EuroSPI² 2012 Proceedings

Proceedings

The papers in this book comprise the industrial proceedings of the EuroSPI² 2012 conference. They reflect the authors' opinions and, in the interests of timely dissemination, are published as presented and without change. Their inclusion in this publication does not necessarily constitute endorsement by EuroSPI² and the publisher.

EuroSPI²

EuroSPI² is a partnership of large Scandinavian research companies and experience networks (SINTEF, DELTA, STTF), iSQI as a large German quality association, the American Society for Quality, and ISCN as the co-ordinating partner.

The EuroSPI² conference presents and discusses results from **systems, software and services process improvement and innovation (SPI)** projects in industry and research, focusing on the gained benefits and the criteria for success. This year's event is the 19th of a series of conferences to which international researchers and professionals contribute their lessons learned and share their knowledge as they work towards the next higher level of software management professionalism.

Since 2009 we have extended the scope of the conference from software process improvement to systems, software and service based process improvement.

The International Network for Terminology (TermNet) and the Vienna University of Technology, Austria are the hosts of the EuroSPI² 2012 conference. TermNet represents a world-wide network and the Vienna University of Technology maintains a network of world-wide research partnerships. EuroSPI² focuses on creating Europe-wide networks for innovation and improvement and thus we are proud of co-organizing this year's event in Vienna, Austria.

EuroSPI² Chairs

General & Workshop Chair	Richard Messnarz, ISCN, Austria/Ireland
EuroSPI² Marketing Chair	Miklós Biró, Software Competence Center Hagenberg (SCCH), Austria
Scientific Programme Chair	Rory O' Connor, Dublin City University, Ireland
Scientific Programme & Local Chair	Dietmar Winkler, Vienna University of Technology, Austria
Industrial Programme Chair	Jørn Johansen, DELTA, Denmark
Industrial Programme Chair	Risto Nevalainen FiSMA and STTF, Finland
Industrial Programme Chair	Stephan Goericke, iSQI, Germany
Industrial Programme Chair	Mads Christiansen, DELTA, Denmark
Industrial Programme Chair	Nils Brede Moe, SINTEF, Norway
Organizing Chair	Adrienne Clarke, ISCN, Ireland
Co-Organizing Chair	Sonja Koinig, ISCN Austria
Local Organizer	Gabriele Sauberer, TermNet, Austria

Industrial Programme Committee

Bachmann Volker	SIBAC GmbH	Germany
Bæk Jørgensen Jens	Mjølner Informatics A/S	Denmark
Breske Eva	Robert Bosch GmbH	Germany
Christiansen Mads	DELTA	Denmark
Daughtrey Taz	James Madison University	USA
Dekkers Carol	Quality Plus Technologies	USA
Dussa-Zieger Klaudia	Method Park Software AG	Germany
Ekert Damjan	ISCN GmbH	Austria
Fehrer Detlef	SICK AG	Germany
Fenz Andrea	Skills International GmbH	Austria
Forselius Pekka	4SUM Partners	Finland
Georicke Stephan	iSQI GmbH	Germany
Hagenmeyer Philipp	ZF Friedrichshafen AG	Germany
Hallikas Jarmo	Falcon Leader Oy	Finland
Hällmayer Frank	Software Factory GmbH	Germany
Hind Tim	AXA	UK
İbrişim Ayşegül	Turkish Standards Institution	Turkey
Johansen Jørn	DELTA Axiom	Denmark
Johansson Mika	FiSMA	Finland
König Frank	ZF Friedrichshafen AG	Germany
Messnarz Richard	ISCN Ltd	Austria/Ireland
Morgenstern Jens	-	Germany
Nevalainen Risto	FiSMA	Finland
O'Leary Eugene	EQN Ltd.	Ireland
Poth Alexander	SQS Software Quality Systems AG	Germany
Renault Samuel	Centre de Recherche Public Henri Tudor	Luxembourg
Romcea Cristina	Conti Temic microelectronic GmbH	Germany
Sauberer Gabriele	TermNet	Austria
Schweigert Tomas	SQS Software Quality Systems AG	Germany
Sechser Bernhard	Method Park	Germany
Spork Gunther	Magna Powertrain	Austria
Stefanova-Pavlova Maria	Center for Innovation and Technology Transfer-Global	Bulgaria
Von Bronk Peter	Systemberatung Software-Qualität	Germany
Wegner Thomas	ZF Sachs AG	Germany

EuroSPI² Board Members

ASQ, http://www.asq.org

DELTA, http://www.delta.dk

FiSMA, http://www.fisma.fi

ISCN, http://www.iscn.com

iSQI, http://www.isqi.de

SINTEF, http://www.sintef.no

Editors of Proceedings

Richard Messnarz, ISCN, Austria/Ireland

Damjan Ekert, ISCN, Austria

Mads Christiansen, DELTA, Denmark

Jørn Johansen, DELTA, Denmark

Sonja Koinig, ISCN, Austria

Welcome Address by the EuroSPI² General Chair

Richard Messnarz

ISCN, Austria/Ireland

EuroSPI² is an initiative with 5 major goals (www.eurospi.net):

1. An annual EuroSPI² conference supported by Software Process Improvement Networks from different European countries.

2. EuroSPI² supported the establishment of a world-wide SPI Manifesto (SPI = Systems, Software and Services Process Improvement) with SPI values and principles agreed among experts world-wide. We build clusters of experts and knowledge libraries for these values and principles.

3. Establishing a web-based experience library based on hundreds of experience reports contributed to EuroSPI² since 1994 and which is continuously extended over the years and is made available to conference attendees.

4. Establishing a European Qualification Framework for a pool of professions related with SPI and management. This is supported by Europe-wide certification for qualifications in the SPI area, exam systems, and online training platforms (European Certification and Qualification Association, www.ecqa.org).

5. Establishing a world-wide newsletter with articles from key industry and European key research associations helping to implement the SPI manifesto world-wide (newsletter.eurospi.net).

EuroSPI² is a partnership of large Scandinavian research companies and experience networks (SINTEF, DELTA, STTF), the iSQI as a large German quality association, the American Society for Quality, and ISCN as the co-coordinating partner. EuroSPI² collaborates with a large number of SPINs (Software Process Improvement Network) in Europe.

EuroSPI^2 conferences present and discuss results from **systems, software and services process improvement (SPI)** projects in industry and research, focussing on the benefits gained and the criteria for success. This year's event is the 19th of a series of conferences to which international researchers contribute their lessons learned and share their knowledge as they work towards the next higher level of software management professionalism.

A typical characterization of EuroSPI² was stated by a company using the following words:

> *"... the biggest value of EuroSPI² lies in its function as a European knowledge and experience exchange mechanism for SPI and innovation".*

A cluster of European projects (supporting ECQA and EuroSPI²) contribute knowledge to the initiative, including currently SafEUr (ECQA Certified Safety Manager), SIMS (ECQA Certified Social Media Expert), VALO (ECQA Certified Valorisation Manager), BPM (ECQA Certified Business Process Manager), BPM- HEI (BPM for Higher Education), GOSPEL (ECQA Certified Trusted Businesses and Governance Control Assessor), mLeMan (ECQA Certified Mobile Learning Manager). A pool of more than 30 qualifications has been set up (see www.ecqa.org).

Join the community of cross-company learning of good practices!

Contact: Richard Messnarz, ISCN, Austria/Ireland, e-mail: rmess@iscn.com

Welcome by DELTA, Editors of the DELTA Improvement Series

Jørn Johansen

DELTA, Denmark

DELTA has been working with Software Process Improvement (SPI) for more than 16 years including maturity assessment according to BOOTSTRAP, SPICE and CMMI. DELTA has also been a partner in the EuroSPI² conference from the very beginning 16 years ago. We are now for the 4th time the publisher of the Industrial Proceedings from EuroSPI² making it part of the DELTA series about Process Improvement.

Jørn Johansen is Senior Technology Specialist of at DELTA. He has an M.Sc.E.E. from Ålborg University and more than 32 years experience in IT. He has worked in a Danish company with embedded and application software as a Developer and Project Manager for 15 years. Mr. Johansen has been involved in all aspects of software development: specification, analysis, design, coding, and quality assurance. Furthermore he has been involved in the company's implementation of an ISO 9001 Quality System and was educated to and functioned as Internal Auditor.

For the last 17 years he has worked at DELTA as a consultant and registered BOOTSTRAP, ISO 15504 Lead Assessor, CMMI Assessor and *ImprovAbility™* Assessor. He has participated in more than 100 assessments in Denmark and abroad for companies of all sizes. He was the Project Manager in the Danish Centre for Software Process Improvement project, a more than 25 person-year SPI project and Talent@IT, a 26 person-year project that involves 4 companies as well as the IT University in Copenhagen and DELTA. Latest Mr. Johansen was the Project Manager of SourceIT an 18 person-year project focusing on outsourcing and maturity. Mr. Johansen is also the co-ordinator of a Danish knowledge exchange group: Improving the Software Development Process, which is the Danish SPIN-group. At the moment Mr. Johansen is lead editor on ISO/IEC 33014 Guide for process improvement.

Contact: Jørn Johansen, DELTA, Denmark, e-mail: joj@delta.dk

Mads Christiansen

DELTA, Denmark

Mads Christiansen has an M.Sc.E.E. from DTU (Danish Technical University) and more than 32 years experience in product development and IT. He has worked for 19 years in a Danish company with embedded and application software as a Developer and Project Manager. Mr. Christiansen has been involved in all aspects of software development: specification, analysis, design, coding, and quality assurance and managing outsourced projects in Denmark and USA.

For the last 14 years he has worked at DELTA as a consultant in SPI (requirements specification, test, design of usable products and development models). Currently Mr. Christiansen works with eBusiness and as Innovation Agent. Mr. Christiansen is also *ImprovAbility™* Assessor and Trainer of *ImprovAbility™* project Assessors.

Contact: Mads Christiansen, DELTA, Denmark, e-mail: mc@delta.dk

Welcome from the Local Organization and Scientific Programme Committee Chair in Austria

Stefan Biffl

Vienna University of Technology, Austria

Dietmar Winkler

Vienna University of Technology, Austria

Welcome to the 19th EuroSPI[2] Conference in Vienna and at the Technical University Vienna (http://www.tuwien.ac.at). The TU Vienna looks back on a long tradition at the leading edge of scientific research and education.

Today the TU Vienna has eight faculties: Architecture and Regional Planning, Technical Chemistry, Civil Engineering, Computer Sciences, Electrical Engineering and Information Technology, Mathematics and Geoinformation, Mechanical Engineering and Business Science, and Physics.
The Institute of Software Technology and Interactive Systems (http://www.isis.tuwien.ac.at) includes four research groups: **Information and Software Engineering**, Electronic and Commerce, Business Informatics, and Interactive Media Systems.

Our team Quality Software Engineering (http://qse.ifs.tuwien.ac.at), embedded within the Information and Software Engineering research group, focuses on research and teaching in Software Engineering, Software Product and Process Improvement, Empirical Software Engineering, Quality Assurance and Quality Management. In 2010 we established a **Christian Doppler Laboratory „Software Engineering Integration for Flexible Automation Systems"** (CDL-Flex) with focus on integrating heterogeneous engineering environments in the automation systems domain with respect to engineering process improvement (http://cdl.ifs.tuwien.ac.at).
Based on our experience in research, education, and consulting in the area of software process improvement and our long tradition with the EuroSPI[2] conference we are proud to act as local organization for EuroSPI[2] in 2012.

Stefan Biffl is an associate professor of software engineering at the Institute of Software Technology and Interactive Systems. He received MS and PhD degrees in computer science from TU Vienna and an MS degree in social and economic sciences from the University of Vienna in 2001. He received an Erwin-Schrödinger research scholarship and spent one year as researcher at the Fraunhofer IESE, focusing on quality management and empirical software engineering. Also, in 2001 he received the Habilitation degree Venia Docendi for his work on empirical software engineering in project management and quality management. In 2006 he worked as guest researcher at Czech Technical University, Department of Cybernetics. Since 2010 Stefan Biffl is the head of the Christian Doppler research laboratory "CDL-Flex".

Dietmar Winkler received an MS in computer science from TU Vienna, Austria, in 2003. He worked as a guest researcher at the Czech Technical University, Department of Cybernetics in 2007 and received a PhD research scholarship at Fraunhofer IESE in Kaiserslautern, Germany, in 2008. Since 2010 he is key researcher for quality management and software process improvement in the CDL-Flex. Moreover, he works as software engineering and process management consultant. His research interests include Software Engineering, Engineering Processes and Process Improvement, Quality Management, and Empirical Software Engineering.

We are happy to welcome an impressive group of international experts in software process improvement at TU Vienna and wish readers and participants interesting presentations, successful networking, and an unforgettable EuroSPI[2] 2012 Conference in Vienna.

Contact Details:
Stefan Biffl (E-Mail: stefan.biffl@tuwien.ac.at)
Dietmar Winkler (E-Mail: dietmar.winkler@tuwien.ac.at)

Welcome from the Local Organization Chair in Austria

Gabriele Sauberer

TermNet, Austria

TermNet, the International Network for Terminology, provides an international co-operation forum for organizations who want to benefit from the knowledge and expertise of a global network.

It was a natural step for us to join forces with the EuroSPI[2] community, whose aim is software process improvement and innovation – quite a similar goal as what we want to achieve through advocating terminology management.

Quality through terminology - automotive industry standard to check quality of translated manuals and product descriptions

(www.termnet-gmbh.at, www.lics-certification.org/tsp/translation_quality)

Quality control and evaluation of technical documentation and translations is a key issue, especially with regard to product liability. Product documentation like manuals and their translations need to be checked for consistent use of terminology. The more target markets and languages served, the more complex it becomes to keep control of all the document versions. The quality of the translation with regard to different client specifications can be assessed through an industry standard: SAE-J2450 - Translation Quality Metric.

TermNet, as a founding partner of the Language Industry Certification System (www.lics-certification.org), is actively involved in certification against this and other relevant standards. TermNet Business Ltd, the commercial part of the TermNet family, has made this service one of its focal businesses. One of our largest target markets is the automotive and aviation industry and we have been collaborating with prominent manufacturers.

Quality through ECQA Certified Terminology Manager - Basic (www.termnet.org/english/products_service/ecqa_ctm-basic)

Through ECQA our collaboration with members of the EuroSPI[2] community started. TermNet developed the ECQA Certified Terminology Manager - Basic in 2010. Training for terminology managers was not a new field for us. We were pretty much the only independent organization with the expertise and capacity to provide for general, basic, open training courses for those who already do – or are "forced" to do - the job of a terminology manager. Demand for training, qualification and certification was and is constantly growing – especially now that ECQA certified Terminology Manager Advanced, Automotive, Health etc. is coming soon.

I wish the readers and participants interesting presentations and fruitful networking during EuroSPI[2] 2012.

Contact: termnet@termnet.org

Welcome from the ECQA President

Michael Reiner

ECQA, Austria

The European Certification and Qualification Association (ECQA) is a not-for-profit association that aims to unify the certification processes for various professions in Europe. It is joining together institutions and thousands of professionals from all over Europe as well as from abroad and offers the certification to participants for numerous professions. Currently, 27 professions are active and some new professions are being developed right now. ECQA services are being offered in 24 countries across Europe by 60 ECQA members. With the help of Ambassadors the ECQA is also enhancing its activities by expanding to all over the world (e.g. USA, Thailand, India, Singapore etc.).

The main objective of the ECQA is to develop and maintain a set of quality criteria and common certification rules across the different European regions. Therefore the ECQA ensures that the same knowledge is presented to participants across Europe and all participants are tested according to the same requirements. The knowledge to be provided and tested for certain professions is defined by experts from industry and research, who know best what the requirements of the market are and what the state of the art knowledge is within certain domains. These experts work in ECQA groups called Job Role Committees. The EQCA coordinates their work and provides the infrastructure and IT support.

The ECQA has developed a set of quality criteria, which are used for the certification of the following types of service providers: trainers, training organizations, exam organizations, and certification organizations. The aim is to ensure the same level of training and certification quality in all participating countries.

Michael Reiner, president of the ECQA and lecturer for Business Administration and E-Business Management at the IMC University of Applied Sciences Krems, has several years of experience in the field of IT, Microsoft Office, Microsoft NAV (ERP), Knowledge Management, Business Intelligence, Web 2.0 and social networks. Moreover Mr. Reiner is member of the Microsoft Dynamics Academics Advisory Board and coordinates and participates in various EU projects.

I wish you a good time at the EuroSPI² 2012, a lot of interesting networking partners and informatory meetings.

Contact: Michael Reiner, IMC University of Applied Sciences Krems, Austria, e-mail: michael.reiner@fh-krems.ac.at

Welcome from the Quality Management Center of the German Association of Automotive Industry (VDA QMC)

Jan Morenzin

VDA QMC, Germany

VDA-QMC is the responsible assessor certification body for Automotive SPICE®, verifying knowledge and experience as crucial qualifications. A training and certification provided by VDA-QMC is a core qualification for software quality professionals in automotive industry

The spectrum of VDA QMC ranges from developing systems and methods to shaping the future of quality management systems in the automotive industry. These developments, as well as the direction of QMC, are steered by the top-level committee regarding quality: the VDA QM Commission.

