and process improvement, is a major problem. This could be addressed by issues discussed in Section 4, such as the use of blended learning courses, MOOCs and webinars.

Professionals and relevant organisations could be asked to assist with the production of suitable case studies, case histories and short YouTube videos lasting possibly a maximum of five or ten minutes. These resources could easily be included in the relevant classes. Students could be motivated by the use of Open Badges. As they gained the relevant process improvement skills, they could be awarded the appropriate e-badge, which are now used by various organisations including some universities and schools as a means of motivation and as an on-line record of CPD (Continuous Professional Development).

Recorded lecture systems such as Panopto can be used to automatically capture the lecturer's explanations, attached to each Power Point slides for use later, possibly remotely. The author currently uses Panopto in short segments with PowerPoint and the use of interactive boards (Griffin and Ross, 2015). These segments are designed for final year students in particular those that have language difficulties or have missed sessions. It allows the students to go directly for further explanations of a particular slide. In addition, the lecturers are provided with a record of which students accessed the system, when it was used and more importantly for which elements of the recording the students found it desirable to re-hear the full commentary. Online capture could be used as part of a MOOC or to develop a "flipped" classroom, as described in Section 4.

The various appropriate Specialist Groups of the professional bodies, including the BCS Quality Specialist

Group and the BCS e-learning Specialist Group, could hold events, both physical and by webinars to increase knowledge of process improvement. The appropriate groups could organise competitions, aimed mainly at students to raise awareness of process improvement.

6. Governments & Professional Bodies

As there is always pressure to include new topics on courses, the professional bodies, such as the BCS, could specify that process improvement should be included in any degree course to be accredited by that body. The governments, through their financial power, could play a major role in encouraging the professional bodies and the universities and colleges to give a higher priority to relevant courses and in particular to quality and process improvement.

The Government and professional bodies influence can also be applied to schools, to control the curriculum. An example of this is the UK computing syllabus, started in September 2014 in the first year of primary schools. These children, as they progress through their schooling, year by year, will follow a new computing curriculum, gradually increasing in depth and breadth, potentially until they reach the age of sixteen, to try to address in the future, the shortage of IT professionals. Interest in this approach has been shown in a number of countries including Denmark, Holland and Japan. By aiming to influence this new curriculum to include quality and process improvement, especially for those pupils in the latter school years, this would influence the potential workforce of the future. The active support by Government and professional bodies to include quality and process improvement in colleges and universities in their units, should be of benefit to future professionals and their employers.

By encouraging these units in these courses to be aligned with the relevant professional syllabus, students could be given the opportunity, in addition to attaining their degrees, to achieve with professional qualifications such as those of the ECQA. This concept has been used successfully over many years to improve students' employability. Networking students can achieve Cisco qualifications while undertaking their computing degrees (Udall and Ross, 2012a). The author implemented a similar arrangement, by aligning the syllabus of the BCS Structured Systems Analysis and Design qualifications with the appropriate second year unit at Southampton Solent University. Students completed the university's unit assessments and then, on completion of that unit, could take the BCS examination, so gaining both academic and professional qualifications (Uhomoibhi and Ross, 2013). The author also ran intensive two week courses for professionals for this BCS SSADM qualification.

Employers could be encouraged to specify in job advertisements, qualifications which have a strong commitment to quality and process improvement, as they already do within Six Sigma, ITIL and Prince2. These could raise the profile of potential employees to the importance of process improvement. The professional bodies could offer, at no cost, to re-publicise these job advertisements on their websites providing they promote process improvement, professional standards or professional membership in the job descriptions.

Influence could be applied by government and professional bodies on the content and organisation of the new higher apprenticeships. These have recently been started in the UK, where a student, instead of attending a full-time university degree course, would be employed by an organisation while at the same time, would study part-time over a period of three or four years for a university degree or part of a university degree. There are special degree courses being designed by some universities for particular employers that would provide a large number of high-level apprentices in a particular discipline. In these cases many of the university degree units would align with the requirements of that industry or organisation.

Other universities, that are expecting the students on apprenticeships to come from a number of SMEs, are organising degree courses on a day release basis. These might possibly have some units run in conjunction with their full-time degree courses. Another structure that has been implemented is for higher apprenticeship employees to study by distance learning on relevant Open University degree units. These apprentices would have time allowed each week for their university studies and have their fees, as with the other models, paid by their employers. These students should complete half of a normal degree course during their three-year apprenticeship.

The SFIAplus industry structure model could be modified to enhance the roles associated with quality and process improvement (SFIAplus, ND). These roles, for the various computer related professions, identify the necessary skills, possible qualifications and possible activities. These are proposed for different levels, with the career progression from starting work at say eighteen years of age to possibly becoming a senior manager of a major organisation. By encouraging process improvement to become a specified role, this would raise the profile as well as clarify the knowledge and experience required at different levels. Within existing roles of SFIAplus, quality and process improvement could be specifically included, as they are relevant to all roles. As this SPIAplus model is used by many organisations, as well as individuals, to plan career progression, it would bring the need for appreciation and implementation of quality and process improvement into the various levels and roles, regardless of the sector.

By aligning the degree courses with the modified SFIAplus which could include more emphasis on quality and process improvement, the skills of students could be more easily identified by potential employers (Udall and Ross, 2012b). This could improve the relationship between universities and colleges with their local organisations. To provide external checking of the level of knowledge of these areas, the professional bodies could audit this, in addition to providing a more general audit of their potentially accredited courses. Various multiple choice online qualifications could be designed, possibly administered by the professional organisations, such as the BCS, to assure the knowledge on quality and process improvement aligns with the relevant different levels of SFIAplus. These could be taken by both students and professionals.

7. Conclusions

To assist the lecturers to inspire their students, in addition to helping with suitable material, opportunities could be provided for lecturers and teachers to gain real life experience by shadowing process improvement professionals, possibly with Certification Bodies, subject to their clients' agreement, and in organisations with quality and process improvement sections. This would enable the lecturers to introduce some real world, even though limited, experience to their discussions with students.

Competitions could be organised, such as part the BCS Quality Specialist Group and e-learning Specialist Group, possibly related to promoting awareness of process improvement, but with the prizes being opportunities of work-experience for the winners to gain real life quality and process improvement knowledge first hand. The relevant organisations would need to be involved in the development, marketing and judging of such competitions, whether aimed at students, lecturers or open to professionals. The author is currently involved in such competitions, aimed at pupils of about school leaving age, and another competition aimed at their teachers, which has been designed in conjunction with the computing department of Hampshire County Council and the main prizes included relevant work experience.

By influencing the syllabus for these courses, and other degree courses, to include quality and process improvement, the future professionals, on entering the various Industries, could act as ambassadors for process improvement for the future.

References

[1] Agile ND, BCS Agile Foundation Professional Qualification, http://certifications.bcs.org/category/17577 (accessed 19/5/2015)

[2] Almpanis T, E Miller, M Ross, D Price, R James, 2011, Evaluation the Use of Web Conferencing Software to Enhance Flexible Curriculum Delivery, Proc IICE (Ireland International Conference in Education) 2011, Dublin, 2011

[3] Almpanis T, E Miller, M Ross, D Price, R James, 2010, Virtual Classrooms for Flexible Curriculum Delivery on MSc Six Sigma Course, Proc INSPIRE 2010, London, ISBN 978-0-9557300-8-8

[4] Dewar E, J Uhomoibhi, M Ross, D Hutty, 2014 MOOCs development and implementation: The challenges and prospects for higher education in emerging countries. Proc INSPIRE 2014, Southampton, ISBN 978-0-9926958-2-8

[5] ECQA, ND, http://www.ecqa.org/fileadmin/documents/professional_leaflets/ECQA-spi-manager-flyer.pdf (accessed 19/5/2015)

[6] EC report, 2014, Report to the European Commission on New Modes of Learning and Teaching in Higher Education, ISBN 978-92-79-39789-9, http://ec.europa.eu/education/library/reports/modernisation-universities_en.pdf, (accessed 10/5/2015)

[7] Georgiadou E, K Siakas, and M Ross, 2014 Innovation in Project Sustainability: The Need for Early Planned Dissemination and Exploitation, Proc 6th World Congress of Software Quality, London, 2014

[8] Griffin B, M Ross, 2015, Speak and be Heard Later: Use of Audio-Visual Support at Southampton Solent University, Proc INSPIRE 2015, ISBN 978-0-9926958-7-3

[9] Protheroe H, T Carrier, E Miller, J Rees, M Ross, 2008, Blended Learning for Six Sigma, Proc SQM 2008, Belfast, ISBN 978-1-906124-05-2

[10] SFIAplus, ND, SFIAplus Industry Structural Model, http://www.bcs.org/category/17784, (accessed 19/5/2015)

[11] Udall M, M Ross, 2012a, Assisting Employability for Undergraduates, Proc INSPIRE 2012, Tampere, Finland, ISBN 978-951-44-8901

[12] Udall M, M Ross, 2012b, SFIAplus in the Curriculum, SFIA in Education and Workplace Learning

conference, Milton Keynes, published BCS e-wic, Open University, Milton Keynes, 2012

[13] Uhomoibhi J, M Ross, 2013 Globalisation and e-Learning: Integrating University and Professional Qualifications for Employability and Lifelong Learning, ICEL 2013 (International Conference on e- learning), Cape Peninsula University of Technology, South Africa, 2013

A Systematic Mapping Study on Software Process Education

Alberto Heredia
Universidad Carlos III de Madrid
Computer Science Department
Leganés 28911, Spain
alberto.heredia@uc3m.es

Ricardo Colomo-Palacios
Østfold University College
Faculty of Computer Sciences
Halden 1783, Norway
ricardo.colomo-palacios@hiof.no

Antonio Amescua-Seco
Universidad Carlos III de Madrid
Computer Science Department
Leganés 28911, Spain
amescua@inf.uc3m.es

Abstract

Software professionals often face trouble when developing software products as it is a highly dynamic, knowledge-intensive complex process. The success of the software process heavily depends on the people involved, among other factors, making their education and training an interesting topic for research. The purpose of this study is to structure and characterize the state of the practice on software process education to help identify best practices and find new challenges. To do so, authors conducted a systematic mapping study to identify primary studies in the existing literature related to software process education. The analysis of results helps clarify the general characteristics of the software process education and training initiatives, the lessons learned in previous research, and the future works proposed by the authors in previous research on software process education.

1 Introduction

Modern societies increasingly depend on the services offered through computerized systems. The advent of smartphones, tablets, wearables and other intelligent devices makes that more and more products embed or take advantage of some piece of software. Unfortunately, software is a complex product, difficult to develop [FuNi14].

Software Engineering has the main goal of creating software products with quality, respecting time and budget constraints [Hump95]. To do so, the software development activity usually follows a software process, which can be defined as the coherent set of policies, organizational structures, technologies, procedures, and artifacts that are needed to conceive, develop, deploy, and maintain a software product [Fugg00], i.e., it describes the approach that is taken as software is engineered.

However, the controversial reports from The Standish Group continuously mention a low percentage of successful projects delivering software on time, on budget, and with required features and functions. Other forums, such as Risk Digest [Acmc15], constantly document numerous examples of software failures that could be harmful for the society, e.g., the accidental erasure of criminal records or the exposure of private data from online customer databases.

Many of these problems found in software products are unintentionally caused by people [KuFM13], as software in the end is developed by individuals and is largely dependent on human capital [CCGG09, CCMS14, CCSG13a, HeCG13]. Thus, it is worth researching how they are educated and trained on the process to follow for the development of a software product[CCSG13b, RoZS14].

Training software engineers in order for them to acquire the knowledge and skills required in professional practice depends on the stage of their careers. As an example, software engineering courses at the university usually consist of lectures along with a small software project [BaOH05], but software process is often treated as an additional module to the core curriculum. Trainings in an industry environment are, on the other hand, organized in a workshop style with theoretical and practical parts interwoven [KuFM13]. Yet it is not clear if this education and training –no matter the way it is provided– effectively prepares software process (improvement) practitioners as skilled and competent professionals for industrial life.

In fact, software engineering professionals are often unsatisfied with their level of preparation for the real-world when they start working in industry [Exte14]. Some authors point out the root of the problem lies in the way software process is typically taught at universities [AlUn14, BaOH05]; due to the time and scope constraints inherent in an academic setting, most

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

course projects leaves little room for experiencing the many facets of the software lifecycle [KoCM14].

Many authors have researched on how to make improvements in software process education and training to overcome this issue using different approaches. The first one lays on a specific subject that is needed but currently missing or not properly addressed [WaSB12]. Another approach aims at bringing the class project closer to a real-world one, for instance, by intentionally applying unexpected complications during the project [Daws00] or involving external organizations [ChCh11]. A third approach uses a simulated environment in conjunction with lectures and projects for enhancing the learning and understanding of complex themes [BaOH05]. Finally, the gamification of learning has emerged as a significant trend in recent years in an effort to make education more attractive by means of incorporating game mechanics and elements [PGBP15].

Regardless of the approach chosen, it is also important to consider how instructors intend their students to learn. The most traditional delivery method consists of a series of lectures and demonstrations in which the teacher presents a particular subject and directly instructs students. This method is often contrasted to experiential learning, which is based upon the premise that the best way to learn how to do something is by actually doing it [BaOH05]. Other methods center learning around an anchor such as a case study or a problem [BSHK90], foster a situated learning in which the learning environment is closer to reality [AnRS96], focus on the aptitude of students and tailor the learning environment to their needs [Yeh12], emphasize a lateral thinking that require students to take different perspectives [Bono09], or just focus on motivating students to learn [Kell87].

We thus need to further study how software engineers learn the software process. The objective of this paper is to structure and characterize the state of the practice on software process education. In consequence, the authors of this study conducted a systematic mapping study to identify, select, classify and analyze primary studies published in scientific journals. To the best of our knowledge, no systematic mapping study on software process education has been published yet.

The remainder of this paper proceeds as follows. Section 2 describes the method followed in this research work. Section 3 analyzes and discusses the results of the systematic mapping study. The paper concludes with the limitations of this research and concluding remarks.

2 Research Method

The purpose of this study is to structure and characterize the state of the practice on software process education, analyzing previous works published in the literature to provide an overview of the topic and to help discover potential gaps for future research. Thus, the main research question driving this study is:

What is the state of the practice of the education on software process?

Due to the breadth of the topic, a systematic mapping study [KiBP11] is used to identify and categorize all relevant research papers (referred to as primary studies) related to software process education. The study follows the guidelines provided by Petersen et al. [PFMM08]. The following sub-sections present the different stages of the mapping study: definition of research questions, conducting the search for primary studies, screening papers based on inclusion/exclusion criteria, classifying the papers, and data extraction and aggregation.

2.1 Research Questions

To answer the main research question driving this mapping study, the authors of this study stated the following specific research questions:

- RQ1. What are the general characteristics of the software process education and training initiatives?
- RQ2. What lessons did researchers learned from previous research on software process education?
- RQ3. What future works did authors propose in previous research on software process education?

The answer to RQ1 will help determine different aspects of the software process education such as which stage of software engineers' career does this education usually focus on, the educational methods that are typically followed, how this education is usually delivered, or which parts of the software process have not received much attention yet with regard to software process education. The aim of RQ2 is to identify best practices on the field. RQ3 gathers challenges identified in the field of software process education.

2.2 Search Strategy

The search strategy is key to ensure a good starting point for the identification of studies and ultimately for the actual outcome of the study. An extensive and broad set of primary studies was needed to answer the research questions. The most popular academic databases in the domain of software engineering were selected to be used in this systematic mapping to search for potentially relevant papers:

- ACM Digital Library (http://dl.acm.org)
- IEEE Xplore Digital Library (http://ieeexplore.ieee.org)
- ScienceDirect (http://www.sciencedirect.com)
- Springer Link (http://link.springer.com)

Regarding the keywords for the search, after some exploratory searches using different combination of keywords, the researchers jointly established the final string to be used in the search for papers in the databases:

"software process" AND (education OR training)

The search was performed at the beginning of 2015. The search string was applied to title, abstract and keywords, and limited to journal papers written in English in the area of Computer Science and published between the years 2000 and 2014. A total of 1450 papers were retrieved from the different databases. Unfortunately, despite using the advanced search, only IEEE's database seems to properly retrieve exact phrases in title, abstract and keywords, so this set had to be revised and only 253 unique papers were finally considered for the study selection (Figure 1).

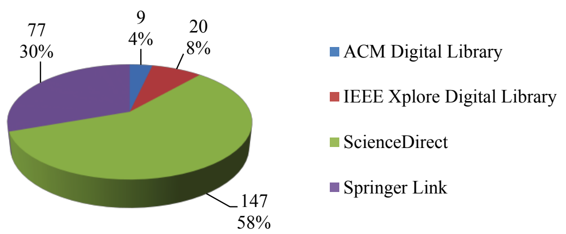


Figure 1: Selected databases and retrieved papers

2.3 Study Selection

The main guiding criterion to include a paper in the study or not was its focus on software process education. To reduce the possibility of researcher bias, the authors jointly agreed the exclusion criteria to be used in the following order:

- Based on title: the title does not suggest that there is any relation to software process education.
- Based on abstract: the abstract shows the paper is not focused on software process education.
- Based on full text: the paper is definitely not related to software process education.

In those cases where there was disagreement between researchers regarding the relevancy of a paper, the paper was not finally excluded.

The authors of this study must point out that the revision of the full text of the primary studies allowed to assure that all of them were relevant for structuring and characterizing the state of the practice of the education on software process. This revision is also important because this study does not contain a formal quality evaluation of the primary studies, which indeed is not essential in mapping studies and could not be properly achieved due to the inclusive nature of the search that includes theoretical studies as well as empirical studies of all types [KiBP11].

After the exclusion of irrelevant papers, the researchers finally agreed on 33 primary studies to be included in the systematic mapping study (Table 1). The full list of primary studies is listed in the appendix.

Table 1: Study selection reading detail

Reading detail	# of studies
Search	253
Title	95
Abstract	53
Full-text	33

2.4 Study classification

A data extraction form was designed to collect relevant information from each one of the selected primary studies. It included the following properties: title, authors, year, journal, number of citations in the ISI Web of Knowledge, type of participants in the educational initiative, educational method, mode of delivery, focus of the initiative, lessons learned in the initiative, and future work proposed.

The authors agreed in classifying primary studies depending on three different types of participants in the educational initiative: undergraduates, graduates and industry professionals.

Regarding the different educational methods and attending to the background of this research described previously, the authors decided to classify the primary studies in these groups: lectures, exercises, project, teaching a missing subject, adding realism to a project, inclusion of simulation in practical classes, and gamification.

Finally, for classifying the main mode of delivery used by the initiative the authors agreed in the following ones: traditional, experiential (learning by doing), anchored instruction, aptitude-treatment interaction, situated learning, lateral thinking, and motivation.

2.5 Data extraction and synthesis of results

This section synthesizes the results produced by the extraction of data from the primary studies according to the protocol described above.

The distribution of primary studies does not vary much throughout the years considered in this mapping study. Number of publications fluctuates mainly between 1 and 3, being 2002 and 2008 the most productive years with 5 publications.

Data extracted from primary studies revealed that a total of 77 different authors published papers on the topic of software process education. It is not a surprise to find W.S. Humphrey is the most prolific author among the primary studies with 3 papers as he created the Personal Software Process (PSP), which is one of the processes often used in software process education.

Regarding the journals that published the primary studies, IEEE Software is the journal that accepted the most publications (7) related to software process education, closely followed by the Journal of Computing Sciences in Colleges, the Journal of Systems and Software, Information and Software Technology, and the Software Quality Journal.

Similarly, papers published in IEEE Software sum the largest amount of citations in ISI (64), given that the Journal of Computing Sciences in Colleges (79 citations in Google Scholar) is not indexed in ISI. Taking into account the number of papers, journals such as IEEE Transactions on Engineering Management, IEEE Transactions on Software Engineering and Annals of Software Engineering have a better average of citations in ISI as they published only one of the primary studies that, however, received a significant amount of citations.

To provide a better overview of the field, Figure 2 depicts the types of students involved in the initiatives described in the primary studies, Figure 3 shows the educational methods followed in the initiatives described in the primary studies, and Figure 4 illustrates the modes used for delivering education in the initiatives described in the primary studies. The authors must point out that some primary studies involved more than one type of students, followed more than one method and/or used more than one mode of delivery in their educational initiatives.

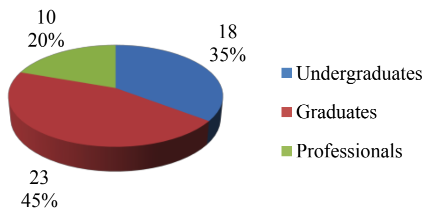


Figure 2: Participants in the primary studies

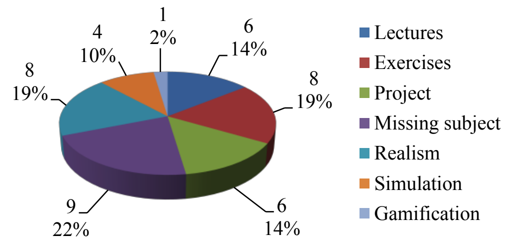


Figure 3: Educational methods in the primary studies

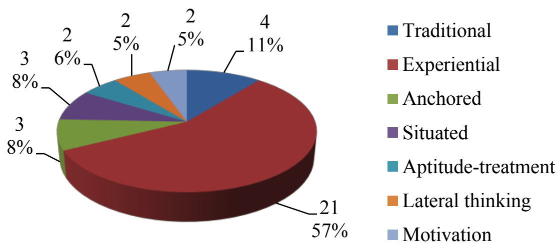


Figure 4: Modes of delivery in the primary studies

3 Analysis and discussion of results

In the following sub-sections the authors analyze and discuss the results produced by conducting the systematic mapping study according to the steps described in the previous section in order to find answers to the research questions of this study.

3.1 What are the general characteristics of the software process education and training initiatives? (RQ1)

According to the results shown in Figure 2, the majority of studies related to software process education focus research on early stages of software engineers' career, graduates (23) and undergraduates (18), while few of them focus on education for industry professionals (10).

Some of the studies represented in Figure 3 follow the most traditional method for educating future software engineers (6) consisting in a series of lectures combined with some exercises and/or a small project to put acquired knowledge into practice; these studies are usually oriented to undergraduates. Most of the studies, however, describe initiatives to cover a subject that is usually missing in software process education curricula (9); these initiatives aimed at completing the education on software process are usually based also in the combination of lectures with exercises and/or a small project. Another method which is broadly used (8) is making students' project experience closer to the real world (e.g. using an external customer [S23]). Several experiences with simulations (4) to improve software process education have been also reported in the last years; these simulations are often oriented to graduates, and especially to industry professionals, in conjunction with other initiatives based on task assignments, tutorials and workshops. Finally, only one primary study is related to gamification [S17].

Given that many of the primary studies report initiatives using exercises, it is not strange finding experiential learning is the most used delivery method by far (21), see Figure 4; as Albert Einstein once said: "Learning is experience, everything else is information". Results also show that modes of delivery such as situational learning or motivation are generally used when adding realism to a project; while the former is basically used with undergraduates, the latter is mainly used when education is oriented to industry professionals.

To conclude this section, the authors found several of these approaches focus on teaching a specific software process such as the PSP (7) or the TSP (5), while others train students in iterative and agile software development methods (10). Still many of them put emphasis on process improvement training (8), mainly related to CMM and CMMI. Only 3 of the primary studies deal with software process education from the point of view of Project Management. Finally, the remainder focuses on specific parts of the software process such as design, programming or document inspection.

3.2 What lessons did researchers learned from previous research on software process education? (RQ2)

Previous research on software process education has provided numerous and various lessons learned. In the following paragraphs the authors cover the most relevant ones found in the scope of this mapping study.

In general, introducing processes into the classroom environment is not easier than injecting them into the workplace, so future researchers should take some considerations into account. Matching the software process weight to the students' abilities, expectations and tolerance is vital for success [S25]. Furthermore, although the use of model representations eases the understanding of the process and increases visibility, giving the students a written process description is not enough; instructors must also provide guidance in the form of mentoring to have a major impact [S6]. Motivation is also essential as engaged and motivated students are more likely to accept the software process [S29].

In addition, if **tools** are used to support process activities, they should be easy to learn and use to create a positive attitude towards their adoption [S2]; despite their learning curve, software process tools have proven to be important to the successful development of projects. In some cases, using a knowledge repository can facilitate the learning process and the transfer of knowledge among students [S20]; low-experienced software engineers can gain experience from more experienced ones and giving them more autonomy [HGAS13].

With regard to software development methods, results point out that **Agile** works well for student projects in an introductory software engineering course

[S26]. Such incremental and iterative approaches allow students to learn from preceding iterations and incorporate previous experience and feedback into the next iteration [S27].

PSP and TSP are also good means to introduce discipline concepts and software process to potential engineers because they show students how to define processes, how to use a defined process, how to plan, measure and track their work, and how to measure and manage quality [S31]. Results gathered from the primary studies confirm the benefits of training students on the PSP. It enhances predictability and reduces the number of (trivial) defects in the code, although students may require more time for finishing tasks because of the error checking that leads to the improved robustness [S8].

The authors in [S28] and [S29] provide some recommendations for using PSP and TSP as discipline drivers in software process education: 1) Customize PSP and TSP courses to the context and the needs of the students; 2) Integrate PSP as part of TSP in order for students to first master PSP techniques before assuming a role in a software development group; 3) Arrange PSP training regularly and continuously to ensure that a student can meet both essential and accidental software challenges (actually, some authors state that learners should apply PSP practices not just in a single course, but as a regular part of their studies for instilling good habits and professional attitudes); 4) Motivate students about the benefits of PSP and TSP; and 5) Let students see their progress through the data collected, but these data should not be used for grading purposes in order to reduce the likelihood of students manipulating the values in an attempt to gain better grades.

Another interesting recommendation found in the primary studies is tailoring the assignments the course to imitate the real-world software projects [S12]. **Realism** has to be seek so that when a process-related problem arises, the process should be improved in order to not repeat the same problem in subsequent projects. To increase reality, instructors can promote collaboration between students and external customers [S33], provided that customers' involvement may help to produce software better adapted to real expectations. However, there is a risk of students giving more attention to the product than the process as customers are interested in the product [S23]. On the other hand, facilities such as studios [S24] not only bring home a great opportunity to take theory into practice, but also provide students with environments and experiences they will encounter or maybe even bring to their future jobs.

When using a **project** to educate on software process or in senior **capstone courses**, students can use everything they already learned [S14]. The use of **dynamic teams** [S27] in these projects is a good experience because it challenges students to adapt to multiple personalities and skill sets; they can learn from one another, they feel more comfortable in rating peers honestly, and it leads to fewer group breakdowns when team members underperform. Other practices such as **pair designing** [S18] may slow down the project, but it is more predictable than individual designing with regards to quality.

To improve the likelihood of successfully design and implement a software project course, researchers should follow several guidelines [S30]: 1) Clearly identify course goals; 2) If the course is time-restricted or represents students' first team project experience, use a modest and well-defined problem; 3) Use a defined team process for the project work; 4) Enforce process discipline; and 5) Instructors should move their role from lecturer to coach.

Concerning **process improvement training**, CMMI-recommended practices are accepted across much of the industry and thus they are a good reference for software process education efforts. Results of previous research [S22] revealed some hot spots that require more training in software process programs (e.g. organizational practices). It may be beneficial to dedicate significant time to provide details about process models such as CMMI at the graduate level, but it may not be appropriate at the undergraduate level [S3]. Therefore, researchers recommend addressing individual skills at the undergraduate level and management skills at the graduate level.

With regard to **gamification**, it proved to have potential to support education [S17], although further research is needed. In this sense, [HCAY14] presents a Gamification approach for software process, but not linked to education or training. Likewise, **simulation** seems to be very useful because allows students to change process settings and helps decide if a process is suitable for a certain context [S10]. When researching on the benefits of simulation for educating on the software process, researchers must take into account that it is not an inexpensive undertaking and students need time for the familiarization with the simulator [S1]. Yet, there is little evidence that process simulation has become an accepted and regularly used tool in industry [S4]. Moreover, the use of simulation techniques like, for instance System Dynamics is well grounded in software engineering education [GCGP08].

To conclude this section, researches should consider learning from practitioners of other engineering disciplines [S19], as their lessons learned can be useful for software engineering too.

3.3 What future works did authors propose in previous research on software process education? (RQ3)

In spite of the large amount of lesson learned gathered from previous research on software process education, not many of the primary studies propose future works. The most common ones proposed exporting described initiatives to other universities [S16] or to the industry [S15].

More interesting proposals, especially those focused on simulation and gamification, suggested enhancing complexity and variability to allow a more dynamic learning experience [S17]. In addition, future research could consider the extension of the single-learner model towards a collaborative learning environment [S1]. Nevertheless, there is still a need for providing evidence of the usefulness of simulation in the real-world and additional studies of long-term evolution from a product and organizational perspective [S4].

Primary studies related to the PSP raise several questions to address with further studies regarding the degree to which PSP students make more balanced estimates, the relationship between productivity and effort estimation accuracy, whether planning time and postmortem time are dependent on project size or whether they are more or less constant and could be viewed as overhead [S28]. Other additional important questions could be: How will defect estimation behave in further studies? How could we prepare a set of exercises that allows us to separate the complexity of exercises from the PSP levels? To what extent is a virtual environment the most appropriate tool for teaching discipline teamwork? What kind of feedback is received best as motivation by the students: defects, size estimation or effort estimation?

To conclude the answer to this research question, studies considering issues related to human capital suggest incorporating ethical and social aspects of ICTs in computer science programs and developing awareness of potential threats posed by new ICTs among today's students [S33]. Others propose analyses of the impact of outdated technology skills or about attitudes toward software process innovations [S5].

4 Limitations

The objective of this study was to structure and characterize the state of the practice on software process education, analyzing previous works published in the literature to provide an overview of the topic and to help discover potential gaps for future research. For that purpose, the authors decided to use a general search string to not bias the study towards any specific educational method or mode of delivery. However, other searches using keywords related to specific educational method, such as realism or simulation, or mode of delivery, such as lateral thinking or situated learning, could provide more primary studies. This limitation makes this study to be a first step towards a future research that could include a systematic literature review centered on new approaches for the education on software process based on trending modes of delivery such as flipped learning or Massive Open Online Courses (MOOCs).

Similarly, due to the specific focus of this 1st International Workshop on Software Process Education, Training and Professionalism, the authors decided to include just the term "software process" and not the term "software engineering" in the search string. Broadening the scope of this research to software engineering education and not focusing only in the software process would have provided a richer set of primary studies and should be considered for a future work.

The exclusion of conference papers and books represent another limitation of this study. This publication bias is based mainly on practical concerns; the amount of primary studies to be included could have been unmanageable and a lot of analysis would be needed to handle the fact that many journal papers are improvements of previously published conference papers. Nevertheless, the inclusion of journal papers guarantees a high scientific quality of the primary studies. However, and in spite of the inclusion of journals, given the composition of databases for the study, some papers published in journals not listed in the databases can also be biased in this study.

Finally, another threat for this study is researcher bias that could have affected the selection of primary studies, their classification and the accuracy in data extraction. To reduce the subjective component of this study, two researchers participated in the selection and classification of primary studies following a multi-staged protocol for the inclusion and exclusion criteria and resolving disagreements by discussion.

5 Conclusions and future work

Software process improvement is considered one of the most important fields in the software engineering discipline. However, and in spite of its importance, increasing its coverage in educational settings is still challenging. The complexity of the subject together with the need of a good background of the discipline is normally pushing subjects into master programs, while PSP and TSP approaches are mostly present in bachelor curricula. This paper is a first effort towards understanding the subject and interpreting its needs and implementation in the academia.

Future works will be twofold. Firstly, it is intended to investigate the use of MOOCs in software process improvement settings and secondly, it is aimed to develop specific gamification strategies and tools for software process improvement education and training.

References

[Acmc15] ACM COMMITTEE ON COMPUTERS AND PUBLIC POLICY: *The Risks Digest*. URL http://catless.ncl.ac.uk/risks. - abgerufen am 2015-02-27

[AlUn14] BIN ALI, NAUMAN ; UNTERKALMSTEINER, MICHAEL: Use and evaluation of simulation for software process education: a case study. In: . Seeon Monastery, Germany : Shaker Verlag, 2014, S. 59–73

[AnRS96] ANDERSON, JOHN R. ; REDER, LYNNE M. ; SIMON, HERBERT A.: Situated Learning and Education. In: *Educational Researcher* Bd. 25 (1996), Nr. 4, S. 5–11

[BaOH05] BAKER, ALEX ; OH NAVARRO, EMILY ; VAN DER HOEK, ANDRÉ: An experimental card game for teaching software engineering processes. In: *Journal of Systems and Software*, *Software Engineering Education and Training*. Bd. 75 (2005), Nr. 1–2, S. 3–16

[Bono09] BONO, EDWARD DE: *Lateral Thinking: A Textbook of Creativity* : Penguin UK, 2009 — ISBN 9780141938318

[BSHK90] BRANSFORD, JOHN D ; SHERWOOD, ROBERT D ; HASSELBRING, TED S ; KINZER, CHARLES K ; WILLIAMS, SUSAN M: Anchored instruction: Why we need it and how technology can help. In: NIX, D. ; SPIRO, R. (Hrsg.): *Cognition, education, and multimedia: Exploring ideas in high technology*. Hillsdale, NJ. : Lawrence Erlbaum, 1990, S. 115–141

[CCGG09] CASADO-LUMBRERAS, C. ; COLOMO-PALACIOS, R. ; GOMEZ-BERBIS, J.M. ; GARCIA-CRESPO, A.: Mentoring programmes: a study of the Spanish software industry. In: *International Journal of Learning and Intellectual Capital* Bd. 6 (2009), Nr. 3, S. 293–302

[CCMS14] COLOMO-PALACIOS, RICARDO ; CASADO-LUMBRERAS, CRISTINA ; MISRA, SANJAY ; SOTO-ACOSTA, PEDRO: Career Abandonment Intentions among Software Workers. In: *Human Factors and Ergonomics in Manufacturing & Service Industries* Bd. 24 (2014), Nr. 6, S. 641–655

[CCSG13a] COLOMO-PALACIOS, RICARDO ; CASADO-LUMBRERAS, CRISTINA ; SOTO-ACOSTA, PEDRO ; GARCÍA-CRESPO, ÁNGEL: Decisions in software development projects management. An exploratory study. In: *Behaviour & Information Technology* Bd. 32 (2013), Nr. 11, S. 1077–1085

[CCSG13b] COLOMO-PALACIOS, R. ; CASADO-LUMBRERAS, CRISTINA ; SOTO-ACOSTA, PEDRO ; GARCÍA-PEÑALVO, FRANCISCO J. ; TOVAR-CARO, EDMUNDO: Competence gaps in software personnel: A multi-organizational study. In: *Computers in Human Behavior*, *Advanced Human-Computer Interaction*. Bd. 29 (2013), Nr. 2, S. 456–461

[ChCh11] CHEN, CHUNG-YANG ; CHONG, P. PETE: Software engineering education: A study on conducting collaborative senior project development. In: *Journal of Systems and Software* Bd. 84 (2011), Nr. 3, S. 479–491

[Daws00] DAWSON, RAY: Twenty Dirty Tricks to Train Software Engineers. In: *Proceedings of the 22Nd International Conference on Software Engineering*, *ICSE '00*. New York, NY, USA : ACM, 2000 — ISBN 1-58113-206-9, S. 209–218

[Exte14] EXTER, MARISA: Comparing Educational Experiences and On-the-job Needs of Educational Software Designers. In: *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, *SIGCSE '14*. New York, NY, USA : ACM, 2014 — ISBN 978-1-4503-2605-6, S. 355–360

[Fugg00] FUGGETTA, ALFONSO: Software Process: A Roadmap. In: *Proceedings of the Conference on The Future of Software Engineering*, *ICSE '00*. New York, NY, USA : ACM, 2000 — ISBN 1-58113-253-0, S. 25–34

[FuNi14] FUGGETTA, ALFONSO ; DI NITTO, ELISABETTA: Software Process. In: *Proceedings of the on Future of Software Engineering*, *FOSE 2014*. New York, NY, USA : ACM, 2014 — ISBN 978-1-4503-2865-4, S. 1–12

[GCGP08] GARCÍA-CRESPO, ÁNGEL ; COLOMO-PALACIOS, R ; GOMEZ-BERBIS, MJ ; PANIAGUA-MARTIN, F: A Case of System Dynamics Education in Software Engineering Courses. In: *IEEE Multidisciplinary EngineeringEducation Magazine* Bd. 32 (2008), S. 52–59

[HCAY14] HERRANZ, EDUARDO ; COLOMO-PALACIOS, RICARDO ; DE AMESCUA SECO, ANTONIO ; YILMAZ, MURAT: Gamification as a Disruptive Factor in Software Process Improvement Initiatives. In: *j-jucs* Bd. 20 (2014), Nr. 6, S. 885–906. — |http://www.jucs.org/jucs_20_6/gamification_as_a_disruptive|

[HeCG13] HERNÁNDEZ-LÓPEZ, ADRIÁN ; COLOMO-PALACIOS, RICARDO ; GARCÍA-CRESPO, ÁNGEL: Software engineering job productivity — a systematic review. In: *International Journal of Software Engineering and Knowledge Engineering* Bd. 23 (2013), Nr. 03, S. 387–406

[HGAS13] HEREDIA, ALBERTO ; GUZMÁN, JAVIER GARCÍA ; AMESCUA, ANTONIO ; SEGURA, MARIA ISABEL SÁNCHEZ: Interactive Knowledge Asset Management: Acquiring and Disseminating Tacit Knowledge. In: *Journal of Information Science and Engineering* Bd. 29 (2013), Nr. 1, S. 133–147

[Hump95] HUMPHREY, WATTS S.: *A Discipline for Software Engineering*. Boston, MA, USA : Addison-Wesley Longman Publishing Co., Inc., 1995 — ISBN 0201847485

[Kell87] KELLER, JOHN M.: Development and use of the ARCS model of instructional design. In: *Journal of instructional development* Bd. 10 (1987), Nr. 3, S. 2–10

[KiBP11] KITCHENHAM, BARBARA A. ; BUDGEN, DAVID ; PEARL BRERETON, O.: Using mapping studies as the basis for further research – A participant-observer case study. In: *Information and Software Technology*, *Special Section: Best papers from the APSEC Best papers from the APSEC*. Bd. 53 (2011), Nr. 6, S. 638–651

[KoCM14] KOHWALTER, T.C. ; CLUA, E.W.G. ; MURTA, L.G.P.: Reinforcing Software Engineering Learning through Provenance. In: *2014 Brazilian Symposium on Software Engineering (SBES)*, 2014, S. 131–140

[KuFM13] KUHRMANN, MARCO ; FERNÁNDEZ, DANIEL MÉNDEZ ; MÜNCH, JÜRGEN: Teaching Software Process Modeling. In: *Proceedings of the 2013 International Conference on Software Engineering*, *ICSE '13*. Piscataway, NJ, USA : IEEE Press, 2013 — ISBN 978-1-4673-3076-3, S. 1138–1147

[PFMM08] PETERSEN, KAI ; FELDT, ROBERT ; MUJTABA, SHAHID ; MATTSSON, MICHAEL: Systematic Mapping Studies in Software Engineering. In: *Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering*, *EASE'08*. Swinton, UK, UK : British Computer Society, 2008, S. 68–77

[PGBP15] PEDREIRA, OSCAR ; GARCÍA, FÉLIX ; BRISABOA, NIEVES ; PIATTINI, MARIO: Gamification in software engineering – A systematic mapping. In: *Information and Software Technology* Bd. 57 (2015), S. 157–168

[RoZS14] RONG, GUOPING ; ZHANG, HE ; SHAO, DONG: Where does experience matter in software process education? An experience report. In: *2014 IEEE 27th Conference on Software Engineering Education and Training (CSEE T)*, 2014, S. 129–138

[WaSB12] VON WANGENHEIM, CHRISTIANE GRESSE ; SAVI, RAFAEL ; BORGATTO, ADRIANO FERRETI: DELIVER! – An educational game for teaching Earned Value Management in computing courses. In: *Information and Software Technology* Bd. 54 (2012), Nr. 3, S. 286–298

[Yeh12] YEH, YU-CHU: Aptitude-Treatment Interaction. In: SEEL, P. D. N. M. (Hrsg.): *Encyclopedia of the Sciences of Learning* : Springer US, 2012 — ISBN 978-1-4419-1427-9, 978-1-4419-1428-6, S. 295–298

Appendix: Primary studies selected for the systematic mapping study

[S1] Pfahl, D., Klemm, M., Ruhe, G.: A CBT module with integrated simulation component for software project management education and training. Journal of Systems and Software. 59, 283–298 (2001).

[S2] Agarwal, R., Prasad, J.: A field study of the adoption of software process innovations by information systems professionals. IEEE Transactions on Engineering Management. 47, 295–308 (2000).

[S3] Biberoglu, E., Haddad, H.: A Survey of Industrial Experiences with CMM and the Teaching of CMM Practices. Journal of Computing Sciences in Colleges. 18, 143–152 (2002).

[S4] Ali, N.B., Petersen, K., Wohlin, C.: A systematic literature review on the industrial use of software process simulation. Journal of Systems and Software. 97, 65–85 (2014).

[S5] Matalonga, S., Solari, M., Feliu, T.S.: An empirically validated simulation for understanding the relationship between process conformance and technology skills. Software Qual J. 22, 593–609 (2013).

[S6] Elliott, M., Dawson, R., Edwards, J.: An evolutionary cultural-change approach to successful software process improvement. Software Qual J. 17, 189–202 (2009).

[S7] Kamatar, J., Hayes, W.: An experience report on the personal software process. IEEE Software. 17, 85–89 (2000).

[S8] Prechelt, L., Unger, B.: An experiment measuring the effects of personal software process (PSP) training. IEEE Transactions on Software Engineering. 27, 465–472 (2001).

[S9] Morisio, M.: Applying the PSP in industry. IEEE Software. 17, 90–95 (2000).

[S10] Hsueh, N.-L., Shen, W.-H., Yang, Z.-W., Yang, D.-L.: Applying UML and software simulation for process definition, verification, and validation. Information and Software Technology. 50, 897–911 (2008).

[S11] Sampaio, A., Vasconcelos, A., Sampaio, P.R.F.: Assessing agile methods: An empirical study. J Braz Comp Soc. 10, 21–48 (2004).

[S12] Shen, W.-H., Hsueh, N.-L., Lee, W.-M.: Assessing PSP effect in training disciplined software development: A Plan–Track–Review model. Information and Software Technology. 53, 137–148 (2011).

[S13] Carver, J., Shull, F., Basili, V.: Can observational techniques help novices overcome the software inspection learning curve? An empirical investigation. Empir Software Eng. 11, 523–539 (2006).

[S14] Beasley, R.E.: Conducting a Successful Senior Capstone Course in Computing. Journal of Computing Sciences in Colleges. 19, 122–131 (2003).

[S15] García, J., Amescua, A., Sánchez, M.-I., Bermón, L.: Design guidelines for software processes knowledge repository development. Information and Software Technology. 53, 834–850 (2011).

[S16] Bagert, D.J., Mengel, S.A.: Developing and using a web-based project process throughout

the software engineering curriculum. Journal of Systems and Software. 74, 113–120 (2005).
[S17] Wangenheim, C.G. von, Thiry, M., Kochanski, D.: Empirical evaluation of an educational game on software measurement. Empir Software Eng. 14, 418–452 (2008).
[S18] Canfora, G., Cimitile, A., Garcia, F., Piattini, M., Visaggio, C.A.: Evaluating performances of pair designing in industry. Journal of Systems and Software. 80, 1317–1327 (2007).
[S19] Hantos, P., Gisbert, M.: Identifying software productivity improvement approaches and risks: construction industry case study. IEEE Software. 17, 48–56 (2000).
[S20] Amescua, A., Bermón, L., García, J., Sánchez-Segura, M.-I.: Knowledge repository to improve agile development processes learning. IET Software. 4, 434–444 (2010).
[S21] Guzmán, J.G., Martín, D., Urbano, J., Amescua, A. de: Practical experiences in modelling software engineering practices: The project patterns approach. Software Qual J. 21, 325–354 (2012).
[S22] Moreno, A.M., Sánchez-Segura, M.-I., Medina-Dominguez, F., Cuevas, G.: Process Improvement from an Academic Perspective: How Could Software Engineering Education Contribute to CMMI Practices? IEEE Software. 31, 91–97 (2014).
[S23] Tadayon, N.: Software Engineering Based on the Team Software Process with a Real World Project. Journal of Computing Sciences in Colleges. 19, 133–142 (2004).
[S24] Niño, J.: Software Engineering Team Studios. Journal of Computing Sciences in Colleges. 23, 59–65 (2008).
[S25] Umphress, D.A., Hendrix, T.D., Cross, J.H.: Software process in the classroom: the Capstone project experience. IEEE Software. 19, 78–81 (2002).
[S26] Hart, D.: Supporting Agile Processes in Software Engineering Courses. Journal of Computing Sciences in Colleges. 25, 136–143 (2010).
[S27] Anewalt, K., Polack-Wahl, J.A.: Teaching an Iterative Approach with Rotating Groups in an Undergraduate Software Engineering Course. Journal of Computing Sciences in Colleges. 25, 144–151 (2010).
[S28] Rombach, D., Münch, J., Ocampo, A., Humphrey, W.S., Burton, D.: Teaching disciplined software development. Journal of Systems and Software. 81, 747–763 (2008).
[S29] Borstler, J., Carrington, D., Hislop, G.W., Lisack, S., Olson, K., Williams, L.: Teaching PSP: challenges and lessons learned. IEEE Software. 19, 42–48 (2002).
[S30] Hilburn, T.B., Humphrey, W.S.: Teaching teamwork. IEEE Software. 19, 72–77 (2002).
[S31] Humphrey, W.S.: Three Process Perspectives: Organizations, Teams, and People. Annals of Software Engineering. 14, 39–72 (2002).
[S32] Germain, É., Robillard, P.N.: Towards software process patterns: An empirical analysis of the behavior of student teams. Information and Software Technology. 50, 1088–1097 (2008).
[S33] Begier, B.: Users' involvement may help respect social and ethical values and improve software quality. Inf Syst Front. 12, 389–397 (2009).

Software Process Improvement in Graduate Software Engineering Programs

Claude Y. Laporte
École de technologie supérieure
Montréal
Canada
claude.laporte@etsmtl.ca

Rory V. O'Connor
Dublin City University
Dublin
Ireland
roconnor@computing.dcu.ie

Abstract

At the École de technologie supérieure (ÉTS), software process improvement (SPI) is taught in lecture format and with a 10-week implementation project in an organization by teams of students of the graduate software engineering curriculum. The SPI course is taught using a 'problem-goal-solution' approach where students learn that any process improvement initiative must be based on issues preventing an organization in achieving its organizational goals whether the organization is a company or a not-for-profit organization. An important aspect of this course is the management of technological change where students learn and put in practice in their project the 'soft' issues which are part of most SPI organizational initiatives.

1. Introduction

As reported by Charette [Cha05], software specialists spend about 40 to 50 percent of their time on avoidable rework. The ability of organizations to compete, adapt, and survive is increasingly dependent on quality, productivity, development time, and cost. Systems and software are growing larger and more complex every year. For example, mainstream cars have between 20 and 30 million lines of code (LOC) and top-of-the-line cars contain 100 million LOC and it is expected that LOC will increase by 50% by 2020 [Fle14]. Software process improvement (SPI) is even more important when we consider all the software development projects that have partially or totally failed, the numerous incidents and the financial losses generated by those failures. As an example, recalls du to software errors, which account for about 60-70% of vehicle recall in European and North American markets, can lead to multimillion and even multibillion dollar losses.

The École de technologie supérieure (ÉTS), a 7,800-student engineering school, began offering its graduate SPI course to professional students in 2000. The aim of this specific SPI course is to ensure that software engineering students are aware of the importance of SPI, and that they understand and are able to manage and apply SPI practices in real organizations. The professor who designed the SPI course has an industrial experience of more than 20 years, mainly in defense and railway sectors. The course is made up of lectures, practical exercises, and a team project in industry. A continuous process of student evaluation is carried out to ensure that the concepts are well understood.

This article is divided into two sections. First, the authors present an overview of the SPI course, in a second section projects performed by students in organizations are briefly described.

2 Software Process Improvement Course

The SPI course, a 3-credit course (i.e. 9 hours per week including 3-hour lectures) over a period of 13 weeks. Each lecture topic is illustrated with industrial examples, international or professional standards, and process improvement model practices. To ensure that students grasp the importance of SPI activities, the business model concept and the cost of quality concept are stressed throughout the course. When performing SPI activities as part of their projects, students must make tradeoffs between technical issues and 'soft' issues such as the management of cultural changes.

There is a wide spectrum of development approaches for organizations developing software. Figure 1 illustrates the spectrum of approaches on 2 axes. The horizontal axis (from left to right) illustrates the level of ceremony, from a low ceremony approach with little documentation (e.g. agile approach) to a high ceremony approach with a comprehensive documentation (e.g. plan driven CMMI approach). The vertical axes illustrate the approaches based on the level of risk. The top axis illustrates a low risk linear

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

approach using a waterfall approach while the lower part of the axis illustrates a risk-driven project using an iterative approach. ISO/IEC 29110 is located at about the center of both axes.

Figure 1: Positioning of the ISO/IEC 29110 (adapted from [Kro03])

Initially, the SPI course used the CMMI model developed by the Software Engineering Institute (SEI). The CMMI was selected because many organizations, especially defense and aerospace enterprises, were already using it and it was available in French at no cost from the SEI. The ÉTS engineering school is located in Montréal. A survey of the software development companies was done a few years ago in order to obtain a picture of this industry in the Montréal area. As illustrated in Table 1, it was found that close to 80% of software development companies have fewer than 25 employees. In addition, over 50% have fewer than 10 employees.

Table 1: Size of software development companies in the Montreal area [Gau04]

Size (Number of employees)	**Software Companies**		**Total Number of Jobs**	
1 to 25	540	78%	5,105	29%
26 to 100	127	18%	6,221	36%
over 100	26	4%	6,056	35%
TOTAL	**693**	**100%**	**17,382**	**100%**

Since a large percentage of students attending the SPI course were working in small organizations, the emphasis on the use of the CMMI framework was gradually reduced to switch to a new ISO set of standards and guides: the recently published family of ISO/IEC 29110. The ISO/IEC 29110 standards and guides have been developed specifically for enterprises, organizations and projects having up to 25 people. The ISO/IEC 29110 management and engineering guides are available in English and French, at no cost, from ISO. They are also available in Portuguese and Spanish.

2.1 ISO/IEC 29110

The ISO/IEC 29110 standard "Lifecycle profiles for Very Small Entities" [Lap08] is aimed at addressing the issues identified above and addresses the specific needs of VSEs [OC11a] [OC11b] and to tackle the issues of poor standards adoption by small companies [Col08] [OC09]. The approach [OC14] [Lap13] used to develop ISO/IEC 29110 started with the pre-existing international standard ISO/IEC 12207 dedicated to software process lifecycles. The overall approach consisted of three steps: (1) Selecting ISO/IEC 12207 [16] process subset applicable to VSEs of up to 25 employees; (2) Tailor the subset to fit VSE needs; and (3) Develop guidelines for VSEs.

The basic requirements of a software development process are that it should fit the needs of the project and aid project success. And this need should be informed by the situational context where in the project must operate and therefore, the most suitable software development process is contingent on the context [Jen13] [Cla12]. The core situational characteristic of the entities targeted by ISO/IEC 29110 is size, however there are other aspects and characteristics of VSEs that may affect profile preparation or selection, such as: Business Models (commercial, contracting, in-house development, etc.); Situational factors (such as criticality, uncertainty environment, etc.); and Risk Levels. Creating one profile for each possible combination of values of the various dimensions introduced above would result in an unmanageable set of profiles. Accordingly VSE's profiles are grouped in such a way as to be applicable to more than one category.

Profile Groups are a collection of profiles which are related either by composition of processes (i.e. activities, tasks), or by capability level, or both. The "Generic" profile group has been defined [OC10] as applicable to a vast majority of VSEs that do not develop critical software and have typical situational factors. This profile group does not imply any specific application domain, however, it is envisaged that in the future new domain-specific sub-profiles may be developed in the future.

Finally, the results obtained from systematic literature review of the ISO/IEC 29110 standard [San15] show that there is an increasing interest on it.

Figure 2 illustrates the activities of the project management and software implementation processes. The Project Management process and the Software Implementation are described in great details in the guides. As an example, each activity is composed of a set of tasks with inputs, outputs and roles. For all inputs and outputs, the guides describe a typical content.

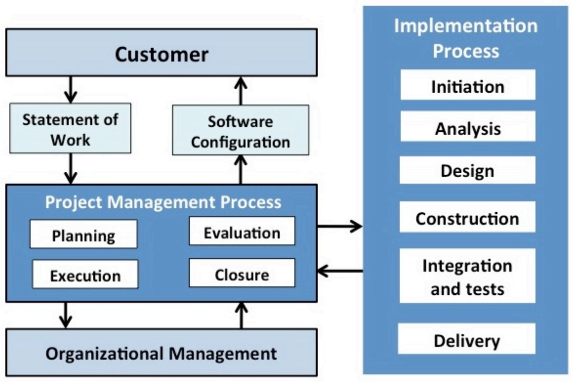


Figure 2: Basic profile processes and activities [Lap15a]

2.2 The Course

The approach used for software process improvement is covered in the book titled 'Making Process Improvement Work' [Pot02]. This approach includes the following four steps: 1) Determine the business goals and the problems that an organization wishes to solve; 2) Determine organizational goals and identify problems; 3) Prioritize identified problems; 4) Develop and implement a SPI plan. The topics presented in class, listed in Table 2, are supported with weekly reading assignments.

Throughout the course, the students are exposed to the management of technological and cultural changes using a book titled 'Managing Transitions: Making the Most of Change' [Bri09]. The book describes a change using a three-phase of transition model as illustrated in figure 3.

Table 2: List of SPI course topics

Theme	Content
Introduction	Challenges faced by organizations developing products comprising software. Benchmarking process performances. Improving process performances (e.g. quality, productivity, turnaround, etc.) Outsourcing and off shoring.
Models, Standards and Methods	The IDEAL improvement model from the SEI. The Capability Maturity Model Integration for Development ISO/IEC 29110 standards and guides for Very Small Entities (VSEs) Process performance assessment methods. Goal-Problem approach.
Management of Organizational Changes	Description of the organizational context of a change project. Organizational culture assessment. Change history assessment. Stress level assessment. Sponsorship assessment. Change agent's capability assessment. Motivational factors assessment. Change readiness assessment. Deployment/installation plan

During the course, students are introduced to the recently published ISO software development standard ISO/IEC 29110 [ISO12] targeting Very Small Entities (VSEs). Students use the engineering and management guide included in ISO 29110 which is freely available in English or French from the ISO, as a framework to help them understand when software quality practices are used in a development project and why. They also use the guide as a framework for their team project.

The course has been designed in such a way that teams of 4 students can apply the SPI practices, presented in the lectures, in an organization. Since many graduate students are already working in an organization, it was quite easy to identify a topic for improvement and obtain the support of the management of the organization. To reduce the burden to the organization, this employee is the interface with management.

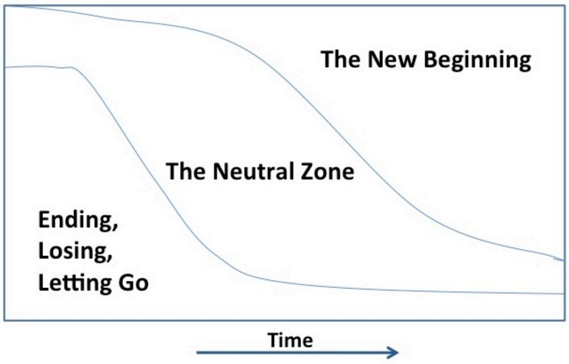


Figure 3: Three-phase transition [Bri09]

As mentioned above, the approach used for the SPI course is focused on solving an organizational problem. The activities performed by the students are:

- Develop a team member's contract (e.g. roles, responsibilities, expectations)
- Define the business context (e.g. type of product, customers, challenges)
- Identify business objectives, challenges and barriers.
- Develop a business case.
- Develop a communication plan.
- Measure and analyze organizational issues related to the management of change.
- Perform a mini-diagnostic of the performance of a process.
- Develop a mini-improvement/installation plan.
- Document the solution (e.g. a process).
- Implement the improvement plan

A vast majority of students are quite knowledgeable about the technical issues but they are not no knowledgeable about the management of technological change. Since the management of change is a key element of a successful process improvement program, a series of actions were done to facilitate the development, implementation and adoption of the processes, methods and tools [Lap98]. A set of assessment tools provided, described in table 3, helped the students understand the 'soft' issues of a technological change in an organization.

Table 3: Elements measured with the IMA Tools

Title of Tool	**Description**
Organization's stress level	Evaluation of the priorities for resources in the organization
Sponsor assessment	Evaluation of the resources, reinforcement (e.g. motivation) and communications commitments made and demonstrated by the sponsor(s) of a change project
Change agent skills	Evaluation of the skills and motivation of those responsible for facilitating the implementation of organizational changes
Individual readiness	Evaluation of the reasons why people may resist an organizational change
Culture assessment	Assessment of the fit between the desired change and the actual organizational culture in order to identify potential barriers and to leverage actual cultural strengths
Implementation history	Assessment of barriers and lessons learned from previous change projects (since past problems are likely to recur, this tool allows identification of the issues that need to be managed for the change project to be successful)

The assessments performed allow the students to better identify potential barriers to a proposed SPI and guide them in developing mitigating actions to increase the likelihood of the success of their improvement project. Table 4 describes the deliverables and the presentations made by students of the SPI course.

3 Software Process Improvement Project in Organizations

To illustrate the SPI projects done, we briefly describe, below, a few organizations where software process implementation projects conducted by graduate students.

<u>Organization 1</u>

Background: The company, of about 140 employees (14 software developers), designs and sells electric powertrain systems in the automotive field. Their products are embedded software that controls the operation of engines in real time and embedded software that controls the interactions between the components of a vehicle. Property management and audit of building health.

SPI Project: A compliance study was conducted to establish the difference between the processes in place and those proposed by the Entry profile of ISO/IEC 29110. An action plan has been developed to organise the software process improvement activities An analysis of differences between ISO/IEC 29110 was conducted An economic impact assessment was conducted using the methodology developed by ISO.

Organization 2

Background: Property management and audit of building health.

SPI Project: document existing business processes, analyze them and identify those with potential for improvement.

Organization 3

Background: An engineering company specialized in the integration of interactive systems, communication and security in the area of public transportation such as trains, subways and buses and railway stations, stations and stops of bus.

SPI Project: Start-up of 4 people in 2011. Many customers in the public transportation ask for a CMMI level 2. Implementing the CMMI® Level 2 Process Areas was too demanding for a start-up of 4 people. ISO/IEC 29110 has been used as the main reference for the development of the management and engineering processes. A gap analysis between CMMI level 2 and the ISO/IEC 29110-based processes was done.

Organization 4

Background: Medical device R&D enterprise.

SPI Project: The Basic profile of ISO/IEC 29110 was used to document and implement a quality management system. ISO 13485 was used as the framework for the quality management system.

Organization 5

Background: A project conducted at ÉTS for a unit responsible to promote activities for graduates and to raise money for the financing of the ÉTS foundation.

SPI Project: The software project developed a web portal using the Basic profile of ISO/IEC 29110. The portal allowed graduates to register to activities, modify their personal information.

Organization 6

Background: An enterprise specialized in industrial process control. A department of 13 employees.

SPI Project: ISO/IEC 29110 Entry profile was used to assess practices in used. The management of requirements was the focus of the project.

Organization 7

Background: An IT start-up involved in the development of web traffic surveillance.

SPI Project: A start-up of 4 employees. Documentation of the software development process using the Basic profile of ISO/IEC 29110.

Organization 8

Background: An IT service department of a large banking institution supporting the work of traders.

SPI Project: A department of 8 employees. Analysis of current practices. Development and implementation of requirements management practices using a traceability matrix.

Organization 9

Background: A large civil engineering and construction firm.

SPI Project: A department of 15 people. Responsible for the development and maintenance of software for the other units of the company. After an analysis of current practices using ISO/IEC 29110, the improvement implemented a change request management process.

In addition to the one-semester graduate SPI course, a few students decided to pursue the work done during the 10-week project as their graduate project. The following paragraphs briefly describe these projects.

An implementation project has been conducted in an IT start-up VSE by a team of two (part-time) developers [Lap4]. Their web application allows users to collaborate, share and plan their trips simply and accessible to all. The use of the Basic profile of ISO/IEC 29110 has guided the start-up to develop an application of high quality while using proven practices of ISO 29110. The total effort of this project was nearly 1000 hours. Using the management and engineering practices of ISO/IEC 29110 enabled the start-up to plan and execute the project expending only 13 percent of the total project effort on rework.

An implementation project has been conducted in an IT start-up founded by a graduate student. The VSE has 5 employees at the Montréal site and 2 employees in a site in Tunisia. The VSE offers software development services, Web solutions, mobile applications as well as consulting services to implement ERP solutions.

Table 4: Topics on course

Deliverables	Value
1. Project Plan and Contract between Team Members	4%
2. Project Overview 3. Business Case 4. Communication Plan 5. Organizational Culture Analysis	5% 5% 5% 5%
6. Diagnostic of the Organization 7. Sponsorship Evaluation 8. Updated contract between team members	5% 5% 0%
9. Analysis of the Motivational Factors 10. Organizational Stress Analysis	5% 5%
11. Change Agent Capability Assessment 12. Change Readiness Assessment	5% 5%
13. Process Description	5%
14. Improvement/Installation Plan	5%

A Canadian division, of about 400 employees, of a large American engineering firm has implemented a program to define and implement project management processes for their small-scale and medium-scale projects. The firm already had a robust and proven process to manage their large-scale projects. Their projects are classified into three categories as illustrated in Table 5.

Table 5: Classification of projects by the engineering firm [Lap15b]

	Small project	Medium project	Large project
Duration	< 2 months	> 2 & < 8 months	> 8 months
Team size	<= 4 people	4-8 people	> 8 people
Number of engineering specialties	1	>1	Many
Engineering fees	$5,000 - $70,000	$50,000 - $350,000	> $350,000
Percentage of projects	70%	25%	5%

The division documented the business goals as well as the problems that it wished to solve. The division used the project management process of the Entry profile of ISO/IEC 29110 to document their small-scale project management process and they used the project management process of the Basic profile to document their medium-scale project management process.

An ISO methodology was used to estimate the anticipated costs and benefits over a period of three years. An estimate of anticipated costs and benefits over a period of three years was made by the improvement program project sponsors. Table 6 shows the results for the first three years of this cost/benefit estimation.

Since the utilization of ISO/IEC 29110 was very successful in the development of their project management processes, the recently published systems engineering ISO/IEC 29110 Entry and Basic profiles will be used to redefine and improve the existing engineering process (ISO 2014, ISO 2015). This process will address the activities required from engineering requirements identification to final product delivery.