Dr. Jan Morenzin is working as a professional for development, functional safety management and quality assurance in automotive embedded computing. He is responsible for the software quality initiatives and the personnel certification activities in VDA QMC. He is a member of the VDA QMC WG 13 and the ISO/IEC JTC1/SC7.

Please contact us for the latest information on Software Process Improvements activities in the German Association of the Automotive Industry.
Meet your personnel certification body and learn about how to become an intacsTM Automotive SPICE® assessor.

Automotive SPICE® is a registered trademark of Verband der Automobilindustrie e.V (VDA).

Table of Contents

Experience Session 4: SPI & Agile 1

Experience Session 5: SPI & Agile 2

Experience Session 6: SPI & Functional Safety

Experience Session 7: SPI & Stakeholder Improvement

Global software projects:
Work package allocation factors

Marcos Ruano-Mayoral, Ricardo Colomo-Palacios, Ángel García-Crespo & Sanjay Misra

Abstract

This paper presents a discussion on the factors affecting task and work package allocation in Global Software Development (GSD) settings. The identification of the factors was achieved by a systematic review of relevant literature prior to a validation by a group of 11 experts using the nominal group technique. The study resulted with a list of factors ordered by relevance in which cost, cultural distance and project duration were in the top. Additionally, the study has shown that the intrinsic relationships among project work packages along with geographical, temporal, cultural and structural factor make the management of GSD projects an extremely complex activity.

Keywords

Software Development, Work package allocation, Metrics, Measurement

1 Introduction

Offshoring outsourcing is the practice of distributing work, particularly in the area of information technology services and development to workers outside the national borders of the host country (Niederman, Kundu & Salas, 2006). Research indicates that offshoring can create wealth for both the countries and companies involved (Farrell & Agrawal, 2003) but in the other hand, the debate about the possible impact of offshoring services on developed country growth rates, wages, and industrial structure is open (Dossani & Kenney, 2007). In any case, the offshoring of software development is a significant global economic phenomenon (Niederman, Kundu & Salas, 2006). But this phenomenon is bound to have a profound effect on the field: not only the obvious human effect, but also a long-lasting influence on the technology itself (Meyer, 2006). The growth of software development offshoring as a strategic option for firms has transformed what was traditionally an internal activity to one driven by external vendors (Carmel & Agarwal, 2002). In the early 1990s, offshoring of software work to development centers in low wage countries pertained to large Western companies such as IBM and SAP who systematically attempted to take a hold of wage differences and resources of a global market (Winkler, Dibbern & Heinzl, 2008). Now, many Fortune 500 companies produce their business information systems in developing countries (such as China and India) to take advantage of their relatively low-cost labor (Sakthivel, 2007) and large telecommunications and software companies have numerous software development groups around the world (Edwards & Sridhar, 2005). Thus, it can be said that software development outsourcing is an integral part of software development projects (Schümmer & Lukosch, 2009). Given this unstoppable trend, there is a clear need to better understand how to manage offshore projects more effectively (Iacovou & Nakatsu, 2008).

Global Software Development (GSD) teams are geographically distributed teams which make use of collaborative technologies to produce software (Herbsleb & Moitra, 2001). These teams can be considered as a specification of virtual teams (Martins, Gilson & Maynard, 2004) and their creation is encouraged by the relationships between customers of software development outsourcing organizations and developers (Heeks et al., 2001).

The adoption of GSD means that software engineers should collaborate over geographic, temporal, cultural and linguistic distance; these characteristics are usually termed as "global distance" (Noll, Beecham & Richardson, 2010). In this scenario, Milewski et al. (2008) stated that GSD is a paradox, while several researchers and practitioners state that some GSD teams are highly productive, others affirm that GSD teams perform sub-optimally. This paradox could be rooted on the global distance and the difficulty in managing such kind of projects.

According to García-Crespo et al. (2010), one of the hard decisions present in the management of GSD projects is Work packages assignation. In this work, authors stated that aspects such as software and task dependencies, the need to preserve core competency in one of the companies, trust and lack of knowledge about real competences from partners are issues that managers must manage to face these decisions. These are some of the aspects present in work-package allocation, but, taking into account the intrinsic complexity of the decision process, this list is just introductory. Thus, this paper presents main factors present in the work-package allocation decision making process. These factors have been derived from a study of the literature as well as a qualitative study.

2 Task and work package allocation

Task allocation in GSD environments has its precedent in manufacturing systems. Not in vain, in the last decade of the 20th century, the role of manufacturing shifted from a producer of goods and services to one that co-ordinates the whole industry value chain (Choy & Lee, 2003). Thus, since that decade, literature has produced a number of publications related to the strategy formulation, evaluation and conceptual frameworks of Global Manufacturing systems (Lee & Lau, 1999). In this scenario, the work of Shi, Gregory and Nailor (1997) proposes a practical self-assessment tool known as the International Manufacturing Configuration Map (IMCM) for supporting a global manufacturing strategy formulation. Adopting, a benchmarking approach, IMCM can be used to analyze international manu-

facturing networks. There are many tools derived from this seminal work present in the literature (e.g. Choy & Lee, 2003; Lam, Kwok & Lee, 2008; Lee & Lau, 1999, Tso, Lau & Ho, 2000).

In this same path, the concept of "Virtual enterprise" emerged. The term "virtual enterprise" has been used in articulating the strategy for the 21st century global manufacturing enterprises (Park & Favrel, 1999). A virtual enterprise is a temporary consortium of autonomous, diverse, and possibly geographically dispersed organizations that pool their resources to meet short-term objectives and exploit fast changing market trends (Ip et al., 2003). Martínez et al. (2001) proposed a control structure for a consortium of virtual enterprises. This framework enabled task distribution by means of a process that includes task decomposition, PERT programming and negotiation. Other efforts include partner selection (e.g. Chen, Chen & Lee, 2007; Jarimo, Jarimo & Salo, 2009; Ip et al., 2003; Mikhailov, 2002; Wu & Su, 2005) or task assignment (e.g. Choi, Kim & Doh, 2007) among other issues related to virtual enterprises consortia management.

Focusing on GSD projects, the allocation of tasks and responsibilities to distributed teams can have a significant impact on GSD project success (Noll, Beecham & Richardson, 2010). While, back in 2008, Barcus & Montibeller (2008) stated that there is a lack of attention to the problem of allocating projects in distributed teams, recent and relevant efforts have been devoted to this field of study. Maybe the most relevant efforts come from the works of Lamersdorf, Munch et al. (e.g. Lamersdorf & Münch, 2010) although the work by Doma et al. (2009) can be considered a good antecedent. However, the work presented in this paper has a different objective. It is aimed to describe factors for work package allocation instead of task allocation and is aimed to investigate if cultural factors and competence paradigm are present in this view.

3 A literature review for allocation factors in GSD

Literature has revealed a set of factors influencing task and work package allocation in GSD environments. In this work, literature review was designed and conducted in order to identify influencing factors. As a result of this study up to 15 influencing factors were identified. There are several sources that explain the process of designing and implementing a literature review, but in order to follow a sound method, authors followed the steps drawn by Kitchenham (2007).The search strategy comprises search terms, literature resources, and search process, which are detailed one by one as follows.

The search string has to be defined based on the population under study, and the keywords and their synonyms. Thus, keywords were "Global Software Development" and "Distributed Software Development". Given the diversity of sources to be consulted electronically via the web, this established literature resources of information for the present SLR were six electronic databases (IEEEXplore, ScienceDirect, Wiley Online, ACM Digital Library, Taylor & Francis). Up to 437 documents were retreived and considered valid. A further selection of papers (selecting only the ones published in Journals or Magazines since 2004) brings a final set of 224 papers. Focusing just on papers published on 2010 and 2011, these papers include: Boden, Müller and Nett. (2011); Johri (2011); Käkölä, Koivulahti-Ojala and Liimatainen (2011); Khan, Niazi and Ahmad (2011); Kwan, Schroter and Damian (2011); Lamersdorf and Münch (2010); Noll, Beecham & Richardson (2010); Palacio et al. (2011); Patil et al. (2011); Serçea et al. (2011); Sidhu, J., & Volberda, H.W. (2011); Smite et al. (2010); Smite and Wohlin (2011).

As a result of this process, the set of factors identified were: Cost, Development Time, Temporal Distance, Geographic Distance, Cultural Distance, Size, Documentation capability, Economic environment, Customer proximity, Collaboration history, Risk reduction, Competence, Strategic Motivations, Maturity, Development quality and Trust.

4 The study

4.1 Design

The aim of Study 1 is to identify factors influencing work package allocation in GSD environments. In the previous section, a set of factors present in the literature have been depicted. The objective of this study is to identify, taking into account these factors, the set of final factors that influence work package allocation in GSD environments. To do so, the Nominal Group Technique (NGT) was employed. This technique is based on a structured focus group meeting (Delbecq, Van de Ven, & Gustafson, 1975). The NGT essentially harnesses group facilitation processes in a manner that structures group interaction in specific tasks to achieve a specific goal. Since responses are generated impartially from each participant and weighted equally, the data obtained with the NGT tend to provide a valid representation of the implicit views of the group (Elliott & Shewchuk, 2002).

Firstly, participants were required to answer the following question: What factors affect in work package assignation in GSD settings? All participants received a worksheet where the question was stated. They spent about 5 minutes generating as many concepts as possible to answer the question. The facilitator asked the participants not to consult or discuss their ideas with others. Secondly, a round-robin listing of ideas, which were recorded in a flip chart visible to the group, was conducted. The round robin process continued until all ideas reported were presented. This stage took around 20 minutes. Thirdly, the group discussion was initiated. In this stage, each proposed strategy was briefly discussed for clarification. The aim was to ensure that the meaning and logic of each response was understood by the group and identify redundant responses. Finally, to conclude, a discussion and evaluation in which group members voted the top ideas or issues was performed to find out the five preferred stages.

4.2 Sample

The selection of study participants was made taking into account that they should be professionals with relevant experience and responsibility in the field of GSD projects. We contacted a total of 20 professionals who possess the required characteristics, out of which 11 agreed to participate in the study. Out of the 11 participants, 9 individuals were men (82%) and 2 were women (18%). The average age of participants is 42.5. Minimum working experience is 12 years and the maximum 25 years. It can be considered the experience of participants as high as the average experience is 17.5 years.

4.3 Results

The expected result after the completion of this study is the list of factors involved in the allocation of work packages in GSD environments. The list of factors evaluated in the study and the votes for each factor (in descending order by number of votes) is: Cost (11); Cultural Distance (11); Development Time (11); Competence (10); Temporal Distance (10); Customer proximity (10); Trust (9); Geographic Distance (9); Maturity (9); Economic environment (8); Strategic Motivations (8); Collaboration history (7); Size (6); Development quality (5); Documentation capability (5) and Risk reduction (3).

5 Discussion and conclusions

As seen in the ranking of the obtained factors, experts have attributed the utmost importance in the process of assigning tasks to cost, cultural distance and time or duration of the project. The relevance of these variables is consistent with other studies discussed below. The study shows that the possibil-

ity of reducing costs is probably the main factor in GSD task allocation. Thus, reducing costs is the main factor for the development of GSD projects (e.g. Smite & Wohlin, 2011). Although by no means the only one. But costs are not the sole purpose in GSD and, as a result, in work package allocation. An important objective is also the possibility of reducing the duration of the project. But, in many cases, changes are more difficult in offshore countries and, as a result of this, those work packages that have less chance of changes would be better candidates for offshoring. The third factor that has received utmost importance by the experts is cultural distance. Literature highlights that cultural distance negatively impacts the level of understanding and appreciation of the activities and efforts of remote teams (Casey, 2009). But it is not necessary to belong to very distant cultures: cultural misunderstandings can also occur between Westerners or between people with common mother tongue. In this sense, some informants mentioned the misunderstandings that some of them have lived with Latin American teams. Despite sharing the same language, differences of perception and communication between both cultures generated friction. For example, the formulas of courtesy or indirect communication used by Latin Americans sometimes incorrectly interpreted as positive or agreement by the Spaniards.

Another factor considered relevant is the team competence. Competence paradigm is key to enable a modern human resource management and, in GSD settings, is based on transparency. Subjects mentioned that such transparency is hard to reach in such scenarios. More precisely, several subjects mentioned People-CMM as an effort that should guide organizations to adopt the practices needed for a proper Competence management.

Temporal and Geographic distances are also present in the study. In short, geographic and temporal distances limit informal communication, which in turn prevents trust building among distributed teams, and also limit the extent to which implicit knowledge is shared among teams, and interfere with the ability to resolve procedural issues.

With respect to maturity, according to Casey (2009), while process models like CMMi realize success in local environments, do not provide explicitly the impact or consequences in GSD environments, especially in relation to social or psychological complexities. In any case, our experts understand that software process maturity along with human resources management maturity is factors for the smooth functioning of the project.

Our informants agree that trust is the basis for successful cooperation, promoting problem solving. Literature confirms the growing importance of the trust variable and its complex relationship with other factors analyzed. In this sense, geographic, temporal and cultural distances have a significant impact on trust among members of GSD teams. Lack of confidence and lack of communication have proved barriers to effective collaboration. In addition, geographic and temporal distances limit informal communication, which in turn prevents the building of trust among distributed teams.

The rest of the factors are considered less important. However, they are also relevant for respondents. The experts mentioned that although the economic environment is not as decisive a factor as costs or the cultural distance, social and economic stability of a host country is an element that promotes the development of projects. Respondents also mentioned that projects are distributed among the participants taking into account strategic decisions whose scope exceeds the project. However, they note that such decisions often weigh down the whole project, and they suggest not considering this as a *winner* factor. Finally, collaboration history determines future decisions: a positive history favor the establishment of new partnerships. But also promotes and reinforces trust, whose importance has already been mentioned in this discussion. Factors with less than seven mentions have not been considered as important (Size; Development quality; Documentation capability and Risk reduction). Many of them have been considered as part of the previous factors by participants.

In short, the highly interdependent, uncertain and demanding tasks, as well as geographical, temporal, structural and cultural, make the management of offshore projects a remarkably complex activity. As a result, the allocation of tasks or work packages becomes a decisive factor in the management of GSD projects, and hence on the distribution of work packages.

Future works include the definition of metrics and assessment methods to all these factors in order to integrate all of them in the framework defined by Ruano-Mayoral et al. (2011). This work draws the construction of a framework for work-package allocation within GSD projects. The framework lies on three main pillars: individual and organizational competency, organizational customization and sound

assessment methods.

Finally, authors aim to link their work to software process improvement efforts. The list of factors presented in this paper considers the corporate and organizational aspects of GSD as the major objective. Maturity is one of the factors and, according to respondents; this factor can be measured using software process initiatives such as CMMi and People-CMM. Thus, authors aim at analyzing this link to provide connections between allocation factors and process maturity initiatives.

6 Literature

Boden, A., Müller, C., & Nett, B. (2011). Conducting a Business Ethnography in Global Software Development projects of small German enterprises. Information and Software Technology, 53(9), 1012–1021.

Carmel, E., & Agarwal, R. (2002). The maturation of offshore sourcing of information technology work. MIS Quarterly Executive, 1(2), 65–78.

Casey, V. (2009). Leveraging or exploiting cultural difference? In Proceedings of the 2009 Fourth IEEE International Conference on Global Software Engineering (ICGSE '09), pages 8–17, Limerick, Ireland, 2009a. IEEE Computer Society.

Chen, Q.X., Chen, X., & Lee, W.B. (2007). Qualitative search algorithms for partner selection and task allocation in the formulation of virtual enterprise. International Journal of Computer Integrated Manufacturing, 20(2-3), 115-126.

Choi, K.H., Kim, D.S., & Doh, Y.H. (2007). Multi-agent-based task assignment system for virtual enterprises. Robotics and Computer-Integrated Manufacturing, 23(6), 624-629.

Choy, K.L., & Lee, W.B. (2003). A generic supplier management tool for outsourcing manufacturing. Supply Chain Management: An International Journal, 8(2), 140-154.

Delbecq, A. L., Van de Ven, A. H., & Gustafson, D. H. (1975). Group techniques for program planning: A guide to nominal group and delphi processes. Glenview, IL: Scott, Foresma.

Doma, S., Gottschalk, L., Uehara, T., & Liu, J. (2009). Resource Allocation Optimization for GSD Projects. In Proceedings of the 3rd International Conference on. Complex Systems and Applications. Lecture Notes in Computer Science, Volume 5593/2009, 13-28.

Dossani, R., & Kenney, M. (2007). The Next Wave of Globalization: Relocating Service Provision to India. World Development, 35(5), 772-791.

Edwards, H.K., & Sridhar, V. (2005). Analysis of Software Requirements Engineering Exercises in a Global Virtual Team Setup. Journal of Global Information Management, 13(2), 21-41.

Elliott, T.R., & Shewchuk, R.M. (2002). Using the nominal group technique to identify the problems experienced by persons living with severe physical disabilities. Journal of Clinical Psychology in Medical Settings, 9(2), 65-76.

Farrell, D., & Agrawal, V. (2003). Offshoring and beyond. The McKinsey Quarterly, 4, 24–35.

García-Crespo, Á., Colomo-Palacios, R., Soto-Acosta, P., & Ruano-Mayoral, M. (2010). A Qualitative Study of Hard Decision Making in Managing Global Software Development Teams. Information Systems Management, 27 (3), 247-252.

Heeks, R., Krishna, S., Nicholsen, B., & Sahay, S. (2001). Synching or sinking: Global software outsourcing relationships, IEEE Software, 18 (2), 2001, pp. 54-60.

Herbsleb, J. D., & Moitra, D. (2001). Global software development, IEEE Software, 18 (2), 16-20.

Hye, P.K., & Joël, F. (1999). Virtual enterprise — Information system and networking solution. Computers & Industrial Engineering, 37(1-2), 441-444.

Iacovou, C.L., & Nakatsu, R. (2008). A risk profile of offshore-outsourced development projects. Communications of the ACM, 51(6), 89—94.

Ip, W.H., Huang, M., Yung, K.L., & Wang, D. (2003). Genetic algorithm solution for a risk-based partner selection problem in a virtual enterprise. Computers & Operations Research, 30(2), 213—231.

Jarimo, T., Jarimo, T. & Salo, A. (2009). Multicriteria Partner Selection in Virtual Organizations With Transportation Costs and Other Network Interdependencies. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, 39(1), 124-129.

Johri, A. (2011). Sociomaterial bricolage: The creation of location-spanning work practices by global software developers. Information and Software Technology, 53(9), 955–968.

Käkölä, T., Koivulahti-Ojala, M., & Liimatainen, J. (2011). An information systems design product theory for the class of integrated requirements and release management systems. Journal of Software Maintenance and Evolution: Research and Practice, 23(6), 443–463.

Khan, S.U., Niazi, M., & Ahmad, R. (2011). Barriers in the selection of offshore software development outsourcing vendors: An exploratory study using a systematic literature review. Information and Software Technology, 53(7), 693–706.

Kitchenham, B.A. (2007). Guidelines for performing Systematic Literature Reviews in Software Engineering. Keele University and University of Durham.

Kwan, I., Schroter, A., & Damian, D. (2011). Does Socio-Technical Congruence Have an Effect on Software Build Success? A Study of Coordination in a Software Project. IEEE Transactions on Software Engineering, 37(3), 307 – 324.

Lam, A.C.Y., Kwok, S.K., & Lee, W.B. (2008). Design of a knowledge-based virtual asset management system in dispersed manufacturing. International Journal of Value Chain Management, 2(4), 468-486.

Lamersdorf, A., & Münch, J. (2010). A multi-criteria distribution model for global software development projects. Journal of the Brazilian Computer Society, 16(2), 1–19.

Lee, W.B., & Lau, H.C.W. (1999). Multi-agent modeling of dispersed manufacturing networks. Expert Systems with Applications, 16(3), 297-306.

Martínez, M.T., Fouletier, P, Park, K.H., & Favrel, J. (2001). Virtual enterprise – organisation, evolution and control. International Journal of Production Economics, 74(1-3), 225-238.

Martins, L. L., Gilson, L. L., & Maynard, M. T. (2004) Virtual Teams: What Do We Know and Where Do We Go From Here?, Journal of Management, 30 (6), 805-835.

Meyer, B. (2006). The unspoken revolution in software engineering. IEEE Computer, 39(1), 121—124.

Mikhailov, L. (2002). Fuzzy analytical approach to partnership selection in formation of virtual enterprises. Omega, 30(5), 393-401.

Milewski, A. E., Tremaine, M., Köbler, F., Egan, R., Zhang, S., & O'Sullivan, P. (2008). Guidelines for Effective Bridging in Global Software Engineering. Software Process: Improvement and Practice, 13(6), 477–492.

Niederman, F., Kundu, S., & Salas, S. (2006). IT Software Development Offshoring: A Multi-Level Theoretical Framework and Research Agenda. Journal of Global Information Management, 14(2), 52-74.

Noll, J., Beecham, S., & Richardson, I. (20110). Global software development and collaboration: barriers and solutions. ACM Inroads, 1 (3), 66-78.

Palacio, R.R., Vizcaíno, A., Morán, A.L., & González, V.M. (2011). Tool to facilitate appropriate interaction in global software development. IET Software, 5(2), 157–171.

Park, K.Y., & Favrel, J. (1999). Virtual enterprise — Information system and networking solution. Computers & Industrial Engineering, 37(1-2), 441-444.

Patil, S., Kobsa, A., John, A., & Seligmann, D. (2011). Methodological reflections on a field study of a globally distributed software project. Information and Software Technology, 53(9), 969–980.

Ruano-Mayoral, M., Colomo-Palacios, R., Fernández-González, J.M., & García-Crespo, A. (2011). Towards a Framework for Work Package Allocation for GSD. In Meersman, Robert; Dillon, Tharam; Herrero, Pilar (Eds.), On the Move to Meaningful Internet Systems: OTM 2011 Workshops, Lecture Notes in Computer Science, Vol. 7046, 200-207.

Sakthivel, S. (2007). Managing risks in offshore systems development. Communications of the ACM, 50 (4), 69-75.

Schümmer, T., & Lukosch, S. (2009). Understanding Tools and Practices for Distributed Pair Programming. Journal of Universal Computer Science, 15 (16), 3101-3125.

Serçea, F.C., Swigger, K., Alpaslan, F.N., Brazile, R., Dafoulas, G., & Lopez, V. (2011). Online collaboration: Collaborative behavior patterns and factors affecting globally distributed team performance. Computers in Human Behavior, 27(1), 490–503.

Shi, Y., Gregory, M., & Naylor, M. (1997). International manufacturing configuration map: a self-assessment tool of international manufacturing capabilities. Integrated Manufacturing Systems, 8(5), 273-282.

Sidhu, J., & Volberda, H.W. (2011). Coordination of globally distributed teams: A co-evolution perspective on offshoring. International Business Review, 20(3), 278–290.

Smite, D., Wohlin, C., Gorschek, T., & Feldt, R. (2010). Empirical evidence in global software engineering: a systematic review. Empirical Software Engineering, 15(1), 91-118.

Smite, D., & Wohlin, C. (2011). A Whisper of Evidence in Global Software Engineering. IEEE Software, 28(4), 15-18.

Tso, S.K., Lau, H., & Ho, J.K.L. (2000). Coordination and monitoring in an intelligent global manufacturing service system. Computers in Industry, 43(1), 83-95.

Winkler, J.K., Dibbern, J., & Heinzl, A. (2008). The impact of cultural differences in offshore outsourcing—Case study results from German–Indian application development projects. Information Systems Frontiers, 10(2), 243-258.

Wu, N., & Su, P. (2005). Selection of partners in virtual enterprise paradigm. Robotics and Computer-Integrated Manufacturing, 21(2), 119-131.

7 Author CVs

Marcos Ruano-Mayoral

Marcos Ruano-Mayoral is a consultant at LowendalMasaï, Spain. Formerly he was a research assistant of the Computer Science Department at Universidad Carlos III de Madrid. He holds a BSc in computer systems from Universidad de Valladolid and a MSc in computer science from Universidad Carlos III de Madrid. He is a Ph.D. candidate from Universidad Carlos III de Madrid, Spain.

Ricardo Colomo-Palacios

Ricardo Colomo-Palacios is an associate professor at the Computer Science Department of the Universidad Carlos III de Madrid. His research interests include applied research in information systems, software project management, people in software projects and social and semantic web. He received his PhD in computer science from the Universidad Politécnica of Madrid (2005). He also holds a MBA from the Instituto de Empresa (2002). He has been working as software engineer, project manager and software engineering consultant in several companies including Spanish IT leader INDRA. He is also an Editorial Board Member and Associate Editor for several international journals and conferences and Editor in Chief of International Journal of Human Capital and Information Technology Professionals.

Ángel García-Crespo

Angel Garcia Crespo is the Head of the SofLab Group at the Computer Science Department in the Universidad Carlos III de Madrid and the Head of the Institute for promotion of Innovation Pedro Juan de Lastanosa. He holds a PhD in Industrial Engineering from the Universidad Politécnica de Madrid (Award from the Instituto J.A. Artigas to the best thesis) and received an Executive MBA from the Instituto de Empresa. Professor García-Crespo has led and actively contributed to large European Projects of the FP Vand VI, and also in many business cooperations. He is the author of more than a hundred publications in conferences, journals and books, both Spanish and international.

Sanjay Misra

Sanjay Misra obtained M.Tech. Degree in Software Engineering from Motilal Nehru National Natioanal Institue of Technology, Allahabad India and D.Phil. from University of Allahabad, India. His current researches cover the areas of: Software quality, software measurement, software metrics, software process improvement, software project management, object oriented technologies, XML, SOA, Web Services, and cognitive informatics. He published more than 70 papers in these areas and got several awards for outstanding publications. He is founder of several international workshops annd is serving as editor in chief, associate editor, and editorial board member of several journals of international repute.

Software Process Improvement Initiatives in Global Software Development: A Systematic Literature Review

Ali Saeed Khan, Muhmmad Younus Javed, Farooque Azam
National University of Sciences & Technology Islamabad, Pakistan
{alisaeed, myjaved, farooq}@ceme.nust.edu.pk

Abstract

The key benefits of distributed project development are the existence of twenty four hour work cycle, cost reduction and availability of larger pool of resources. These benefits have steered software organizations around the globe to adopt the practices of global software development (GSD). The development process involved in distributed software development is different than collocated software development process (CSD). Traditional software process improvement initiatives are not very effective in distributed software development (DSD). The literature of Global Software Process Improvement (GSPI) has steadily been growing. However, it is very difficult to find a complete systematic and conclusive literature review on GSPI initiatives. We have conducted a systematic literature review on the studies, reporting global software process improvement initiatives to identify current factors and challenges involved in global software process improvement. This review is based on a selection process which comprises of three phases of study selection (i.e. primary, secondary and final). The finally selected publications are used for the data extraction based on the research questions. The primary focus of this review was to extract the data for the identification of existing GSPI initiatives (both proposed and implemented). The secondary focus was to identify the key challenges and factors involved in improvement of global software development process. This paper presents the literature overview to provide an insight to the researchers and practitioners into current GSD and GSPI initiatives. The extracted data is synthesized to report key factors and initiatives. Based on the evidence from the literature a small set of Factors is proposed. A time line is presented in the end to depict the research trend in the area of GSPI.

Keywords

Global Software Process Improvement Initiatives, Global Software Development, Systematic Literature Review.

1 Introduction

The growth and progression in global communication systems has enabled organizations to utilize most suitable resources around the globe to develop high quality products. Software industry is widely using distributed development style [1, 2, 3, 4, 5, 6, 7]. The skill and cost limitations of collocated software development can be easily skipped by distributed software development. Global software development provides a larger domain of software experts from all around the world and at the same time reduces the cost of development process.

Product quality is mostly dependent upon the quality of process followed to develop it [8]. Process improvement is usually a recurrent method which is based on knowledge of current process and past experiences. The definition of process is critical for its improvement. Software development process has varied over the last decade with the concept of business process outsourcing (BPO). This paper is an effort to explore the existing research on global software process improvement.

Globalization of software development is inevitable [9] and every process needs to be improved with each project. Practitioners are widely adapting distributed development so as the research community is working to design strategies to improve this process [10, 11]. This paper is the first step as an effort to design a framework for GSPI. The data about the challenges in the process and the existing strategies has to be extracted and analyzed to identify the areas where further improvement can be achieved.

The prime objective of this research paper is the identification and classification of current initiatives used in the improvement of global software development process. The secondary objective is to highlight all factors involved in the progression or retrogression of process improvement.

The next section gives an overview of the literature covering related topics to the area of interest. Section 3 outlines the research methodology that we have used. In section 4 we have presented the analysis and results. Conclusion of the paper is paragraphed in section 5. References are provided in the last section.

2 Literature Overview

The prime objective of this literature review is to search for existing literature available on Global Software Process Improvement Initiatives or on global software development. The improvement of a process needs a consolidated definition of that process. In the literature there is no study which formally defines a global software development process. The only definition found is presented by Mariangela, Marcelo, and Rafael [12] which is guided by the analyses of a set of practices followed by an organization at various distributed locations. This definition is role-based and each team has a focal point defined individually for each site by the team leader. The authors have explored a case study for data collection and analysis.

Gucegliogl and Demirors [8] have developed and applied a process quality measurement model in a software process improvement initiative. Setamanit, Wakeland and Raffo have developed a model and simulated it to study the effects of Global Software Development and to highlight important factors [6]. These factors are critical to the improvement and management of the global development process. the simulation model presented in this study is hybrid in nature which combines system dynamics and discrete-event paradigms to represent GSD projects. Prikladnicki, Nicolas and Evaristo have presented a reference model (based on results of a case study to) for GSD [11]. Habra et.al has presented a methodology to initiate software process improvement in small enterprises [13]. Such initiatives can be tailored by practitioners to their needs for software process improvement on distributed sites.

Kitchenham et.al [14] has provided a very complete and reasonable definition of Systematic Literature Review, i.e. an SLR is a methodologically rigorous review of research results. An SLR should not only look for answers to the research questions, it should also provide a set of guidelines to the practitioners. Ali Babar and Mahmood Niazi [10] have conducted an empirical study of Vietnamese global software market to explore practitioner's experiences and implementation perceptions of SPI initiatives. Authors have conducted a questionnaire survey of companies and practitioners. Similar kind of work is done by Rauf, Anwar, Ramzan and Shahid to analyze the SPI efforts in Pakistan [15]. Kitchenham et.al has carried out a study to find SLR's on software engineering published from 2004 to 2008 [16]. This study has proved to be a very helpful catalogue. The results from this study highlight that only two SLRs on Software Process have been published in this time period. This focus needs to be increased in order to achieve better improvement in global software process. Siffat-ullah, Niazi and Ahmad have identified success factors for offshore software development outsourcing vendors in an empirical study [17]. This study can prove to be very helpful for practitioners (vendors) to standardize their organization for better business development. The same authors have identified barriers in the selection of outsourcing vendors in an exploratory study [18].

The most relevant study we found is a systematic literature review carried out by Unterkalmsteiner et.al on evaluation and measurement of software process improvement [19]. Hossain, Babar and Paik have carried out a SLR covering literature on usage of Scrum in global software development [20]. They have uncovered some key factors which can be used in the improvement of GSPI. A similar study is published by Jalali and Wohlin which covers implementation of agile practices in global software engineering [21]. Harter, Kemerer and Slaughter have conducted a longitudinal field study to analyze the effects of software process improvement on severity of defects [22]. Smite, Wohlin, Feldt

and Gorschek has presented a classification scheme for describing the context of a GSE study [23]. Jung and Goldenson have provided empirical evidence in their study [24] to support a proposition that process maturity is associated with project performance and product quality. In adaptation of an initiative there are always a set of de-motivators which should be considered in design of an initiative. In persuading an organization to become accustomed to new initiatives these motivators must be brought into focus. Baddoo and Hall have presented a study which lists a set of de-motivators from practitioner's perspective [25]. Babar, Kitchenham, Zhu, Gorton and Jeffery have proposed the concept of software architecture evaluation based on Internet-based collaborative technologies [26]. The study provides preliminary findings on the viability of groupware-supported evaluation process. Wohlin, Martin and Henningsson have presented four methods of empirical research in software engineering [27] any one of them can be used in the context of global process improvement.

Krishnan and Kompali have developed and implemented a process maturity framework which they have based on 24 key process areas [3]. Software architecture plays an important role in the development process. Babar has presented a framework to support software architecture evaluation in global software development [28]. Ribeiro, Czekster and Webber have proposed a simplified process which can be used by collocated team in a global development [29].

Coordination plays an important role in the success of a globally distributed project. Research community has focused on this aspect in the recent years. Gupta and Fernandez have identified patterns of collaboration among distributed team members [4] and analyzed the effectiveness of collaboration mechanisms in specific scenarios. Supported on their analysis authors have recommended a planning based approach which can be used to enhance collaboration effectiveness. Fabinder and Henz have shared their experiences from industry in implementing global development as combination of collaboration across functions and locations [30].

Klein, Rausch and Fischer have presented an approach that bridges the semantic gap between processes by construction of syntactically correct processes [31]. The authors have elaborated their approach through the use of collaboration scenarios in globally distributed organizations. Magdaleno, Werner and Araujo have argued that the collaboration in globally developed projects can be analyzed through the use of social networks [32]. They have discussed the social network tools against the collaboration requirements which could be helpful for researchers and practitioners in tailoring processes. One of the most challenging steps in GSP is the elicitation of requirements. Gabriela, Vizcaino and Piattini have proposed a framework which analyzes the hindering factors in global software development and suggests strategies for the improvement of these factors [33]. A comparison of multi-site development processes has been presented by Avritzer and Paulish [34].

3 Research Strategy and Procedure

The standard procedure to carry out a systematic literature review in software engineering is outlined by Barbara Kitchenham [35, 36, 37] which is followed in this review. The procedure followed for literature review is outlined in Fig 1. This literature review doesn't follow those guidelines to the knot. We have also followed the SLR patterns presented in studies by Kitchenham, Unterkalmsteiner, Hossain and Babar [14, 16, 19 and 20].