Table 6: Costs (in $CAD) and benefits estimations from implementing ISO/IEC 29110 [Lap15b]

	Year 1	**Year 2**	**Year 3**	**Total**
Cost to implement & maintain	59,600	50 100	50,100	159,800
Net benefits	255,500	265,000	265,000	785,500

3 Conclusions

Many changes have been made to the SPI course since it was initially set up in 2000. The challenge was to ensure that all these improvements met the objectives of the course. At ÉTS, students evaluate both the course and the professor. Following the improvements, the course scored 4.34 for a maximum score of 5, while the average score for the courses of the graduate software engineering program was 3.83. The use of ISO/IEC 29110 instead of the CMMI greatly facilitated the understanding and implementation of a software engineering framework suitable for most VSEs of the Montreal area.

The authors think that the current SPI course lectures and projects in industry provide a solid foundation for software engineers, even though SPI is still perceived as a low priority by most SMEs and VSEs. However, the profession of software engineering is still young...and Rome was not built in a day.

References

[Bri05] Bridges, William, Managing Transitions: Making the Most of Change. Da Capo Press, Cambridge, 2009.

[Cha05] Charette, R., Why software fails. IEEE Spectrum, pp. 42-49, September 2005.

[Col08] Coleman, G., O'Connor, R.: Investigating software process in practice: A grounded theory perspective. Journal of Systems and Software. 81, 772–784 (2008).

[Cla12] Clarke, P., O'Connor, R.V.: The situational factors that affect the software development process: Towards a comprehensive reference framework. Journal of Information and Software Technology. 54, 433–447 (2012).

[Fle14] Fleming, W., An Overview of Advances in Automotive Electronics, IEEE Veh. Technol. Mag., vol. 9, no. 1, pp. 4–9, March 2014.

[Gau04] Gauthier R., Une force en mouvement, La Boule de Cristal, Centre de recherche informatique de Montréal, January 22, 2004.

[ISO11] ISO/IEC TR 29110-5-1-2:2011 – Software Engineering - Lifecycle Profiles for Very Small Entities (VSEs) - Part 5-1-2: Management and engineering guide - Generic profile group: Basic profile, International Organization for Standardization/International Electrotechnical Commission: Geneva, Switzerland. Available at no cost from ISO: http://standards.iso.org/ittf/PubliclyAvailableStandards/c051153_ISO_IEC_TR_29110-5-1_2011.zip

[ISO12] ISO/IEC TR 29110-5-1-1:2012 – Software Engineering - Lifecycle Profiles for Very Small Entities (VSEs) - Part 5-1-1: Management and engineering guide - Generic profile group: Entry profile, International Organization for Standardization/International Electrotechnical Commission: Geneva, Switzerland. Available at no cost from ISO: http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html

[Jen13] Jeners, S., O'Connor, R.V., Clarke, P., Lichter, H., Lepmets, M., Buglione, L.: Harnessing software development contexts to inform software process selection decisions. Software Quality Professional. 16, 35–36 (2013).

[Kro03] Kroll, P., and P. Kruchten. 2003. The Rational unified process made easy: A practitioner's guide to the RUP. Boston: Addison-Wesley Longman Publishing Co.

[Lap12] Laporte, C.Y., Palza Vargas, E., "The Development of International Standards to facilitate Process Improvements for Very Small Enterprises," in Software Process Improvement and Management: Approaches and Tools for Practical Development, IGI Global Publisher, pp. 34-61, 2012.

[Lap98] Laporte, C.Y., Trudel, S. Addressing the people issues of process improvement activities at Oerlikon Aerospace. Software Process-Improvement and Practice, 4(1), pp 187-198 (1998).

[Lap08] Laporte, C.Y., Alexandre, S., O'Connor, R.V.: A Software Engineering Lifecycle Standard for Very Small Enterprises. In: O'Connor, R., Baddoo, N., Smolander, K., and Messnarz, R. (eds.) Proceedings of EuroSPI. pp. 129–141. Springer, Heidelberg (2008)

[Lap13] Laporte, C.Y., O'Connor, R., Fanmuy, G.: International Systems and Software Engineering Standards for Very Small Entities. CrossTalk - The Journal of Defense Software Engineering. 26, 28–33 (2013).

[Lap15a] Laporte, C.Y., Chevalier, F., An Innovative Approach to the Development of Project Management Processes for Small-scale Projects in a large Engineering Company, 25th Annual International Symposium of INCOSE (International Council on Systems Engineering), Seattle, US, July 13-16, 2015.

[Lap14] Laporte, C.Y., Hébert, C., Mineau, C., Development of a Social Network Website

Using the New ISO/IEC 29110 Standard Developed Specifically for Very Small Entities, Software Quality Professional Journal, ASQ, vol. 16, no. 4, 2014, pp. 4-25.

[Lap15b] Laporte, C.Y., O'Connor, R., Garcia, L., Software Engineering Standards and Guides for Very Small Entities: Implementation in two start-ups, 10th International Conference on Evolution of Novel Approaches to Software Engineering (ENASE 2015), Barcelone, Spain, May 29-30, 2015.

[OC09] O'Connor, R., Coleman, G.: Ignoring“ Best Practice”: Why Irish Software SMEs are Rejecting CMMI and ISO 9000. Australasian Journal of Information Systems. 16, (2009).

[OC10] O'Connor, R., Laporte, C.Y.: Towards the provision of assistance for very small entities in deploying software lifecycle standards. Proceedings of the 11th International Conference on Product Focused Software (PROFES '10). pp. 4–7. ACM (2010).

[OC11a] O'Connor, R., Laporte, C.: Deploying Lifecycle Profiles for Very Small Entities: An Early Stage Industry View. In: O'Connor, R., Rout, T., McCaffery, F., and Dorling, A. (eds.) Software Process Improvement and Capability Determination. pp. 227–230. Springer-Verlag, Heidelberg (2011).

[OC11b] O'Connor, R., Laporte, C.Y.: Using ISO/IEC 29110 to Harness Process Improvement in Very Small Entities. In: O'Connor, R.V., Pries-Heje, J., and Messnarz, R. (eds.) Workshop on SPI in SMEs, 18th European Software Process Improvement Conference. pp. 225–235. Springer-Verlag, Heidelberg (2011).

[OC14] O'Connor, R.V., Laporte, C.Y.: An Innovative Approach to the Development of an International Software Process Lifecycle Standard for Very Small Entities. International Journal of Information Technology Systems Approach. 7, 1–22 (2014).

[Pot092] Potter, N., Sakry, M., Making Process Improvement Work. Addison-Wesley-Pearson Education, 2002.

[Tru14] Trudeau, P.O., Laporte, C.Y., Lévesque, S., Enseignement de la norme ISO/CEI 29110 aux étudiants en technique informatique d'un collège technique québécois, Revue Génie Logiciel, Numéro 110, septembre 2014, pp 33-45.

[San15] Sanchez-Gordon, M.L., O'Connor R.V. and Colomo-Palacios, R., Evaluating VSEs Viewpoint and Sentiment Towards the ISO/IEC 29110 Standard: A Two Country Grounded Theory Study, In Rout, O'Connor, R.V. and Dorling. (Eds), Software Process Improvement and Capability Determination, CCIS 526, Springer-Verlag, 2015.

Process Assessment Issues in a Bachelor Capstone Project

Vincent Ribaud
Univ. Bretagne Occidentale,
UMR 6285, Lab-STICC
CS 93837, F-29200 Brest, France
ribaud@univ-brest.fr

Alexandre Bescond, Matthieu Gourvenec, Joël Gueguen, Victorien Lamour, Alexandre Levieux, Thomas Parvillers
Univ. Bretagne Occidentale, FR, Bachelor of Computer Science
{FirstName.LastName}@etudiant.univ-brest.fr

Rory V. O'Connor
School of Computing, Dublin City University
Glasnevin, Dublin 9, Ireland
roconnor@computing.dcu.ie

Abstract

Based on a small subset of ISO/IEC 15504:2006, a Process Assessment was performed in the capstone project of a Bachelor in Computer Science. Parallel to this assessment, students performed a continuous self-assessment using an ability model based on 15504 Base Practices and Work Products. This paper highlights how students' self-assessment and teacher's assessment are correlated. The capstone project itself implements major constructivism principles. This paper presents also the students' point of view through different questionnaires and students' participation to the paper.

1. Introduction

The ACM Computing Curricula [ACM05] establishes the following requirement for a Bachelor curriculum: *"Demonstration that each student has integrated the various elements of the undergraduate experience by undertaking, completing, and presenting a capstone project."* The capstone project is intended to provide students with a learning by doing approach about software development, from requirements to qualification testing. Indeed, the project progress is sustained by software processes. It helps students to be conscious about and improve what they are doing when processes are replaced in a whole picture and when a continuous assessment provide them with objective feedback. Hence, a main capstone teacher's activity is to assist students with appreciation and guidance, a task that relies on the assessment of students' practices and students' products. Students were encouraged to perform a self-assessment in parallel of the teacher's assessment. Consequently, we implemented an experimental protocol to observe how students' self-assessment and teacher's assessment are correlated.

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

Our implementation of a capstone project results from a twenty years experience about project and problem-based learning for software development. From the designer's side - the teacher, most constructivism principles are taken in account and implemented. However, what's up from the constructors' side – the students – The question was raised to the class using several questionnaires and several students accepted to anonymize answers and to analyze results. Hence they are co-authors of this paper whose structure is: section II presents process assessment, section III statistics and pedagogical practices, section IV the practicum, students and teacher roles. Questionnaires results are intertwined in the sections and commented by students and teacher.

2. Process assessment

The main goal of the capstone project is to learn by doing a simplified cycle of software development through a somewhat realistic project. Until this year, students worked in small teams (2-3 people). Thanks to doubling the hours allocated to the project this year and to avoid too much behaviorist division of labor between students, the capstone project was performed individually from A to Z.

2.1 Software processes

A side-effect goal of the capstone project is to be exposed to some kind of process assessment. We choose a small subset of the ISO/IEC 15504:2006 Process Reference Model, mainly the Software-related Processes of the ENG Process Group [15504-Part 5]: ENG.3 System architectural design, ENG.4 Software requirements analysis, ENG.5 Software design, ENG.6 Software construction, ENG.7 Software integration, ENG.8 Software testing. Process Purpose, Process Objectives and Base Practices have been kept without any modification; Input and Outputs Work Products have been reduced to the main products.

We recall some definitions from the ISO/IEC 15504 standard [15504]: "processes are grouped according to the type of activity they address: the processes included in the same group contribute to a complementary area", "a process is a set of interrelated or interacting activities which transforms inputs into outputs", "a base practice is an activity that, when consistently performed, contributes to achieving a specific process purpose", and "a work product is an artifact associated with the execution of a process".

2.2 Ability model

From an individual human perspective, the 15504 Exemplar Process Model can be seen as a competencies model related to the knowledge, skills and attitudes involved in a software project. A competencies model defines and organizes the elements of a curriculum (or a professional baseline) and their relationships. During the education period, all the students use the same model but they can individually change afterwards.

A hierarchical model is easier to manage and use. We kept the hierarchical decomposition issued from the 15504: process groups –process – base practices and products. A competency model is decomposed into competency areas (mapping to process groups); each area roughly corresponding to one of the main division of the profession or of a curriculum. Each area organizes the competencies into families (mapping to processes). A family roughly corresponds to main activities of the area. Each family is made of a set of knowledge and abilities (mapping to base practices), eventually called competencies; each of these entities being represented by a designation and a detailed description.

2.3 Process Assessment

ISO 15504 [15504] defines a measurement framework for the assessment of process capability defined on a six point ordinal scale which represents increasing capability of the implemented process, from not achieving the process purpose through to meeting current and projected business goals. [15504-2]. Within this measurement framework, the measure of capability is based upon a set of process attributes (PA). Each attribute defines a particular aspect of process capability. The extent of process attribute achievement is characterized on a defined rating scale: N Not Achieved, P Partially Achieved, L Largely Achieved, F Fully Achieved. Capability Level 0 denotes an incomplete process, either not performed at all, or for which there is little or no evidence of systematic achievement of the process purpose [15504-3]. Capability Level 1 denotes a performed process that achieves its process purpose through the performance of necessary actions and the presence of appropriate input and output work products which, collectively, ensure that the process purpose is achieved [15504-3]. Higher levels denote higher process maturity: the process is managed (Level 2), established (Level 3), predictable (Level 4), optimizing (Level 5).

If students are able to perform a process, it denotes a successful learning of software processes, and teachers' assessments rate this capability. Because we believe that learning is sustained by continuous assessment, self-directed, done by teachers or a third-party, the research question aims to state how students' self-assessment and teacher's assessment are correlated and if self-assessment for performing BP and delivering WP is an alternative to external assessment about 15504 Level 1. Obviously, the assessment main goal is students' ability to perform the selected processes set.

3. The capstone project

This section overviews the project and assessment results, then presents each process with assessment details, teacher's analysis and students' comments.

3.1 Overview

3.1.1 Schedule

The curriculum is a 3-year Bachelor of Computer Science. The project happens the third year before students' internship. The project is performed during a period of 2 weeks. Before the dedicated weeks, 40 lecture hours are dispatched all the semester along and some homework is required. Ideally, students should be familiar with the Author-Reader cycle as the project starts and have performed the requirements and architectural design processes. Each deliverable can be reviewed as much as needed by the teacher that provides students with comments and suggestions.

3.1.2 System architecture

The system is made of 2 sub-systems:

• PocketAgenda (PA) for address books and agenda management and interface with a central directory;

• WhoIsWho (WIW) for managing the directory and a social network.

PocketAgenda is implemented with Java, JSF relying on a Oracle RDBMS. WhoIsWho is implemented in C or Java using a small RDBMS or files. Both sub-systems communicate with a protocol to establish using UDP.

The system is delivered in two batches. Batch 1 scope is: PocketAgenda – address book and directory interface; WhoIsWho - directory management. Batch 2 scope is: PocketAgenda – agenda and social network interface; WhoIsWho – social network management.

3.1.3 Rating scheme

Table 1 presents the rating scheme. Students' assessment was continuous and communicated to students regularly; hence they have been made aware of their progression each day and adjusted their effort.

Table 1: Rating scheme

Process	Work product	Pt.
	Batch 1	
ENG.4	Use cases –Social network	1
ENG.3	Interfaces specification	1
ENG.5	Detailed Design Document	2
ENG.6	4GL applications	3
ENG.6	Network application	3
	Batch 2	
ENG.4	Use cases	3
ENG.6	4GL applications	2
ENG.6	Java/SQL application	1
	Project	
ENG.7	Integration schema, configuration, version sheet	1
ENG.8	Test reports	1
Attitude	Assiduity, commitment, organization	2
Total		**20**

3.1.4 Statistics

Table 2 presents teacher's assessment. BP and WP rating are aggregated using an all-or-none principle: if all BP or WP in a process are rated at least Largely (or Fully), the process is rated Largely (or Fully)[1]. At the two-third of the project, students have been made aware of the Level 2 and its attributes. However, the teacher has not enough time to track the PA 2.1 Performance management and only the PA 2.2 Work product management was tracked for the most advanced students: those who were assessed by the teacher for all processes at L or F; it represents 7 students over 23.

Table 2: Teacher's assessment

	BP level 1		*WP level 1*		*WP2*
	L	*F*	*L*	*F*	*L*
ENG.4 Requirement	3	13	7	10	3
ENG.3/5 Design	7	3	8	8	2
ENG.4 Construction	4	6	10	6	4
ENG.7 Integration	7	0	5	2	1
ENG.8 Testing	6	10	12	3	1

As the project ends, students have to complete a summary including hour's breakdown and self-assessment of achievement for each process. Summary was mandatory and 22 students over 23 completed it. Table 3 presents students' self-assessment and the average hours spent for each process. Last column indicates the number of times where the teacher's assessment matches the student's self-assessment.

Table 3: Overview of self-assessment and match

	Hrs	*N*	*P*	*L*	*F*	*Match*
ENG.4 Requirements	20	0	0	8	14	18
ENG.3/5 Design	19	0	0	11	11	16
ENG.4 Construction	48	0	6	9	7	14
ENG.7 Integration	9	4	7	9	2	11
ENG.8 Testing	5	2	7	9	4	10

3.1.5 Information about students

The class comprises 24 students. One gave up in the middle of the project. Among 23 remaining, 3 students' projects failed, 4 projects were barely satisfactory, 9 good, 5 very good and 2 excellent. Questionnaires were

[1] BPs that are a kind of Develop test criteria or Develop test procedures, are out of scope and excluded from aggregates.

completed by 22 students. 6 students have participated to the writing of this paper and were classified as: 1 project failed, 1 was barely satisfactory, 1 good, 2 very good and 1 excellent.

A unique student works in parallel. 20 completed first and second year in our Bachelor. 17 were assiduous. 15 repeated at least a class before the Bachelor final year (in high school or at the university). 15 were able to perform the project outside the university labs. 10 claimed to have a good knowledge of SQL and Java before the project.

3.2 Project progress

Students were advised that they can freely participate to the following experiment: they will have to regularly update a competency model comprising the ENG process group, the 6 processes above and their Base Practices and main Work Products and self-assess on the N-P-L-F scale. The teacher will also assess the same BPs and WPs and volunteers students will correlate self-assessment and teacher's assessment and deliver anonymous results for the paper. All students did agree with the experiment but only 18 delivered the completed competency model to volunteers. The data distribution is presented in tables in each process subsection. The match with teacher's assessment is indicated as the last column of each table. Teacher analysis and comments made by students co-authoring the paper are reported at end of process subsection.

3.2.1 Requirements

According to students' estimates average, they spent 20 hours over 102 total hours to capture, write and manage requirements through use cases. A 4-hour lecture about use cases was delivered in January at the beginning of the semester, then the iterative process of writing and being reviewed by the teacher started. When the project full-time period had started, 6-7 students over 22 have completed the requirement process and produced the requirement specification WP. The remaining completed theses tasks during the project. Without surprise, the more backward students (for this task or the following one) failed.

Table 4 presents main Base Practices (ENG.4.BP1: Specify software requirements; ENG.4.BP3: Develop criteria for software testing; ENG.4.BP4: Ensure consistency) and main Work Products (17-11 Software requirements) for the ENG.4 Software requirements analysis process.

Table 4: ENG.4 assessment (self and teacher)

	N	*P*	*L*	*F*	*Match*
BP1. SW requirements	0	2	9	7	6
BP3. Test criteria	0	5	7	5	2
BP4. Consistency	1	3	7	7	8
17-8 Interface requirements	0	3	1	8	6
17-11 SW requirements	0	1	10	7	9

Thanks to the Author-Reader cycle, specification writing iterates several time during the semester and the final mark given to almost SW requirement document was Fully Achieved. However matching between students and teacher assessments is poor. A deeper look on data yields a possible explanation: “good” students, despite the excellent final mark, were aware of the cycle and the improvement suggested by the teacher at each iteration, hence they self-assess generally as Largely Achieved whereas the teacher rated a Fully Achieved; “normal” students took the final mark as the level they achieved and self-assessed as F whereas the teacher rated a L.

Clearly, students did not understand the ENG.4.BP3: Develop criteria for software testing and failed the self-assessment. The definition is “*Use the software requirements to define acceptance criteria for the software product tests. Software product tests should demonstrate compliance with the software requirements*. [15504-Part 5]” The teacher defined acceptance criteria and students were not aware of this topic, however they confused “develop criteria for SW testing” and “testing SW” and self-assessed at a much higher level that the teacher did.

Students' comment. It was the first time that we have to write use cases from a statement of work. Eliciting and writing requirements were difficult and the Author-Reader cycle helped to produce complete and usable use cases and to acquire a writing style. Because of the novelty of the task and to achieve a certain maturity degree, it is required to start the writing task early in the semester.

3.2.2 Architectural and detailed design

On average, students spent 19 hours over 102 total hours to perform architectural and detailed design. Design is split in data modeling, Web-based design and oriented-object design. The PocketAgenda subsystem is structured around a database schema. Modeling is performed using SQL Developer Data Modeler, freely available through the Oracle Academy program. Data architectural design results in a Logical model, data detailed design (obvious in that case) is performed through automated forward engineering and results in a Relational model. A 2-hour lecture about Data Modeler was delivered in February after the use cases phase. then the iterative Author/Reader cycle started.

Jdeveloper is a Java IDE for the Oracle Application Development Framework (ADF). ADF is an end-to-end development framework, built on top of the Enterprise Java platform, and providing integrated solutions including data access, business services development, a controller layer, a JSF tag library implementation. 12 labs hour were devoted to learning the framework, insufficient for mastering the IDE but enough for a quick start.

UML modeling and object-oriented design are taught in dedicated lectures during the curriculum (30 hours each). However, nearly all students had no idea how to perform the design. Design was taught by example: students have developed a component of the batch 1 from a design document provided by the teacher. Then they had to develop another batch 1 components and retro-design their development. Finally they had to establish the design of remaining components.

Architectural design was also shown by example: a complete cycle was provided for one networked function: use case, interface specification, design for the client and server sides, client and server stubs program. Students reproduced the scheme.

Table 5: ENG.3 and 5 assessments (self and teacher)

	N	*P*	*L*	*F*	*Match*
BP1. Describe syst. arch.	0	4	10	5	6
BP3. Define interfaces	0	4	6	8	6
BP3. Detailed design	0	2	9	7	7
BP4. Consistency	1	2	9	6	8
04-01 Database design	0	1	6	11	13
04-04 High level design	0	4	6	8	8
04-05 Low level design	0	2	8	8	11

Table 5 presents main Base Practices (ENG.3.BP3: Define interfaces; ENG.5.BP3: Develop detailed design; ENG.5.BP5: Ensure consistency) and main Work Products (04-01 Database design; 04-04/05 High/low level SW design) for the ENG.3 et 5 System and software design process.

Again, matching is poor, except maybe for technical design. A similar concern to requirements arose with design: a few students were aware of the improvement cycle performed by the Author-Reader cycle and took the Work Product (Design Document) as an indication of their achievement. Another explanation is related to the fact that bachelor students are focused on technology, hence there are more able to self-assess on technical tasks (Database or Detailed Design).

Students' comment. Requirement specifications greatly helped to figure out the system behavior and facilitated the design phase and interface specification. However, students had never learnt architectural design and interfaces between sub-systems. Design time has to be immediately followed by coding time and could not spread along the semester as we did it for requirements. Students performed high level design for a batch and low level design for the other, and both have advantages depending on the student's personality: either creative or preferring to be guided.

3.2.3 Construction

On average, students spent 48 hours over 102 total hours to develop the software. Java, network programming and database / SQL programming are taught in dedicated lectures during the curriculum (60 hours each). Despite of this amount, 12 students self-judged as having a poor knowledge of SQL and Java, and 10 students were unable to develop the client-server application although a Java server skeleton has been provided. Time constraints also played their role: because the network component was perceived by difficult by some students, they did not commit to the work and invested others more cost-effective tasks. Students have almost no idea of test-driven development and a lack of a test strategy; hence unit were poorly tested. This point has to be addressed in the next edition.

Table 6 presents main Base Practices (ENG.6.BP1: Develop unit verification procedures; ENG.6.BP2: Develop SW units; ENG.6.BP3: Ensure consistency; ENG.6.BP4: Verify SW units) and main Work Products (11-05 Software unit; 14-04 Test log) for the ENG.6 Construction process.

Table 6: ENG.6 assessment (self and teacher)

	N	*P*	*L*	*F*	*Match*
BP1. Verification procedures	2	3	10	3	6
BP2. Develop units	0	5	8	5	8
BP3. Consistency	0	7	8	3	9
BP4. Verify units	0	7	8	3	6
11-05 Software unit	0	4	9	5	8

Unit testing is a little more familiar to students, and although they probably misunderstood the ENG.6.BP1: Develop unit verification procedures; the matching is not so worse that for the ENG.4.BP3: Develop criteria for software testing. The discrepancy between students and teacher assessments about ENG.6.BP2: Develop software units stems from the “goggle-paste” phenomena; only a few students writes his/her own code and has been assessed at the Largely or Full level by the teacher; most students adapt code from others without a real understanding of the programming activity and over-assess themselves.

Students' comment. This process raised a certain anxiety because students had doubt about their ability to develop a stand-alone server interoperating with a JDeveloper application. Students had never learnt a 4GL (fourth generation language) environment such as JDeveloper, hence they reported that the switch from a 3GL to a 4GL was difficult but once understood, they appreciated the power leverage of such environments. The majority of students whose successfully developed the client-server component reported that they could not achieve it without the help of the skeleton provided by the teacher. For some students, a poor Java literacy prevent them to struggle with the network part. Some students failed because they jumped to code before having any draft or idea to realize it.

3.2.4 Integration and tests

On average, students spent 15 hours over 102 total hours to integrate and perform qualification tests of the software. These topics are unaddressed in the curriculum and because they mostly occur at the end of the project, no time was available to complete the learning. In the best cases, students have respected their interfaces specification and few problems arose when they had to integrate the Java client program within the JDeveloper application. In other cases, they were unable to perform the integration and the assessment was partial and based on the Java client code. Test cases specification stemmed from use cases, hence no test plan was required. Test procedure was reduced to test each use case - success scenario and main extensions, to verify the conformity to use cases and the results achieved.

Table 7 presents main Base Practices (ENG.7.BP3: Integrate software item; ENG.7.BP5: Ensure consistency; ENG.8.BP1: Develop tests for integrated software product; ENG.8.BP2: Test integrated software product) and main Work Products (08-21 Software test plan; 11-01 Software product; 14-04 Test log) for the ENG.7 et 8 Software integration and software testing process.

Table 7: ENG.7 and 8 assessments (self and teacher)

	N	*P*	*L*	*F*	*Match*
BP3. Integrate SW items	2	4	10	1	9
BP5. Consistency	2	3	11	1	9
BP1. Develop tests	2	5	10	2	4
BP2. Test product	0	5	9	3	9
08-21 Software test plan	0	4	11	2	3
11-01 SW product	0	4	9	4	9
14-04 Test log	0	4	11	2	10

We observe the same poor correlation for the ENG.8.BP1: Develop tests for integrated software product and the WP 08-21 Software test plan, indicating that students are not aware of the test definition and planning activity, a common hole in a Bachelor curriculum although testing is an ability strongly required by employers.

Integration is also an uncovered topic and students are not aware of the subject: for the ENG.7.BP3: Integrate software item, 11 students (over 18) were assessed by the teacher as Not or Partially whereas they only 6 self-assessed N or P.

Students' comment. Some students were aware of the poor maturity of the integrated product, partly due to the lack of testing. Although the Junit framework has been taught during the first semester, some students did not see the point to use it while some others did not see how to use it for the project. Students that did not develop the server had no integration to perform.

4. Students and teacher roles

Constructivism can be summed up with two fundamental statements [Duf96]: (i) learning is defined as an active process for knowledge building rather than a knowledge acquisition process; (ii) teaching is essentially aimed at helping students in this process rather than transmitting knowledge.

Among practices belonging to the constructivist stream, Dwyer [Dwy94] and Tardif [Tar98] define a learning paradigm, in opposition with the main teaching paradigm. The learning paradigm provides a framework which allows the school to constitute a learners' community for the pupils as well as the teachers and the other staff members.

This section aims to relate the educational system with the new roles required in a constructivism approach. The questionnaire collects anonymously students' perception about roles. Teacher's role has to be rated on the scale used to rate practices and products: Not achieved, Partially Achieved, Largely Achieved, Fully Achieved. Students' self-opinion about their roles and about the practicum are expressed on a 5-point Likert scale from Strongly Agree to Strongly Disagree.

4.1 Teachers' role

Tardif [Tar98] defines teachers' roles as creators of pedagogical environments; interdependent, open-minded, critical professionals; development instigators; mediators between knowledge and students; coaches; collaborators for the students' success of a whole school. The first role was questioned in a special part of the questionnaire related to the educational system and is presented in section 4.3. Table 8 presents students' rating about the teacher's roles.

Table 8: Students' rating about teacher's roles

	?	N	P	L	F
a professional, open-minded and open to criticism	1			4	17
a development instigator			1	8	13
a mediator between knowledge and students			2	7	13
a coach	1		4	6	11

Students' comment. Students agree with the teacher's roles required. The majority of students want to be instigated but not directed to a solution. Some students stated that teachers fall into two categories: those that don't care of students and those that help too much and deprive them of autonomy because they want to control the learning results. They appreciated the balanced teacher's attitude and to be on his or her own but also to have a teacher in case of emergency. Students noticed that the teacher wanted that everyone speak, discuss and compare points of view and aimed an active participation. Some students complained that the teacher did not share his time equally between students and pointed out that a second teacher will be useful.

4.2 Students' role

Tardif [Tar98] defines students' roles as investigators; co-operators sometimes experts; clarifying actors; strategic users of available resources. The questionnaire set the following definitions: *investigator*: I discussed with other students my questions about the project and/or I defended my solutions; *co-operators sometimes experts*: I explained some project points to other students and/or I had myself explanations from others; *clarifying actors*: I asked the teacher or other students in order to insure my good project understanding and to verify the adequacy of my proposals; *strategic users of available resources*: I used the available resources and/or supplementary resources and I verified their relevance. Table 9 presents students' perception about their roles.

Table 9: Students' self-perception about their roles

	strg agr	agr	neu-tral	dsgr	strg dsgr
investigator	12	8	2		
co-operators - experts	10	11		1	
clarifying actors	14	7	1		
strategic users	7	8	6		

Students' comment. Some students underestimate themselves and some definitions (strategic users, for instance) were seen as out of the reach and they could not use it to qualify themselves. However students have learnt to debate, find and explain solutions. Students learnt a lot about to work with consistency and traceability, to respond to demands within the recommended time and to log his or her work in order to notice the project progress.

4.3 The practicum

Tardif [Tar98] defines the characteristics of a pedagogical environment (the practicum) consistent with the learning paradigm: constancy of learning and time variability; cognitive imbalance; authenticity of learning situations; transdisciplinarity; interactions between theory and practice; embedment of assessment within the learning situations. The last part of the questionnaire let students express their opinions about the practicum, which are presented in Table 10.

Table 10: Students' self-perception about the practicum

The Agenda project	strg agr	agr	neu-tral	dsgr	strg dsgr
I had the time to learn and do the project.	6	11	2	3	
I found the project complex.	5	12	2	3	
I committed to perform the project.	14	6	1	1	
I found the project realistic.	11	8	1	1	1
I understand relationships between specifications, design, building and tests.	15	6	1		
I had to deepen my knowledge and skills to perform the project.	10	12			
My work for the project helped me to understand lectures.	5	6	8	1	2
I used a lot the reviewing facilities.	7	7	1	5	2
I made progress thanks to the reviewing facilities.	12	5	2	1	2
I improved my working methods thanks to the project.	6	9	7		

Although one project objective is to relate to previous lectures and to mobilize knowledge and skills gained during the bachelor studies, it was not effective and rather seen as a new learning experience, although some students have enjoyed the project as an experience to deepen the different notions of program seen and learned during lectures. We were surprised with the relatively poor use of reviewing.

Students' comment. Students appreciated that each project phase has been explained from experience and through examples. Students have been convinced of the usefulness of the different phases performed in a software project and that it might be applied to other type of projects. Generally speaking, students prefer project to labs. Using on-line tutorials as a learning support is appreciated, but some students complain about the quality of some tutorials written by the teacher. A forum could be useful to share knowledge and help others people. Shared documents could be an alternative to mail exchange and might trigger the use of reviewing facilities that some students misused. Students asked to be exposed to a whole picture of the project at the beginning and to start the project having all project documents at their disposal. Some students found the work load too heavy and time devoted to the project too short. As a student said, all students learned something during the project, and some students have learned more than others!

5. Conclusion

The research question aims to see how students' self-assessment and external assessment [by a teacher] are correlated. This is not true for topics not addressed in the curriculum or unknown by students. For more classical topics, assessments are correlated roughly for the half of the study population. However, the study is a suffering from a bias due to the learning process: deliverables go through a Author-Reader cycle that leads to improve them sufficiently to achieve a Largely Achieved or Fully Achieved level but only "good" students are aware of the help provided by the teacher at each iteration. Hence "good" students under assess themselves whereas "normal" students over assess themselves considering that the resulting deliverable is a witness of their achievement level. The bias invalidates partially the experiment that has to be set again outside of a learning situation.

Questionnaire and students-authors contribution indicates that the system favors knowledge building, encourage students to work in an active way, develop autonomy and success feeling, improve assessment and may develop mutual help; what is expected in a successful project-based learning situation. Process learning seems to be effective for requirements, design and building but we need to improve the system for the ENG.7 SW integration and ENG.8 SW testing process.

6. Acknowledgments

We thank all the students of the 2014-2015 final year of Bachelor in Computer Science to their agreement to participate to this study.

References

[Duf96] T. M. Duffy, D. J. Cunningham. Constructivism : Implications for the design and delivery of instruction. *Handbook of Research for Educational Communications and Technology*, MacMillan, 1996.

[Dwy94] D. Dwyer. Apple Classrooms of Tomorrow: What we have learned. *Educational Leadership*, 54(7), 1994.

[15504] ISO/IEC 15504:2004. *Information technology -- Process assessment*. ISO, Geneva, 2006.

[Tar98] J. Tardif. *Intégrer les nouvelles technologies de l'information – Quel cadre pédagogique ?*. ESF, 1998

The SPI Manifesto and the corresponding ECQA certified SPI manager Training

Tomas Schweigert
Software Quality Systems AG
Cologne
Germany
tomas.schweigert@sqs.com

Miklos Biro
Software Competence Center Hagenberg GmbH,
Hagenberg
Austria
miklos.biro@scch.at

Abstract

In order to define a modern approach for Software Process Improvement (SPI), the SPI Manifesto was developed, discussed, and finalized at the 16th EuroSPI Conference in 2009 in Alcala, Spain. The common understanding gained during the discussion and usage of the manifesto formed a nucleus from which the new ISO/IEC 33014[ISO01] was derived. In parallel to the development of the SPI Manifesto, the SPI manager Training was developed by the European Certification and Quality Association (ECQA) under Sponsorship of the European Commission. The goal of the training was to address all topics mentioned in the SPI manifesto and also include additional topics seen as useful for mangers dealing with SPI.

Key-words: Software Process Improvement. ISO 33014, SPI Manifesto, SPI Manager Training, Agile manifesto.

1. Introduction

In the 1st decade of the 21st century, it became clear that system and software development are going to face a massive change of paradigms. Following ISO 9001 and / or CMMI level change campaigns many organisations implemented Software Engineering Process Groups (SEPG)[Kasse04] that created a huge amount of process bureaucracy. And a lot of these SEPG's failed when challenged with the speed requirements of the internet age. So in 2001 a group of experienced software managers developed and published the agile manifesto. While first implementations like extreme programming (XP) did not seem to have real influence on SPI, things changed radically when Scrum became part of the game. Even if some SEPG's recognized that a well organised agile development could be rated as CMMI level 3, they did not find an approach to help organisations speeding up.

This situation forced a complete rethinking of what SPI should be and could be. Instead of bureaucracy and intranet publishing business success, change and people involvement were seen as the new cornerstones of SPI management.

In order to enable SPI responsibles to step up with these new paradigms an SPI manager training was created by a team of experts forming the SPI manager Job Role Committee (JRC) at ECQA. 1st trainings were launched in 2010 making the background knowledge of the SPI manifesto available to the market [Schweigert09], [Korsaa12].

2. The SPI Manifesto

In this section, the SPI manifesto [PriesHeje10] will be presented, values and principles will be described.

2.1 Background

When we look at sources like the Chaos Report also referred to in [Standish2014], we have a stable set of up to 40 % of large software projects that fail and another 50% with delay, poor product quality and budget overrun. The typical reaction of organisations is:

- Invest in new technologies
- Formalize processes

In most cases, this approach ends up in a mail stating that the projects have to use a new tool and that new and mandatory processes are published on the intranet.

At the end, the projects realise –if they are able to find the new processes in the intranet – that the new tool provides poor support to the new processes and has insufficient interfaces to the rest of the software development environment.

As a result, the new tool and processes are ignored or additional time and budget is spent by the project to make the tool and processes suitable for the needs of the project. The effect is delay and budget overrun. At the very end, process performers and management loose confidence in process improvement.

Similarly to the Chaos Report, Joyce Statz, Don Oxley and Patrick O'Toole analyzed the major risks for SPI failure

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

[Statz97]. They found out, that there are some typical risks for SPI projects like

- Organization instability, such as high employee turnover.
- Frequent structural changes in the organization (every six months or less for the last several years).
- Significant change in business focus in the last three to six months.
- Threat of or impending sale (or merger) of the organization.
- Lack of management commitment (lack of personal time invested in SPI).
- Unfounded expectation of a fast pace of change.
- Lack of knowledge about how to proceed.
- Project management and project team resistance to change.
- Project schedules that restrict time for learning about new processes.
- Difficulty in measuring the impact of changes because of a previously chaotic environment.

Even if this analysis was done several years ago, from practical experience it still addresses the main risks of SPI projects. We can't prove that 70% of all SPI initiatives still fail [Gilb04], [Centraline04] but we did not find younger studies that deliver more optimistic data either. Indeed, there are other sources that add some risks to the portfolio

- Poor communication
- Critical success factors not identified

As we can see, there are a lot of risks and challenges in SPI projects, some more generic as they might apply in all projects, others very specific to SPI. For readers interested in detailed further reading, there is a compilation of references developed by Mark Paulk [Paulk10].

But there are not only issues which we would attribute to management techniques, there are also personal demotivators which have to be taken into account [Baddoo08], [Standish2014], e.g.

- Budget Constraints
- Commercial pressures
- Cumbersome processes
- Customers
- Fire Fighting
- Imposition
- Inertia
- Irrelevant objectives
- Isolated best practices
- Lack of evidence of direct benefits
- Lack of feedback
- Lack of SPI management skills
- Lack of standards
- Large Scale programmes
- Low process priority
- Negative / bad experience
- Organisational changes
- Personality clashes
- Reduced creativity
- Workload

On the other hand, we find lots of success stories. So in principle, successful SPI must be achievable. We think one key factor is a skilled SPI management. Experts like Tim Kasse have always stated that there is a need for the Role Process Improvement Manager. A more detailed organisation of SPI management can be found in the book by Johansen & Pries-Heje [Pries-Heje13].

We see that there is need for SPI management skills and a consensus that there should be formal SPI related roles in SPI projects. To satisfy this need, a formal SPI manager education is needed beyond standard project management or standard process engineering. Properly educated SPI management experts will make sure that the money spent by the management and the support of the process performers does lead to success and not to frustration.

2.2 Values of the SPI Manifesto

The SPI Manifesto states 3 core principles of modern SPI:

- We truly believe that SPI must involve people actively and affect their daily activities
- We truly believe that SPI is what you do to make business successful
- We truly believe that SPI is inherently linked with change [PriesHeje10]

2.3 Principles of the SPI Manifesto

The principles of the SPI manifesto were organised around the given values.

2.3.1 People Involvement

The need for people involvement was declared by the SPI manifesto using the following arguments:

- A.1 Context and problem In the last decade we have seen the growing of ivory towers in many organisations, using 'magic' tools and models to paint process diagrams. However, in most of these organisations, the projects and services did not really use these processes. So the ivory towers have had limited success as drivers of SPI, and it is now time to bring SPI to the people who will be most affected.

- A.2 Value explained Business success depends on the competitiveness of an organisation. The competitiveness of every organisation is based on the knowledge, engagement and commitment of the people working in it. SPI is a tool to improve the competitiveness of organisations. Bringing this together, we believe it becomes clear, that only active involvement of the people working in an organisation ensures the success of an SPI initiative from the business perspective! Successful SPI is based on actively involved people having sufficient information and training.
- A.3 Hints and examples The modern organisation paradigm is a change from experts solving problems and trying to force change on organisations to the organisation's people solving problems and changing the organisation together. Japanese improvement efforts such as Kaizen have demonstrated this convincingly in the 20th century. More recently, we also see this in the growing success of agile development approaches. Enablers for success in modern organisations include people making full use of their experience, taking responsibility for change on their project and throughout their organisation, and using and improving the processes they have helped to define.

Given that explanation the following Principles were added to this value:

- Know the culture and focus on needs
- Motivate all people involved
- Base improvement on experience and measurements
- Create a learning organisation

Even if the value and the associated principles seem to be clear and concise, creating an associated training forced the JRC team to deal with the work of Hofstede, Zur Bonsen, Robert W. Jacobs, and the ISO/IEC 15939 [ISO02].

2.3.2 Business Success

The focus on business success was declared by the SPI manifesto using the following arguments:

- B.1 Context and problem: The software process creates software. Software Process Improvement means activities that improve the way of creating and implementing software. However, many people believe that they don't need processes in order to build and ship software products. This belief may be the source of most resistance to change met by SPI professionals. But the fact is that you cannot create software without process. Another problem is when 'process' is seen as 'somebody else's process description.' This again leads to the misconception that one can do without process. Software should not be created without process; however, what is important is that you have process that fits the need of your projects and your business.
- B.2 Value explained: Process descriptions are just words – we believe the process should bring value to the business. To have success with SPI we must ensure that improvement recommendations are targeted to the actual business-related objectives, rather than compliant with a generic standard. We must also close the gap between 'the process' and 'how the work is really being done'; we believe that words and actions consistently should communicate the unity of the two – not the decoupling. B.3 Hints and examples: Use today's implemented processes as an agreed baseline for process improvements. Understand the business objectives in order to ensure that suggested improvements will be effective in supporting these. Always refer to the process description as a representation of the process. Communicate how standards and models is meant to support SPI. This continuous communication at all levels of management and practitioners helps managers and practitioners to understand how and why they need to support the SPI activities. If you are using a maturity model to inspire improvement, you should respect that at maturity level 3, the process belongs to the organisation. At maturity level 2, the process belongs to the project. And at maturity level 1, the process belongs and exists at the individual level.[PriesHeje10]

Given that explanation the following Principles were added to this value:

- Support the organisation's vision and objectives
- Use dynamic and adaptable models as needed
- Apply risk management

2.3.3 Change

The relevance of change was declared by the SPI manifesto using the following arguments:

- C.1 Context and problem: Only in a perfect world is there nothing to improve. We believe that all improvement involves change; for the individual, the project, and the organisation. We know that it is difficult for people to accept or adopt change, because they are comfortable doing things the way they always have, even if it costs them overtime or loss of social interaction. Never the less, we need to face the need for change when doing SPI.

- C.2 Values explained and Interpreted: So Software Process Improvement means change! Realising this means an organisation must ensure that the process improvement infrastructure has a change management component in it. It is essential for an organisation to launch a process improvement initiative and to obtain measurable business results together with satisfied employees.
- C.3 Example :An IT organisation in a predominantly Asian culture wanted to enact a SPI program and achieve CMMI Maturity level 3 at the same time. One change required was to institutionalise peer reviews. But practitioners did not want to review colleagues' work and offer input that suggested major defects were found and needed to be corrected. Peer review training was repeated every six months, while videotaping the consultant coaching a live peer review. After three years, the results of using peer reviews could not be cost justified. The consultant explained to the CEO that if major defects were not found in peer reviews, but by the organisation's customers, everyone would lose face, including the top managers. Jobs could be lost as well. The CEO then appointed top middle managers to serve as coaches, and encouraged all project members to participate in peer reviews, concentrating on the most costly major defects. When the practitioners saw management's commitment to change, and saw that no one was getting fired or being demoted because they found and reported major defects, they participated willingly. The product quality went up, jobs were kept, profits increased, and lifestyles improved due to less time needed in finding defects. After a successful assessment, the CEO declared that this cultural change was the most significant event in the process improvement initiative.[PriesHeje10]

Given that explanation the following Principles were added to this value:

- Manage the organisational change in your improvement effort
- Ensure all parties understand and agree on process
- Do not lose focus

These values and principles of the SPI Manifesto cover all aspects of modern SPI

3. The European Certification and Qualification Association (ECQA)

ECQA is a result of a number of EU supported initiatives in the last ten years where in the European Union Life Long Learning Programme different educational developments decided to follow a joint process for the certification of persons in the industry [ECQA10]

Through the ECQA it becomes possible that a person attends course for a specific profession in e.g. Slovenia and performs a European wide agreed test at the end of the course. The certificate will then be recognized by European training organizations and institutions currently in 18 member countries.

To make sure that this works a rigorous approach was implemented based on the components Skill Set, European Test Pool and The ECQA Website www.ecqa.org

- Skills Sets: A defined set of quality criteria that has to be followed to create the learning objectives and syllabus for new professions. Only skills sets which fulfil the defined criteria are accepted by the ECQA.
- European Test Pool: Assuming that a group of training bodies agree on the same skills set, then students must be able to pass a test independently from the region or country in a Europe wide scope. This is the reason why a Europe wide pool of test questions plus European test portals have been set up and allow a cross-European Internet based collaboration. The system is based on the results from the former EQN project 2005 – 2007, and supported and automated by an online system.

ECQA supports currently more than 21 professions in Europe. ECQA offers certification for professions like IT Security Manager, Innovation Manager, EU project manager, E-security Manager, E-Business Manager, E-Strategy Manager, SW Architect, SW Project Manager, IT Consultant for COTS selection, Internal Financial Control Assessor (COSO/COBIT based), Interpersonal Skills, Scope Manager (Estimation Processes), Configuration Manager, and SPI Manager. Currently new professions such as Integrated Mechatronics Designer, E-Learning Manager, and Terminology Manager are being integrated until 2010 /19/

To enable an effective and scalable certification scheme, ECQA has defined common principles /8, 5,18/. The main idea is that the certification body, the training provider and the examination holder are independent entities. They use a common administrative and technical infrastructure which includes course management, eLearning environment and self-evaluation/examination system. Each skills set is described as a skill card. The most essential management concept of ECQA is the Job Role Committee, established separately for each job role. As a summary, the leading principles are /5/:

- Job Role Committee (JRC): One European consortium is built per accepted profession to annually update the skills set and create a European wide test questions pool.
- Defined Certification Rules and Procedures: The acceptance of professions and skills sets and the certification of students are based on defined quality rules and certification procedures.

4. The SPI Manager Training

In parallel to the development of the SPI manifesto the SPI Manager Training was developed under the governance of the ECQA.

4.1 Background of the SPI manager Initiative

As discussed in the context of the SPI Manifesto, a high proportion of software projects actually fail. Most common reasons are known to be the lack of management commitment and unrealistic expectations. This is unfortunately a statement which is too general to be useful for avoiding failures. Digging one step deeper, we find poor understanding of competences, roles and responsibilities of process improvement activities and tasks which led to inadequate training and qualification of various PI professionals. [Messnarz08]

ISO/IEC 15504 Part 4[ISO03] was published in 2004. It became the first nucleus of a body of knowledge of process improvement. Process improvement was described in a cycle of 8 steps. In the following period, many ideas were launched.

In 2008, DELTA Axiom published the results of a three year research program, in which DELTA, the IT University of Copenhagen and four main players in the financial sector studied successes and failures in process improvement. ImprovAbility™ is a powerful model to support improvements by analyzing the need for the improvement, the project, the deployment and the organisation. [Pries-Heje07]

Based on the indentified risks and the defined "nature" of the organisation, ImprovAbility™ can recommend actions and the most suitable change strategy.

The change strategy is recommended out of a set of change strategies grouped into ten families. Each one will be more or less effective depending on the specific context.

- Commanding

Change is driven and dictated by (top) management (owner, sponsor and change agent)

- Employee driven

Change is driven from the bottom of the organizational hierarchy

- Exploration

Change is driven by the need for flexibility, agility, or a need to explore new approaches.

- Learning driven

Change is driven by a focus on organizational learning, individual learning.

- Metrics driven

Change is driven by metrics and measurements

- Optionality

Change is driven by the motivation and need of the individual or group.

- Production organized

Change is driven by the need for optimization and/or cost reduction

- Reengineering (BPR)

Change is driven by fundamentally rethinking and redesigning the organization.

- Socializing

Change in organizational capabilities is driven by working through social relationships. Diffusion happens through personal contacts rather than through plans and dictates.

- Specialist driven

Change is driven by specialists (professional, technical, or domain experts). [Pries-Heje13]

The team behind the ImprovAbility Model also played a central role in the development of ISO/IEC 33014 also the research experience and the ImprovAbility™ practices are incorporated into the SPI manager training.

In parallel, the authors who also contributed to the SPI Manifesto, developed a syllabus for the SPI Manager Training. This development was organized by the EU Cert project since late 2008. The SPI Manager training and certification scheme addresses al lot of process management areas (Software, Systems, Services but also Testing). The vision is that it will cover the full set of PI professions and will be suitable in the future for other domains than software, systems and services. The certification is managed in a non-profit association, called European Certification and Qualification Association ECQA [Schweigert09], [Korsaa12].

4.2 The content of the SPI Manager Training

The SPI Manager Training strongly focusses on the needs of people managing SPI. It was assumed that people manging PI need a broader view on the PI related issues such as culture and change.

The training consists of the following topics:

- Alignment of PI Goals to Business Goals
- Capability and Maturity Models
- Deployment of SPI
- Experience and Good Practice Sharing
- Generic Process Description Models
- Multi Cultural Teams
- Organisational Culture
- PI Change Strategies
- PI Facilitation Techniques
- PI Leadership
- PI Measurement and Analysis
- PI Reporting
- PI Team Communication
- Planning PI
- Process Improvement Models
- Process Measurement
- Process Thinking
- Software Process Design and Process Description Models
- Supporting Top Management [Korsaa12]

These learning blocks have mutual cross references to the SPI Manifesto.

One of the main experiences from training delivery is, that there is real relevance of the role of the attendee.

Attendees with management roles show a much better performance in training and testing than people with technical roles.

Each Topic is linked to a part of the SPI manager Skill set and to the skill requirements of the SPI manager Exam

Even if the Job Role Committee(JCR) recommends to undergo the training before examination it is possible to undergo the SPI Manager exam by just studying the skill set and reading the SPI-Manifesto. It is not required that a formal SPI manager Training was attended.

5. Discussion

The SPI Manager Qualification was presented in several conferences, and also several publications were developed. Participants on management level were very happy with the training.

But even if the relevance and validity of the content were proven by these attendees the sales figures in Germany show that the SPI Manager training did not completely reach its intended audience.

There were also other issues mentioned in the JRC:

- one issue is that maybe people believe SPI is for software, but the PI part is working on nearly all areas [van Loon2012]
- one issue is that professions like SW tester (there are many SW testers in the world) rather earn with exams than with the training because skills are there but there was no way to certify it before
- one issue is that our PI thinking is very much European driven and Japanese, Chinese, USA approaches are not equally represented.

The JRC is currently analyzing this situation and checking if traditional classroom training is adequate for the target group or if new didactical concepts have to be taken into account.

Taking the experience of the day to day work in PI into account, the SPI Manager Skill Set is still valid even in times of SCRUM.

References

[Schweigert09] Tomas Schweigert, Richard Messnarz, Morten Korsaa, Jorn Johansen, Risto Nevalainen, Miklós Biró (2009): The SPI Professional Qualification Scheme. In: Richard Messnarz, Sonja Koinig, Mads Christiansen, Jorn Johansen, Juan Cuadrado Gallego (ed.) EuroSPI'2009 Industrial Proceedings. Alcalá de Henares, Spain, 02/09/2009-04/09/2009. Copenhagen: Delta, 2009. pp. 3.13-3.22. (ISBN:978-87-7398-151-1)

[ISO01] ISO/IEC TR 33014:2013 Information technology -- Process assessment -- Guide for process improvement

[ISO02] ISO/IEC 15939:2007 Systems and software engineering -- Measurement process

[ISO03] ISO/IEC 15504-4:2004 Information technology -- Process assessment -- Part 4: Guidance on use for process improvement and process capability determination

[Korsaa12] Korsaa Morten, Biro Miklos, Messnarz Richard, Johansen Jörn, Vohwinkel Detlef, Nevalainen Risto, Schweigert Tomas (2012): The SPI manifesto and the ECQA SPI manager certification scheme. JOURNAL OF SOFTWARE: EVOLUTION AND PROCESS 24: pp. 525-540. (2012) Link: http://onlinelibrary.wiley.com/doi/10.1002/smr.502/abstract

[Korsaa13] Korsaa Morten, Johansen Jörn, Schweigert Tomas, Vohwinkel Detlef, Messnarz Richard, Nevalainen Risto, Biro Miklos (2013): The people aspects in modern process improvement management approaches. JOURNAL OF SOFTWARE: EVOLUTION AND PROCESS 25:(4) pp. 381-391. (2013) Link: http://onlinelibrary.wiley.com/doi/10.1002/smr.570/abstract

[Baddoo08] Nathan Baddoo, Tracy Hall, Ciaran O'Keeffe: Using Multi Dimensional Scaling to analyse Software Engineers' Demotivators for SPI, EUROSPI 2008 Proceedings p. 5.1 – 5.16.

[ECQA10] ECQA Guide: ECQA European Certification and Qualification Association Guide, in http:/www.ecqa.org/. [Accessed 3 March 2010]

[Pries-Heje13] Jan Pries-Heje, Jørn Johansen: ImprovAbility. Success with process improvement. DELTA, 2013, ISBN 978-87-7398-139-9.

[Mathiassen02] Lars Mathiassen, Jan Pries-Heje, Ojelanki Ngwenyama: Improving Software Organizations: From Principles to Practice, Addison-Wesley, 2002, ISBN 0-201-75820-2.

[Messnarz08] Messnarz Richard, Ekert Damjan, Reiner Michael, O'Suilleabhain Gearoid. Human resources based improvement strategies - the learning factor. Software Process: Improvement and Practice 2008:4, pages 297 - 382 (July/August 2008).

[Standish2014] "Big Bang Boom". The Standish Group www.standishgroup.com/sample_research_files/BigBangBoom.pdf.

[Statz97] Statz, Joyce, Don Oxley, and Patrick O'Toole: "Identifying and Managing Risks for Software Process Improvement," CrossTalk (April 1997), pp. 13-18. In http:// web: stsc.hill.af.mil/crosstalk/1997/04/identifying.asp

[Kasse04] Tim Kasse, Practical insight into CMMI: The look and feel of a successful implementation, Artech House Publishers, 2004, Boston Massachusetts USA. ISBN 1-58053-625-5.

[Gilb04] Tom Gilb: Project Failure: Some Causes and Cures By Tom Gilb; Web publishing 2004. In http://www.webster.edu/ftleonardwood/COMP5940/Student_Files/Project_Failure/ProjectFailure.pdf

[Paulk10] Mark C Paulk, A (Software) Process Bibliography, last update January 2009. In http://www.cs.cmu.edu/~mcp/papers/biblio.pdf. [Accessed 3 March 2010]

[Centreline04] 10 Major Causes of Project Failure. Centreline Solutions Inc. Web publishing, 2004.

[PriesHeje10] Jan Pries-Heje, Jörn Johansen et. al: The SPI Manifesto, In http://newsletter.eurospi.net , European Systems and Software Process Improvement and Innovation [Accessed 3 March 2010]

[van Loon 2012] Using Target profiles in the real world, 12th international Conference SPICE 2012 Proceedins P 286-288

An Education-oriented ISO 26262 Interpretation Combined with Constructive Alignment

Barbara Gallina
IDT, Mälardalen University,
Västerås, Sweden
barbara.gallina@mdh.se

Abstract

Safety standards (e.g., ISO 26262) define safety life-cycles to be adopted for the development of safety-critical systems. Professionals (i.e., safety engineers, safety managers, and, more broadly safety culture-aware personnel) who are responsible of the development of such systems can be, in turn, considered as safety-critical systems. Course-modules aimed at forming such professionals are critical. Given the criticality of such modules, the intended learning outcomes, before being constructively aligned [Biggs07] with teaching / learning / examination activities, should be derived by applying an education-oriented risk-driven process. The typical "what if" questions aimed at brainstorming on what if something goes wrong become essential to establish the expected stringency related to the knowledge and skills that personnel involved in the development of safety-critical systems should have. ISO 26262 defines a risk-driven safety life-cycle for developing safety-critical systems. In this paper, we give an education-oriented ISO 26262 interpretation and then we combine it with constructive alignment principles and we introduce SCA, Safety-critical Constructive Alignment, a new process to design Master's level safety-critical courses or modules. To illustrate SCA and its potential effectiveness, we then apply it to design a specific module.

Key-words: Safety standards, ISO 26262, Automotive Safety Integrity Levels, safety-critical systems, safety life-cycles, Education Safety Integrity Levels, education, Safety Element out of Context, Safety Educatee out of Context, Structure of Observed Learning Outocomes (SOLO) taxonomy, interactive lectures, constructive alignment.

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

1. Introduction

A qualified personnel is necessary and strategic for the development of safety-critical (software) systems. The personnel and the safety-critical system constitute a safety-critical socio-technical system. Private enterprises that manufacture/supply safety-critical systems/components should promote a deep safety culture to be spread throughout all the phases of the safety life-cycle. This promotion can be performed internally (e.g., via in-house training) or out-sourced (e.g., by taking courses).

In the context of the KKS PROMPT project [PROMPT], which aims at establishing a national educational alternative targeting industry, we offer a five-module-based course (DVA433) on safety-critical software. *Safety standards* is one module within DVA433.

Given the criticality of forming such qualified personnel, current methods aimed at engineering new courses should be further developed to make sure that such criticality is taken into consideration.

Safety-critical courses are expected to make educatees transit from an (un)consciously incompetent status to a consciously safety-competent status. In this paper, first of all, we propose to equate safety-critical competences with safety-critical elements/components to be developed via course-modules aimed at forming qualified personnel, responsible of the development of safety-critical systems. Then, we build on top of our experience related to automotive safety-critical systems engineering ([Gallina13], [GRSC]) and we propose to combine corresponding best practices with best practices in courses engineering.

More specifically, to develop safety-critical courses, we propose a novel method that combines constructive alignment principles with an education-oriented interpretation of ISO 26262 main principles related to the development of safety-critical elements out of context (SEooC) to be used in context for the actual

development of safety critical systems. We call this combination SCA (Safety-critical Constructive Alignment).
We then illustrate SCA and its potential effectiveness for developing a course-module in the framework of the PROMPT project.

The rest of the paper is organized as follows. In Section 2, we provide essential background information. In Section 3 we present Safety-critical Constructive Alignment and in Section 4, we apply it. In Section 5, we discuss related work. Finally, in Section 6 we present some concluding remarks and future work.

2. Background

In this section, we present the background information on which we base our work. In particular, in Section 2.1 we provide essential information concerning constructive alignment. In Section 2.2, we briefly present the skeleton of a typical safety life-cycle for the development of SEooCs. Finally, in Section 2.3, we briefly recall the definition of socio-technical systems.

2.1 Constructive alignment

Biggs and Tang [Biggs07] propose an interesting teaching approach aimed at improving the quality of learning at the university level. Their approach is in line with the Bologna process[1] and can be synthetized with the acronym OBTL, which stands for Outcome-Based Teaching and Learning. OBTL in essence consists of the constructive alignment of:

- Intended Learning Outcomes (ILOs), which are defined as statements that stipulate the skills in terms of actions plus content (formulated via verbs plus objects), ability level (e.g. deep), and context;
- Teaching and Learning tasks, which require students to apply, invent, generate new ideas, diagnose and solve problems;
- Assessment tasks, which require students to enact the verbs that characterize the ILOs.

The necessity of the alignment stems from the recognition that to achieve intended learning outcomes the focus must be student-centered and as a consequence what the student does in terms of actions is of paramount importance. Students' actions, indeed, should mirror the skills in terms of actions that we expect students to acquire during the learning process. Since however not all students are autonomous, proactive, goal-oriented, and highly or better intrinsically motivated; teachers still have a significant role to play. Teachers thus should practice reflection or better transformative reflection allowing them to make emerge what they might be in terms of role-models in triggering students to act or better enact what is needed to achieve the intended outcomes. Transformative reflection allows teaching practices that lead to surface learning (e.g. behaviorism-oriented teaching based on punishment/premium stimulating only extrinsic motivation) to be first identified and then changed. Changes should promote the introduction of practices that make the students feel the value of the teaching material and the personal relevance (towards the development of social motivation and then gradually intrinsic motivation) as well as the chances of success/the possibility of ownership.
Teachers' actions are crucial "in setting the stage for effective learning". To encourage deep learning and thus achieve quality and not quantitative-oriented learning outcomes, teachers should be aware that their actions have an impact on the way the students' brain is activated. Teachers should succeed in triggering the activation of complex cognitive areas permitting students to exercise the actions that are part of the intended outcomes during the learning process. Coherently, the same actions should be assessed during the examination. An abstract exemplification of constructive alignment is:

- ILO: apply (expected deep ability) methods M to solve small-sized problems;
- Teaching tasks: expository lecturing + interactive lecturing aimed at providing the context that requires the students to apply the methods M;
- Learning tasks: listening to/reading material provided via expository lectures + enacting/constructively applying methods M during the interactive lectures jointly with peers/teacher(s);
- Assessment tasks: summative assessment tasks offering contexts similar to the ones proposed during the teaching tasks, aimed at requiring enactment of ILO-verb (i.e., apply methods M), possibly in non-invigilated settings but asking for personal reflections to monitor/dissuade plagiarism-oriented behaviour.

To succeed in proposing an approach towards the standardization of the outcomes, Biggs and Tang

[1] http://www.ehea.info

provide taxonomies (e.g., SOLO taxonomy) containing actions (verbs) to be used during the outcome definitions according to the kind of knowledge that we as teachers expect students to acquire. These actions should then be kept in mind to define aligned teaching/learning tasks as well as assessment tasks.

To OBTL the expression constructive alignment is associated since the intention besides the alignment is to allow students to construct their knowledge based on their experience. The constructive aspect keeps open the possibility for desirable but unintended outcomes.

As a positive side-effect their approach can be used to better formalize and standardize the outcomes at the institution level and thus allow clear and outcome-based interfaces to be defined aimed at easing students mobility as well as students integration in the job market after graduation.

2.2 ISO 26262

ISO 26262 is the functional safety standard within the automotive domain. This standard introduces Automotive-specific Safety Integrity Levels (ASIL) and a safety life-cycle that guides the system development from inception to commissioning and whose stringency can be tailored according to the ASILs. ASILs are associated to the hazardous events that, if they occur, may lead to hazards. ASILs are also associated to safety goals, which are requirements aimed at preventing the hazardous event from happening. ASILs represent confidence measures.

In this section we briefly recall the skeleton of ISO 26262 V-model-based life-cycle. The top-level left-handed safety life-cycle activity consists of the definition of the system to be developed, followed by the identification and categorization of the hazards and risk assessment procedures. Once hazards are identified (e.g., via HAZOP-HAZard and Operability- analysis), they are categorized by assigning an *ASIL,* which can assume one out of five values, ranging from negligible QM and A to D, where D represents a hazard that may lead to catastrophic consequences. An ASIL is obtained based on values assigned to three different attributes (namely, severity, exposure and controllability).

Once hazards are categorized, safety requirements aimed at reducing risk are elicited as well as traced throughout the traditional development steps (specification, design, implementation, etc.). Safety requirements are named differently with respect to the abstraction level. At the highest abstraction level (i.e., at item-level), they are named *Safety Goals*, then when the functionalities to achieve those safety goals are revealed, *Functional safety requirements* can be formulated. The formulation of the functional safety requirements concludes the *concept phase,* given is Section 3.

The following phase, called *product development phase*, given is Section 4, begins with the formulation of *Technical safety requirements* related to the architectural components aimed at implementing the Functional safety requirements. Once software as well as hardware components (onto which technical safety requirements are allocated) become clear, software as well as hardware requirements are formulated.