The process is divided into three sub-processes, Pre-Review, Review and Post-Review processes. The pre-review sub process is the preparation which includes some important decisions. We started from definition of problem domain and area of interest. The motivation for selecting this research area is the increasing global trend towards distributed development processes.

```
                           ⟨ Start ⟩
                              │
   ┌───────────────────────────────────────┐
   │  ┌─────────────────────────────────┐  │         ┌──────────────────┐
   │  │  Defining Problem Domain and    │  │         │     Review       │
   │  │     Area of Interest            │  │         │   Preparation    │
   │  └─────────────────────────────────┘  │         │                  │
   │  ┌─────────────────────────────────┐  │         │  1. Pre – Review │
   │  │  Outlining Research Questions   │  │         │      Step        │
   │  └─────────────────────────────────┘  │         └──────────────────┘
   │  ┌─────────────────────────────────┐  │
   │  │  Defining a Homogeneous Study   │  │
   │  │     Selection Methodology       │  │
   │  └─────────────────────────────────┘  │
   │  ┌─────────────────────────────────┐  │
   │  │  Selecting Literature Sources   │  │
   │  └─────────────────────────────────┘  │
   │  ┌─────────────────────────────────┐  │
   │  │      Defining Keywords          │  │
   │  └─────────────────────────────────┘  │
   └───────────────────────────────────────┘
                              │
   ┌───────────────────────────────────────┐
   │  ┌─────────────────────────────────┐  │         ┌──────────────────┐
   │  │   Primary Study Selection       │  │         │  Study Selection │
   │  └─────────────────────────────────┘  │         │    Methodology   │
   │  ┌─────────────────────────────────┐  │         │                  │
   │  │   Secondary Study Selection     │  │         │  2. Review Step  │
   │  └─────────────────────────────────┘  │         └──────────────────┘
   │  ┌─────────────────────────────────┐  │
   │  │     Final Study Selection       │  │
   │  └─────────────────────────────────┘  │
   └───────────────────────────────────────┘
                              │
   ┌───────────────────────────────────────┐
   │  ┌─────────────────────────────────┐  │         ┌──────────────────┐
   │  │      Data Extraction            │  │         │     Results      │
   │  └─────────────────────────────────┘  │         │   Engenderment   │
   │  ┌─────────────────────────────────┐  │         │                  │
   │  │  Data Analysis & Data Synthesis │  │         │  3. Post –Review │
   │  └─────────────────────────────────┘  │         │      Step        │
   │  ┌─────────────────────────────────┐  │         └──────────────────┘
   │  │     Results Documentation       │  │
   │  └─────────────────────────────────┘  │
   └───────────────────────────────────────┘
                              │
                         ( Finish )
```

Figure 1: Systematic Literature Review Procedure

An important process of planning a literature review is the definition of research questions. These questions are mapped from the problem domain and highlight the area of interest. The research questions selected for this literature review are outlined in Table 1. To review the literature a methodology (protocol) is defined. The protocol is uniformly outlined and homogeneously followed by all three authors for the literature review. The quality of the literature is critical in a review to ensure the quality the depth and breadth of the literature is very important. The depth depends on the search criteria and breadth depends on the sources.

To ensure the quality top three internationally acclaimed bodies are selected i.e. IEEE[1], Springer[2] and Elsevier[3]. Google Scholar[4] was also used to search the literature. In the last step of preparation key phrases were derived from the keywords in different combinations using logical operators. These keywords cover research questions and the area of interest. The digital sources are searched with these key phrases.

The second sub-process is the execution of protocol. This phase is divided into three steps. The first is the primary study selection where each digital source is searched with the key phrases. English is used as language for searching and the duration is kept from year 2000 to 2011. The first step returned a large number of studies. This number is further reduced by filtering out irrelevant literature in second phase.

---

[1] http://ieeexplore.ieee.org/Xplore/dynhome.jsp
[2] http://www.springerlink.com
[3] http://www.sciencedirect.com
[4] http://scholar.google.com.pk

Table 1: Research Questions

ID	Research Question	Aim
RQ1	What are the current global software process improvement initiatives?	Classification of the current initiatives (in the literature) used for standardization of global software process improvement.
RQ2	Can Collocated Initiatives are applicable in Global?	Collocated software development process is much matured and initiatives can be found at different levels of implementation. It will be very effortless if these initiatives can be used for global processes. The purpose of this question is to look for any evidence in the literature which can prove this implementation.
RQ3	Do the improvement initiatives really affect the maturity of the process?	The objective here is to find empirical evidence regarding the effectiveness of improvement initiatives.

The second step is secondary study selection which is based on selection filters. These filters include publication year, content type, subject, and specialization area. The inclusion or exclusion of a study is based on a key factor named as "Relevance". This factor is calculated for each study and scaled as low, medium and high. Papers with medium and high relevance are only included for final study selection otherwise they are excluded. The relevance for the secondary selection study is calculated on certain factors like title, authors, content type, publication date and maturity of publisher.

The third step is final study selection where the most relevant literature is selected and other is excluded from secondary study. The criterion for study selection in this step is also 'Relevance' however the calculation of relevance is based on slightly different factors. In this step papers are reviewed in more depth i.e. abstract, introduction and result sections are overviewed by each author and then relevance is assigned to it. The final set of publications is used for data extraction and analysis. The partially reviewed studies provided a wide range of knowledge for research trends in global software development process.

4 Data Analysis & Results

4.1 Data Extraction & Research Answers

The research shows that the focused area of interest (i.e. Global Software Development Process Improvement Initiatives) is very new and pre-mature. The main digital sources were searched for the relevant literature but the initial outcome was very low number of studies. Therefore we had to broaden our area of interest. This area of global software development is developing. It needs a lot of focus from industry, academia and research communities.

There are standardized initiatives like CMMI [38] and ISO 9001: 2000 [39] for collocated software development process but no such internationally acknowledged initiative could be found for global software process improvement. The research community needs to work on the standardization of such initiative. The main outcome objective of this literature review is to look for the answers of research questions (Table 1).

Answers to RQ1. The literature review shows that currently there is not globally accepted standard for global process improvement. Still the initiatives found in the review are tabulated in table2. In [12] authors have presented a practice for global software process definition using data from a case study. The practice is simply based on two roles, Process Owner (PO) and Process Reviewer (PR) and a Process Change Request Documentation (PCRD). The PO defines global process and the artifacts, schedules the meetings and updates the artifacts. The PR revises and validates the process artifacts and suggests the changes. The review process achieves a level of process improvement which is restricted to the site only.

Answer to RQ2. Analysis of extracted data from the research literature clearly specifies that we can't use the initiatives used for collocated software process for global processes. Global software devel-

opment challenges the techniques of traditional software engineering and requires new solutions [11, 40]. The practices, organizational structures and initiatives used for collocated development are often not ample for GSD projects [9].

Answer to RQ3.The answer is 'Yes', and is supported by the literature. A survey result presented in [10] states that only 4% practitioners have the opinion that SPI initiatives in their organizations have not provided the desired results. This rate is further reduced in south Asian countries.

4.2   Data Synthesis

The main objective of this review is to highlight the existing initiatives and factors in the current literature. Table 2 classifies the initiatives found in the research. In the first part only empirically tested frameworks, models, techniques and me-thodologies are presented. Proposed or only reviewed initiatives are excluded.

Table 2: Initiatives

Initiative	Classification	Reference(s)
Model	Process Evaluation	[8, 11, 6]
Model	Process Collaboration	[31]
Model	Process Improvement	[13]
Framework	Process Maturity	[3]
Framework	Software Architecture Evaluation	[28]
Framework	Requirement Elicitation	[33]

We have identified some key factors effecting process improvement and have classified them in table 3. The three basic classifications (with respect to development process) used for these factors are Communication, Coordination and Control. These factors can be used by practitioners to help them select an appropriate improvement initiative. The researchers can benefit from the Reference column to overview the source of a certain impact factor. Impact of each factor on the globally distributed development process is specified as positive, negative and both. A factor can have a positive or negative impact based on its usage.

Table 2: Identified Key Factors

Key Factors	Classification	Impact	Reference(s)
Cultural Differences	Coordination (Fundamental)	Negative	[12, 11, 41, 6, 17, 33, 18]
Different Time Zone	Coordination (Fundamental)	Both	[12, 11, 41, 6, 33]
Language Difference	Communication (Fundamental)	Negative	[12, 11, 4, 6, 17, 18]
Pre-Existent Processes	Control (Fundamental)	Positive	[12]
Communication Infrastructure	Fundamental (Communication)	Both	[12, 11, 41, 33, 18, 25]
Organizational Structure	Control (Organizational)	Both	[12, 11, 41]
Team Trust	Coordination (Organizational)	Positive	[12, 11, 5]
Team Size	Control (Strategic)	Both	[12, 11, 17]
Development Site	Control (Strategic)	Both	[6]
Product Architecture	Control (Strategic)	Both	[6]
Task Allocation Strategy	Control (Strategic)	Both	[6]
Distribution Overhead	Coordination (Strategic)	Negative	[6]
Distribution Effort Loss	Coordination (Strategic)	Negative	[6]
Impacts from Virtual Teams	Coordination (Organizational)	Negative	[6]
Team Formulation	Control (Organizational)	Both	[6]

Team Dynamics	Control (Organizational)	Both	[6]
Inertia – Laziness	Control (Strategic)	Negative	[10, 25]
Staff Training	Control (Organizational)	Both	[20, 10,11, 25]
Formal Procedures	Control (Strategic)	Positive	[10]
Lack of Tools	Control (Organizational)	Negative	[10, 25]
SPI Awareness	Control (Organizational)	Both	[10]
Shared availability of participants during interactions	Coordination (Organizational)	Positive	[4]
Shared familiarity of participant's knowledge/skills	Coordination (Strategic)	Positive	[4, 33]
Shared Context	Coordination (Strategic)	Positive	[11, 4]
Shared motivation for win-win	Coordination (Strategic)	Positive	[4]
Shared portal for content sharing without confusion	Coordination (Strategic)	Positive	[4]
Lack of knowledge on the expectations and processes of different teams	Coordination (Strategic)	Negative	[11, 4]
Differing Technical and Domain Vocabularies	Coordination (Strategic)	Negative	[9]
Coherence	Coordination (Fundamental)	Positive	[40]
Interpersonal Relationships between stakeholders	Communication (Fundamental)	Positive	[11]
Lack of Knowledge / Expertise Exchange	Communication (Organizational)	Negative	[17]
Hidden Costs	Control (Organizational)	Negative	[18]
Imposition	Coordination (Strategic)	Negative	[25]
Lack of SPI management skills	Control (Organizational)	Negative	[25]

Communication Infrastructure, Cultural Differences, Time Zone and language barrier are very foundational factors of global development process. The structure of the organization has an immense impact on the process. The team size and trust are two factors which are found in multiple studies.

The knowledge of driving factors is crucial for design and implementation any improvement initiative. We have presented all the factors involved in global software development which are collected during our literature review. In [11] authors have presented factors in a conceptual map of project development.

Based on the evidence collected during the literature review we have came up with some factors that may play an important role in distributed process improvement. These factors are just proposed here. The impact of these factors can only be analyzed on implementation. The factors are presented in Table 4.

Table 3: Set of Proposed Factors

	Proposed Factor	Aim
1.	Different Work Ethics	Teams working in different working environment and culture may develop variant working ethics.
2.	Professional & Personal priorities	People from different backgrounds usually have different professional and personal priorities.
3.	Long term staff hiring	Coordination developed can be sustained for a long time.
4.	Change Announcement	This factor must be considered by management. When a change occurs at one location it should be announced to all the differentially located teams.

The research trend in global software development is depicted in table 5. The time line is constructed from the filtered literature from secondary study selection. Hence this timeline covers literature narrowed down to area of interest. This time line is restricted to the last six years because the literature available on the previous years is not very relevant. It emphasized more on global software development, global architecture and distributed requirement elicitation.

Table 4: Research Trend Timeline

	2006	2007	2008	2009	2010	2011
Journal	08	09	09	07	10	14
Conference	07	07	10	17	13	13
Book Chapters	01	07	03	06	07	06
Reports (Technical/Academic)	00	00	00	01	00	00
Total	16	23	22	31	30	33

5 Conclusion a& Future Work

The driving factors in a process are cost, quality and schedule. The improvement is usually incurred to reduce cost, increase quality and meet the schedule. The global process must be defined and this definition should be homogeneous and clearly understood by all the distributed on and off-sites involved in development.

This research paper has presented a systematic literature review carried out to find the existing initiatives for global software process improvement and key factors involved in its success or failure. The data extracted from the literature is synthesized to present the findings of the study. The outcome of this review is a guideline for the researchers and practitioners to design and implement new GSPI initiatives. A research trend time line is formed stretching over the last six years to show the increasing trend in GSD.

A questionnaire based empirical study is lining up after this review to study the key factors in the local small to medium sized organizations.

Acknowledgements

6 References

1. Seija Komi-Sirvio, Development and Evaluation of Software Process Improvement Methods (June, 2004)
2. Ismo Lehtonen , Communication Challenges in Agile Global Software Development. Report, Department of Computer Science, University of Helsinki (2009)
3. Narayan, M.S. Krishnan, Prasad Kompalli, Leveraging Global Resources: A Process Maturity Framework for Managing Distributed Development, IEEE Computer Society, pp. 80 – 86. (2005)
4. Mayank Gupta, Jude Fernandez, How Globally Distributed Software Teams Can Improve their Collaboration Effectiveness?, Proceedings of 6th International Conference on Global Software Engineering, pp. 185 – 189. (2011)
5. Gil Taran, Lynn Robert Carter, Improving Distance Mentoring: Challenges and How to Deal with them in Global Development Project Courses, Proceedings of 23rd IEEE Conference on Software Engineering Education and Training, pp. 97 – 104. (2010)
6. Setamanit, S.-o., Wakeland, W., Raffo, D., Improving Global Software Development Project Performance Using Simulation, Portland International Center for Management of Engineering and Technology, USA, pp. 2458 – 2466. (2007)
7. Par J Agerfalk, Brian Fitzgerald, Helena Holmstrom Olsson, Eoin O Conchuir, Benefits of Global Software Development: The Known and Unknown, Springer ICSP 2008, LNCS 5007, pp. 1 – 9, 2008.
8. A. Selcuk Guceglioglu and Onur Demirors, The application of a New Process Quality Measurement Model for Software Process Improvement Initiatives, 11th IEEE International Conference on Quality Software (2011)

9. James D. Herbsleb, Global Software Engineering: The Future of Socio-technical Coordination, Proceedings of Future of Software Engineering (FOSE 07), pp. 188 – 198. (2007)

10. M. Ali Babar, Mahmood Niazi, Implementing Software Process Improvement Initiatives: An Analysis of Vietnamese Practitioners' Views, proceedings of 2008 IEEE International Conference on Global Software Engineering, pp. 67 – 76. (2008)

11. Rafael Prikladnicki, Jorge Luis, Roberto Evaristo, A Reference Model for Global Software Development: Findings from a Case Study, Proceedings of IEEE International Conference on Global Software Engineering (ICGSE 2006)

12. Mariangela Vanzin, Marcelo Ribeiro, Rafael Prikladnicki, Global Software Process Definition in a Distributed Environment, Proceedings of 29th IEEE/NASA Software Engineering Workshop (2005)

13. Naji Habra, Simon Alexandre, Jean-Marc Desharnais, Claude Y. Laporte, Alian Renault, Initiating Software Process Improvement in very Small Enterprisees Experience with a Light Assessment Tool, Information and Software Technology, Volume 50, Issues 7–8, pp.763 – 771, June 2008.

14. Barbara Kitchenham, O. Pearl Brereton, David Budgen, Mark Turner, John Bailey, Stephen Linkman, Systematic Literature Reviews in Software Engineering – A systematic literature review, Information and Software Technology, Volume 51, Issue 1, pp. 7-15 (2009)

15. Rauf, Abdul, Ramzan Muhammad, Anwar Sajid, Shahid A. Ali, Analysis of Software Process Improvement Efforts in Pakistan, Proceedings of 2nd International Conference on Computer and Automation Engineering (ICCAE), Volume 1, pp. 375 – 379. (2010)

16. Barbara Kitchenham, Rialette Pretorious, David Budgen, O. Pearl Brereton, Mark Turner, Systematic Literature Reviews in Software Engineering – A tertiary study, Information and Software Technology, Volume 52, Issue 8, pp. 792-805 (August 2010)

17. Siffat Ullah Khan, Mahmood Niazi, Rashid Ahmad, Critical Success Factors for Offshore Software Development Outsourcing Vendors: An Empirical Study, Springer LNCS 6156, pp. 146-160, 2010.

18. Siffat Ullah Khan, Mahmood Niazi, Rashid Ahmad, Barriers in the Selection of Offshore Software Development Outsourcing Vendors: An Exploratory Study using a Systematic Literature Review, Information and Software Technology, Volume 53, Issue 7, pp. 693-706, July 2011.