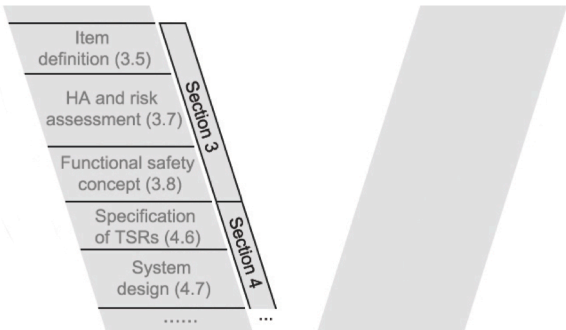


Figure 1 Portion of the ISO 26262 life-cycle

Figure 1 shows the previously textually described portion the ISO 26262 life-cycle.

On the right-hand side of the V-model, verification and validation activities are carried out to check that the elicited safety requirements are correctly specified, designed, implemented and deployed. Since these activities are not in focus within the scope of this paper, they are not detailed in Figure 1.

The novelty of ISO 26262 is also represented by the notion of *Safety Element out of Context* (SEooC)

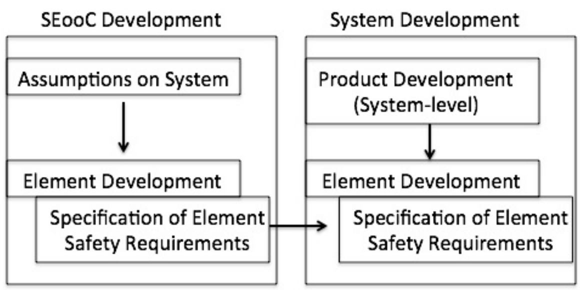


Figure 2 Reusable SEooC Development [DIS11]

A SEooC represents an element that is not developed for a specific item and thus its safety requirements are assumed during its development (SEooC development in Figure 2). Once the SEooC is developed (SEooC-related requirements specified, designed, implemented, and tested) it can be reused within a specific context to compose a system (System development in Figure 2). To be able to reuse a SEooC, assumed safety requirements should match with the actual safety requirements.

2.3 Socio-technical Systems

In this section, we briefly recall essential information on socio-technical systems aimed at enabling the reader in recognizing professionals involved in the development of safety-critical systems as part of enclosing socio-technical systems. As previously summarized by Gallina et al. [Gallina14b], socio (of people and society) and technical (of machines and technology) is combined to give socio-technical. Socio-technical refers to the interrelatedness of 'social' and 'technical' [Walker07]. Successful (or unsuccessful) system performance depends on this interrelatedness. As the SERA (Systematic Error and Risk Analysis) [Hendy03] taxonomy highlights humans may fail for various reasons including lack of training as well as absence of qualifications. A knowledge-related failure, for instance, may occur when the human does not have the pre-existing baseline knowledge or skills required to adequately or correctly interpret the situation.

Adequate training and qualification may contribute in avoiding/mitigating such failures.

3. Safety-critical Constructive Alignment

In this section, we introduce a novel approach for designing courses targeting safety-critical competences. This approach, called Safety-critical Constructive Alignment (SCA), stems from the combination of constructive alignment and ISO 26262 main principles translated in the education-oriented semantic domain. More specifically, in this section, first of all we give a motivation for the introduction of SCA; then we provide an interpretation of ISO 26262 in the semantic domain of education. Then, we combine such interpretation with constructive alignment to enable an acceptably safe formulation of Intended Learning Outcomes as well as a corresponding design of activities aimed at achieving and assessing those outcomes.

3.1 Motivation

Similarly to safety-critical systems, safety-critical competences should be developed in compliance with high-quality standards. Educatees/Employees are expected to execute crucial tasks during the life-cycle of safety-critical systems. Educatees/Employees can be seen as components/elements of an enclosing system, the socio-technical system that encloses employees, technology and regulatory/organizational procedures. Their training and qualifications are crucial to reduce certain types of failure [Hendy03]. Their training is either performed in context (e.g., within the enterprise, based on actual requirements) or out of context, based on assumed requirements. Assumed requirements (i.e., assumed intended learning outcomes) should stem from a thorough risk-driven-based requirements engineering process. Thus in Section 3.2, we interpret ISO 26262 within the education domain in order to engineer (we especially focus on the requirements and design phases) a course module aimed at forming educatees/employees that compose safety-critical socio-technical systems.

By proposing SCA, we aim at taking part to the debate [Baldwin13] around the efficacy of Bologna process' aspect concerning ILOs. More specifically, we intend to mediate and re-contextualize this aspect in our own field of practice.

3.2 From ASIL-SGs to ESIL-ILOs

In this subsection, we provide an education-oriented interpretation of ISO 26262. The idea is to translate crucial concepts in the educational semantic domain and then maintain the typical ISO 26262 process.

In particular, we are interested in introducing education-specific safety integrity levels that we call Education Safety Integrity Levels (ESIL). Similarly to ASIL, ESIL can be derived based on the severity, exposure and controllability related to the hazardous events. The hazardous events might be perceived differently according to the domain in which the employees are expected to work.

Similarly, we are interested in introducing the notion of Safety Employee/Educatee out of Context, called PROMPT-M3 SEooC, which translates the notion of Safety Element out of Context. Figure 3 builds on top of Figure 2 and shows the development of a Safety Educatee/Employee.

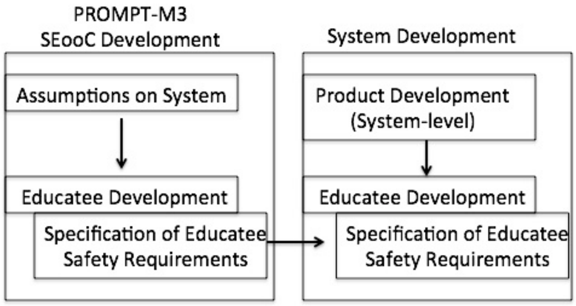


Figure 3 Reusable PROMPT-M3 Educatee Development

Safety Goals, which are ASIL classified requirements at the system level, are refined and broken down into functional/technical safety requirements (FSRs and TSRs respectively). ASIL classified TSR can be translated into ESIL-classified ILOs, more precisely into assumed ESIL-classified ILOs.

Our interpretation is limited to a few concepts, since the idea is to pioneer the application of safety standards within the education domain. An ISO 26262 expert would be certainly disappointed by this initial effort, however since ISO 26262 is currently under revision [ISO18], an in-depth interpretation would risk becoming obsolete rather soon. Moreover, a consensus has not been achieved and various interpretations are currently coexisting due to "different cultural approaches to the standard across the globe" [ISO18] Further clarifications on concepts are also expected to avoid agony while classifying hazardous events [Ellims12].

3.3 Constructively aligned ESIL-activities

Once ESIL-ILOs are formulated, teaching/learning/assessment activities have to be conceived to achieve them. These activities are aimed at design/implement the course module as well as at testing/assessing that the design/implementation meets the ILOs. Thus, constructive alignment inherently is a V-model and we customize it according to the ESIL.

4. Applying SCA to design DVA433-M3

In this section we apply SCA for designing Safety standards, the third module of DVA433. Then, we discuss our findings.

4.1 DVA433-M3

DVA433-M3 is a 1.5 credit /40 hour effort module, which at the time of writing (Spring 2015) is being offered for the first time in the framework of the PROMPT initiative. This module is supposed to be taken by personnel working in enterprises that either manufacture (or supply) safety critical (sub) systems. Bombardier Transportation, Volvo Trucks, Volvo Construction Equipment, Saab are examples of such enterprises. This module provides a panorama concerning safety standards and then focuses on safety life-cycles and development processes from various perspectives. Primarily, the module aims at forming process engineers, who have the responsibility of planning, executing and assessing safety processes for the development of safety-critical systems. The module, however, could and should be taken by those other roles that are expected to interact with process engineers. Understanding the relevance of a structured way of working via a well-defined process is the first step towards a crucial mentality change, which was also advocated by Parnas [Parnas86]: from a self-fulfilling prophecy stating that processes are not useful to a shared safety culture that spreads the relevance and potential gain of well-defined and rational-explicit processes; from a consequent tick-box mentality to a rational-based execution of (tailored) process steps.

4.2 ESIL-ILOs formulation

To formulate the ILOs and assign an ESIL, it is necessary to perform an investigation of the knowledge and skills that educates/employees are expected to offer (functional learning outcomes, after having defined the employee out of context) within the targeted safety-critical systems manufacturers/suppliers. To do that, the following questions require an answer:

- What a process engineer is expected to perform and know?
- What employees interacting with process engineers are expected to know with respect to process engineering?
- Is efficiency via intra/cross domain reuse of process elements a viable way?

By reading the standards and by interviewing industrial personnel, we realized that crucial skills and knowledge include: capability of 1) comparing/aligning safety standards and reusing process elements, 2) awareness concerning the strategic interrelatedness of roles as well as the necessity of increasing safety culture and effective communication, 3) planning, executing, and documenting processes as well as process compliance with standards. Thus the corresponding ILOs are:

• compare and contrast software safety standards;
• create a risk-based software development plan;
• apply selected process-steps;
• create typical conformance documentation.

To formulate additional ILOs (safety-related ILOs), the following questions require an answer:

- What if reuse is not systematically introduced?
- What if the relevance of a structured process is not understood?
- What if documentation does not conform to the expected requirements?
- What if an employee is not a teamplayer?

These above-formulated additional questions originate via a HAZOP analysis-like brainstorming process. From this brainstorming process, possible answers associated to these questions also originate. In particular, we realize that additional crucial skills/capabilities and knowledge include: systematizing reusable process elements, mastering process-related terminology and reference models, modeling processes, documenting process compliance, working in teams.

The ESIL to be associated to these skills and knowledge is D, the highest. Since if employees fail in guaranteeing the expected skills, catastrophic consequences may occur in terms of harm to people or environment or in terms of loss of money.

Anyway, the D level is not inherited as it is. It can be lowered (ESIL decomposition, to be performed similarly to the ASIL decomposition [ISO11] rules) if domain/context specific justifications exist. The decomposition rules are expected to customized and conceived in cooperation with industrial partners and should ease the matching between the PROMPT-M3 SEooCs and the elements that are actually needed by the industries.

4.3 ESIL-activities

To achieve the ILOs, teaching, learning and assessing activities are designed. Concerning the teaching activities, delivery of educational content aimed at supporting the achievement of the all the ILOs is performed via video-recorded lectures, physical lectures, virtual learning environments and video-conferencing. Reading material, lecture notes and examples contribute in teaching the required skills.

The learning activities aimed at exercising the ILOs-related verbs consist of: contrast standards via cooperation with other attendees of other domains, plan/apply/document processes and their compliance in cooperation with other attendees.

Cooperation is highly encouraged and with respect to limited learning activities even enforced for a twofold motivation: 1) increase the chances of forming team players; 2) enable co-construction of knowledge via peer-to-peer interaction [Webb08].

Finally, concerning assessing the ILOs, students are expected to: a) reflect on the lectures via provision of pro-memoria, b) joining threads of peer-to-peer discussion on virtual learning environments (more precisely via the Blackboard Learning System), c) execute a group-based project that includes tasks aligned with the ILOs, which are:

1. create a (portion of a) software development plan;
2. apply selected development/tool qualification process activities/steps;
3. create typical conformance documentation;
4. identify variation points that could be introduced to move either from standard X to standard Y (while performing 1-3) either from standard X/safety integrity Y to standard X/safety integrity Z (while performing 1-3);
5. present the work performed.

4.4 Discussion

In this section we discuss the findings related to the application of SCA for the design of M3. The discussion covers the following two main bolded aspects: **General soundness** - The application of SCA for the design of courses aimed at forming qualified personnel is sound since beyond the traditional constructive alignment permits course-designers to carefully consider safety concerns and thus has the potential to increase the quality of the formative offer. **Maturity** – SCA is still in its embryo stage. SCA is a process that combines systems/software development processes with constructive alignment, educational courses development process. SEI-CMMI [CMMI] is a process improvement approach that defines criteria to evaluate the maturity of a process. Thus, in our discussion, we use and elaborate on those criteria to evaluate SCA's maturity. SCA has the potential of being effective, but right now from a CMMI perspective its level of maturity can be considered to be in between Level-1 (Initial) and Level-2 (Managed). SCA ensures that requirements are managed and that teaching/learning/assessment activities are planned and performed to take care of functional as well as safety-related learning outcomes. However, no measurement, and control is ensured yet.

5. Related work

In the literature, various methods for designing courses have been proposed and discussed. Fink [Fink03], for instance, proposes a method for designing courses for significant learning. Fink's method differs from Biggs's constructive alignment in mainly two aspects. The first aspect is represented by the taxonomy. Instead of using the SOLO taxonomy, Fink proposes a new taxonomy called *a taxonomy of significant learning* where other dimensions are considered such has caring and human dimension. The second aspect is the consideration of situational factors e.g., characteristics of the learners as well as of the teacher.

To our knowledge, SCA represents a novelty. No related research work has explored the application of safety life-cycles within the education domain. Even tough SCA builds on top of Biggs'method, elements of the Fink's method are included via the combination of ISO 26262, which recommends situational analysis and is permeated by the caring and the human dimension via safety culture management.

6. Conclusion and Future work

In this paper, we have introduced SCA (Safecty-critical Constructive Alignment), a new process for designing courses targeting safety-critical competences. The process, which builds on top of ISO 26262 and constructive alignment principles, is aimed at offering a means for risk-driven course development. We have then applied SCA to design a course module related to Safety Standards. The illustration has been limited to the left-hand side of the ISO 26262 V-model.

Since the Swedish enterprises that expressed a concrete interest in this initiative are now well known, in the short-term future, based on the gathered experience related to the first edition, we aim at analyzing the characteristics of the various industrial contexts (avionics, automotive, railways, etc.) to identify commonalities and variabilities. Once commonalities and variabilities are identified, we combine SCA with VROOM & cC [Gallina13], our previously proposed method that aligns ISO 26262 and product line engineering practices. The idea is that since the entire set of attendees/educatees can be seen as a cross-domain product line, where the product is the educatee, commonalities and variabilities could be systematized. The combination of VROOM & cC and SCA would reduce the frequency of mismatch between assumed and actual requirements, at least for those educatees that belong to the educatee-line.

In a short-term future we also aim at elaborating a more in-depth ISO 26262 interpretation covering the left as well as the right-hand side of the V-model. Moreover, since the course is offered on-line, we are interested in applying SCA by taking into consideration the challenges that need to be faced in these circumstances to ensure an effective and safer delivery.

In the mid-term future, the idea is to perform an experimental evaluation of this new process.

Finally, in the long-term future, the intention is to explore other standards (e.g. Automotive SPICE [ASPICE10]) and see if other process elements could be considered of relevance within the education domain. To do this, a relevant starting point is the systematization of commonality and variability within safety-oriented processes, presented by Gallina et al. [Gallina12], [Gallina14a].

7. Acknowledgement

This work has been financially supported by the Knowledge Foundation via the KKS PROMPT project [PROMPT] and by the Swedish Foundation for Strategic Research via the SSF Gen&ReuseSafetyCases project [GRSC].

References

[Biggs07] J. B. Biggs, C. Tang. Teaching for Quality Learning at University: what the student does. Maidenhead: McGraw-Hill/Society for Research into Higher Education, 2007.

[ISO11] ISO 26262. Road vehicles – Functional safety. International Standard, November 2011.

[Parnas86] D. Parnas, P. Clements. A rational design process: How and why to fake it. IEEE Trans. Software Engineering 12(2), 251–257, 1986.

[ASPICE10] Automotive SPICE Process Assessment Model, Automotive SIG, 2010, www.automotivespice.com

[Gallina12] B. Gallina, I. Sljivo, and O. Jaradat. Towards a Safety-oriented Process Line for Enabling Reuse in Safety Critical Systems Development and Certification. Post-proceedings of the 35th IEEE Software Engineering Workshop (SEW-35), IEEE Computer Society, Heraclion, Crete (Greece), 2012.

[Gallina14a] B. Gallina, S. Kashiyarandi, H. Martin and R. Bramberger. Modeling a Safety- and Automotive-oriented Process Line to Enable Reuse and Flexible Process Derivation. Proceedings of the 8th IEEE International Workshop on Quality-Oriented Reuse of Software (QUORS), joint workshop at COMPSAC conference, IEEE Computer Society, pp. 504-509,Västerås (Sweden), 2014.

[PROMPT] PROMPT - Professionell masterutbildning i programvaruteknik. http://www.mrtc.mdh.se/projects/prompt/

[Walker07] G. Walker, N. Stanton, P. Salmon and D. Jenkins. A Review of Sociotechnical Systems Theory: A Classic Concept for New Command and Control Paradigms, Human Factors Integration Defence Technology Centre, U.K. Ministry of Defence Scientific Research Programme, HFIDTC/2/WP1.1.1/2, 2007.

[DIS11]DRAFT INTERNATIONAL STANDARD ISO/DIS 26262-10.2, 2011.

[Gallina13] B. Gallina, A. Gallucci, K. Lundqvist, M. Nyberg. VROOM & cC: a Method to Build Safety Cases for ISO 26262-compliant Product Lines. Proceedings of the Workshop on Next Generation of System Assurance Approaches for Safety-Critical Systems (SASSUR), joint event of SAFECOMP, CNRS report (HAL/Arxiv), hal-00848479, version 1, Toulouse, France, 24 September 2013.

[Hendy03] K. C. Hendy. A tool for Human Factors Accident Investigation, Classification and Risk Management. Defence R&D Canada, Toronto, DRDC Toronto TR 2002-057, March 2003.

[Gallina14b] B. Gallina, E. Sefer, and A. Refsdal. Towards Safety Risk Assessment of Socio-technical Systems via Failure Logic Analysis. 2nd IEEE International Workshop on Risk Assessment and Risk-driven Testing (RISK), joint event of ISSRE, pp.287-292, Naples, Italy, November 3-6, 2014.

[Baldwin13] R. Baldwin. Changing practice by reform. The recontextualisation of the Bologna process in teacher education. PhD Thesis, University of Gothenburg, Gothenburg, Sweden, ISBN: 978-91-7346-764-3, 2013.

[CMMI] CMMI-Levels: http://partners.clearmodel.com/cmmi-appraisals/cmmi-levels/

[GRSC] Gen&ReuseSafetyCases http://www.es.mdh.se/projects/393-Gen_ReuseSafetyCases

[ISO18] 5th International Conference on ISO 26262. http://www.iso26262-conference.com

[Ellims12] M. Ellims, H. Monkhouse, “AGONISING OVER ASILS: Controllability and the In-Wheel Motor”, The 7th IET International System Safety Conference, Edinburgh UK, 2012.

[Webb08] N. M. Webb. Learning in small groups. In. T. L. Good. 21st Century Education: A Reference Handbook, 2008.

[Fink03] D.L. Fink. Creating Significant Learning Experiences. San Francisco, Jossey-Bass, 2003.

The Teacher's Role in Gamification in Software Engineering at Universities (Field Report) - or how geeks can be inspired to sing

Martine Herpers
Computer Science Dept.
Fulda, Germany 36037
martine.herpers@cs.hs-fulda.de

Abstract

New educational methods require new competences from university teachers. In a non-technical seminar with included SE aspects for bachelor computer science student's self-determined learning and gamification could improve student's engagement significantly. Both methods seem to be a good candidates for technical courses in software engineering, too. The article highlights the new role and competences required for university teachers using gamification.

1. Introduction

Computer technology is changing fast and learned know-how is becoming outdated very soon. Industry very often complains about graduates with low key qualifications. The missing skills concern the ability to analyze and reflect independently, to write and communicate research results, to be team minded and able to solve problems. For universities this means that teaching student's common problem solving methods and self-determined learning becomes more and more important.

A special challenge for university teachers is teaching software engineering (SE). At universities of applied sciences some of the students have first experiences with real product development and can at least imaging the importance of working processes. Most of the students, however, have no idea about the challenges in real software development projects. Also university teachers often have only restricted experiences in real life software development. Most universities request that students are getting work experience by performing hands-on trainings. The success of internships in companies, however, depends heavily on the chosen company and its capability level in software development. For teaching software engineering at university it is very important to motivate students and to develop the key qualifications in analyzing, reflecting and communicating.

University education in computer science is changing since several years and in the meantime it is well accepted, that just offering lectures, exercises or software development projects is not enough. With activating mechanisms like online queries, brief teamwork practices, letting students evaluate each other, etc. student's ability to concentrate and learn can be improved [Rac05]. Such mechanisms are very useful, but are more or less just islands of improved learning experience.

A broader mechanism is the **problem based learning** (PBL)[1]. PBL gives students the responsibility for their learning success supported by the teacher's short lectures about the techniques they need to solve the problem. A very similar method with focus on making students wishing to learn is the **gamification**[2] of lectures. Gamification doesn't mean only to add the aspect of fun to the lectures, it can be used to get everyone involved and to let students learn from each other. One goal of a gamification can be that the best students demonstrate and prove their talents in a contest at the end of the course.

For teaching software engineering at universities two different approaches seem to be promising:

- Using PBL on a project level, where some project steps are allowed to fail, if needed working processes have to be ignored (e.g. project feasibility gate fails, because a stakeholder has been ignored)
- Using gamification for strengthening particular capabilities for improving software quality like finding errors, deriving test cases from given specifications or test driven coding.

There might be other approaches for an activating and intensive learning of working processes at universities, but all these new ways of teaching require a change in the teacher's role [Fle07].

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

[1] A good explanation of PBL can be found in Wikipedia: http://en.wikipedia.org/wiki/Problem-based_learning

[2] Gamification is used since several years in industry to motivate employees. A list of gamification projects can be found at [Bol12].

Teachers must develop management skills enabling them to lead the students through such courses. This article shows the challenges teachers are facing by using these new teaching methods. The course "Presentation and Communication" for computer science students of the University of Applied Computer Science in Fulda, Germany, was held in winter semester 2014/2015 and shows all the requirements which are needed to meet for a successful teaching process. Although soft skills are not the main topic of software engineering, they are very important for every project and therefore relevant for all working activities. Because I am convinced, that teaching soft skills requires similar educational capabilities as teaching technical skills, I recommend the experiences I made during the above-mentioned course for teaching all kinds of competences.

As far as my experience goes soft skills are seen as a "not so important" subject for most of the computer science students. Thus there was a high demand of motivation by the teacher to get the students willing to learn it. My experiences show that using activating teaching methods like group work and telling stories along the software development process worked very well. The gamification I used, however, made students enthusiastic learners showing outstanding results at the end.

2. New skills needed: managing instead of lecturing

At the beginning all my colleagues told me, that I would not be able to activate the students for participating on a communication contest (gamification). In particular, that I wanted them – besides to give a 10-minute presentation, to perform an escalator pitch and to draw a visualization on a flip chart – to sing a song as a choir, was considered to be impossible.

I asked my husband and my daughter (student of mechanical engineering at TU Chemnitz), what they thought about the idea to offer such a challenge at the end of the course. My husband thought, that I was very ambiguous, but that the students couldn't be motivated to sing, perhaps for the other contest categories. My daughter's first reaction was: "No, never!". Her second reaction was: "Well, perhaps it might be fun at least."

In literature emotional and enjoyable learning is recommended [Rac07] and after some discussions with a gamification expert, I was convinced, that gamification would result in great learning experiences for the students. So I was not sure, that it will work, but I tried trusting on my managing[3] experience for many years.[4]

In the following the different phases and the corresponding challenges for me as the teacher during the course are explained in more detail. First it is explained, how the students had been motivated to try something unusual. Finally the challenges for teaching it are described in detail.

2.1 The seminar outline

The seminar "presentation and communication" is a required course for bachelor computer science students at the university at Fulda. It takes place in the third semester. So all students attend to seminars in groups of about twenty persons for four hours a week. The winter semester 2014/15 started in October 2014, paused two weeks for Christmas break and finished in February 2015.

Three groups were trained by me, which offered the possibility to organize a contest at the end of the seminar to find out which group got the best communication skills. Two other colleagues, training two other groups, were asked to participate with their groups, but they had no interest in attending the contest.

2.2 How geeks can be motivated

As professor at a university of applied sciences the question for me was: What is motivating for young geeks coming from high school and for those with working experiences? For both groups it is important to understand the relationship of the topic to be learned to the working practice in their future life. So it was easy to motivate the students along the software development process. Telling stories and work experiences from real life as a software manager I got always highest attention by the students. Below you see the topics of the course and how it can be mapped to steps in the software development process.

---

[3] Managing is understood as modern, team-oriented managing. Often coaching competences are required for effective teachers. Coaching, however, would be too week for the tasks of introducing new educational methods like gamification. A clear goal setting capability with the appropriate authority is essential.

[4] Technical team manager in software development, head of department for consulting software quality have been some career stages before my assignment to professor at the university of applied computer science in Fulda.

Table 1: Motivation by references to software development tasks

Learning goal	Motivation by references to software development tasks
Presentation / elevator pitch	Presentation of technical ideas/products in the feasibility phase or for customers
Visualization on flip charts	Discussions during all phases
Feedback giving and taking	Performing a review or inspection, Useful for pair programming or in discussions between quality engineer and developer
Conflict resolution	Very important in communication with customers and between developers and testers
Negotiation	Same as conflict resolution
Intercultural communication	Distributed development teams, interdisciplinary teams

Introducing and using up-to-date educational teaching methods needs, however, additional motivation supporting techniques regarding the group processes. In my case it was very important to form the group to become a team ready for fighting against other teams. I trained three groups. All groups had been set together by random, so there were unfortunately only a few already established friendships among the group members. So I started right from the beginning to strengthen trust between the group members by organizing randomly smaller learning groups (about 5 persons) and to allow self-determined learning experiences. In the mid-seminar review all students mentioned that the learning atmosphere allowed them to get in closer contact to all other group members and they want to continue to work using the activating learning methods. Trust among the group members and me as the teacher was well established at this point in time.

The groups were trained well separated from each other. Every group was different in size, in composition of talents and willingness to actively participate in the communication contest at the end:

- **group A**: 23 persons, medium willingness to participate
- **group B**: 16 persons, low willingness to participate
- **group C**: 21 persons, high willingness to participate

It showed that the following aspects were important to form teams accepting the new and challenging learning environment:

- **Openness**: right from the beginning the uncommon combination of learning and setting a goal to win a contest was clearly communicated and explained. Some were really surprised by the expectation to do a contest and in particular to sing in a choir. I got a lot of comments like: “Don’t want to sing” and only very few saying “Singing is fun”. The contest itself seemed to be mostly accepted right from the beginning, the choir performance wasn’t.
- **No doubt about the new format**: The contest was not discussed in the course, neither the choir. As the teacher I outlined the learning effects, generated an officially looking announcement of the contest and organized the eLearning video team to record the contest. In the first half of the seminar the contest was no topic. This time was used to build trust among the students and to learn communication skills. In the middle of the seminar I set up four teams one for each category (presentation, elevator pitch, visualization, choir) of the contest.
- **Creating trust among the group members**: In order to get familiar with each other different team set ups were done. After getting familiar with the new way of working in always new team set ups, the students looked forward to the tasks and the set ups of the next lesson.
- **Supporting the learning activities**: The very active learning style was new for the students. So it was necessary to support them at the beginning by showing them that supporting each other is a wished behavior and nobody is losing his/her dignity in doing unpopular or uncommon things for students like fetching the flip chart or summarizing the results. After the first phase, where I organized the paper for the flip chart or looked for the beamer, all students became familiar with all needed tools and started to support each other.
- **Allowing not to participate at the contest**: the course was organized as a seminar with

compulsory attendance for at least 80% of all lessons. So the students had the possibility of not attending the contest, which was just one of the lessons. At the beginning some of the students planned not to attend. But with growing trust in the other group members and in the teacher, that the contest is not relevant for the grade at the end of the seminar, they accepted the new format.[5]

- **Making it a big event**: The event had to be moderated appropriately with some fun and the without the feeling that somebody could fail. Due to this it was explained, that in the eyes of the teacher contributing was more important than winning and only the fact of contribution would be clearly valued. Furthermore we organized some soft drinks and cookies for our convenience. The importance of the performance war underlined by the fact that all should be recorded. That enabled me also to analyze the activities afterwards and to celebrate the performance with the students.

At the end only two persons didn't attend the contest and two more hadn't been on the stage singing in the choir. Even group B was attending. But the winning team was the one with the highest willingness to participate right from the beginning.

2.3 Challenges for the teacher

As described above the activating teaching methods in particular the gamification need new modern educational approaches of course organizing. Particularly the organization of the contest itself as a big event demands more effort from the teacher as performing a normal lecture. Organizing the team work during the course requires as much time and efforts as developing conventional exercises and correcting them.

In terms of the overall effort the gamification requires perhaps some more preparation as a normal lecture.

The real challenges are others. The teacher's role has to be enhanced by the following two aspects:

- The teacher has to be **really persuaded about the new teaching method**. In case of doubting the success, the student's behavior will probably change immediately and become inactive again.
- Show **veritable management skills**: Setting clearly the goal, guide the team to become clear about the expectations they have to fulfil and believe in the team. In some situations it can become very difficult to still believe that the contest will be a success, because the willing-ness to participate are changing from week to week. The week before the contest seemed to be the most critical. In this week unexpected issues and conflicts were popping up, because everything was becoming more concrete.

At the beginning it is important to allow the denial of all new methods. The following figures show the comments (in German) of one group about what the group wanted to do and what they didn't want. A majority of voices were raised against the very unusual requirement to sing in a choir and one comment says: "no new learning methods".

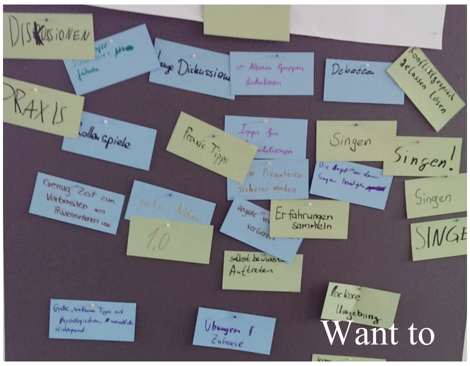
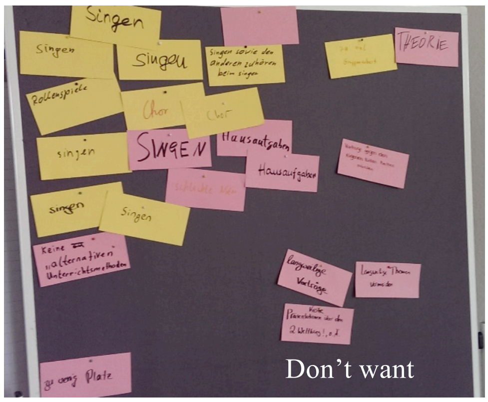


Figure 1: Comments of group A about what they want to be happen in the seminar.



[5] The importance, that the own learning is supported by the teacher and the peers, is also highlighted by [Pra01].

Figure 2: Comments of group A about what they do not want to be happen in the seminar.

During the seminar it was very difficult to judge the probability that the contest at the end will be attended by the students. Some few students seemed to like the idea of a contest, others were still against it and others seemed to wait and see what will happen.

After the contest took place, I asked the students to show their curves of willingness to attend the contest over the time of the seminar.