19. Michael Unterkalmsteiner, Tony Gorschek, A. K. M. Moinul Islam, Chow Kian Cheng, Rahadian Bayu Permadi, Robert Feldt, Evaluation and Measurement of Software Process Improvement – A Systematic Literature Review, IEEE Transactions on Software Engineering (2011)

20. Emam Hossain, M. Ali Babar and Hye-young Paik, Using Scrum in Global Software Development: A Systematic Literature Review, Fourth IEEE International Conference on Global Software Engineering, pp. 175 - 184. (2009)

21. Samireh Jalali, Claes Wohlin, Agile Practices in Global Software Engineering – A Systematic Map, Proceedings International Conference on Global Software Engineering, pp. 45-54, Princetown, USA, (August 2010)

22. Harter, D., Kemerer, C., Slaughter, S., Does Software Improvement Reduce the Severity of Defects? A Longitudinal Field Study, IEEE Transactions on Software Engineering (2011)

23. Darja Smite, Claes Wohlin, Robert Fledt, TonY Gorschek, Reporting Empirical Research in Global Software Engineering: a classification scheme, IEEE International Conference on Global Software Engineering (ICGSE) October, 2008.

24. Ho-Won Jung, Dennis R. Goldenson, Evaluating the Relationship between Process Improvement and Schedule Deviation in Software Maintenance, Information and Software Technology, Volume 51, Issue 2, pp.351-361, February 2009.

25. Nathan Baddoo, Tracy Hall, De-Motivators for Software Process Improvement: An Analysis of Practitioner's Views, Journal of Systems and Software, Volume 66, Issue 1, 15, pp. 23-33, April 2003.

26. Muhammad Ali Babar, Barbara Kitchenham, Liming Zhu, Ian Gorton, Ross Jeffery, An Empirical Study of Groupware Support for Distributed Software Architecture Evaluation Process, Journal of Systems and Software, Volume 79, Issue 7, pp. 912-925, July 2006.

27. C. Wohlin, M. Höst, K. Henningsson, Empirical Research Methods in Software Engineering, Lecture Notes in Computer Science. Berlin, Germany: Springer, vol. 2765, pp. 7–23. (2003)

28. M. Ali Babar, A Framework for Supporting the Software Architecture Evaluation Process in Global Software Development, 4th IEEE International Conference on Global Software Engineering, pp. 93 – 102. (2009)

29. Marcelo Blois, Ricardo M. Czekster, Thais Webber, Improving Productivity of Local Software Development Teams in a Global Software Development Environment, Proceedings of IEEE International Conference on Global Software Engineering (ICGSE 2006)

30. Fassbinder, P., Henz, V., Improving Global System Development and Collaboration across Functions Experiences from Industry, Proceedings of Fourth IEEE International Conference on Global Software Engineering, pp. 262 – 266. (2009)

31. Herald Klein, Andreas Rausch, Edward Fischer, Collaboration in Global Software Engineering Based on Process Description Integration, Springer LNCS 5738, pp. 1 – 8, 2009.

32. Andrea magalhaes Magdaleno, Claudia Maria Lima Werner, Renata Mendes de Arauja, Analyzing Collaboration in Software Development Processes through Social Networks, Springer LNCS 6415, pp. 435 – 446, 2010.
33. Gabriela N. Aranda, Aurora Vizcaino, Mario Piattni, A Framework to Improve Communication during the Requirements Elicitation Process in GSD Projects, Requirements Engineering, Volume 15, Number 4, pp. 397-417, 2010.
34. Alberto Avritzer, Daniel J. Paulish, A Comparison of Commonly Used Processes for Multi-Site Software Development, Collaborative Software Engineering (Springer), pp. 285-302. (2010)
35. B. Kitchenham, S. Charters, Guidelines for performing systematic literature reviews in software engineering, Software Engineering Group, Keele University and Department of Computer Science, University of Durham, United Kingdom, Technical Report EBSE-2007-01, 2007.
36. P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, M. Khalil, Lessons from applying the systematic literature review process within the software engineering domain, Journal of Systems and Software, vol. 80, no. 4, pp. 571–583. (2007).
37. Barbara Kitchenham, Procedures for performing Systematic Reviews, Technical Report, July 2004.
38. CMMI Development Team, Capability Maturity Model Integrated, Version 1, Software Engineering Institute, 2001.
39. International Organization for Standardization, ISO Standard 9001: Quality Management Systems, 2000.
40. Christof Ebert, Casimiro Hernandez Parro, Roland Suttels, Harald Kolarczyk, Improving Validation Activities in a Global Software Development, Alcatel, Switching and Routing Division, Spain, Belgium, Germany.
41. James D. Herbsleb and Deependra Moitra, Global Software Development, IEEE Software Magazine, pp. 16 – 20, (March/April 2001)

A Methology for IT Outsourcing Services Provision Management[*]

Fco. Javier Sáenz[1], Mercedes de la Cámara[1], J.A. Calvo-Manzano[2], Eugenio Fernández[3]

[1]Escuela Universitaria de Informática, Universidad Politécnica de Madrid, Madrid, España
{*jsaenz@eui.upm.es, mcamara@eui.upm.es*}
[2] Facultad de Informática, Universidad Politécnica de Madrid, Madrid, España
jacalvo@fi.upm.es
[3] Facultad de Documentación, Universidad Alcalá de Henares, Madrid, España
eugenio.fernandez@uah.es

Abstract

Current technological advances that have led to a globally connected economy, coupled with the increasing trend towards privatization and deregulation, are leading to new organizational models and increased collaboration between suppliers and customers, sharing information and process flows. This has directly contributed to the great expansion of outsourcing, considering it as a strategic tool for organizations. In this environment, and given the increasing organizational complexity, the use of a methodology to assist the implementation of outsourcing projects has become almost necessary. In recent years several methodologies have been proposed, especially for supporting client outsourcing organizations, but we do not consider them to be complete, they do not cover all necessary aspects to manage an outsourcing project. That is the reason because we, in this paper, propose a methodology for outsourcing projects management from the standpoint of the provider that was complete and easy to apply.

Keywords

Outsourcing; Methodology; IT Service Management; Project Management

---

[*] Authors are ordered by University & School

1 Introduction

Business organizational models that have traditionally existed in the last century are undergoing a profound transformation [7]. Technological advances that have led to a globally connected economy and the increasing trend towards privatization and deregulation, are leading to new organizational models. There is an increasing tendency to fragmentation of the processes that make up the value chain, breaking the hierarchical organizational structure and treating each of its fragments as capable of producing benefits by themselves to the organization, increasing the flexibility of it, reducing administrative costs, and even having them without being their owner. On the other hand the collaboration between suppliers and clients to share information and process flows is increased, reaching at times to form a "virtual organization chart". In this model the manager's role also changes, to the extent that not only manages a fully integrated organization with a culture and values clear and consistent, but he is responsible for a network of specialized suppliers that support every fragment of the value chain. This leads to face a set of important decisions that will have to manage [7]. As the process of fragmentation of the traditional business model is increasingly fragmented, the outsourcing of these elements is spreading to the same extent, affecting a wider range of business functions. And what is outsourcing? There are various definitions in [7], [16], [18], [6], [13], [11], [22], [3], [25], but all of them being around the same ideas, which can be summarized as: transfer of activities from one organization to an external provider, these activities are not key to the organization, supplier is a specialist in these activities, the provision is for a certain period of time, the organization pays an agreed amount of money for this provision, and a contract between the parties is signed. Why do the organizations outsource? Simply because there are others who can do the same cheaper, faster and better, and because organizations have to engage themselves in the key activities of their business. The organization transferring the business activity is the "client", the one to which the activity is transferred and takes decisions about it is "the supplier", and all of this considered as a project.

Every organization that intends to outsource all or part of its activities should be aware of the risks it is exposed, and be able to analyze, evaluate and manage them properly. In [1] are described some risks. To mitigate or avoid those risks and succeed in the outsourcing of activities of an organization is important to use a methodology or good practices that lead the process from its inception to completion, with activities, roles and responsibilities defined. This is true for both the client and the supplier, since both have to manage the project from different points of view. As indicated in [5], "the application of a methodology, with proper management and implementation process, should ensure the success of an outsourcing project in the same way that incorrect application of this methodology can lead to failure". With the increasing of outsourcing in recent years have also appeared methodologies and good practices that seek to establish a framework that helps to perform successful outsourcing projects, their control and good management, but there is not a standard. Note that all the methodologies or good practices except eSCM-SP [15], and to a lesser extent CMMI-ACQ [4], are client-oriented. And only those sponsored by Carnegie Mellon University are more oriented to the management of IT outsourcing. In this article it is proposed a methodology for helping suppliers to manage outsourcing projects, establishing the main roles that will be involved in the project, defining the responsibilities of each and the activities to perform, since a market opportunity is identified to the project completion.

2 Review of Existing Methodologies for Outsourcing Management

In this section we review the literature about methodologies and good practices of outsourcing, with special emphasis on those focused on providing IT outsourcing services.

With the expansion of outsourcing in recent years have also appeared different methodological proposals that seek to establish a framework for carrying out successful outsourcing projects, their control and good management. As noted in [14], organizations have been using outsourcing management strategies since the early 60's of last century, but outsourcing practices have grown in complexity due to the global environment in which organizations are, and the same for the methodologies. Below are exposed some of them, indicating for each the set of activities proposed to achieve success in outsourcing projects. More of them are guidelines that the client can use to drive its projects. In general,

the activities of all these proposals, organized in more or less stages, are: strategy, planning, requirements analysis, cost analysis and service levels, market review of suppliers, evaluation of the proposals, selecting one or more suppliers, contract execution, transition to service provider, monitoring and termination of service. As shown in [17], the project's success and a good relationship between the parties involved will always depend on the establishment of a win-win relationship between them, measurable objectives and goals, mutual respect and the election of good managers. A brief description of the methodologies and best known practices is exposed:

- (Corbett, 2004)[5], proposes a methodology based in five stages: idea, evaluation, implementation, transition and management. The first three prior to the signing of the contract.

- (Garret, Sep / Oct 2003)[8], considers four stages as necessary to carry out a successful outsourcing project: risk analysis, due diligence and supplier selection, design and contract signing, and performance monitoring. The first three prior to the signing of the contract.

- (Zhu, Hsu, & Lillie, 2001)[25], also based on four stages: planning, development, implementation and monitoring. The first two prior to the signing of the contract.

- (Brown & Wilsons, 2005)[2], based on seven stages: strategy, RFP (Request For Proposals), negotiation, implementation, management, termination, maintenance and support. The first three prior to the signing of the contract.

- (Glen, July / August, 2002)[9], as the previous proposal, this methodology is based on seven stages: identification of requirements, review of suppliers, tendering, evaluation, negotiation, implementation and contract management. Only the last two are subsequent to the contract.

- (Pallesen, 2005)[20], also proposes a methodology based on seven stages: identifying needs, RFP, evaluation, selection, contract, definition of administrative tasks and closing. The last two after the signing of the contract.

- (Greaver, 1999)[10], like previous ones, proposes seven stages: planning initiatives, analyze the strategic implications, analyze costs and returns, vendor selection, negotiation of contract terms, resources transition and relationship management.

- (TRG, 2003)[24], proposes an eight-step methodology: analysis and planning, RFT (Request For Tender), proposal preparation, bid evaluation, negotiation, contract, transition and completion.

- The Australian banking group ANZ, as described in [19], proposes a methodology based on the activities of the banking sector and based on nine stages: selection of candidates for outsourcing services, feasibility study, evaluation of market environment, due diligence, contract negotiation, reporting to APRA (Australian Prudential Regulation Authority), transition of the service, contract management and performance monitoring.

In all of these methodologies the outsourcing project planning and the signing of the contract with the supplier have great importance. In fact, the signing of the contract is the milestone in the outsourcing life cycle. There are other methodologies focused on specific aspects of the outsourcing life cycle, for example [23] considers the selection of suppliers as the main activity, categorizing them into groups according to their performance and making a selection based on this classification. In addition to the above, and giving them a special consideration because they are the most widespread, there are two set of best practices developed by working groups at Carnegie Mellon University: CMMI-ADQ and eSCM:

- CMMI for Acquisition (CMMI-ACQ) ([4], which can´t be considered a methodology but a set of best practices to help client organizations of outsourcing to improve their processes. It is organized in process areas, specifically 22, as defined in [4], each area being a set of practices so that, collectively implemented, satisfy a set of goals considered important to achieve an improvement in that area. CMMI-ACQ describes four levels of maturity as the evolutionary path that an organization wishing to improve their skills acquisition processes must follow: 0-incomplete, 1-made, 2-managed and 3-defined.

- eSCM-CL (eSourcing Capability Model for The Client Organizations) [12] is totally oriented to outsourcing activities, in this case to the client's ones. It is a set of best practices to improve the capacity of organizations in the outsourcing life cycle and provide them with objective means to measure that ability. The model is constituted by a total of 95 practices that reflect the critical

capabilities that a client organization of outsourcing needs for a good management of their projects. It is based on a five-level structure which determines the way for an organization to improve their outsourcing activities, from the lowest level, based on the desire to outsource a service, to the highest, in which the organization maintains and demonstrates excellence in the management of outsourcing projects.

- eSCM-SP (eSourcing Capability Model for The Service Providers) [15] is also a set of best practices, to help organizations that provide IT outsourcing services assess and improve their capabilities in service provision, management and client relationships. The model is organized in 84 practices, each viewed from three dimensions: outsourcing life cycle, area, capacity and ability level. Regarding the life cycle are considered only four stages: preparation, initiation, delivery and completion. The main objectives for this model are to provide:

  o to the suppliers of IT services a guidance to help them improve their skills during the IT outsourcing life cycle and a standard that allows them to differentiate themselves from competitors.

  o to the clients an objective means to assess the ability of suppliers.

Of all the methodologies and best practices described before, eSCM-SP is the one that better match the needs of a service provider of IT outsourcing, In the following section it will be presented a methodology for helping outsourcing services providers that tries to address these issues.It do not intends to replace any other but it treat in a more detailed way each of these aspects mentioned and could perfectly be complemented by eSCM-SP as a model for improving the activities of the methodology and achieve excellence.

3 Proposal of a Methodology for IT Outsourcing Services Provision

The methodology proposed, taking into account how methodologies and good practices described above are organized, has five stages: preparation, due diligence, transition, service delivery and completion. These stages follow a parallel path to those explained for the client with contact points along the life cycle. While the client is deciding what to outsource, what staff and resources will be transferred, and evaluating potential suppliers, suppliers are detecting opportunities and preparing offers for the clients. There is a first point of contact in the due diligence stage, if it is necessary, in which the two parties refine the proposal and understand the way of working of the other. The most important point of contact is the signature of the contract, which establishes a formal legal basis for the commitments approved by both parties. From this moment on, the two parties share the same stages, each managing them from their own point of view: the client controlling the service and the supplier providing it. Finally, at the stage of completion it will proceed to a reverse transition, the vendor delivers the service to the client, which assumes full responsibility. In this section are first described the different stages of the methodology, then the roles and responsibilities involved in the activities, and finally a set of management issues which are important for a good service provision.

3.1 Stages

The methodology consists of five stages, two of which are prior to signing the contract and one of them, due diligence, not mandatory, depending on the type of service and the client organization. Between some of the stages there is a transfer of responsibilities among different roles that should be formalized and implemented in internal meetings, namely:

- From account manager to Offer Manager. In this case, the Offer Manager knows everything about the opportunity detected, so it is not necessary a meeting to formalize the transfer.

- From Offer Manager to Transition Manager. After the preparation phase or due diligence (if there is one) and after signing the contract.

- From Transition Manager to Service Manager.

- From Service Manager at the end of the contract, if it is not going to be extended, to the client or the new provider of outsourcing.

At these meetings, in addition to the responsibilities transfer, the work carried out is reviewed and details which are still pending are assumed by the client.

3.1.1. Preparation of the Offer

It begins when the Account Manager detects a business opportunity to provide a service. From that moment on the proposal the risks are evaluated, and the organization decides whether participate or not in the bidding. The objectives of this stage are: to involve different areas of the organization with regard to the service in the assessment of the opportunity, to analyze the risks, and to formalize the process and determine its viability. And the main activities to be performed include: to name the person responsible of the Offer Manager, to assess of the opportunity and the risks associated by the Offer Manager, and decision meeting, where taking into account the proposal and the identified risks it is decided whether to bid or not. The offer should specify what is being offered, the provision and management methods (how), people and entities involved in the service and its responsibilities (who), the global planning (when), if it is the case, references to other similar services provided by the organization, and the appropriate Annexes to help a better understanding of the terms of the offer. Once the proposal is accepted by the client, the contract is drafted, including all legal clauses and paragraphs to be agreed with the client: introduction, objectives of the service, scope of service, service organization, methodology to be followed in each stage, critical success factors, planning, economic conditions, references to similar services, other important aspects, annexes and date of review and approval. For the bid preparation it is proposed the use of support tools that facilitate decision making, such as Offer Review Document, which contains the results of the final review meeting, and Economic Assessment Tool for calculating the costs and profit margins associated with the service to be provided.

3.1.2. Due Diligence

It is a stage prior to the signing of the contract and takes place when there is a clear commitment by the client to accept the offer, it is not a mandatory stage although it is recommended in all cases to better define the service contract, to design a more accurate and real contract, and to assure the success in the service provision. Its goals are: clarify the economic aspects that have not been identified in the preparation phase and have economic impact on the offer, confirm the offer and develop the proposal and the contract with the client, review and verify the estimations, analyze client's infrastructure and the risks that may affect the service, collect of information on the transfer of staff from client to supplier, and check the estimated prices. In the case that the client does not accept the due diligence, additional guarantees should be included in the contract to mitigate risks. If the client requests to be carried out the due diligence before accepting the offer, the cost will be charged to it if it finally does not accept the offer.

3.1.3. Transition

The main objectives of this stage are taking control of the service in a systematic way and developing transformation projects requested by the client, all within the parameters of cost, time and user defined performance. For this, the Transition Manager will develop the Transition Plan, which is organized in three stages:

- Initial: definition of objectives, cost estimation, first meeting, identification of needed equipment and responsibilities, constraints and risks, identification of critical issues and development of the Transition Act.

- Planning: meeting with the Offer Manager, definition of the transition plan, scope of deliverables and detailed planning of activities and tasks.

- Implementation and control: defining the relationship model with the client, staff transfer (if necessary), communication plan, taking control of the service, third party management, knowledge transfer, technical deliverables, procedures and documentation, management and control, and completion.

The Service Quality Plan is a document containing information related to service management: scope, objectives, how it will carry out the planning, design, development and implementation of the solution, what are the supplier roles and responsibilities and what methods, tools and processes will be used to deliver a quality service.

3.1.4. Regular service provision

At this stage the main objective is a continued provision of service according to the commitments with the client. This will carry out activities like: planning and control of all activities necessary for the service, identify problems that may affect the service and take appropriate preventive actions, management of the finances associated with the service, identification and control of the necessary changes for the service, and provide training to clients and end users on the service if necessary.

3.1.5 Completion

This stage can take two paths: continuing providing the service by the same supplier or the contract resolution and the subsequent return of service to the client or a new outsourcing supplier. The Service Manager will be responsible for planning and coordinating the return of the service, ensuring that the objectives have been met according to the established indicators and the client satisfaction surveys. He will also be responsible for training the staff that will operate the service, to warn about known problems and propose appropriate changes for improvement. He should also be responsible for coordinating the transfer of resources, both human (if applicable) and material, ensuring continuity of service, and the knowledge acquired during the performance of it.

3.2　Roles

Possibly not all the proposed roles in this methodology exist in an organization and should be covered with external staff or moving personal from other areas. The main roles are:

3.2.1 Offer Manager

Participates in the preparation and due diligence stages, and that will surely be a new role for most organizations. He is responsible for the offer, all the necessary activities for preparing it and the personnel involved in it. While preparing the offer his responsibilities are: leading the working group, identify risks, calculate costs, assess the feasibility of the proposal and the necessary investments, identification and assessment of suppliers in case it was necessary to outsource an activity or a part of it and decide the participation or not in the bidding. In the due diligence stage, the tasks for which he is responsible are: leading the team, plan and supervision of all the activities and identify the basic aspects to be taken into account for this stage. As main responsible for the above activities and the people who will carry them out, the Offer Manager must have knowledge about offer preparation, the production model of the organization, the infrastructure that will support the services, or risk analysis.

3.2.2 Account Manager

It is the main interlocutor with the client and responsible for the management and development of its account. In the preparation stage is the initiator of the offer because he is responsible for identifying opportunities due to its proximity to the client's activity, it is also responsible for negotiating the economic aspects and participates in the final decision on whether or not to go to the bidding. He also participates in the completion stage and in any of the other if necessary. He must have business and negotiation skills, consistent knowledge about making offers, and the client business.

3.2.3 Technical Manager

It is the technical manager of a service line, he is also the interlocutor to the Service Manager, supports the Offer Manager in the service definition, presents technical solutions and he is responsible for

managing the external technology providers. He must have good technical knowledge, know about the model of production and operation of the organization and cost estimation.

3.2.4 Technical Team

It supports service provision and should know about the techniques, processes and tools required for it.

3.2.5 Finantial Manager

He helps the Offer Manager on the economic valuation of the service and participates in the rest of the stages as responsible for the economic management.

3.2.6 Purchasing Manager

This role will participate in the project depending on the type of service to be provided and the needs of product acquisitions or contracting external suppliers.

3.2.7 Human Resources

This role is one of those who participate in the project or not depending on the type of service to be provided, especially those involving staff transfer.

3.2.8 Transition Manager

It is responsible for the service in the transition phase and the main interface to the client. It is also responsible for directing and coordinating the teams involved in the service transition, for developing the Transition Plan, risk management at this stage, changes... Must have extensive knowledge of the production model of the organization and operation of the infrastructure that supports the services, project planning, financial management, SLAs, risk management,...

3.2.9 Technical Coordinator

He is responsible for coordinating the technical team that operates the service, interacts with the Operation Manager, produces reports, and manages assets, infrastructure and incidents. He must have extensive technical knowledge, about making offers, the pattern of production and operation of the organization, infrastructure, standards, methodologies and best practices, and incident management.

3.2.10 Service Manager

He is the main responsible at the stage of regular service provision and main interlocutor with the client. It is also responsible for directing and coordinating the teams involved, monitoring of service and compliance with SLAs, approving the billing, risk management,... Must have extensive knowledge about the model of production and operation of the organization, infrastructure, risk management, client business and reporting.

3.2.11 Operation Manager

He is the technique responsible for the operation of the service, interlocutor between the technical coordinator and the transition and Service Managers, depending on the stage, he is involved in the monitoring committees and supports decision making. He must have an expertise similar to the Service Manager.

3.3   Manamegent Support

During the different stages some activities that require specific management are carried out, and are considered in the proposed methodology by their capital importance for achieving the success in out-

sourcing projects. All of them are necessary except supplier management, which will be implemented only in the case of outsourcing to third parties of all or part of the service.

3.3.1 Risk Management

Risk management consists of anticipating the onset of the problem, assess its impact on the service and the probability of occurrence, and design a specific plan to mitigate it. It is transverse to all the stages of the project. The risks to consider are related to: the client, technical solution, financial management, staff, third parties or planning.

3.3.2 Economic Management

Objectives of economic management are basically two: make a correct cost estimation associated with service during the preparation, transition and service provision stages, and a reliable tracking of the service contribution to the supplier business.

3.3.3 Change Management

Changes may occur at any time during the life of a service and usually are changes in the agreement between the client and the provider (new components, change in scope or range). Change management helps to formalize the implementation of these changes and to make them in a controlled manner.

3.3.4 Client expectations management

Client expectations means the expected benefit an organization hopes to achieve from the service provided by the supplier while paying for it. Managing these expectations have to be done, based on satisfaction surveys, in the transition, service regular provision and completion stages.

3.3.5 Client-relationship management

It is convenient to define and approve, by both parties, a model of relationship between them to regulate the daily contact of its representatives, defining the roles and responsibilities of each party and reach the commitment to meet its obligations in order to achieve the objectives expected from the service. This model must be defined relative to the stages of transition and regular service provision differently, since each stage involves different roles and activities are also different.

3.3.6 Supplier Management

When it is necessary to subcontract all or part of the service to one or more external suppliers, a specific management of them should be performed for obtaining the best performance, getting an overview of the relationship with suppliers looking for synergies between the various services provided by them, establishing a model of relationships with external suppliers and monitor them and clearly defining the responsibilities of each party.

3.3.7 Reports

Reports must be submitted for monitoring the service and have to collect all those indicators that provide information on the status and progress of it, both qualitative and about the activities, financial control and risk management. Some of these reports will be for the client and others for the supplier organization.

4 Conclusions

Outsourcing business activities have become, in the globalized and increasingly deregulated environment of organizations today, a strategic tool for them. But outsourcing projects are increasingly com-

plex, in correspondence with the increased complexity of organizations, and this has led to the emergence of methodologies and good practices for managing outsourcing projects, some of which are a mere description of the life cycle stages of these projects and the major activities to do in each. And in most cases, these methodologies and good practices are oriented to help those client organizations that outsource their services manage their outsourcing projects. Only two of them do so from the viewpoint of the outsourcing supplier and are mainly focused on process improvement rather than being a guide in carrying out these projects from the identification of the opportunity until the end of service delivery by the supplier. This article proposes a methodology based on the existing ones that attempts to go further, not only setting the stage life cycle for a project of IT outsourcing service provision, with the activities carried out in each of them, but also the main roles involved, their responsibilities, and other management tasks required at each stage to ensure successful completion of the service.

5 Literature

1. Axelrod, C. (2004). *Outsourcing information security.* Artech House.
2. Brown, D., & Wilsons, S. (2005). The black book of outsourcing: how to manage the changes, challenges, and opportunities. Hoboken: John Wiley & Sons.
3. Chase, R., Jacobs, F., & Aquilano, R. (2004). Operations management for competitive advantage. 10th ed. Irwin / McGraw Hill.
4. CMMI PRODUCT TEAM. (2010). CMMI® for Acquisition, Version 1.3. Pittsburgh: Software Engineering Institute.
5. Corbett, M. (2004). The outsourcing revolution: why it makes sense an how to do it right. Chicago: Dearbon Trade Publishing.
6. Dominguez, L. R. (2006). The manager´s step by step guide to outsourcing. McGraw-Hill.
7. Ernst & Young´s. (2008). Outsourcing: the rise of fragmentation. Ernst & Young´s.
8. Garret, K. (Sep/Oct 2003). Four steps to successful vendor management. ABA Bank Compliance, vol. 24, nº 9 , 32-39.
9. Glen, D. (July/August, 2002). Keeping an eye on the purchasing ball. The British Journal of Administrative Management, vol. 32, 14-16.
10. Greaver, M. (1999). Strategic Outsourcing. A structured approach to outsourcing decision and initiatives. New York: American Management Association
11. Griffiths, D. (2001). The theory and practice of outsourcing. STC Proceedings. Society for Technical Communication.
12. Hefley, W., & Loesche, E. (2006). The eSourcing Capability Model for Client Organizations (eSCM-CL) v1.1. Pittsburgh: Carnegie Mellon.
13. Heywood, J. (2001). The outsourcing dilemma. London, UK: Pearson Education Limited.
14. Hurley, M., & Costa, C. (2001). The blurring boundary of the organisation. Outsourcing comes of age. Melbourne, Australia: KPMG Consulting.
15. Hyder, E., Heston, K., & Paulk, M. (2006). The eSource Capability Model for Service Providers (eSCM-SP) v2.01. Pittsburgh: Carnegie Mellon University.
16. IT Governance Institute. (2005). Governance of Outsourcing. IT Governance Institute.
17. Kakabadse, A., & Kakabadse, N. (2000). Outsourcing: a paradigm shift. Journal of Management Development, vol.19, nº 8 , 670-728.
18. Latty, R. (1999). Outsourcing: the benefits and sacrifices. Bethesda, Maryland: Solutions Engineering Corporation.
19. McMahon, P. (2008). Tendering for outsourced services in a large Australian bank: the drivers of success (Ph. D). Melbourne.
20. Pallesen, K. (2005). Seven steps to a successful outsourcing process. Recuperado el 15 de 02 de 2006, de Outsourcing Institute: http://www.outsourcinginstitute.com/oi_index/seven_steps.html
21. Power, M., Desouza, K., & Bonifaci, C. (2006). The outsourcing handbook. London, UK: Kogan Page.
22. Schniederjans, M., Schniederjans, A., & Schniederjans, J. (2005). Outsourcing and insourcing in an international context. Armont, NY: M.E. Sharpe.
23. Talluri, S., & Narasimhan, R. (2002). A methodology for strategic sourcing. European Journal for Operational Research, nº 154, 236-250.
24. TRG. (2003). The Technology Resources Group. Obtenido de http://www.tek-resources.com/outsourc.htm

25. *Zhu, H., Hsu, K., & Lillie, J. (2001). Outsourcing. A strategic move: the process and the ingredients for success. Management Decision, vol. 39, nº 5/6 , 373-378.*

6 Author CVs

Javier Sáenz

He is Computer Science Engineering. He is a titular teacher in the Escuela Universitaria de Informática at Universidad Politécnica de Madrid. He is teaching in the area of Languages, and Computer Systems, specifically in the domains of Information System and IT Governance and Management. He has participated in more than 10 projects (European and Spanish Public Administration). He is author of several national and international papers. He is coauthor of books related to IT service management and IT security. He is a member of itSMF Spain university committee. He is a member of the "Research Chair in Software Process Improvement for Spain and Latin American Region". He is a member of GICAC research group, where his main research field is about models assurance and quality certification, accreditation models and adaptation to the EHEA, focusing in the topics of project management, governance and IT management and audit.

Mercedes de la Cámara

She is a titular teacher in the Escuela Universitaria de Informática at Universidad Politécnica de Madrid. She is teaching in the area of Languages, and Computer Systems, specifically in the domains of Information System and IT Governance and Management. She has participated in more than ten projects (European and Spanish Public Administration). She is author of several national and international papers. She is coauthor of books related to IT service management and IT security. She is a member of itSMF Spain university committee. She is a member of the "Research Chair in Software Process Improvement for Spain and Latin American Region". She is a member of GICAC research group, where her main research field is about models assurance and quality certification, accreditation models and adaptation to the EHEA, focusing in the topics of project management, governance and IT management and audit.

Jose A. Calvo-Manzano

He holds a PhD in Computer Science. He is assistant professor in the Facultad de Informática at Universidad Politécnica de Madrid. He is teaching in the area of Software Engineering, specifically in the domain of software process management and improvement. He has participated in more than 20 research projects (European and Spanish Public Administration). He is author of more than 50 international papers. He is author of books related to software process improvement and software engineering topics also. He is a member of SOMEPRO research group, where his main research field is process improvement, focusing in the topics of requirements management, project management, process deployment, and solicitation and supplier agreement development. He is the head of the "Research Chair in Software Process Improvement for Spain and Latin American Region", where he has been a member of the team that has translated CMMI-DEV v1.2 and v1.3 to Spanish.

Eugenio Fernández

He holds a PhD in Computer Science. He is a titular teacher in the Escuela Técnica Superior de Ingeniería Informática at Universidad de Alcalá de Henares. He is teaching in the area of Software Engineering. He is author of more than 20 international papers. He has worked as CIO at several Spanish universities for more than ten years and director of several masters and courses related to IT management.

Quantitative benefits of model-based improvement in a SME unit

Arthur Maria do Valle[1,2], José Marcelo Almeida Prado Cestari[1,3], Edson Pinheiro[1], Eduardo Portela[1]

[1] Programa de Pós-graduação em Engenharia de Produção e Sistemas, PUCPR. Rua Imaculada Conceição, 1155 – Prado Velho
CEP 80.215-901 – Curitiba-PR
{arthur.maria, jose.cestari, e.pinheiro, eduardo.portela}@pucpr.br

[2] ISD Brasil, Av Cidade Jardim. 400,
01454-901 São Paulo-SP, Brazil
arthur@isdbrasil.com.br

[3] Sofhar Gestão & Tecnologia S.A – Rua Imaculada Conceição, 1430 – Prado Velho
CEP 80.215-901 – Curitiba-PR
josep@sofhar.com.br

Abstract

This paper presents the approach adopted by a software development unit in order to achieve the maturity level 3 of CMMI-DEV and therefore obtaining better performance. Through historical research and secondary data analysis of the organization, the paper intends to answer the following research question: "Could the adoption of maturity/best practices models bring better performance results to small and medium organizations?" Data and analysis conducted shows that, besides the creation of indicator's based management, there are some quantitative performance benefits such as: Schedule Deviation Rate, Effort Deviation Rate, Percent Late Delivery, Productivity Deviation and Internal Rework Rate.

Keywords

Performance management, CMMI, process improvement, quantitative benefits, performance indicators.

1 Introduction

There is a lot of doubt regarding the benefits of adopting a formal approach (model based) for software development projects. Researchers and academy, which defend the use of a software engineering approach, methodology, best practices, maturity models and so on, are usually criticized by companies and professionals because of the excess of work, process bureaucracy and lack of evidence correlating a maturity process and the achievement of performance. In that way, this a problem that deserve to be studied and verified, once companies are always looking for better productivity, quality, efficiency, etc.

People who are studying and contributing with the question are usually related to research institutes and/or practitioners around the world. They are releasing new models, courses, papers and best practices that help the projects to achieve better results around the world, and consolidating the idea that it is a good thing to have an organized and formal process, managed in order to obtain performance (Goldenson and Gibson, 2003)(Gibson, Goldenson and Kost, 2006).

2 Theorical background

In general ways, performance measurement can be basically divided in two periods (Gomes, Yasin, and Lisboa, 2004): the first period (known as the "traditional measurement systems") begun around 1880, and the measures were pretty much related to accounting (e. g., operations costs) and financial control. This approach suffered criticisms because the focus was given only in the financial aspects of a company. After 1980, the second period begins. The researchers realized the importance of measure other areas (besides financial), such as quality, customer satisfaction, process and intellectual capital. A large number of performance measurement systems (PMSs) have been proposed. Among the most widely cited of these PMSs are: the SMART (Cross and Lynch, 1988-1989; Lynch and Cross, 1991), the performance measurement matrix (Keegan et al., 1989), the Balanced Scorecard (Kaplan and Norton, 1992), and the integrated dynamic PMS (Ghalayini et al., 1997).