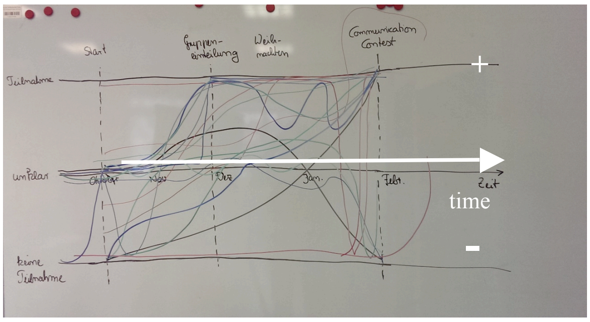


Figure 3: Willingness graphs of the winning group, Group C, 2015

The figure above shows all ups and downs of the willingness of every single team member in attending the contest. The upper bound (+) means that the student wanted to attend the contest, the lower bound (-) stands for not attending the contest. The major events as the start of the course, the sub team's set up, Christmas and the contest were given on the time line. In the lesson after the contest every student was asked to draw a line showing his/her willingness level and its development to attend the contest over the time. Below you see the willingness graphs of the winning team (group C). Only one student drew the line right from the beginning until the end on the Attending bound. All other curves show, that a lot of students underwent massive ups and downs. The teacher must be aware about these ups and downs and he / she has to tackle with it always trusting and supporting the group.

3. Results

Using new educational methods is no end in itself. The learning effects for the students have to be considered at the end. In this course students learned a lot by preparing the contest and showed really great contest acts. For such who had no active part in the contest, the contest has been a good repetition showing how excellent students are performing in different categories.

Most of the students enjoyed the course and learned a lot – not only in the core subject presentation and communication, but also in group dynamics.

4. Conclusion

For software engineering lectures the results of this nontechnical course mean that giving students examples where and how they can make use of the applied techniques is very motivating for them. Furthermore gamification creates a high dynamic environment for learning and at the end the best possible performance in the core subject (whatever the subject is) is shown by the students.

Using activating educational methods works very well, but requires additional soft skills of the teacher. University teachers with low managing experience might make difficulties in guiding the team through the forming and storming phases. In particular for software engineering or quality courses, this shouldn't be any insurmountable barrier. A good solution could be to combine the technic know-how of the university teacher with managing experiences of a practitioner used to guide software teams. Another possibility could be to provide coaching by gamification experts at least for the first time gamification is used.

In other courses, like test-oriented software development, I gained first good experiences with activating educational methods, too. After the success in the nontechnical course I'm convinced that these techniques can also be used in technical courses. PBL and gamification, however, can't be used in all courses in parallel. So the faculty or the university department has to decide how courses can make use of these new educational methods.

5. References

[Bol12] Boller, S. (2012). *100 Great Game Based Learning and Gamification Resources*. (B.-L. P. Inc., Herausgeber) Accessed at 22. March 2015 from http://www.theknowledgeguru.com/100-great-game-based-learning-and-gamification-resources/

[Fle07] Fleischmann, P., & Geupel, H. L. (2007). Lernteamcoaching. In *Neues Handbuch Hochschullehre* (S. C2.5). Berlin: Raabe.

[Pra01] Pratt, D., & al., e. (2001). Development and Use of The Teaching Perspectives Inventory (TPI). *AERA Annual Meeting 2001*. Seattle, USA: AERA.

[Rac05] Race, P. (2005). *Making Learning Happen.* London: Sage.

[Rac07] Race, P., & Pickford, R. (2007). *Making Teaching Work.* London: Sage.

Continuous Learning Process Assessment Model

Oleg Mirzianov
Software Engineering Dept.
Vilnius University, Lithuania
oleg.mirzianov@gmail.com

Antanas Mitasiunas
Computer Science Dept.
Vilnius University, MIT-SOFT Ltd. Lithuania
antanas.mitasiunas@mif.vu.lt

Abstract

Process capability maturity modeling elaborated by the Software Engineering community became applicable for any process-oriented activity assessment and improvement. The purpose of this paper is to contribute to the solution of learning improvement problem based on process quality attributes modeling approach. Two-dimensional Learning process model is developed based on R. Marzano taxonomy of learning objectives and on the staged Learning process maturity model. The consciousness as a learning process quality characteristic is introduced.

1 Introduction

Does any learning improvement problem exist? Unfortunately, in contrast to the natural ability to breathe, walk or digest, humans do not possess neither equal abilities to learn, nor uniform understanding of what is learning. There is quite widespread attitude that a capacity to learn is the ability to memorize.

Communications with students before and after exams during several decades repeatedly were coming to the situation when students were saying: "I knew everything, but I've got this one question which I did not know". In the beginning such mismatch of student's self-assessment and professor's assessment seemed as a student's self-defense to keep moral comfort, but later, due to permanent character of the situation described, there came an idea that another reason exists – students don't know what the target state of the knowledge is. If a student doesn't know what the target state is, he can't reach it.

Usually Universities do not have such lectures for learning to learn, where students could understand, what the target knowledge status is. There are a lot of lectures to deliver knowledge. It is up to the student to find what status to achieve and how to achieve it.

The situation is at some extent different in mathematics study programmes. The mathematics study programs like others do not address explicitly the problem of learning to learn. However implicitly a successful graduate in mathematics during regular mathematical studies acquires the ability to learn without any additional efforts outside mathematical subjects. Mathematical approach to learning has one deficiency – internal demand to learn everything starting from axioms. But usually learning time and efforts are limited, particularly in IT area.

What is the percent of knowledge that has been acquired at the University comparing to the all knowledge to be acquired by a graduate during whole professional career in IT area up to the retirement? Having in mind that IT technologies change every 5 years, during his professional career he may face such changes 8 times. Simple calculation allows to conclude, that knowledge gained at University comprise about 10% of the total professional knowledge. It means that 90% of knowledge must be acquired outside the University by means of one's own efforts. It is a disaster for a graduate, who has not learned to learn at University. Most probably he will be obliged to change his profession.

A lifelong learning is not regular attending in training courses. The lifelong learning is a compulsory part of a regular daily work enabled by the ability to learn acquired at University.

Therefore, the main task for a student at University is the learning to learn, but not the knowledge acquisition as it is frequently understood. The learning to learn is a learning improvement. It is a real pity that such an approach is not widely recognized at University's environment.

The main process at University is not teaching but learning. It should not be left to laissez-faire, traditionally to the mechanical operation with text fragments.

The true learning is the consciously performed structured activity resulting to the creation of mental hierarchical aggregated model as an adequate representation of the learned subject.

The goal of research provided in this paper is to create adequate learning process model as the basis for learning process improvement. The purpose of such model is to transform a learning as a "black box" into transparent box with the internals of learning seen.

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

2015-06-15 published at http://ceur-ws.org

The consciousness of process performance is considered as the essential measurable learning processes characteristic determining learning success.

This research was inspired by the own experience learning and teaching during 40 years and by the experience of creation of several process capability models for creative activities, and enforced by research results in education and psychology, first of all Marzano's New Taxonomy [MAR01].

Software process community can contribute to the recognition and solution of learning improvement problem by applying methods, which were elaborated for software crisis solution and turned up being much wider applicable than software area. It has been already proven that process capability maturity modeling approach became applicable for any process oriented activity assessment and improvement, including such creative activities as software development [15504], innovation [BES12] and learning [MAR14].

Process maturity modeling is based on processes grouping into maturity levels that reflect generic process improvement path. The requirements for processes in maturity modeling are described in terms of process performance and achievement of the process goals.

Process capability modeling is related with predefined process feature – process results predictability. Process capability characteristic is standardized by ISO/IEC 33020 [33020] in terms of process capability levels and process attributes defined by process achievements. Process capability attributes, for instance, PA 2.2. – Work product management attribute or PA 3.2 – Process deployment attribute by default are targeted to processes performed by organization.

The learning processes are performed by a single learner mentally. The results of learning processes performance are knowledge acquired. Process capability characteristics, at least some process attributes are not applicable for learning processes.

In such situation the applicability of process capability maturity modeling approach for learning process improvement is "legitimated" by ISO/IEC 33003 [33003], which allows to define own process quality characteristics.

The idea of modeling process characteristics other than process capability is analyzed in [WEL03].

In the context of standardization by ISO/IEC 330xx of the new process characteristics definition, the process agility characteristic introduced in [OZC14] demonstrates the relevance of approach to employ other process quality characteristics.

The idea of this work is to create a Marzano taxonomy based adequate learning process model for the process quality characteristic – consciousness using ISO/IEC 330xx modelling technics.

The state of the art learning process capability maturity modeling is provided in the Sections 2. The Section 3 contains authors' contribution to learning process modeling - development of two-dimensional learning process model. The last Section concludes the results achieved and provides ideas for the future work to be done to complete the solution of the problem addressed.

2 Learning process Capability Maturity Modeling

Capability maturity modeling at organizational learning level is well elaborated in [PEO09]. Process capability or organizational maturity improvement is widely understood as an organizational learning. But here are few more or less direct attempts to touch capability maturity modeling at individual learning level.

Personal software process [HUM97] can be considered as learning how to improve personal performance based on planning, measurement and tracking, i.e., understanding the process performed.

Capability maturity modeling in e-Learning area [MAR04], [NOV09] gains increasing attention of researchers. E-Learning is situated in between of education as organizational activity and learning as individual activity. E-Learning creates conditions for learner centric education. Education process itself is an organizational process [MIT12] which can be modelled using ISO/IEC 15504 conformant and Enterprise SPICE based model [ENT10].

Learning process maturity model [THO04], [THO06] is oriented to software development learning and is based on the idea that learning improvement can be achieved using the same concepts as improvement in software development.

The learning as an education area stresses on mental process of a learner. This area first of all is represented by the Bloom's Taxonomy [BLO56] and followed learning models and approaches [BIG82].

Particular place among them takes Marzano's New Taxonomy [MAR01], [MAR08] because of its process orientation. According to Marzano's Taxonomy learning is conditioned by three systems of mental activity: ego system, metacognitive system and cognitive system. Ego system is responsible for decision making in learning. Metacognitive system defines the goals and its achievement strategy. Cognitive system is responsible for effective performance of the tasks related to information processing: comparison, classification, conclusion, etc. All these systems use knowledge possessed by a learner. Cognitive system consists of processes grouped into four levels of knowledge processing: retrieval, synthesis, analysis and use.

3 New approach to Learning Process Modeling

Staged Learning process maturity model for learning process assessment and improvement based on R. Marzano taxonomy of learning objectives is proposed and partially validated in [MAR14].

Staged architecture of the model fits well for the sequentially layered cognitive processes in the learning improvement path. These cognitive processes within learning activity can be treated as primary or engineering or life cycle processes. The purpose of cognitive processes is to build mentally aggregated knowledge artefacts. Similarly to engineering mental knowledge building operate with constructs and the rules of their composition into aggregates. An adequate mental model must be built before creation of everyone engineering aggregate.

Ego process is responsible for decision-making, the decision to learn or not to learn is taken at some extent quite unconsciously before learning planning and execution. However in later stages of learning proficiency the decision-making is consciously performed. For this reason continuous architecture of the model is preferred to allow to reflect the performance of learning processes at various levels of consciousness as learning quality characteristics.

The measurement framework of learning process quality characteristic - consciousness is defined based on [33003] by tailoring process capability measurement framework [33020].

3.1 Process consciousness levels and process attributes

Process consciousness is defined on a four point ordinal scale that enables consciousness to be assessed from the bottom of the scale, **Incomplete**, through to the top end of the scale, **Conscious**. The scale allows to evaluate increasing consciousness of the implemented processes, from failing to achieve the process purpose through to continually improving consciousness.

An example of two-dimensional representation of learning processes performance consciousness profile is provided in Fig. 1. It consists of learning processes dimension and process performance consciousness dimension. The process dimension is represented by 7 learning processes. Each of them can be performed at various levels of consciousness from level 0 – Incomplete to level 3 – Conscious defined here bellow.

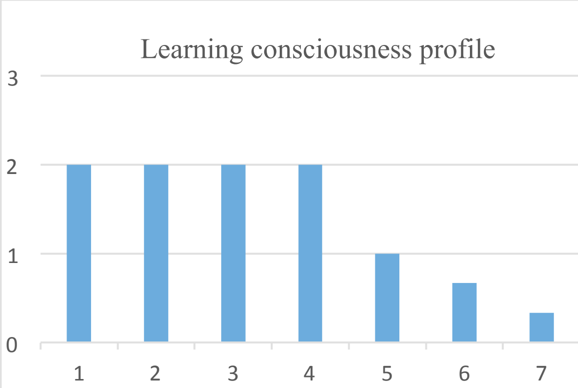


Figure 1: Learning processes consciousness profile

3.1.1 Process consciousness Level 0: Incomplete process

The process is not implemented, or fails to achieve its process purpose. At this level there is little or no evidence of any systematic achievement of the process purpose.

3.1.2 Process consciousness Level 1: Performed process

The implemented process achieves its process purpose. The following process attribute demonstrates the achievement of this level.

3.1.2.1 Process performance attribute PA1.1

The performed process attribute is a measure of the extent to which the process purpose is achieved. As a result of the full achievement of this process attribute:

a) The process achieves its defined process outcomes.

3.1.3 Process consciousness Level 2: Motivated process

The previously described performed process is now implemented as a motivated (important, effective, emotional).

The following process attribute, together with previously defined process attribute, demonstrate the achievement of this level:

3.1.3.1 Motivated process performance attribute PA 2.1

The motivated process performance process attribute is a measure of the extent to which the process performance is motivated. As a result of the full achievement of this process attribute:

a) The importance of process performed is assessed by a learner.
b) The ability of the learner to perform process effectively is assessed.
c) The emotions of the learner concerning process performed are assessed.
d) The motivation to perform process is assessed and the decision to perform process is made.

3.1.4 Process consciousness Level 3: Conscious process

The previously described motivated process is now implemented as a planned and tracked process.

The following process attribute, together with previously defined process attributes, demonstrate the achievement of this level:

3.1.4.1 Planned process performance attribute PA 3.1

The planned process performance process attribute is a measure of the extent to which the process performance is planned. As a result of full achievement of this process attribute:

a) The clear goal of the process performed and the target knowledge state is defined by a learner.
b) The strategy to achieve process goal is created.
c) The plan to achieve the target goal is developed by the learner.
d) The resources, milestones and schedule of the target knowledge state achievement are determined.

3.1.4.2 Tracked process performance attribute PA 3.2

The tracked process performance attribute is a measure of the extent to which the process performance is tracked. As a result of full achievement of this process attribute:

a) The process performance against process plan is tracked.
b) The clarity of the knowledge learned is assessed by the learner.
c) The precision and trustworthiness of the knowledge learned is assessed by the learner.

3.2 Process attribute rating scale

Within this process measurement framework, a process attribute is a measurable property of process consciousness. A process attribute rating is a judgment of the degree of achievement of the process attribute for the assessed process.

As it is indicated in the introductory part of section 3 the measurement framework of learning process quality characteristic - consciousness is defined based on requirements [33003] by tailoring process capability measurement framework [33020]. A process attribute is measured using an ordinal scale as defined below.

N Not achieved: there is little or no evidence of achievement of the defined process attribute in the assessed process.

P- Partially achieved: there is some evidence of an approach to and some achievement of the defined process attribute in the assessed process. Many aspects of achievement of the process attribute may be unpredictable.

P+ Partially achieved: there is some evidence of an approach to and some achievement of the defined process attribute in the assessed process. Some aspects of achievement of the process attribute may be unpredictable.

L- Largely achieved: there is an evidence of the systematic approach to and significant achievement of the defined process attribute in the assessed process. Many weaknesses related to this process attribute may exist in the assessed process.

L+ Largely achieved: there is an evidence of the systematic approach to and significant achievement of the defined process attribute in the assessed process. Some weaknesses related to this process attribute may exist in the assessed process.

F Fully achieved: there is an evidence of the complete and systematic approach to and full achievement of the defined process attribute in the assessed process. No significant weaknesses related to this process attribute exist in the assessed process.

The ordinal scale defined above shall be understood in terms of achievement in percent of a process attribute.

The corresponding percentages shall be:

N Not achieved 0 to ≤ 15 % achievement

P- Partially achieved-> 15 to ≤ 32,5 % achievement

P+ Partially achieved+>32,5 to ≤ 50 % achievement

L- Largely achieved-> 50 to ≤ 67,5 % achievement

L+ Largely achieved+> 67,5 to ≤ 85 % achievement

F Fully achieved > 85 to ≤ 100 % achievement

3.3 Process attribute rating method

A process outcome is an observable result of the successful achievement of the process purpose.

A process attribute outcome is an observable result of the achievement of this process attribute.

Process outcomes and process attribute outcomes may be characterized as an intermediate step to providing a process attribute rating.

3.3.1 Rating method

The approach to process attribute rating shall satisfy the following conditions:

a) Each process attribute for each process within the scope of the assessment shall be characterized for each process instance, based on validated data.
b) Process attribute characterization for all assessed process instances shall be aggregated to provide a process attribute achievement rating.
c) The assessor may choose to apply expert judgement to summarize the ratings without employing a formal mathematical approach, alternatively an aggregation method may be used.

3.4 Learning Process consciousness level model

The learning process consciousness level shall be derived from the process attribute ratings for that process accordingly to the process consciousness level model defined in Table 1.

Table 1: Learning process consciousness level ratings Table 1:

Scale	Process attributes	Rating
Level 1	Process Performance	Largely or fully
Level 2	Process Performance Motivated Process Performance	Fully Largely or fully
Level 3	Process Performance Motivated Process Performance Planned Process Performance Tracked Process Performance	Fully Fully Largely or fully Largely or fully

3.5 Learning Process Reference model

Provided here Learning Process Reference model reuses processes from [MAR14] and forms process dimension, which consists of 7 processes to be performed by a learner. The description of these processes satisfies the ISO/IEC 33004 requirements for Process Reference Model [33004] and is provided in Table 2.

Table 2: Learning Process Reference Model

LEAR.1. Knowledge Retrieve Ability Development	
Purpose	**Outcomes**
To acquire ability to recognize and reproduce target knowledge	1) Learner is able to identify and recognize knowledge items. 2) Learner is able to reproduce and perform a procedure.
LEAR.2. Knowledge Synthesis Ability Development	
Purpose	**Outcomes**
To develop ability to abstract and aggregate knowledge	1) Learner is able to recognize essential and non-essential features of a knowledge item. 2) Learner is able to generalize a set of knowledge items with identic essential features by a single abstract notion. 3) Learner is able to represent, recognize and operate with abstract notions. 4) Learner is able to aggregate

	knowledge items and structures.
LEAR.3. Knowledge Analysis Ability Development	
Purpose	**Outcomes**
To develop ability to verify consistency of aggregated knowledge and matching of new knowledge item to aggregate created.	1) Learner is able to identify similarities and differences of knowledge items. 2) Learner is able to identify knowledge items subsets and supersets. 3) Learner is able to identify mistakes in knowledge presentation. 4) Learner is able to identify special cases and derive related conclusions. 5) Learner is able to foresee possible circumstances.
LEAR.4. Knowledge Application Ability Development	
Purpose	**Outcomes**
To develop ability to apply aggregated knowledge in solving new tasks.	1) Learner is able to derive task solution based on possessed knowledge aggregate. 2) Learner is able to identify and assess solution's alternatives. 3) Learner is able to use knowledge and skills acquired as a tool for hypothesis investigation. 4) The ability to verify the trustworthiness of external information is acquired.
LEAR.5. Motivation Assessment	
Purpose	**Outcomes**
To assess motivation to learn and identify reasons for motivation	1) The importance for learner of knowledge to be acquired is assessed by learner. 2) Learner's opinion about his own ability to acquire identified knowledge and skills is self-evaluated. 3) Emotions related to knowledge and skills to be acquired and to their acquisition are identified. 4) The drives that condition learner's motivation to learn are identified.
LEAR.6. Learning Goals Definition	
Purpose	**Outcomes**
To define learning goals, level of knowledge acquisition and to select suitable strategy to reach learning goals, and to develop learning plan	1) Based on motivation target the knowledge level to be achieved (knowledge retrieve, synthesis, analysis or application ability) is identified by learner. 2) Learning goals are defined. 3) Strategy to achieve learning goals is selected. 4) Learning plan is developed. 5) Learning sources are selected.
LEAR.7. Learning Results Tracking	
Purpose	**Outcomes**
To assess acquired knowledge and skills, and to compare learning achievements with learning goals.	1) Learner is able to track the acquisition efficiency (to assess learning actions for learning goals achievement) of knowledge and skills being learned. 2) Learner is able to track the consistency and precision of knowledge and skills being learned. 3) Learner is able to track the trustworthiness of knowledge and skills being learned.

Learning process model is defined at PRM and PAM levels.

4 Model Validation

Partial validation of the model created was done for Learning process performance at the Level 1. There were selected students for learning process assessment belonging to two different groups, ten students in each group, according to exams results. The average of students learning process performance assessment results was calculated per process for each group. The comparison of learning process profiles composed for two groups is provided in Fig.2.

As it is shown in Fig.2 the classification of the groups according to exam grades was repeated for all processes of Learning process model according to model based learning assessment results.

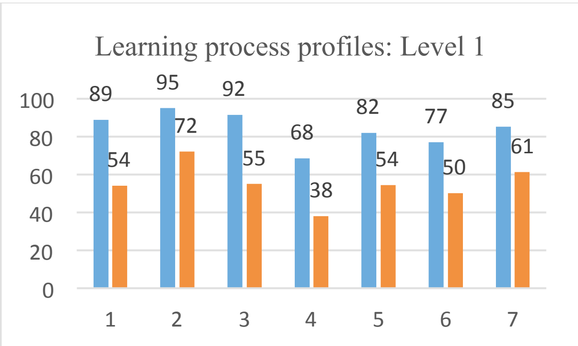


Figure 2: Comparison of Learning process profiles of two students groups

5 Conclusions and Future Work

The paper provides two-dimensional Learning process model for learning process assessment and improvement based on R. Marzano taxonomy of learning objectives and on the staged Learning process maturity model. The measurement framework of learning process quality characteristics - consciousness of learning process performance is tailored based on ISO/IEC 330xx.

Learning process model is defined at PRM and PAM levels.

An adequacy of the Learning process model developed to real learning activity should be validated in future research.

Acknowledgements

The research has been partially funded by the Seventh Framework Programme's project No. 600680 "ICT Transfer Concept for Adaptation, Dissemination and Local Exploitation of European Research Results in Central Asia's Countries" (eINTERASIA)

References

[BES12] Besson J., Woronowicz, T., Mitasiunas A., Boronowsky, M. Innovation, Knowledge- and Technology Transfer Process Capability Model – innoSPICE™. The Proceedings of the 12th International Conference, SPICE 2012, Palma, Spain, May 29-31, 2012: Software Process Improvement and Capability Determination, CCIS, Volume 290, 2012, pp 75-84.

[BIG82] Biggs, J.B., Collis, K.F. Evaluating the quality of learning: the SOLO taxonomy (structure of the observed learning outcome). New York: Academic Press, 1982.

[BLU56] Bloom, B.S., Engelhart, M.D., Frust, E.J., Hill, W.H., Krathwohl, D.R. (Eds.) Taxonomy of educational objectives: The Classification of Educational Goals. Handbook I: Cognitive Domain, 1956.

[ENT10] Enterprise SPICE® An Integrated Model for Enterprise-wide Assessment and Improvement. Technical Report - Issue 1. The Enterprise SPICE Project Team, September 2010, 184 p., http://www.enterprisespice.com/page/publication-1

[HUM97] Humphrey, W.S. Introduction to the personal software process ™. Reading, MA: Addison Wesley Longman, 1997.

[15504] ISO/IEC 15504-5. Information Technology – Process Assessment – Part 5: An exemplar software life cycle process assessment model, 2012.

[33003] ISO/IEC TR 33003 Information technology – Process Assessment – Requirements for process measurement frameworks, 2013

[33004] ISO/IEC TR 33004 Information technology – Process Assessment – Requirements for process reference, process assessment and maturity models, 2013

[33020] ISO/IEC TR 33020 Information technology – Process Assessment – Process measurement framework for assessment of process capability, 2013

[MAR14] Marcinka, J., Mirzianov, O.,Mitasiunas, A.: Learning Process Maturity Model. In: Software Process Improvement and Capability Determination, Communications in Computer and Information Science, Volume 477, pp. 261-267 (2014)

[MAR04] Marshall, S., Mitchell, G. Applying SPICE to e-Learning: An e-Learning Maturity Model? ACE'04: Proceedings of the Sixth Conference of Australasian Computing Education, Australian Computer Society, pp. 185-191.

[MAR01] Marzano R.J. Designing a New Taxonomy of Educational Objectives. Corwin Press, 2001.

[MAR08] Marzano R.J. Kendall J.S. Designing & Assessing Educational Objectives: applying the new taxonomy. Thousand Oaks, CA 91320, USA Designing 2008.

[MIT12] Mitasiunas Antanas, Novickis Leonids. Enterprise SPICE based education capability maturity model. Workshops on business

Informatics research. Proceedings. Series: Lecture Notes in Business Information processing (ISSN 1865-1348), Vol. 106, ISBN 9783642292309, p. 102-116, Springer-Verlag Berlin Haidelberg 2012.

[NOV11] Novickis, Leonids, Mitašiūnas, Antanas, Rikure, Tatiana, Jurenoks, Aleksejs. Promotion of e-learning solutions via information technology transfer concept and Baltic regional competence network. VARE 2011: Virtual and augmented reality in education: proceedings of annual international conference, March 18, 2011, Latvia. ISBN 9789984633183 p. 71-80.

[OZC14] Ozcan, O., Demirors, O.: Assessing Software Agility: An Exploratory Case Study. In: Software Process Improvement and Capability Determination, Communications in Computer and Information Science, Volume 477, pp. 202-213 (2014)

[PEO09] People Capability Maturity Model (P-CMM), Version 2.0, Second Edition, SEI CMU, 2009. CMU/SEI-2009-TR-003, ESC-TR-2009-003, 2009.

[THO04] Thompson, E. Towards a learning process maturity model. In Freyberg, C., Hartmann, S., Kaschek, R., Kinshuk, Schewe, K.-D. & Turull Torres, J. M. (Eds.) *PhD Workshop 2004.* Palmerston North, Department of Information Systems, Massey University.

[THO06] Thompson, E. Using a Subject Area Model as a Learning Improvement Model. The Eighth Australasian Computing Education Conference (ACE2006), Hobart, Tasmania, Australia, January 2006.

[WEL03] Wells, C., Ibrahim, L., LaBruyere, L. A New Approach to Generic Attributes. Systems Engineering, Vol. 6, No.4, 2003, Wiley Periodicals, 2003.

Teaching Process Improvement by establishing Process Modeling Profile to drive Process Improvement – The PRO2PI-WORK4E Method

Clenio F. Salviano
Centro de Tecnologia da Informação Renato Archer
Rodovia D. Pedro I, km 143.6, CEP 13069-901
Campinas, SP, Brazil
clenio.salviano@cti.gov.br

Abstract

A method has been defined and used to guide teaching and learning on Software Process Improvement (SPI) courses. This method is a customization for education of a method for initiating a process improvement cycle in an organization. This method had been used in twenty-three SPI courses. During the classes, each student learns basic concepts of Software Process Improvement and selected reference models, related these concepts and models with his or her actual work environment and work processes, and constructs a proposal for a process improvement.

1. Introduction

Around 2003, a group of Software Process Improvement (SPI) specialists created a specialization postgraduate courses *lato sensu* on SPI. The objective was to disseminate its concepts, techniques, methods and reference models to professionals all over Brazil. This type of course has minimum load 360 hours and only allow the admission of graduates of higher education. It is an alternative to a master degree. The specialization course was offered as distance e-learning from the Federal University of Lavras (*Universidade Federal de Lavras – UFLA*) with eight specific courses. For each specific course, support materials were produced, including a reference book and exercises. Each specific course lasted one month. Each specific course was completed with a four hours classroom lecture at UFLA Campus in the city of Lavras once a year. The specialization course was completed with a monograph.

For this specialization course, I was responsible for the specific course of Software Process Assessment and Improvement with ISO/IEC 15504-5 model. The actual objective of this specific course was to introduce SPI. I thought over SPI, teaching and learning processes and how the teach SPI.

Teaching SPI is a challenging effort. When we extend SPI from software related processes to knowledge working related process, we understand that teaching SPI can be considered as SPI. Knowledge worker, as defined first by Drucker, is a worker that thinks for a living [Dru59]. Software related worker is a knowledge worker. Teaching is a knowledge worker activity. Therefore teaching SPI is a process that, using the concept of process capability, should be performed, managed, established, predicable and improved. Hence, a teacher teaching SPI should follow a method for teaching process.

Conversely, during a course, the students are in a learning process. In order to guide the learning process, a constructivist-based process can ne used. Constructivism is a psychological theory of knowledge (epistemology) that argues that humans construct knowledge and meaning from their experiences. Paulo Freire, a Brazilian educator, declared in his Pedagogy of Freedom that “knowledge cannot be transferred, knowledge must be constructed” and therefore “to teach is not to transfer knowledge but to create the possibilities for the production or construction of knowledge” [Fre98]. Therefore to teach SPI we need to create the possibilities for the production or construction of SPI knowledge.

Therefore, I decided to create this possibility by relating the teaching process with my experience in

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

helping organizations to perform process improvement cycles and the learning process with the students' actual experiences in their work processes.

The students in SPI courses are Information Technology professionals. They should learn SPI by a combination of studying and doing SPI. Hence, to teach SPI we customized a SPI Method in order to improve both teaching and learning. The method guides the students in a SPI experience related with starting a SPI cycle in their actual work.

From the experiences of the process used in this specific course, a method was consolidated to guide new editions of this specific course. The objective of this article is to share the experiences on using this method.

The customized method is named as PRO2PI-WORK4E. It is part of an innovative process improvement methodology: PRO2PI (Process Modeling Profile to drive Process Improvement) [Sal04] [Sal09a]. A methodological element of PRO2PI is a PRO2PI-WORK method. This method guides a workshop to establish a Process Modeling Profile in order to start a process improvement cycle. PRO2PI-WORK4E ("*for* education") is a customized version of this method to be used to teach process improvement.

This article is organized in six sections. This Section 1 is an introduction and a contextualization of the article. Section 2 provides a brief overview of PRO2PI Methodology. Section 3 introduces the PRO2PI-WORK4E method. Section 4 presents information about applications of this method in SPI courses. Section 5 presents further work. Finally Section 6 presents conclusions.

2. PRO2PI Methodology

PRO2PI (Process Modeling Profile to drive Process Improvement) is a methodology for software and other knowledge worker process improvement driven by Process Modeling Profile with elements from multiple reference models.

A Process Modeling Profile is a set of specification and descriptive models of knowledge worker processes. Each model is from one of three types of models: Process Capability Profile, Process Enactment Description and Process Performance Indicator. As the inclusion of Process Enactment Description and Process Performance Indicator are research proposals, for teaching purpose, only Process Capability Profile is used. Therefore, from now on, Process Capability Profile is used instead of Process Modeling Profile.

There are two types or representations of reference models for SPI: staged and continuous. Staged reference models define maturity levels. Continuous defines processes or process areas and capability levels. A Process Capability Profile is a set of processes or process areas in capability levels. A maturity level is an example of a Process Capability Profile.

PRO2PI is defined as a methodology following the meaning of the term methodology used by Schreiber et al [Sch00] in Knowledge Engineering. Schreiber et al presents the elements and their relationships of a methodology as a pyramid with feedbacks cycles. A methodology is a sequence of feedbacks cycles with a worldview based on a set of principles that form the baseline of a methodology. This worldview is grounded in theories that provide the essential concepts for establishing the methodology. The methods (and models, meta-models and other methodological components) and tools provide the key to enable the practical application of the methodology. The use of this methodology (the experiences) produces feedback that feeds the other "layers" of the methodology and enables the evolution of the methodology.

As a multi-model methodology, PRO2PI supports process improvement using elements from multiples process capability models and other sources. These elements are selected or defined and are integrated as a Process Capability Profile. A Process Capability Profile that drives a process improvement under PRO2PI methodology is also named as a PRO2PI.

The current version of PRO2PI methodology has four groups of methodological elements:

- Process Modeling Profile Metamodels,
- PRO2PI Quality Models,
- Process Improvement Methods, and
- Method Framework for Process Models.