In order to create measurements to other than financial areas, especially because of the increasing IT aspects, some IT frameworks and models were created in the beginning of the 1990's, and among then, we must cite the CMM (Capability Maturity Model) and its successor, the CMMI-DEV (Capability Maturity Model Integration for Development), focused on software/systems development. According to CMMI Product Team (2010), CMMI is a maturity model for process improvement and it is a composition of best practices that address development and maintenance activities for the product lifecycle, since its inception until its deploy and maintenance. The model was created basically because of a need from the DoD (Department of Defense – USA). The DoD was dealing with suppliers that were not providing quality (on time, and on budget) software projects and the DoD begun a partnership and sponsorship with Carnegie Mellon University, located in Pittsburg. As a result of this collaboration, the Software Engineering Institute (SEI) was created in order to research and develop frameworks, models and good practices, based on the concepts of Crosby (1979), Juran (1988), Deming (1986) and Humphrey (1989). The idea was that the DoD suppliers could follow these practices and be adherent to the model, reducing the risks of poor quality of software development supply to DoD.

In one of its representations (called staged), the CMMI-DEV model defines five maturity levels, as it follows:

•   Level 1: Initial. Ad hoc and chaotic process. Usually the organization does not have a stable environment and the success of the projects depends of the "heroism" and competence of some employees. The organizations are hardly able to repeat the past success of projects.

•   Level 2: Managed. The requirements and the projects are managed. There are measurement analysis, control and planning of the activities. There is a process for managing the projects, including

the organization of the work products and its control. Management team has visibility about the status, and stakeholders involved are also managed and there is a commitment established with then.

- Level 3: Defined. Well understood, defined and formalized process for the organization. They are formally described and the use of patterns, procedures, tools and methods are institutionalized. Engineering processes are also enforced in this level.

- Level 4: Quantitatively Managed. Some processes are chosen so they can be statistically and quantitatively controlled and managed. Special causes of variation are identified and analyzed.

- Level 5: Optimizing. The processes are continually improved through incremental actions and innovations. Quantitative objectives are established and reviewed for the process improvement. Focus on analyze the common causes of variation.

Each of the maturity levels above cited contains process areas (PAs) describing practices, activities and artifacts that should be addressed in order to achieve that specific maturity level. The levels are cumulative, i. e., to achieve the Level 3, one organization must be compliant with Level 2 and 3 PAs.

According to the above, and aiming to investigate if process maturity brings better operations performance results, we describe a brief history of the effort to implement the CMMI Level 3 maturity level and the results (in terms of performance measurement) founded in an organization that was formally assessed as CMMI Level 3 compliant. Note that a CMMI appraisal is performed by an authorized company and auditor, which can officially assess maturity levels, according to the rules and procedures created by SEI, and registered in the SCAMPI-Standard CMMI® Appraisal Method for Process Improvement (2011).

The remainder of this paper is as follows: In section 3 we present our method and research protocol. In section 4 we present the organization and the organizational unit. In section 5 and 6 we present the process improvement history and approach, respectively. In section 7 the results and in section 8 conclusion and future works.

3 Methodology and Research Protocol

As a research question we defined the following: "Could the adoption of maturity/best practices models bring better performance results to small and medium organizations?" An organization, for our purposes, is "an administrative structure in which people collectively manage one or more projects or work groups as a whole, share a senior manager, and operate under the same policies" (CMMI Product Team, 2010) and, in this context, the organization is the Software Development Area.

In order to evaluate this question, a field study using secondary data was conducted. As a field study, this research has two phases, a) historical data collection and b) data analysis and report, and four activities, as can be seen in figure 1:
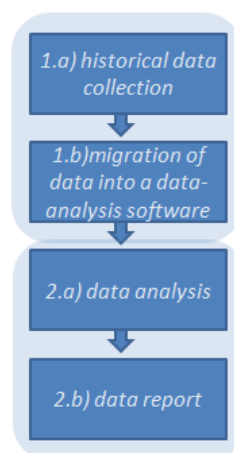
Fig. 1 – Research Methodology

Since the organization had a performance measurement repository, data collection was the gathering of historical data and migration of it into a data-analysis software tool. In terms of the analysis, as part of the organization's performance measurement repository, there are measures that can also be used to measure the progress and benefits of the process improvement program. These measures were applied to legacy software development projects as well as for new projects that used the organization's CMMI-based processes and assets. In this sense, some project-based indicators were selected and analyzed via control charts – in terms of mean and variation improvement rates: Schedule Deviation Rate (%); Effort Deviation Rate (%); Productivity Deviation (%); %Late Delivery (%); and Internal Rework Rate (%).

4 Case description: Company and Organizational Unit

Sofhar Gestão & Tecnologia S.A. is a Brazilian company located in Curitiba (capital city of Paraná state). Sofhar was founded in 1986 and since then helps its clients to achieve success in national and international markets. Sofhar has a complete structure to meet the market needs and these demands are met by providing the following offers (through services and products): Consultancy, Software Development, Infrastructure, Training and Product Sales and Licensing.

The Organization Unit is the Software Development Area composed by 15 people (at the time of the process improvement program), including the area manager and those individuals directly working with software development. Together they implement the roles and processes such EPG-Engineering Process Group, CCB-Control Change Board, QA-Quality Assurance, organizational training, process improvement, and so on, according to the needs and requirements of CMMI maturity level 3.

The organization unit implemented a balanced PMS in order to measure its processes and results during (and after) a software development project. The measures (for projects and organization) were created basically according to the structure, recommendations and relevance proposed in Neely at al. (1997), and also according to the (best) practices specified in the Measurement and Analysis PA of the CMMI Maturity Level 2.

Some of the performance measures created are represented in the following dashboard figure (Fig. 2):

Fig. 2 – Project Measures Dashboard

In the dashboard we can see, for each indicator, the related organizational goal, the measure status and associated risk category. The status are usually represented with colours, where the green stands for "ok", the yellow for "alert" and the red for "critical". In case of a "red light", some corrective action is expected in the project. All the measures were defined and are managed according to a "measure framework", containing, among other items: name of the measure, goal, unity (e. g., days, hours, etc),

formula, procedures for analyze, who measures, who collects, frequency of measurement, frequency of analyses and so on.

All of these project measures are grouped (along with other specific organizational measures) in order that the management can have a general view regarding organizational risks and performance.

5 Process Improvement History

In September/2008 the software development area began a CMMI-based process improvement program in order to enhance the quality of its software projects (and products), especially to achieve more predictability and improve indicators such as SPI (Schedule Performance Index) and CPI (Cost Performance Index). The SEI partner called ISD Brasil helped Sofhar to achieve its goals, as a consultancy company in the program.

6 Process improvement program approach

As part of the program approach, the partnership with ISD brought agility to Sofhar, especially because it was agreed between both companies that effort and schedule dedicated to process definition phase (writing processes, creating templates and putting all together) should be minimized. One of Sofhar's business goals was to achieve CMMI level 3 in about one year after the beginning of the process improvement project. In fact, the SCAMPI Class A CMMI ML3 was conducted in November, 2009 (about thirteen months after the beginning of the process improvement program).

In order to help Sofhar with this goal, ISD proposed a new approach called "ISD CMMI PME", where PME stands for small and medium business, in Portuguese.

After a (initial) SCAMPI C event (where a gap analysis was made and data about the Organizational Unit was raised) ISD took its historical information about processes (common indicators, common workflows and so on) and tailored (together with Sofhar) its set of organizational process assets considering Sofhar context and needs.

These assets are based on ISD's processes descriptions and templates that cover all CMMI level 3 process areas and were specially designed for small and medium organizations. After few interactions between de companies, process definition phase was declared finished and Sofhar was ready to use its new organizational processes and assets for software development projects. Some pilot projects were conducted and, in the sequence, another amount of projects were conducted using the processes. Meanwhile, ISD provided consultancy in order to verify Sofhar progress to CMMI ML 3 compliance. In this sense, some events, including a preparatory SCAMPI class B, were conducted.

Concerning to continuous improvement activities and culture, Sofhar assets have naturally evolved (organizational unit is using the eighteenth baseline of its process). In truth Sofhar had an enormous gain of time, effort and knowledge using (and customizing) the processes and templates delivered by ISD.

7 Results and Benefits

As stated before, in order to address our research question, comparisons of mean and variation improvement rates in the following two scenarios were performed: scenario a) all development projects before (16 projects) and after (14 projects) CMMI-based process institutionalization and scenario b) .Net development projects before (3 projects) and after (5 projects) institutionalization of CMMI-based process the organization, including also projects conducted after being formally assessed as CMMI Level 3. Although some "before/after" effects are not statistically proven yet, due to small sampling, there are some measurable benefits related to these indicators:

In the first scenario (scenario a), there are relevant improvement of mean and variation reduction for Schedule Deviation Rate (see Fig. 3), Effort Deviation Rate (see Fig. 4); %Late Delivery (see Fig. 5); Productivity Deviation (see Fig. 6); Internal Rework Rate (see Fig 7).

Fig. 3 – Schedule Deviation Rate for scenario a)

Fig. 4 – Effort Deviation Rate for scenario a)
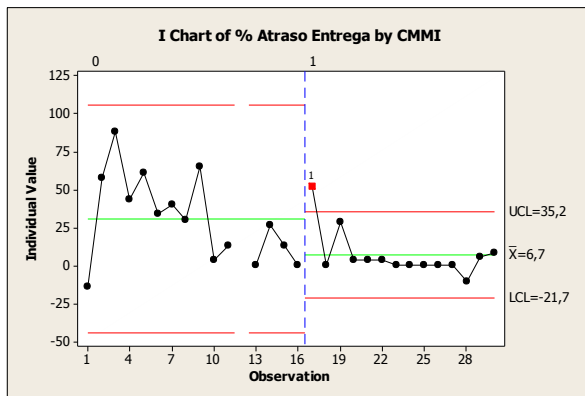
Fig. 5 – %Late Delivery for scenario a)
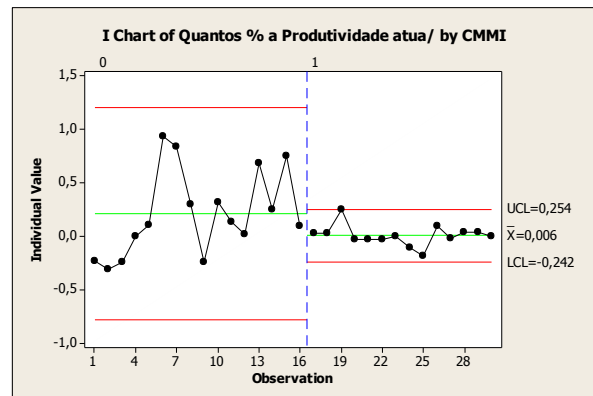
Fig. 6 – Productivity Deviation for scenario a)
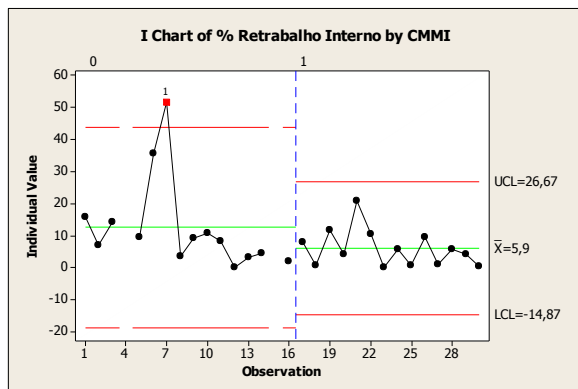
Fig. 7 – Internal Rework Rate for scenario a)

The left column of the graphs (left of the dashed line) shows project data prior to the process improvement program. It is possible to see that the variation is very high, showing difficulties related to the "repetition" and "harmonization" of the different projects conducted by the software development area. That is, regarding to a set of indicators, each project had a very different behaviour, and the predictability was very poor.

In other hand, the right pane (right of the dashed line) shows the improvement of the projects behaviour: the measures deviation has decreased a lot, the superior and inferior limits were flattened and also the mean was improved. The projects became much more predictable, and the "repetition" of values is one of the characteristics of maturity provided by CMMI Level 2 and consolidated in Level 3.

It is possible to deduce (in our case, affirm), that this repetition is due to the use of a formally described and institutionalized procedures, patterns, tools and methods.

This view of the left and right column also applies for the following scenario described (scenario b).

In scenario b), where only projects that used the same technology (i.e .Net) were analyzed, and relevant mean and variation improvement were noticed for Internal Rework Rate (see Fig. 8) and Effort Deviation Rate (see Fig. 9), Schedule Deviation Rate (see Fig. 10); %Late Delivery (see Fig. 11) and Productivity Deviation (see Fig. 12).
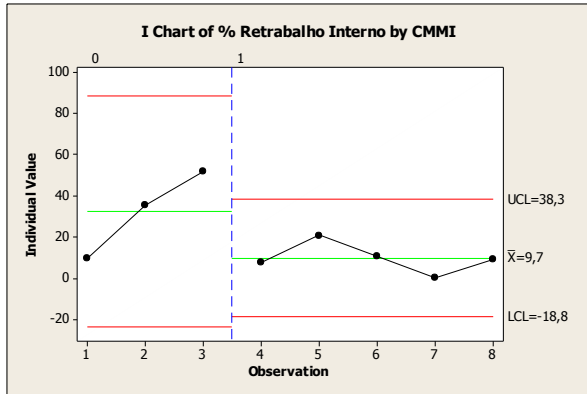
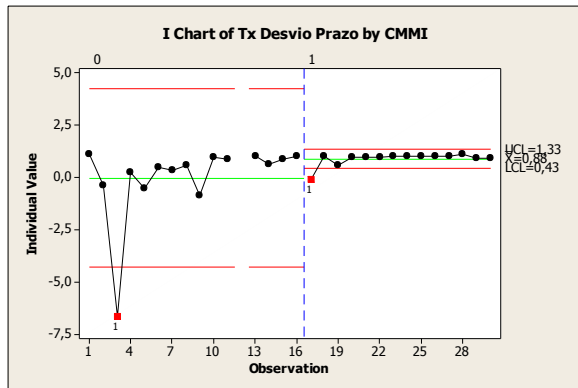**Fig. 09 – Internal Rework Rate for scenario b)**

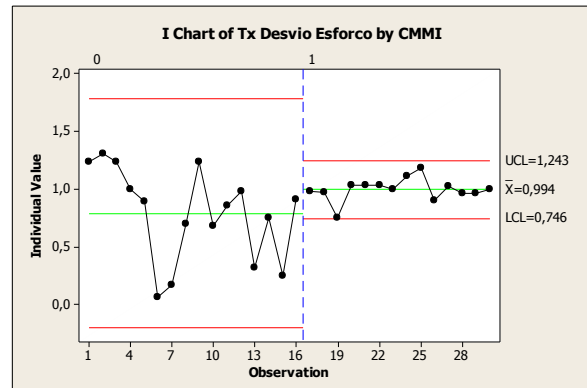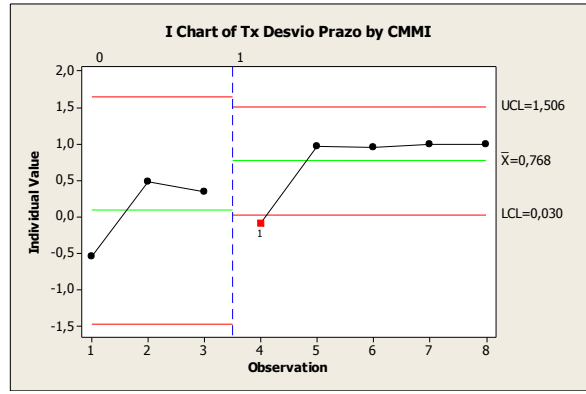**Fig. 10 – Schedule Deviation Rate for scenario b)**
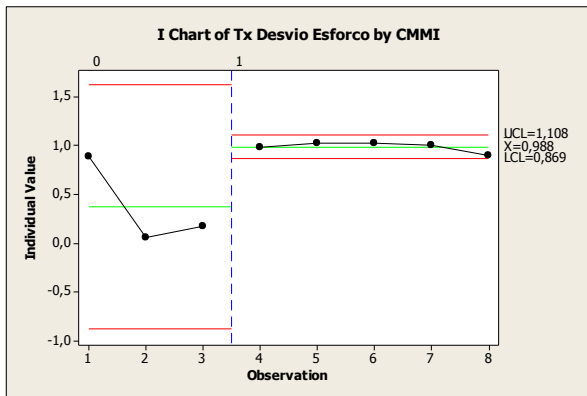
Fig. 10 – Effort Deviation Rate for scenario b)
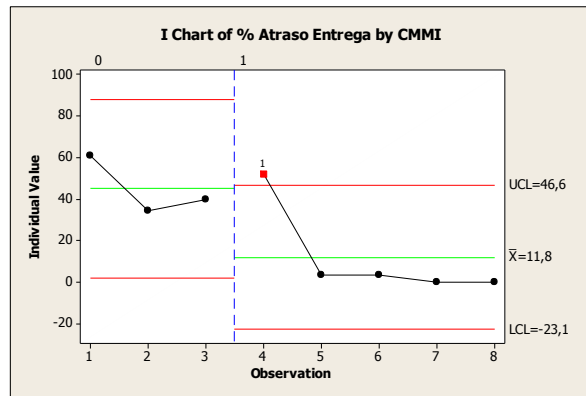
Fig. 11 – %Late Delivery for scenario b)
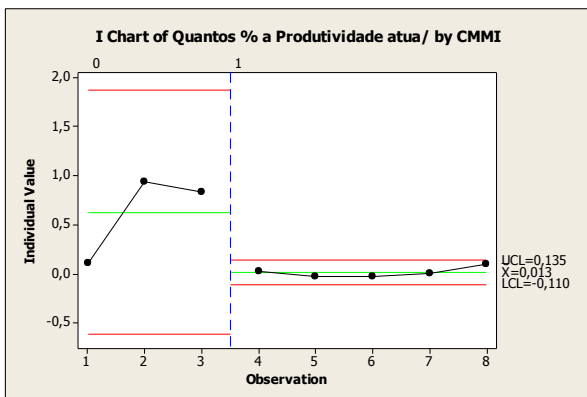
Fig. 12 – Productivity Deviation for scenario b)

8 Conclusion, limitations and future work

Based on the improvement rates of mean and variation of each selected measure in table 1 – where for the majority of indicators a improvement rate range of 25 to 100% was achieved - we can conclude that, at least for Sofhar, a more mature process, with disciplined and managed activities, and compliant to CMMI-DEV Level 3, reveals an improvement of software development projects performance results, such as quality, effort, rework, productivity and schedule.