Process Improvement Methods is centered in PRO2PI-CYCLE. PRO2PI-CYCLE is a method to guide a process for process improvement cycles including a function to define and use a PRO2PI. PRO2PI-WORK defines six phases. The first phase is Prepare for improvement cycle. It starts after a decision and commitment for improvement. The second phase is Establish improvement references. The third phase is Prepare for improvement actions. The fourth phase is Implement improvement actions. The fifth phase is Prepare improvements institutionalization. The sixth phase Institutionalize improvements produces an Improved organization. In the first phase, a first version of a PRO2PI is defined. Then, in each one of the following phases, the PRO2PI can be revised and updated and its current version is always used to drive the actions. An article presents detailed information and examples about this cycle as a modeling view of SPI driven by a PRO2PI [Sal11].

For small organizations, I realized that the improvements actions should starts as soon as possible and should produce visible results soon. In order to start improvements action (the fifth phase) I defined a specific method (PRO2PI-WORK) to implement the first four phases as a workshop. This workshop lasts two or three days.

Therefore PRO2PI-WORK is a method for workshop to establish a PRO2PI. This method has been developed to be used in traditional process improvement cycle methods, as, for example, IDEAL and ISO/IEC 15504 cycle, or in a PRO2PI-CYCLE process improvement cycle. PRO2PI-WORK method is composed of four phases: Preparation, Analysis, Consolidation and Conclusion.

There are two customized variations of PRO2PI-WORK method. One of them is PRO2PI-WORK4A (PRO2PI-WORK for Assessment) for a workshop with emphasis in the assessment of current practices. The other one is PRO2PI-WORK4E (PRO2PI-WORK for Education) for a workshop with emphasis in education on process improvement.

3. PRO2PI-WORK for Education

PRO2PI-WORK4E is method to guide SPI teaching and learning processes during a classroom or distance-learning course.

During the classes, each student learned basic concepts of Software Process Improvement in general, including its history, objectives and definition, process assessment, process enactment description and reference models for SPI, as, for example, CMMI-DEV [Cmm10] and ISO/IEC 15504-5 [Iso06]. Each student also related these topics with the work environment and work processes, and construct a proposal for a process improvement of his/her work processes. This proposal is documented as an article.

As a customization of PRO2PI-WORK Method, PRO2PI-WORK4E is composed of four phases:

- Preparation,
- Analysis (in this case by teaching SPI and Models with an analysis of an Organizational Unit to prepare for improvement),
- Consolidation (in this case by teaching Process Assessment and Improvement with a consolidation of a PRO2PI) and
- Conclusion.

PRO2PI-WORK4E is defined with four phases and twenty-one activities:

Phase 1: Preparation
A.1.1 Analyze information about the specific course
A.1.2 Select process areas from reference models
A.1.3 Select and customize teaching materials
Phase 2: Analysis
A.2.1 Introduce the specific course
A.2.2 Present an introduction to SPI
A.2.3 Identify an Organizational Unit (OU) <W>
A.2.4 Describe a current macro OU process<W>
A.2.5 Identify business factors and goals <W>
A.2.6 Present selected process areas
A.2.7 Identify process areas relevance <W>
A.2.8 Review work and propose PRO2PI <W>
Phase 3: Consolidation
A.3.1 Present process capability and assessment
A.3.2 Present examples of capability levels
A.3.3 Estimate process capability <W>
A.3.4 Present improvement cycle methods
A.3.5 Propose improvement goals/actions <W>
A.3.6 Review work and PRO2PI <W>
A.3.7 Present research directions on SPI
A.3.8 Each student presents proposed PRO2PI<W>
Phase 4: Conclusion
A4.1 Conclude PRO2PI and article <W>
A4.2 Conclude specific course

In the activities identified with <W> the emphases is in the practical work by the students with presentation with concepts, examples and orientation for the practical work.

For each activity, there are artifact templates and examples. One of them is a template and guidelines for the article with a correspondence between each section and each practical work result. The article is from 6 to 12 pages long. In addition to title, authors names, authors affiliation, abstract and introduction, in the beginning, and the references in the end, the article should contains the following sections:

Section 1. Introduction to the article
Section 2. Context, with a description about the organizational unit;
Section 3. Conceptual references, with a introductory view on software process improvement, the selected model and the method used;
Section 4. Related work, with identification and comments on related work to this work;
Section 5. Process used, with a description of how the work was developed;
Section 6. Business factors, and business goals of the organizational unit;
Section 7. Description of the macro process of the organizational unit;
Section 8. Statement about the relevance and risk for selected process areas for the organizational unit;

Section 9. Process Capability Profile for process improvement Section 10. Improvement goals and improvement actions for the organizational unit; Section 11. Conclusions.

This article is constructed during the classes, using specific techniques and templates for each practical work activity.

In Activity A.2.2 - Present an introduction to SPI, the SPI manifesto is used to communicate the values and principles of SPI [Pri10]. In Activity A.2.4 - Describe a current macro OU process, the instructor presents objectives, concepts, notations and examples of process enactment descriptions. Then each student identifies and describes a macro process to be improved. In Activity A.2.5 - Identify business factors and goals, instructor presents objectives, concepts, the SWOT (Strengths, Weaknesses, Opportunities, Threats) technique [Hum04] for business factors, and the approach by Potter and Sakry [Nei02] for business goals definition. Then each student identifies business factors and describes business goals.

In Activity A.2.6 - Present selected process areas, the instructor presents each selected process area with: the concept behind it; its definition from its model; general comments about it; symptoms that are often seen when its practices are missing; and reasons why it may be important. The idea and examples of presenting symptoms and reasons are from a presentation by Garcia et al [Gar08].

In order to guide the understanding of each presented process area, each student relates it to his/her work environment and defines their relevance for process improvement. Hence, in Activity A.2.7 - Identify process areas relevance, for each presented process area, each student defines:

a) How it is performed in the OU, including an identification of the actual process or group of processes that correspond to the process area presented, and information about how it is performed;
b) What is the relative importance of this process area for the business goals, expressed in a three-value scale: low, medium and high; and
c) What is the relative risk for the organization if it continues to perform this process area as it is now, expressed in a three-value scale: low, medium and high.

After the identification of relevance of all presented process areas, each student constructs a three by three bi-dimensional matrix with relative importance and relative risk.

In Activity A.2.9 - Review work and propose PRO2PI, each student first identifies in the result of previous activity, the process areas that are in higher importance and higher risk for the OU, analyze the business factors and goals, and then select two or three process areas that could guides an improvement cycle in this OU. These process areas are the first version of the proposed PRO2PI. The instructor provides further orientations and examples for this activity.

In Activity A.3.3 - Estimate process capability, each student first estimate the current process capability level of each process area in the proposed PRO2PI and then propose a level to be achieved after the improvement cycle. The proposed levels became part of the PRO2PI. In Activity A.3.5 - Propose improvement goals/actions, each student proposes improvement goals and actions to achieve these goals and include these goals and actions in the PRO2PI. The approach by Potter and Sakry [Nei02] for define compelling improvement goals and improvement actions is used.

In Activity A.3.6 - Review work and PRO2PI, each student revises again the activity results and consolidates a version of the PRO2PI.

4. Using PRO2PI-WORK4E

In the last twelve years, I used PRO2PI-WORK4E method to guide twenty-three SPI teaching experiences. Table 1 presents data on these twenty-three applications of PRO2PI-WORK4E Method.

In Table 1 each application of PRO2PI-WORK4E Method is characterized by an identification, from C01 to C07, of the course in which a specific course was teach, the month and year when it was teach, the number of students and the number of articles produced. Usually the number of articles is smaller than the number of students because some articles were produced by groups of students.

C01 is the specific course “Introduction to SPI using ISO/IEC 15504-5 (SPICE)” of “Software Process Improvement” Specialization pos-graduate course at Federal University of Lavras (*Universidade Federal de Lavras – UFLA*). The specific course is 36 hours of distance e-learning with material, orientations, exercises and chats using the Moodle software system during one month. The specific course is completed with 4 hours of a classroom lecture at UFLA Campus in the city of Lavras.

C02 is the specific course “Standards for Software Process - ISO/IEC 15504-5 (SPICE)” of “Quality Software Development” Specialization pos-graduate course at SENAC School of Exact Sciences and

Technology (*Faculdade SENAC de Ciências Exatas e Tecnologia*). The specific course is total of 40 hours with 10 classroom lectures at SENAC Campus in the city of São Paulo.

Table 1 – Twenty-three applications

ID	Course	Year/month	#students	#articles
01	C01	2004/05	18	18
02	C02	2004/09	22	10
03	C01	2004/11	37	31
04	C02	2005/02	11	4
05	C01	2005/05	27	20
06	C03	2005/06	31	13
07	C03	2005/06	24	9
08	C01	2005/10	42	32
09	C04	2005/11	22	17
10	C05	2006/11	27	10
11	C01	2006/03	30	11
12	C01	2007/05	32	20
13	C01	2007/10	36	20
14	C01	2008/05	32	19
15	C01	2008/11	25	18
16	C06	2008/06	28	8
17	C06	2009/06	20	10
18	C05	2010/11	24	12
19	C05	2011/11	22	4
20	C05	2012/11	14	4
21	C07	2013/09	46	17
22	C08	2014/09	22	17
23	C07	2015/04	36	10
		TOTAL	**628**	**334**

C03 is the specific course "Software Process Quality" of "Software Engineering" Specialization pos-graduate course at São Judas Tadeu University (*Universidade São Judas Tadeu - USJT*). The specific course is total of 12 hours with 4 classroom lectures at USJT Campus in the city of São Paulo.

C04 is the specific course "Introduction to SPI with CMMI" of "Capability Maturity Model Integration" Specialization pos-graduate course at Federal University of Lavras (*Universidade Federal de Lavras – UFLA*). The specific course is 36 hours of distance e-learning with material, orientations, exercises and chats using the Moodle software system during one month. The specific course is completed with 4 hours of a classroom lecture at UFLA Campus in the city of Lavras.

C05 is the specific course "Software Process Quality" of "Software Engineering" Specialization pos-graduate course at Piracicaba Methodist University (*Universidade Metodista de Piracicaba – UNIMEP*). The specific course is total of 24 hours with 4 classroom lectures at UNIMEP Campus in the city of Piracicaba.

C06 is the specific course "Software Process Improvement" of "Software Quality Management" Specialization pos-graduate course at Paulista Informatics and Management School (*Faculdade de Informática e Administração Paulista - FIAP*). The specific course is total of 24 hours with 6 classroom lectures at Aclimação Campus in the city of São Paulo.

C07 is the specific course "Software Process Models" of "Information Technology Governance" Specialization pos-graduate course at Unicamp Technological School (FT Unicamp). The specific course is total of 24 hours with 4 classroom lectures at FT Unicamp Campus in the city of Limeira.

C08 is the specific course "Software Process Models and Assessment" of "Software Process Improvement" Specialization pos-graduate course at Vale dos Sinos University (*Universidade do Vale dos Sinos – Unisinos*). The specific course is 24 hours of distance e-learning with material, orientations, exercises and chats using the Moodle software system during one month. The specific course is completed with 4 hours of a classroom lecture at Unisinos Campus in the city of São Leopoldo.

In spite of the different names of each specific course, all of them are about an Introduction to Software Process Improvement. In each of them, a specific Reference Model is presented and other models are also commented. Given the dissemination in Brazil, four reference models are used: the ISO/IEC 15504-5 Exemplar Process Assessment Model (ISO/IEC 15504-5) [Iso06], Capability Maturity Model Integration for Development (CMMI-DEV) [Cmm10a] and for Services (CMMI-SRV) [Cmm10b], Brazilian Software Process Improvement Reference Model (*Modelo de Referência da Melhoria de Processo do Software Brasileiro – MR-MPS.BR*) [Mon09]. In a more recent application, at *Vale dos Sinos* University, CERTICS

Reference Model for Assessment (*Modelo de Referência para Avaliação da CERTICS*) [Sal14] was also used as reference.

5. Further Work

For each application, I get feedback from the students and analyses the results. Minor adjustments have been made for each application to implement minor improvements. Although there are 23 applications in 12 years with participation of 628 students and production of 334 articles, PRO2PI-WORK4E is a work in progress because I neither analyze these data in a systematized fashion nor transfer this method to another instructor. Slides, templates and results, including articles from students, for each specific course are registered. A further work is to analyze these results.

Another further work is to identify an appropriate pedagogical reference and to analyze and improve PRO2PI-WORK4E from this reference. A candidate reference is andragogy. According to the article Malcolm Knowles an American practitioner and theorist of adult education, andragogy as “the art and science of helping adults learn”. Knowles identified the six principles of adult learning as: (a) Adults are internally motivated and self-directed; (b) Adults bring life experiences and knowledge to learning experiences; (c) Adults are goal oriented (d) Adults are relevancy oriented; (e) Adults are practical; and (f) Adult learners like to be respected. A preliminary analysis of these principles indicated that they are relevant for PRO2PI-WORK4E.

PRO2PI-WORK4E is described in Portuguese language as all slides and other support materials because all applications are in Brazil. There is an English version of slides for a tutorial on PRO2PI-WORK (and PRO2PI-WORK4E) that has been presented in international conferences, as, for example, Euromicro SEAA 2012, SPICE 2008 and EuroSPI 2009 [Salb09]. These tutorials and now this article increase the dissemination of PRO2PI-WORK4E.

6. Conclusions

This article presented a method and a balance about experiences with constructivist-based education on Software Process Improvement. The education experiences were guided by PRO2PI-WORK4E method. The twenty-three post graduate courses, the participation of 628 students in those courses, the production of 334 articles with proposals for process improvements for the actual work processes, and the positive feedbacks from them, gives us confidence that this is a valid experience and it deserves to be disseminated. Due to limitations, the experiences were more in identify and planning process improvement actions then actually implement them. Some students continued these experiences after the classes, completing the process improvement cycle.

Acknowledgments

The author would like to thank all students in the various specific courses where PRO2PI-WORK4E was used and the anonymous reviewers of SPETP 2015 Workshop for their comments and suggestions for this article.

References

[Cmm10a] CMMI Product Team, CMMI® for Development, Version 1.3, Improving processes for developing better products and services, Technical Report, CMU/SEI-2010-TR-033, ESC-TR-2010-033, Software Engineering Process Management Program, November 2010.

[Cmm10b] CMMI Product Team, CMMI® for Services, Version 1.3, Improving processes for providing better services, Technical Report, CMU/SEI-2010-TR-034, ESC-TR-2010-034, Software Engineering Process Management Program, November 2010.

[Dru59] P. Drucker, Landmarks of Tomorrow - A Report on the New 'Post-Modern' World, Harper & Row, New York, 1959.

[Fre98] P. Freire, Pedagogy Of Freedom: Ethics, Democracy, And Civic Courage. Lanham : Rowman & Littlefield Publishers, 1998. Print.

[Gar08] S. Garcia, S. Cepeda, G. Miluk, M. J. Staley, Adopting CMMI for Small Organizations, slides presented at Fourth Annual CMMI Technology Conference and Users Group, Denver, USA, November 2004 (available at http://www.dtic.mil/ndia/2004/CMMIT2Mon/110504Cepeda.pdf, last accessed in 17/02/2005)

[Hum04] A. S. Humphrey, The origins of the SWOT analysis model, in SWOT Analysis, by Alan Chapman, www.bussinessballs.com, 2004.

[Iso06] The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC 15504-5 - Information technology —

Process assessment, Exemplar Process Assessment Model - 2006.

[Kno12] M. S. Knowles, E. F Holton III, R. A Swanson, The Adult Learner: The definitive classic in adult education and human resource development. New York, NY, Routledge, 2012.

[Mon09] M. A. Montoni, A. R. Rocha, and K. C. Weber, MPS.BR: a successful program for software process improvement in Brazil, in Software Process: Improvement and Practice, Volume 14 Issue 5, September 2009, pages 289-300, DOI 10.1002/spip.v14:5.

[Nei02] N. S. Potter and M. E. Sakry, Making Process Improvement Work: A Concise Action Guide for Software Managers and Practitioners, Addison-Wesley Professional, ISBN 0201775778, 2002.

[Pri10] J. Pries-Heje and J. Johansen (Chief Editors), SPI Manifesto, eurospi.net, version A.1.2.2010.

[Sal04] C. F. Salviano, M. Jino and M. J. Mendes, Towards an ISO/IEC 15504-Based Process Capability Profile Methodology for Process Improvement (PRO2PI), International SPICE Conference Proc., Lisbon, Portugal, p. 77-84, April 2004.

[Sal09a] C. F. Salviano, A Multi-Model Process Improvement Methodology Driven by Capability Profiles, In Proc. of IEEE COMPSAC, Seattle, USA, p. 636–637, DOI 10.1109/COMPSAC.2009.94, 2009.

[Sal09b] C. F. Salviano, Establishing ISO/IEC 15504-Based Process Capability Profile to Process Improvement, slides for a tutorial presented at 16th EuroSPI 2009, 2-4 September 2009, University of Alcala, Madrid, Spain (available from http://pro2pi-english.wikidot.com/publications-about-the-methodology, last accessed in 17/02/2005).

[Sal11] C. F. Salviano, A Modeling View of Process Improvement, in SPICE Conference - Software Process Improvement and Capability dEtermination, May 31, 2011, pp. 1-12.

[Sal14] C. F. Salviano, A. M. Alves, G. Stefanuto, S. T. Maintinguer, C. V. Mattos, C. Zeitoum, CERTICS - An ISO/IEC 15504 Conformance Model for Software Technological Development and Innovation, In 14th SPICE International Conference, 2014, Vilnius, Lituania, v. 477. p. 48-58. DOI 10.1007/2F978-3-319-13036-1_5.

[Sch00] G. Schreiber, H. Akkermans, W. V. Shadbolt, and B. Vielinga, Knowledge Engineering and Management – The CommonKADS Methodology. USA: The MIT Press, 2000.

Software Process Education Oriented to Software Industry Needs

Carlos S. Portela
Federal University of Pernambuco
Recife, PE 50740-560
csp3@cin.ufpe.br

Alexandre M. L. Vasconcelos
Federal University of Pernambuco
Recife, PE 50740-560
amlv@cin.ufpe.br

Sandro R. B. Oliveira
Federal University of Pará
Belém, PA 66075-110
srbo@ufpa.br

Abstract

Typically, software professionals are trained in undergraduate courses as a way of preparing to the industry. However, there is a shortage of qualified professionals in relation to Software Process field in the Brazilian industry. Thus, software companies have to provide skills related to the Software Process areas through training. To address this problem, this PhD research aims to analyze the recommendations for the Software Process education in curriculum guidelines of computer courses in Brazil. In addition, we will examine which of these recommendations are relevant to software professionals. Furthermore, we propose a teaching approach that aims to meet the goals of these curriculum guidelines through models and quality standards for the process and software product widely adopted in the Brazilian software industry.

1. Motivation

The Software Engineering (SE) teaching is one of the topics of greatest importance in courses in the field of computing [Iee14]. This arises from both the relevance of the software itself, which has a strategic role in modern society, and the challenges related to the complete formation of a professional who will work in companies that consume or produce information technology resources. The result is an increase in demand well-qualified SE professionals in the software industry [Dul03]. It believed that, in the future, all general-purpose software will constructed by a software engineer [Nun10], because is a matter of obtain quality and reliability of the developed software product.

1.1 Gap Area

Bachelors of computer courses working as software professionals in Brazil learn more about Software Process topics after undergraduate study because the necessary skills are not adequately addressed in graduate [Wan09]. The Brazilian software industry presents a shortage of suitably qualified professionals to work in professions that involve stages of the software development process, encompassed by SE [Abe14].

The root of the problem may be in the training of these professionals, i.e., the approach adopted for the teaching of Software Process during undergraduate education. Several authors have reported difficulties found in the teaching of SE, as Soares [Soa04], Castro, Gimenes, and Maldonado [Cas00], and Hazzan and Dubinsky [Haz03]: (i) too much content being given in a short time; (ii) low motivation that the students have to study the theoretical concepts of SE; (iii) difficulties in preparing students for professional practice within academic environments.

1.2 Research Scope

According to the ACM/IEEE [Acm13], SE is a discipline concerned with the application of theory, knowledge and practice for the effective and efficient development of software systems that meet user's requirements. To fulfill the users' needs, SE professionals must to ensure deadlines, costs, and quality of the product developed. However, defining a software process is not a trivial activity, especially when the objective is to ensure high quality products and a competitive level of productivity [Mac05].

In order to meet customer requirements, with respect to the product generated, the industry has adopted the ISO/IEC 25000 standard [Iso14], which specifies quality attributes that monitor throughout the software development process [Mac05]. Regarding the quality of the development process, the Brazilian software industry has followed several reference models, such as CMMI-DEV [Cmm10] and MR-MPS-SW [Sof12], which define processes areas that comprise maturity and capacity profiles.

Thus, this PhD research will examine the recommendations for the teaching of Software Process available in the curriculum guidelines of computer courses in Brazil, more specifically with regard to the software process area. This analysis aims to obtain qualitative data about: (i) teaching and learning of

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

software process from teaching approaches applied in undergraduate computer courses; and (ii) the knowledge about software process, considered necessary, for professionals in this area. The results of this analysis will used to define a new teaching approach that aims to meet the goals of these curriculum guidelines through quality models and standards widely adopted in the software industry to define and improve software process.

2 Problem Statement and Related Work

We can classify the problems of this PhD research (Research Problems - RP) in three groups. (RP-I) First, there is a need to analyze in detail the curriculum guidelines of undergraduate computer courses in Brazil, in order to identify which the software process activities are contemplated. After that, there is the need to analyze how these curricular recommendations are being implemented in Brazilian computer courses to identify the teaching approaches adopted by teachers of SE. (RP-II) Subsequently, there is a need to investigate what activities are actually relevant to SE professionals to define your software process. (RP-III) Furthermore, we will investigate the correlation between activities of process and quality characteristics of software product. The aim is to analyze what activities affect certain characteristics of product quality.

2.1 Problems Area

SE professionals working in the industry have dissatisfaction regarding the level of preparedness of recently graduated students entering the job market [Let07] [Hil07]. Software companies have to complement the knowledge of recent graduates with training and have to provide technical and non-technical skills related to the software development process [Bes12].

According to Lethbridge, Diaz-Herrera, LeBlanc, and Thompson [Let07], this deficiency in the formation of graduates in the SE area is the result of an inadequate education. This finding may be reinforced by the research done by Sargent [Sar04], which reveals that: (i) only 40% of IT professionals in the United States have training in this area; (ii) 40% of those are aware of the main fields of SE, such as requirements, architecture, testing, human factors, and project management.

Although we did not find statistical data in relation to Brazil, it believe that the reality of SE professionals in this country should not be different, given the scenario observed by the authors of this paper on numerous consulting assignments involving the implementation of Software Process Improvement.

2.2 Limitations of Related Work

There are several works dealing with teaching SE approaches [Nun10] [Cas00] [Kit07] [Kit08]. However, these studies do not directly address RP-I. So far, all identified studies that relate to RP-I tend to perform a restricted analysis, focusing primarily on proposing teaching approaches for a particular institution without examining the main curriculum guidelines of the area.

Unlike these works, this research will cover the main curriculum guidelines of the computing area [Acm13] [Sbc05] [Mec03]. In addition, we will conduct an empirical study about the teaching approaches adopted by teachers in applying the Software Process topics proposed in these curriculum guidelines.

Regarding the RP-II, there are few studies that investigate the relevance of process activities to SE professionals [Wan09] [Soa04] [Let07]. From these studies results, it is not possible to state if the problem is in the recommendations of curriculum guidelines, or in the SE teaching approaches.

Differently from these works, besides consulting the SE professionals' opinions about the relevant topics of software process, this PhD research will consider the teaching approaches used by the teachers and their effectiveness by consulting the students of these disciplines.

Finally, among the problems of RP-III, we intend to investigate the correlation between process activities and quality characteristics of the software product. Maciel [Mac05] does a mapping between the process activities contained in the ISO/IEC 12207 and ISO/IEC 15504 standards, and the product quality characteristics from the ISO/IEC 9126. However, the ISO/IEC 9126 standard is outdated. This PhD research will consider in its correlation analysis the product quality characteristics from the predecessor of the ISO/IEC 9126 standard, the ISO/IEC 25000 [Iso14].

3 Questions and Hypotheses

Our main goal is to propose an approach based on quality standards widely adopted in the software industry in order to support the teaching and learning of Software Process in computer courses. In this context, we define our research questions as follows.

RQ1. *What are the Software Process topics covered in the curriculum guidelines of the computer courses?*

RQ2. *What are the Software Process topics covered in computer courses curricula?*

RQ3. *What are the Software Process topics effectively learned by computer students?*

RQ4. *What are the Software Process skills required by the software industry and which of them were acquired in the computer courses?*

We defined our research questions to try refuting the following null hypothesis:

H0. *The current approaches to teaching Software Process meets the software industry needs.*

If the null hypothesis is refuted, we intend to test our alternative hypothesis:

H1. *The current approaches to teaching Software Process does not meet the software industry needs due to misalignment between the curriculum of Software Engineering discipline and the real industry needs.*

The research methods used to answer the research questions and to test H0 and H1 will presented in the next section.

4 Research Method and Progress

In order to answer our research questions, first we are conducting some exploratory studies to understand the curriculum guidelines proposed by ACM/IEEE [Acm13], the Brazilian Computer Society (SBC) [Sbc05] and the Brazilian Ministry of Education (MEC) [Mec03].

Subsequently, we have to address the problems that arise in applying these curriculum guidelines in undergraduate courses. These approaches will analyzed in order to identify strengths and weaknesses. This analysis, as well as the surveys conducted with teachers, students, and industry professionals, will be inputs to the definition of an approach oriented to product and process quality profiles. This profile represents the set of characteristics/process areas that be refined and institutionalized in the organizational environment. The process profile will based on the CMMI-DEV [Cmm10] and MR-MPS-SW [Sof12] quality models and the product profile will based on ISO/IEC 25000 [Iso14]. We chose these models and standards because of their wide acceptance in the efforts of the Brazilian software industry in obtaining quality in software development. For example, the Brazilian MR-MPS-SW model was officially deploying 593 companies in Brazil[1]. In addition, the international CMMI-DEV model was officially implanted in 203 Brazilian companies[2].

4.1 Identifying Process Activities Included in Computer Courses

To answer RQ1, we will conduct a literature review in the curriculum guidelines from ACM/IEEE [Acm13], SBC [Sbc05] and MEC [Mec03] aiming to identify which Software Process topics are contemplated in these guidelines. The results of this review may support (or refute) the hypothesis H0, giving us evidence that the process activities suggested in the curriculum guidelines meet (or not) the software industry demands.

Then, in order to answer RQ2, a survey (S-I) will be conducted with the teachers of undergraduate Computer Science courses. The goal of this survey is to analyze which process activities identified in the literature review are included in the SE curricula disciplines. These results may validate the H0 hypothesis. If this confirmed, the problem may be in the curricula adopted in the SE disciplines.

Finally, answering RQ3, a survey (S-II) will be conducted with students that concluded the Software Engineering discipline. The aim of this survey is to assess whether the students are learning the process activities contemplated in SE disciplines. The results of S-II can validate H0 too, giving us, evidence that the problem may be in the teaching approaches adopted in the classrooms. We are currently working on this phase of the research.

Both S-I and S-II will be applied to undergraduate Computer Science courses from public and private universities in Brazil and will follow the guidelines of Kitchenham and Pfleeger [Kit08].

4.2 Identifying Process Activities Relevant for the Software Industry

To answer RQ4, our goal is, through a survey (S-III), to consult industry professionals about which of their skills to perform process activities were acquired during undergraduate study. In this survey, the goal is

[1] http://www.softex.br/mpsbr/, accessed in May 2015.

[2] http://cmmiinstitute.com/, accessed in May 2015.

to find information on the relevance of the topics covered in the SE disciplines according to the opinion of professionals in this area. The results of S-III can refute the H0, giving us evidence that the process activities suggested in curriculum guidelines dot meet the software industry demands reported by industry professionals. In this way, we can work on validation of H1.

S-III will applied in public and private software companies in Brazil and will follow the guidelines of Kitchenham and Pfleeger [Kit08].

4.3 Defining an Approach to Teaching Software Process Oriented to Quality Standards

After conducting exploratory studies for answering the four research questions, we will obtain: (i) the recommendations of curriculum guidelines; the teaching approaches of SE teachers; and (ii) the considerations of students and industry professionals. These results will considered in the definition of the teaching approach proposed by this research.

Additionally, we plan to conduct a systematic mapping of the relationship between the SE practices recommended by the CMMI-DEV [Cmm10] and MR-MPS-SW [Sof12] models and the practices contained in ISO/IEC 25000 [Iso14] product quality standard. This systematic mapping will follow the guidelines of Kitchenham and Charters [Kit07]. Based on the understanding of this relationship, it will be possible to integrate the concepts of process maturity and capability profiles, and product quality, which will be the basis of the methodology that will compose the teaching approach proposed in this PhD research.

The teaching approach will adopt the Problem Based Learning educational method that uses problems to initiate, motivate, and focus the knowledge acquisition, and to encourage the development of skills and attitudes in students that will be useful in a professional context [Bes12].

In order to validate our proposed teaching approach, we plan to conduct a controlled experiment in an SE discipline in an undergraduate Computer Science course. This experiment will follow the guidelines proposed by Wohlin [Woh00].

5 Expected Contributions and Partial Results

In summary, our PhD research is intended to: (i) identify process activities and analyze their relevance for the software industry; (ii) improve the understanding of curriculum guidelines of the SE disciplines through the analysis of these curricula; (iii) identify the potential problems in the approaches to teaching Software Process through the analysis of the implementation of curriculum guidelines; and (iv) furthermore, provide an educational approach to meet the training demands of SE professionals during undergraduate study.

The three surveys (S-I, S-II and S-III) are being applied in undergraduate computing of public and private universities in Brazil. These surveys are releasing in e-mail list, SE groups on social networks and in loco on public and private universities. The surveys is available in http://goo.gl/vn5jHS and the survey protocol is available in http://goo.gl/gqzMrP. By the time, we have obtained 42 responses.

Finally, we intend to publish the results to stimulate replication of this type of research and hence solve the gaps of the teaching SE area. In this context, we emphasize that the proper SE education is important to improve the current state of software development and help mitigate many of the traditional problems associated with the software industry.

Acknowledgments

I, Carlos Portela, would like to thank my supervisors Alexandre Vasconcelos and Sandro Oliveira for all the support in this PhD research. This research has financially supported by Improvement Coordination of Postgraduate People (CAPES).

References

[Iee14] IEEE Computer Society, *Guide to the Software Engineering Body of Knowledge – SWEBOK*, Version 3.0, January 2014; www.swebok.org.

[Dul03] R. Duley, G. Hislop, T. Hilburn, and A. Sobel, Engineering an introductory software engineering curriculum, *16th Conference on Software Engineering Education and Training*, Madrid, Spain, 2003, pp.99-106.

[Nun10] D. Nunes, C. Reis, and R. Reis, *Education in software engineering*, in The career of researcher in software engineering: principles, concepts and directions, 1st ed. Salvador, Brazil, 2010, pp. 132-181.

[Wan09] C. Wangenheim, and D. Silva, What software engineering knowledge is important for the professional software?, *II Forum on Education in Software Engineering*, Fortaleza, Brazil, 2009.

[Abe14] ABES, *Brazilian software market: scenario and trends*, in Brazilian Software Market and Services 2014, 1st ed. São Paulo, Brazil, 2014.

[Soa04] M. Soares, An experience in teaching software engineering oriented practical work, *I Workshop on Computer Education*, Vitória / Rio das Ostras, Brazil, 2004.

[Cas00] J. Castro, I. Gimenes, and J. Maldonado, A proposal for pedagogical plan for the Software Engineering discipline, *II quality course of undergraduate of Computing and Informatics area*, Curitiba, Brazil, 2000, pp. 251-270.

[Haz03] O. Hazzan, and Y. Dubinsky, Teaching a software development methodology: The case of Extreme Programming, *16th Conference on Software Engineering Education and Training*, Madrid, Spain, 2003, pp. 176-184.

[Acm13] ACM/IEEE. *Computer science curricula 2013*. Curriculum guidelines for undergraduate degree programs in Computer Science. December 20, 2013.

[Mac05] T. Maciel, *Definition and adapting of software process based on a Hierarchical Process Model*, M. S. thesis, Informatics Center (CIn), Federal University of Pernambuco (UFPE), Recife, Brazil, 2005.

[Iso14] ISO/IEC 25000 - Software Engineering – *Software Product Quality Requirements and Evaluation (SQuaRE)* - Guide to SQuaRE. Geneva, Switzerland, 2014.

[Cmm10] CMMI Product Team, *CMMI for Development, version 1.3*, Carnegie Mellon University, Pittsburgh, Software Engineering Institute, Pennsylvania, Technical Report CMU/SEI-2010-TR-033, November 2010.

[Sof12] SOFTEX, *SPI Reference Model for Software, version 2012*, General Guide SPI for Software, December 2012.

[Let07] T. Lethbridge, J. Diaz-Herrera, R. LeBlanc, and J. Thompson, Improving software practice through education: Challenges and future trends, *Conference Future of Software Engineering*, Minneapolis, MN, 2007, pp.12-28.

[Hil07] T. B. Hilburn, and M. Towhidnejad, A case for software engineering, *20h Conference on Software Engineering Education and Training*, Dublin, Ireland, 2007, pp. 107-114.

[Bes12] B. Bessa, M. Cunha, and F. Furtado, ENGSOFT: Simulation tool for real environments to support the Problem Based Learning (PBL) in teaching Software Engineering, *XX Workshop on Computer Education*, Curitiba, Brazil, 2012.

[Sar04] J. Sargent, An overview of past and projected employment changes in the professional IT occupations, *Computing Research News*, vol. 16, no. 3, pp. 1-21, May 2004.

[Sbc05] SBC. *Reference curriculum for undergraduate courses in Bachelor in Computer Science and Computer Engineering*, 2005.

[Mec03] MEC. *Curriculum Guidelines for Undergraduate courses in Computer Science, Computer Engineering, Software Engineering and Information Systems and the courses in Computer Science Degree*, 2003.

[Kit08] B. Kitchenham, and S. Pfleeger, Personal Opinion Surveys, in *Guide to Advanced Empirical Software Engineering*, Springer, 2008, ch. 3, pp. 63-92.

[Kit07] B. Kitchenham, and S. Charters, *Guidelines for Performing Systematic Literature Reviews version 2.3*, Keele University, University of Durham, Keele/Durham, UK, Technical Report EBSE-2007-01. July 2007.

[Woh00] C. Wohlin et al., Experimentation in software engineering: an introduction, in *Kluwer Academic Publishers*, Norwell, MA, 2000.

Establishing Long-lasting Relationships between Industry and Academia

Patricia McQuaid,
California Polytechnic State University
USA
pmcquaid@calpoly.edu

Ritendra Banerjee
School of Applied Learning in Testing
India

Abstract

So how can industry and academia work better together to produce graduates who not only understand the theory and practice of software engineering and process improvement, but understand the challenges, and have ideas for solutions? This paper will address these issues and serve as a basis to generate additional ideas.

1. The Issues

All too often while speaking at conferences, we hear people from industry saying that "the students coming out of college these days do not have the skills we need...". While that often is true, we propose a partial solution to this problem that is actually quite straightforward. Industry and academia need to establish long-lasting relationships, so academia can educate the emerging workforce in what industry needs. Industry needs college graduates who understand what quality software means, and to not only understand software processes, but how to improve them.

One element that universities and businesses have in common is budget constraints, especially in today's economy. Just like many businesses have cut back drastically with discretionary spending, so have many universities, perhaps even more so. As we know, professional conferences are very expensive; so are specialized training courses in what-ever your field of expertise may be. Now, in computer science and other information technology-related fields, the body of knowledge we need to know is extremely dynamic and very quickly becomes outdated. The technology used now, both hardware and software, was not even invented when many of us professors were in graduate school. So, much of what we teach in our current curricula is mostly self-taught. If we are lucky, we have been able to attend a training course on the topic we are teaching, but more likely, we had to learn it on our own. So how can dedicated faculty get the training they need to properly educate the future software engineering and software process professionals? By having industry establish more partnerships with universities, both informally and formally

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

Informal Alliance

Industry spends a great deal of money on training their personnel, from sending employees to conferences and to professional development seminars, to paying for a formal college degree. Many of the seminars held to train employees are held on-site, in which case either consultants are brought in to provide the training classes or they are held by the company's own staff. Very often, consultants charge on a per-person basis, but they also may charge based on a range of people, for example a fixed fee for between 20 and 25 attendees. It is this last case that we would like to address here. Would you consider that the next time you have an on-site training class, you invite your local professor to attend at no charge? What better way to help ensure that the college you recruit from teaches the material that you want your future employees to know? What would it cost you? Another set of training materials and a lunch? Or if they do charge by the person, is it worth it to you to pay this incremental cost of educating a professor? Probably so.

Formal Alliance

Establishing a more formal alliance is another option that may be even more worthwhile for everyone involved. This type of an alliance can take numerous forms. Would you consider funding a trip for a professor to a conference or for a course? Would you consider having your organizations' personnel guest-lecture at the university? How about providing tours of your company to faculty and students, so they can better see what you do, and see the environment in which they might work? Have you considered having the faculty work at your company for a period of time, perhaps over the Summer break?

One of a faculty member's challenges is developing real-world examples and exercises that are both meaningful and challenging to the students. Do you think your organization could provide material and examples that professors can use in their courses? That would be extremely valuable, to allow professors to use rich, meaningful material as example, assignments, and case studies. You could remove anything confidential, but provide information that we can incorporate into our classes.

Some companies have formal programs that involve professors, from establishing visiting positions ranging from a several month appointment to lasting several years. One of the authors was fortunate to be involved in such an alliance several years ago, spending five weeks during the Summer as part of a major organization's faculty partnership program, working in one of their major software testing labs. This experience was invaluable.

2 Benefits to the Faculty and to Industry

At the time, one of the authors was considering offering a new class for the university, a class related to software testing and process improvement. As part of the preparation for the class, a goal was to learn current practices and to become familiar with the processes and tools that were being used in industry. What was it like being a full-time software engineer in industry – what challenges do they face, which tools are being used, what is it like being a project manager of software engineering initiatives? How did the organization integrate process improvement initiatives into their business processes? What were some of the best practices to facilitate process improvements in organizations? What were some of the biggest obstacles to impede change? And, what were they doing to overcome these challenges?

A major goal was also to strengthen the relationship between my university and the organization. We could read the books, but wanted to know more, to improve understanding and to bring this knowledge into the classroom.

As part of the company's faculty partnership program, I was able to interview and observe the software engineers in their daily work. I was also able to discuss issues with software project managers. In addition to process improvement, I was interested in software quality and software testing. One of the highlights was that I spent several hours learning about usability testing from their usability specialist. An unexpected benefit was that he taped me in their usability lab performing a usability test on a web site, and I use the tape in my classes to illustrate the process of a formal usability test. The company also paid for me to attend several training classes to learn to use two vendors' software products.

As a result of the alliance, I was able to obtain the training I needed to prepare to offer the new course. In addition to the technical training I received, I obtained a better understanding of current business software process improvement and software testing challenges. I offered the new class the following term, with the class comprised of students from both the information systems in the college of business and from computer science in the college of engineering. They worked in teams and I made sure that two students were from information systems (business) and the other two students were from computer science (engineering). The students from both colleges had to work together as a team. This class proved to be invaluable to the students, and to me. In addition to the technical and business knowledge each student gained, the students learned to gain a new understanding and appreciation of each area (business and engineering), and learned to communicate in ways they did not do before.

The organization that provided the "faculty internship" benefitted by strengthening their alliance with my university and increasing their recruiting success through better student awareness of not only how their company operates, but process improvement and testing-related careers. Students then considered these areas as career choices, which they had not done in the past. Potential hires became more suited to the organization's needs, with students obtaining offers for summer internships and for full-time positions upon graduation.

An immediate benefit to the organization was that they now had potential employees already interested in them, and had a good understanding of their processes and expectations of that company. Plus, if I were using an example in class, why would I not use what I learned for that particular company? So, it turned out to be a good advertising and recruiting extra for their company.

Related benefits.

During my stay in industry, one of the managers contacted the chief operating officer at one of the leading providers of automated software testing tools, to ask them to donate software to my university. By doing so, everyone involved would benefit: the university would receive top-of-the-line testing tools to use in my course, the organization would be able to hire recruits that have experience with automated testing tools, and the provider of the automated testing tools would be getting their software in the hands of future software testing decision makers. Meanwhile, independently, the software company began to establish a formal program with additional universities, using my work as a model.

After working with this organization for about a year, they announced their new program, a program designed to provide software and training materials to assist academic institutions to develop their technical curricula. My university was the first institution to become part of the program and the first to receive their donation. In my class, I used their web applications testing software, their web load testing software, and their product that manages requirements, holds test plans, and tracks defects. Other companies have donated software (and hardware) to my university for us to use in our classes. The point is that many companies are kind enough to donate their products to colleges and universities for classroom use, but may not do so unless we ask. So, why don't we ask? We should.

3 Challenges and Suggestions

It was an enormous task, and it was just me doing all this. So, if software companies would like to donate software to a university, they must be prepared to provide extensive training and support. Otherwise, the program will probably fail. In universities where there are multiple faculty involved, it would improve the chances of success, but this will not always be possible. It might also be attractive to have faculty from other disciplines, for example business information systems and computer science.

To successfully build a course and incorporate the software into the curriculum, it takes a great deal of time. At least in my case, I did this in addition to my normal teaching load. Needless to say, I was very busy. So, what would also greatly help the probability of success, is for those organizations interested in working with their local professors, to provide funding to the university for release time. What this means, is that if the professor is teaching two or three classes per term, funding could be provided to the university to pay the professor‘s salary for one course, so that they will be teaching one less course. What I need most, is time. At a number of universities, if the economy is in a downturn, teaching loads (the number of courses a professor teaches per term) increase. So we may be teaching even more than usual, not less. Will this cost industry money? Absolutely. But, what is it worth to you to be able to locate qualified college graduates? Think of all the money industry spends on training, once employees are hired. Why not spend the money up front, as an investment in both your organization's future and the future of the emerging college graduates? (Depending on your country, there may even be tax advantages for donations…)

Another option is for industry to provide money for faculty positions. This would effectively pay a salary (or salaries) for a year. But this type of situation is not permanent, so if you are trying to attract new faculty, you may not be as successful as if you had permanent funding. Having said that, some faculty like to work in visiting positions for a year or two, perhaps taking leave from their university. A permanent option is to establish an Endowed Chair position. This position would be funded by a company, and is one of the more expensive options, but perhaps more successful. In such a case, an organization would donate an enormous amount of money, and the interest on that amount funds the position, on an annual basis.

If your university has a Master‘s Degree or Ph.D. program, then graduate students might be able to work with the faculty members on projects. Not only would such endeavors aid the success of the project, these projects could, in turn, become a masters or doctoral thesis. It might even help to attract grants to your program.

4 It Is Not Just Industry That Needs to be Agile

Agile development, process improvement, and testing methodologies are practiced around the world in organizations. But, it is not just industry that should be paying attention – academia should be involved, as well.

"All over the world, universities and colleges have been gradually rethinking how their organizations and infrastructures can be more agile. The thought is that if institutions are more flexible, they will be better able to support and promote entrepreneurial thinking — a long-term trend. At the University of Florida, the Innovation Academy acts as an incubator for students to plan and develop products and businesses, and even seek external funding." [NMC15]

Entrepreneurship

Additionally, an increasing trend in universities is focusing on Entrepreneurship, providing "incubators" and "hot-houses". These are areas where typically students involved in software engineering, can work (and sometimes live). They are not in a classroom, but are located buildings that are set up with significant technology resources, and advisors, to help them think through their ideas for creating products, and bringing them to market.

What a collaboration opportunity! Who could be involved? There are so many possibilities:

- college of engineering: developing the software
- college of business: marketing, accounting, finance, information systems
- industry: providing guidance, partnering, providing funding. Then, maybe hiring them, or buying out their product…

5 Additional Initiatives and Ideas

We now describe several additional examples of initiatives that the authors have been involved in, to help generate further collaboration ideas. While some of these real examples focus more on software testing than process improvement, the core ideas are there, to generate thought to help academia and industry strengthen their ties, for the benefit of both.

An ERP vendor provided their software, and a major consulting firm hosted it, and provided training materials for the college.

The Chief Financial Officer of a major multinational computer technology corporation was speaking with the Dean of the college, and asked how he could help the college. This company specializes in developing and marketing computer hardware systems and enterprise software products, particularly its own brands of database management systems. Currently, the faculty have been struggling with their individual classes, trying to develop an ERP system for their own course. Some faculty developed their own version of an ERP system, with limited success.

The company provides technical training, but not business process training. So, they partnered with a major consulting firm to help the college, and the consulting firm is not only hosting the site for the college, but working with the college to provide training materials for us to use in our classes. Over the Summer, faculty from all disciplines of the college will work together to establish a plan to have the key courses build on each other, not duplicate material, and expand and integrate the material for a progression of courses throughout the college. This not only takes a commitment from these two companies, but also commitment from the college. This does directly translate into a financial commitment from all involved. The ERP focus will allow students to truly understand how processes work and interrelate. Only then can they even think about improving them.

Leveraging academia for reviewing and enhancing competency development

The Quality Assurance and Testing business unit of a global Systems Integrator was being regularly challenged by its clients to showcase productivity improvements by designing and deploying solution accelerators, test harnesses and automation capabilities aligned to the client's need and environment.

While traditionally, the business unit was investing close to 5% of its revenues on the effort and productivity improvements were being 'seen at places', it was still grappling with finding a scientific way to establish a meta framework for competency development, that not only brought to clients 'state of the art' practices and tools, but also allowed the business unit to have a standardized, consistent approach.

A distinguished retired Professor from one of Asia's leading engineering schools was engaged on a retainer basis to study projects on the ground, map specific competency needs that would create value for clients, and then engage with the business unit's Project Managers and the relevant client's 'Single Point of Contact(s)' to establish more 'bespoke' frameworks. Through a detailed process of shadowing, the Professor was able to quickly perform time and motion studies, understand competency gaps resulting in delivery gaps of services and eventually ways to address and fix them. In the long run, this engagement not only enhanced the overall maturity of the business unit, but also helped it realize its full potential of capabilities.

An Independent Testing Organization's dilemma about bringing a 'Business Face' to testing

Having clocked phenomenal growth, specializing in the banking and financial services testing services, the firm was challenged with finding a way to build more 'Business Leadership' amongst its testers. This was all the more critical given that most key IT leadership roles (those who bought and consumed these services) amongst its customers, namely banks and insurance companies, were held by people who have had a formal business leadership role in the past.

What emerged was a focused MBA in Management of large QA organizations, with specific emphasis on banks and financial institutions.

Partnering with a leading business school, the Program was designed to be an ‘industry first’. Half of the graduates were employed by the independent testing organization, while the others were recruited by the competition firms in the industry.

In the words of one of the key leaders, this was a way of giving back to the industry. Also, doing so, enabled the organization to position itself as a ‘true Market leader’.

Joining hands with academia to get engineers to be industry ready.

A global start up that focusses in bridging the Knowing-Doing Gap amongst software testers, has started exploring and extensively collaborating with leading engineering and technical schools to ensure that these graduates are industry ready. In the past, once the people graduate, it would take 3-6 months before they could be considered productive. In this ‘new model’, that duration is expected to be shortened to 1 to 2 months. The collaboration includes design and launch of a common Test Lab, where students get assessed and get to work on real work like projects, after the assessment. They are also expected to shortly benefit from a curriculum that focusses on Quality Assurance and Testing as a major discipline, even at the undergraduate level.

6. Conclusions

Clearly, both industry and academia can benefit by working together more closely.

What better way is there to get to know the faculty at your local colleges and universities? What better way is there to forge alliances between industry and academia? What better way is there to help ensure that you can recruit students that will better meet your needs? There are probably a lot of professors that would like to work with you!

References

[NMC15] The NMC Horizon Report: 2015 Higher Education Edition.

http://www.nmc.org/publication/nmc-horizon-report-2015-higher-education-edition/

Process Education Training and Professionalism – Let's Bring Together Process Improvement Knowledge

Linda Ibrahim
Enterprise SPICE
Washington, DC 20016
rlibrahim@aol.com

Abstract

Reasons frequently given for process improvement failure include lack of training, education, awareness of the principles and value of process improvement, or how to do it. Whereas there is a growing body of knowledge regarding process improvement, this information is scattered and sometimes inconsistent. This paper examines issues, challenges and opportunities regarding process improvement education and training and recommends bringing together all stakeholders to develop a coherent body of knowledge to serve as a basis for process improvement education and training

1. Introduction

Reasons frequently given for process improvement failure include lack of training, education, awareness of the principles and value of process improvement, or how to do it. Whereas there is a growing body of knowledge regarding process improvement, this information is scattered and sometimes inconsistent. This paper examines issues, challenges and opportunities regarding process improvement education and training and recommends bringing together all stakeholders to develop a coherent body of knowledge to serve as a first step, as a basis for process improvement education and training.

It is hoped these thoughts can help us develop a strategy in pursuit of a common vision regarding process education, training and professionalism.

Proceedings of the International Workshop on Software Process Education, Training and Professionalism, Gothenburg, Sweden

20015-06-15 published at http://ceur-ws.org

2. Background

Process improvement experts, consultants and practitioners have accumulated enormous experience and practical knowledge of what needs to be done for success in process improvement. See for example [Das13], [Ibr08], [ISO13], [Nia15] and [SPI10].

The IEEE Computer Society developed a Guide to the Software Engineering Body of Knowledge (SWEBOK) [IEE04] which includes one knowledge area focusing on Software Engineering Process.

Most major process models and standards include a process (or process area or clause or set of practices) pertaining to process improvement / process definition / process assessment / continual improvement as a (brief) part of the document.

There are curriculum guidelines for undergraduate and graduate software engineering programs (e.g. [ACM14] and [Pys09]) which address the entire software engineering discipline and include some information on processes, but do not focus specifically on process improvement education.

The Software Engineering Institute embarked on a major initiative to describe the subject matter of process improvement. The subject matter is intended for use in academic, industrial and governmental settings. See [Ibr95].

Professional organizations and professional certifiers have process improvement knowledge and information that they use in their certification programs. Some are standards based (such as SPICE training and certification) or best practice based (such as ITIL training and certification) or model-based (such as CMMI training) or methodology based (such as Six-Sigma or Lean Six-Sigma certification programs). See e.g. [Ent11], [ITI11], [ISO11], [CMM15] and [ASQ15].

In-house process improvement training and education might typically include courses for executives, practitioners and assessors, often grounded in the domain and culture of the organization and offered by in-house staff. See e.g. [FAA06].

A variety of mechanisms are in use for delivering process improvement education and training: such as on-line education and training from various universities and colleges and institutes; on-site training; centralized classes in various locales, Massive Open Online Course (MOOC). These efforts are helping but we still have shortages of educated and trained professionals. See [Den15].

Despite all these activities there are difficulties that our customers have in sorting through the various educational and training opportunities, and attending classes.

And the vast process improvement knowledge we have is scattered. It needs to be consolidated, integrated and structured.

A first step to meet this challenge is to bring all this wisdom into an internationally recognized process improvement (PI) body of knowledge (BoK) that is endorsed and used for process improvement education, training and certification. The BoK would articulate what process improvement professionals should know and serve as a basis for skills development and continual learning. The time is clearly ripe.

3. Challenges

Several things need to come about to be successful in this effort, each with challenges.

- **Standards** - to be developed for content of process improvement education and training. But will BoK developers and owners of PI knowledge recognize the value of consolidation and integration and collaborate on its development? (Our customers may insist on this.)
- **Usage -** of the BoK**.** Will educators and trainers use the BoK? Will degrees and certifications be based on the BoK?
- **Organizations** - to value process and support process improvement training and education. Are our leaders ready to drive improvement across the enterprise? Is the organizational culture ready? Do we have strategic-minded executives to lead and support the path to improved performance?
- **Practitioners -** to have appropriate skills and competencies to help organizations. Will they be equipped to use process improvement standards and models and best practices? Will they be able to tailor PI information to the business needs of their customers? Will they foster implementation of basic principles?

4. The Current Environment

Below are some SWOT observations on our current situation. *(Hopefully we can build further on such an analysis at our Workshop to help us develop a strategy for moving forward.)*

4.1 Strengths

What are our strengths? (*We need to maintain, build on and leverage these.)*

- Extensive community of people working in process improvement education and training and professional certification – including Universities, Colleges, Professional Societies, Institutes
- Courses offered using various delivery mechanisms e.g. on-line, instructor-led, in-house, off-site, options for self-study
- Process improvement knowledge captured via various initiatives

4.2 Weaknesses

What are our weaknesses? *(We need to remedy, change, stop and overcome these.)*

- Dwindling, sporadic interest in process improvement in industry and government
- Confusion in terminology regarding training and certification offerings e.g. Business Process Management (BPM), Quality Training, BPI (I=Innovation or Improvement), black belt, 6-sigma, Lean 6-sigma, ITIL, Business Process Re-engineering, TQM, etc.

- Confusion regarding which process improvement approach might help the most e.g., Model-based, SPICE, Six-sigma, Lean, IDEAL, black belt, Lean Six-Sigma, ITIL
- Insufficient attention to process in university courses to ground the fundamentals
- Lack of standardization regarding process education and training content – similar topics, overlap, inconsistency, various bodies of knowledge
- Customer confusion … do I want/need model-based training, SPICE, CMMI, Six-sigma, Lean, IDEAL, black belt, ITIL, … and who is best qualified to offer this training …
- Is professional certification really needed or important? To whom?
- Stove-piped professional courses not recognizing the needs to integrate PI approaches, or the value obtained from various approaches
- Training and education too expensive, too time-consuming

4.3 Opportunities

What are our opportunities? *(We need to prioritize, capture, build on and optimize these.)*

- Clarification and standardization of subject matter, body of knowledge for process improvement
- Internationally recognized common content and authorization for professional certifications
- Curriculum guidelines for process improvement education and training
- Undergraduate capstone projects as well as graduate projects in industry
- Collection and publication of data on availability and effectiveness of education, training and professional programs
- Reduce training costs, distance learning, Massive Open Online Course (MOOC)
- Work to ensure executives and decision-makers understand the value of process improvement to address dwindling interest in process improvement
- Bridge the gap between education and training
- We need to provide education, training and guidance to our customers based on the accumulating codified wealth of process knowledge and information available

4.4 Threats

What are our threats? *(We need to counter, minimize or manage these.)*

- Lack of buy-in from customers regarding the need for process improvement and hence education and training
- Lack of cooperation and buy-in from education and training institutions to work together to improve the quality and available of process improvement education and training
- Dwindling, sporadic interest in process improvement in industry and government
- Competing training organizations

5. Issues and Concerns

5.1 Issues

Through my experiences in process improvement, I find recurring issues such as:

- A certification may help a practitioner get a job, but the employer may not be interested in using the skills acquired by the practitioner. Do practitioners have the skills and competencies they need? If they do, do they have the chance to use them?
- Are professional certifications providing needed competencies, and offering recognized subject matter? Are certifications sought just to check the box, and don't really address customer needs?
- The organization may not be ready or interested in process improvement.
- Our executives need to know process improvement principles and concepts to lead us effectively. There is a need to educate

- executives and decision-makers on the value of process improvement. How can we reach leaders and influence them to effect process improvement?
- Is it clear to the customer what education and training will really help them in their process improvement quests? What training should I invest in? The market is too confusing.
- Training is too expensive and too time consuming.
- Do education and training endeavors address essential process improvement concepts and principles in the field?

What can we do to help our customers and professionals rectify these situations?

5.2 Concerns

There are also several concerns I have regarding our efforts in process improvement education and training, such as:

- **Lack of integration**: I worry that separate process improvement approaches are taught and followed diligently without realizing the value of each and how they can and need to work together for optimal customer benefit. Practitioners need to recognize what is gained from focusing on model-based best practices, what quality systems offer, what measurement focused problem-solving techniques offer, and how a recognition of all of these will help the organization. How they interact. Black-belts can find issues in a process but are not inclined to improve the process using best practices … just find what's broken in the existing process, but the process itself may not recognize or use the best practices available in models and standards. Professional certifications are typically stove-piped on a particular method or model that may not reap the true benefits of process improvement when pursued in isolation.
- **Standards:** We don't want to have the model-wars that have gone on in the process model endeavors. Will this happen in the process improvement education arena if we decide to develop a standard body of knowledge for PI? Will the stakeholders collaborate?
- **Organizations**: Process improvement is not yet fully ingrained in many organizational cultures. We need to reach enterprise executives, via education and training, so we can help organizations improve their performance via strong leadership and strategic vision tied to process improvement. Social and cultural changes are needed to bring about the full benefits of process improvement.

6. Body of Knowledge – Previous Efforts

6.1 Software Engineering Body of Knowledge

The Guide to the Software Engineering Body of Knowledge (SWEBOK) [IEE04] broadly addresses ten knowledge areas (KA) describing the discipline of software engineering. One of these KAs is called Software Engineering Process, which includes some topics relevant to our process improvement workshop. This KA is structured into 4 topics: Process Implementation and Change; Process Definition; Process Assessment; and Process and Product Measurement. Each topic is broken down into 2 to 5 subtopics with a brief description and references for each.

6.2 Software Quality Engineer Body of Knowledge

The Software Quality Engineer Body of Knowledge [ASQ08] includes seven parts: General Knowledge; Software Quality Management; Systems and Software Engineering Processes; Project Management; Software Metrics and Analysis; Software Verification and Validation; and Software Configuration Management. This BoK is used to certify quality engineers.

6.3 The Subject Matter of Process Improvement

The need for process improvement education and training has been recognized for some time. To address this need I led an initiative while working at the Software Engineering Institute of Carnegie Mellon University focused on developing a description of the subject matter of process improvement. The purpose of this work was to assist software engineering educators and trainers in selecting topics for curricula or training programs in the process improvement arena.

Data were collected from a variety of sources including courses, workshops, tutorials and documents relating to various aspects of process improvement; selected literature including published standards, certification and professional society publications; customer views, including experiences, viewpoints and documents provided by change agents, educators and trainers in industry, government and academia. Several surveys and focus group sessions were carried out to gather subject matter content as well as issues relating to education and training. Approximately 100 professionals participated in the initiative.

The subject matter is presented in a framework describing:

- what you need to *Know* (Process Fundamentals, Process Improvement Fundamentals);
- what you need to *Do* (Process and Process Improvement Management, Culture Change);
- what you need to *Use* (Tools and Techniques, Pervasive Supporting Skills).

Each topic area contains annotated subtopics with references. In addition the report aligns the subject matter with general audiences across academic and industry/government domains, proposing the extent of mastery that might be required for proficiency.

For further information see [Ibr95].

6.4 Practitioner Knowledge Collection

Throughout the years, process improvement practitioners have built up a broad body of knowledge regarding process improvement. Some examples are provided below. *(Note that several journals (e.g. [ASQ15]) regularly offer articles on process improvement experiences but this paper does not intend to bring together all these sources of information.)*

6.4.1 SPI Manifesto

The SPI Manifesto [SPI10], developed by a group of international SPI experts, provides a wealth of SPI knowledge and experience. The Manifesto brings together three core SPI values, 10 principles supporting the values, with examples for each principle.

6.4.2 Success Factors

Another example is the following set of known factors that are deemed critical for successful process improvement, consolidated from numerous publications and sources (extracted from [Ibr08]).

- Support, commitment and involvement
  e.g., visible support and sustained commitment from senior management; middle management support and commitment; grass roots support and involvement; technical staff involvement
- Showing measurable, observable results
  e.g., observable results backed with data to sustain interest and motivation; process improvement measured, results made visible
- Process improvement management
  e.g., effort must be planned, managed; senior management actively monitors progress; adequate staff time/resources dedicated; clear assignment of responsibility; process group staffed by highly respected people; risks recognized and mitigated as necessary
- Goals and alignment
  e.g., clearly stated, communicated, well understood, appropriate process improvement goals aligned with the business; shared values and goals, improvement in everyone’s performance plan; sustained focus and follow through; no constant shifting of priorities
- Knowledge
  e.g., having ability, skills, knowledge; sufficient education about process and process improvement; for managers, learn enough to manage it and to have confidence in methods used
- Culture
  e.g., open communication; teamwork; mutual trust; respect for the individual; investment in people; quality orientation, customer focus; continuous learning; NOT: belief that PI gets in the way of real work; NOT: cynicism from

previous unsuccessful PI efforts

7. Conclusions and Recommendations

This paper recommends working together to develop a process improvement body of knowledge for use in education and training. It has brought together some thoughts and issues and challenges we face in process improvement education, training and professionalism. It provides some views on our current environment, and summarizes some previous initiatives. The paper calls for bringing together process improvement knowledge into an internationally recognized and endorsed standard.

We need to build on and improve previous initiatives; continue to integrate subject matter content from broad international sources; engage stakeholders to pilot the subject matter content in various venues and report experiences and lessons learned.

Such an effort might be part of a strategy developed by the broad community of process improvement professionals concerned about improving process education, training and professionalism.

References

[ACM14] Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, ACM/IEEE, 2014.

[ASQ08] Body of Knowledge-Software Quality Engineer, American Society for Quality, 2008, www.asq.org

[ASQ15] Software Quality Professional, American Society for Quality, www.asq.org

[CMM15] CMMI Institute, www.cmmiinstitute.com

[Das13] *Guide to Applying the Enterprise SPICE Model*, W. Daschner, L. Ibrahim, W. Henschelchen and E. Wallmueller, May 2013, Enterprise SPICE, SPICE Users Group – available at www.enterprisespice.com

[Den15] *The Profession of IT - A Technician Shortage*, P.J. Denning and E. E. Gordon, Communications of the ACM, March 2015, Vol 58, No 3, pp. 28-30.

[Ent11] Enterprise SPICE Practitioner Training – Syllabus, Enterprise SPICE Project, 2011.

[FAA06] Federal Aviation Administration Integrated Capability Maturity Model (FAA-iCMM) Appraisal Method, v2, (Appendix: Lead Appraiser Process), FAA, September 2006.

[Ibr08] *A Process Improvement Commentary*, L. Ibrahim, Crosstalk, August 2008, pp. 26-29

[Ibr95] The Subject Matter of Process Improvement: A Topic and Reference Source for Software Engineering Educators and Trainers, R. L. Ibrahim and I. Hirmanpour, Technical Report CMU/SEI-95-TR-003, ESC-TR-95-003, available at http://www.sei.cmu.edu/reports/95tr003.pdf

[IEE04] Guide to the Software Engineering Body of Knowledge (SWEBOK), IEEE Computer Society, 2004.

[ISO11] ISO 13053:2011 Quantitative methods in process improvement – Six Sigma, 2011.

[ISO13] ISO/IEC TR 33014:2013 Information technology –Process Assessment – Guide for process improvement, 2013.

[ITI11] ITIL (IT Infrastructure Library), 2011, www.ITIL.org.uk

[Nia15] *A Comparative Study of Software Process Improvement Implementation Success Factors*, M. Niazi, Journal of Software: Evolution and Process, February, 2015.

[Pys09] Graduate Software Engineering 2009 (GSwE2009): Curriculum Guidelines for Graduate Degree Programs in Software Engineering, A. Pyster (ed), Stevens Institute of Technology, 2009.

[SPI10] SPI Manifesto, 2010.