	% Internal Rework	% Productivity Deviation	% Effort deviation	% Schedule Deviation	% Late Delivery
variation_improvement (%)	35,3	75,1	75,0	89,5	62,0
mean_improvement (%)	52,7	97,1	26,3	3679,1	78,1

Tab. 01 – Improvement rates of mean and variation

Apart from the quantitative benefits obtained, this research has some limitations, and we could mention at least three:

- Only one organization was studied, so we cannot do a generalization of benefits and results.

- The amount of project assessed is not so high.

- There are no formally ROI (Return of Investment) numbers tracked by the management.

In despite of the lack of ROI numbers, during data analysis activities, some informal interviews were conducted with the management and the company pointed at least two financial benefits:

- The reduction of rework caused a direct reduction of cost;

- The predictability (including regarding to the effort and size estimation process) gave a better budget view and price composition.

For future work, a deeper analysis on the performance measure database could be conducted in order to discover, quantify and prove cause-effect relationships that will be a basis to create and use process performance models to better estimate and quantitatively manage development projects as well as organizational performance.

9 References

Gomes, C. F., Yasin, M. M., and Lisboa, J. V. (2004), "A literature review of manufacturing performance measures and measurement in an organizational context: a framework and direction for future research", Journal of Manufacturing Technology Management, Vol. 15, No. 6, pp. 511-530.

Cross, K.F. and Lynch, R.L. (1988-1989), "The SMART way to define and sustain success", National Productivity Review, Vol. 9 No. 1, pp. 23-33.

Lynch, R.L. and Cross, K.F. (1991), Measure up: The Essential Guide to Measuring Business Performance, Mandarin, London.

Keegan, D.P., Eiler, R.G. and Jones, C.R. (1989), "Are your performance measures obsolete?", Management Accounting, Vol. 71, pp. 45-50.

Kaplan, R.S. and Norton, D.P. (1992), "The balanced scorecard: measures that drive performance", Harvard Business Review, Vol. 70 No. 1, pp. 71-9.

Ghalayini, A.M., Noble, J.S. and Crowe, T.J. (1997), "An integrated dynamic performance measurement system for improving manufacturing competitiveness", International Journal of Production Economics, Vol. 48 No. 3, pp. 207-25.

CMMI Product Team. (2010), "CMMI® for Development, Version 1.3". Technical Report – Software Engineering Institute (SEI). Retrieved from http://www.sei.cmu.edu/library/abstracts/reports/10tr033.cfm

Crosby, Philip B. (1979), "Quality Is Free: The Art of Making Quality Certain". New York: McGraw-Hill.

Deming, W. Edwards. (1986), "Out of the Crisis". Cambridge, MA: MIT Center for Advanced Engineering.

Juran, Joseph M. (1988), "Juran on Planning for Quality". New York: Macmillan.

Humphrey, Watts S. (1989), "Managing the Software Process". Reading, MA: Addison-Wesley.

Neely, A., Richards, H., Mills, J., Platts, K. and Bourne, M. (1997), "Designing performance measures: a structured approach", International Journal of Operations & Production Management, Vol. 17 No. 11, pp. 1131-1152.

Standard CMMI Appraisal Method for Process Improvement (SCAMPI) A, Version 1.3 (2011): Method Definition Document. Retrieved from http://www.sei.cmu.edu/library/abstracts/reports/11hb001.cfm

Goldenson , D. R., Gibson, D. L. (2003), "Demonstrating the Impact and Benefits of CMMI®: An Update and Preliminary Results", Carnegie Mellon University: Software Engineering Institute.

Gibson, D. L., Goldenson , D. R., Kost, K., (2006), "Performance Results of CMMI® - Based Process Improvement", Carnegie Mellon University: Software Engineering Institute.

10 Authors CVs

Arthur Maria do Valle

PhD student at Industrial and Systems Engineering Post Graduate Program, MSc in Computer Science. Consultant on projects related to performance improvement. Has published and presented papers in journals and conferences specialized in process improvement, including the 18th EuroSPI Conference in 2011 in Denmark. Official instructor of "Introduction to CMMI" course. BlackBelt SixSigma. Professor at PUCPR.

José Marcelo Almeida Prado Cestari

PhD student at Industrial and Systems Engineering Post Graduate Program, MSc in Computer Science. Certified in PMP, Lead Auditor ISO-9001:2000, ITIL and COBIT Foundation, IBM, MCTS and Information Security Foundation. Works with software engineering, including software process improvement and implementation of maturity models. Currently is the development manager of Sofhar Gestão & Tecnologia S.A. and responsable for the CMMI-DEV Level 3. Lecturer and professor related to software engineering.

Edson Pinheiro de Lima

Professor at Industrial and Systems Engineering Post Graduate Program, Pontifical Catholic University of Paraná State. Electrical Engineer by the Technological Federal University of Paraná State, Master in Electrical Engineering by the Regional University of Campinas. PhD in Industrial Engineering by the Federal University of Santa Catarina State (2001). Doctoral degree advisor.

Eduardo Alves Portela Santos

Professor at Industrial and Systems Engineering Post Graduate Program, Pontifical Catholic University of Paraná. Mechanical Engineer by the Federal University of Bahia State, Master in Mechanical Engineering by the Federal University of Santa Catarina State. PhD in Electrical Engineering by the Federal University of Santa Catarina State (2003). Doctoral degree advisor.

Impact of Standards on the Role and Application of Traceability in the Medical Device Domain

Gilbert Regan, Fergal Mc Caffery, Kevin Mc Daid, Derek Flood

{gilbert.regan,fergal.mccaffery, kevin.mcdaid, derek.flood}@dkit.ie

Abstract

Software failure in the medical device domain can lead to injury or death. Controlling this risk is fundamental to producing quality software. To produce quality software, an effective requirements and hazards traceability process is required. Hence traceability is central to medical device software development. It is also an essential requirement for regulatory approval. The necessity for traceability is reinforced by the medical device standards and guidelines. In this paper we outline how traceability is an important part of medical device software development, what standards contain reference to traceability, and which specific clauses within those standards companies should refer to when defining their traceability processes. We also summarise the findings obtained when a lightweight assessment method (Med-Trace), that we created, based upon the traceability practices within these standards, was implemented in two SME organizations.

Keywords

Med-Trace, Medical Device Standard, Software Traceability, Software Process Assessment and Improvement

1   Introduction: Background to Medical device software

Software-based medical devices are playing an increasingly important part in healthcare. Many medical devices must interface with other equipment, connect to hospital and laboratory information systems, and work in high-stress situations. The increased demands on such devices has resulted in increased software complexity and has created formidable development challenges for their manufacturers [1]. This increased complexity has resulted in the need for increased traceability and risk control measures.

In order to market their devices within a country, a medical device development company must comply with the regulatory requirements of that country [2]. Although guidance exists from regulatory bodies on what software activities must be performed, no specific method for performing these activities is outlined or enforced [3].

In 1993 the European Council released Medical Device Directive (MDD) 93/42/EC [4]. The purpose of this directive was to ensure the safety of medical devices placed on the European market. This

directive has been amended by Directives 2000/70/EC [5], 2001/104/EC [6], 2003/32/EC [7] and 2007/47/EC [8].

To this end, in the USA, the Food and Drug Administration (FDA) Center for Devices and Radiological Health (CDRH), independently from the European Council, published guidance papers which include risk based activities to be performed when using off-the-shelf software [9], during software validation [10], and for pre-market submission [11]. These documents however did not enforce any specific activity for performing the software activities, hence manufacturers could fail to comply with expected requirements.

Therefore the medical device industry decided to recognise ISO/IEC 12207 [12] (general software engineering process standard) as suitable for general medical device software development. However the Association for the Advancement of Medical Instrumentation (AAMI) identified pitfalls in ISO/IEC 12207 and produced AAMI SW68 [13] (Medical Device- Software Lifecycle Processes) which was based on ISO/IEC 12207. In 2006 a new standard AAMI/IEC 62304 [14] was released which replaced AAMI SW68.

The remainder of this paper is structured as follows; Section 2 discusses the importance of traceability in all domains culminating with the medical device domain. Section 3 identifies traceability requirements within the medical device standards. Section 4 considers the implementation and findings of Med-trace, a traceability assessment model. In Section 5 we draw our conclusions.

2 Traceability

2.1 Introduction

In engineering terms a trace is comprised of a source artifact, a target artifact and the link between them [15]. Traceability is the ability to establish and use these traces. Numerous definitions for traceability exist in the literature but one of the most popular and encompassing is:

"Requirements traceability refers to the ability to describe and follow the life of a requirement, in both a forwards and backwards direction (i.e., from its origins through its development and specification to its subsequent deployment and use, and through all periods of on-going refinement and iteration in any of these phases "[16].

In general, traceability is about understanding a design right through from the origin of the requirement to its implementation, test and maintenance. Traceability allows us to understand aspects such as to whether the customers' requirements are being met, the specific requirements that an artefact relates to, and the origins and motivation of a requirement. Traceability helps ensure that 'quality' software is developed.

2.2 Traceability in all domains

Software systems are becoming increasingly complex. Artefacts such as test cases, requirements documents, source code, design documents, bug reports etc, and the links between them are created over long periods of time by different people. Creating and maintaining these links is a difficult and expensive task. Therefore most existing software systems lack explicit traceability links between artefacts [17].

Traceability was initially used to trace requirements from their source to implementation and test, but now plays an increasing role in defect management, change management and project management. Increasingly software development is globally distributed across multiple teams and sites which makes traceability even more important [18]. As traceability provides an essential support for developing high quality software systems [19], it is vital to engage an efficient traceability process.

Traceability implementation is mandated in many software development standards and many industries, in particular the safety critical industries [20]. For example in the US the Food and Drugs

Administration states that code must be linked to requirements and test cases. Safety critical products can be dangerous because failure can result in loss of life, significant environmental damage, or major financial loss [21]. Safety critical systems must satisfy a range of functional and non-functional requirements, including safety, reliability and availability. Regulation normally requires safety critical systems are certified before entering service. This involves submission to the appropriate regulator of a safety case (a reasoned argument that safety requirements have been met and the system is acceptably safe to operate) must be made for a safety critical systems as regulation requires these systems are certified before entering service [22].

2.3 Traceability in the medical device domain

In the medical device domain, software development is a difficult and complex endeavor. Defective medical device software can cause serious injury or death. Therefore safety is a key concern [18]. In the period from 7th Feb 2011 and 7th Feb 2012 the FDA recorded 151 medical device recalls and state software as the cause [23]. The number of devices that have recently been recalled due to software and hardware problems is increasing at an alarming rate [24]. During 2009 the FDA recalled 63 medical devices because of software issues. During 2010 they recalled 107 medical devices for the same reason.

It is incumbent on medical device manufacturers to ensure, to the best of their ability, that software-based medical devices are safe and effective. Meeting this responsibility requires expertise in effective risk management practices, familiarity with software safety, and the adoption of a risk management mind-set [1]. Manufacturers must establish effective software development processes that are based on recognized engineering principles appropriate for safety critical systems. At the heart of such processes, they must incorporate traceability.

Generally there is a lack of published material regarding traceability in medical device software in addition to a lack of guidance on how to implement traceability effectively in organisations [18]. As traceability is central to the development of medical device software, a traceability assessment and improvement method called Med-Trace [20] has been developed (See section 4).

3   Medical device software standards and guidelines

Software traceability is central to medical device software development and essential for regulatory compliance. The need for this compliance is highlighted in many of the medical device software standards and guidelines. In order to understand the generic requirements for traceability and in particular the requirements for traceability within the medical device domain, a literature review of generic, safety critical and medical device domains was conducted. Detailed requirements for traceability, as expressed by the medical device standards and guidelines, are summarised in this section. Table 1, details the number of times (including section numbers for each instance) each standard identifies traceability. Table 2 provides an example of two of these references.

Table 1: Number of times (and section numbers) each standard impacts traceability

Standard Title	No.	Section Numbers
ANSI/AAMI/IEC 62304:2006 Medical device software—Software life cycle processes (2006)	**6**	5.1.1; 5.2.6; 5.7.4; 7.3.3; 8.2.4; B.6.2;
Medical Device Directive 2007/47/EC	**1**	ANNEX I 12.1
General Principles of Software Validation ; U.S. Food and Drug Administration 2002	6	3.1.2; 3.2; 5.2.2; 5.2.3; 5.2.4; 5.2.5;
Guidance for the Content of Premarket Submissions for Software Contained in	**2**	Page 11; Page 16;

Medical Devices : US FDA 2005		
Off-The-Shelf Software Use in Medical Devices: US FDA 1999	**1**	5.5.1
ISO 14971:2007(E) - Medical devices — Application of risk management to medical devices	**1**	3.5;
IEC/TR 80002-1:2009 - Medical device software Part 1: Guidance on the application of ISO 14971	**8**	3.5; 6.3; Table C; Table D
ISO 13485 (2003) Medical devices — Quality management systems — Requirements for regulatory purposes	**2**	7.5.3.2.1; 7.5.3.2.2;

Table 2: An example of Practice content relating to traceability taken from two standards as referred to in Table 1

Standard Title	Process	Practice	Practice Content
ANSI/AAMI/IEC 62304:2006 Medical device software— Software life cycle processes (2006)	Software development Process 5.0	Software development planning 5.1 Software development Plan 5.1.1	The manufacturer shall establish a software development plan which should ensure TRACEABILITY between SYSTEM requirements, software requirements, SOFTWARE SYSTEM test, and RISK CONTROL measures implemented in software;
ANSI/AAMI/IEC 62304:2006	Software configuration management process 8.0	Change control 8.2 Traceability change 7.3.3	The MANUFACTURER shall create an audit trail whereby each: a) CHANGE REQUEST; b) relevant PROBLEM REPORT; and c) approval of the CHANGE REQUEST can be traced. [Class A, B, C]

Failure in medical device software can have fatal consequences. The gravity of these consequences is highlighted in the medical device standards through reiteration of the necessity to control risks. Traceability can control risk. For example the **General Principles of Software Validation (GPSV) [10]** states that a software requirements traceability analysis should be conducted to trace software requirements to (and from) system requirements and to risk analysis results. Moreover **ISO 14971:2007 [25]** requires the manufacturer to establish and maintain a risk control file which shall provide traceability for each identified hazard to a)risk analysis, b)risk evaluation, c)implementation and verification of risk control measures and d)the assessment of the acceptability of any residual risks. The documentation of risk control measures is emphasised by **ANSI/AAMI/IEC 62304 [14]**which directs the manufacturer to document traceability of the software hazards: from hazard situation to software item; from the software item to the specific software cause; from the software cause to the risk control measure; and from the risk control measure to the verification of the risk control measure. The imperative for risk control is further called for in **Off-The-Shelf Software Use in Medical Devices [9], Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices [11]** and **IEC/TR 80002-1 [26]** .

There is considerable variance in the level of traceability detail required within the standards. Some of the standards provide very little detail as to between which stages of the Software Development Life

Cycle (SDLC) traceability should be provided e.g. **ISO 13485** requires an organization to establish documented procedures for traceability and that such procedures shall define the extent of product traceability and the records required. However other standards provide a greater level of required traceability detail such as **GPSV** which requires traceability from system requirements to software requirements and through each stage of the SDLC including design, code (including modules and functions) and test (traceability from test to detail design, high level design and to software requirements). Moreover the **Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices** state that explicit traceability must exist among requirements, specifications, identified hazards and mitigations and among verification and validation testing.

The ability to trace change is good practice for software development in general and of necessity for medical device software development. The necessity for change management is emphasised in **ANSI/AAMI/IEC 62304** when it states that the manufacturer shall create an audit trail whereby each a)Change request b)Problem report and c)Approval of change request, can be traced. It further requires that approved change requests are made traceable to the actual modification and verification of the software.

4 Med-Trace assessments and findings

4.1 Development of the Med-Trace Assessment Method

Due to the safety critical nature of medical device software, a company must meet 'country specific' regulatory requirements in order to market their product in that country. An effective traceability process is a crucial requirement to achieving regulatory compliance. Due to a lack of specific guidance within the medical device standards and documentation, achieving an effective traceability process is problematic, resulting in many medical device companies engaging inefficient traceability processes [20]. Consequently, a method (known as Med-Trace [20]) of assisting medical device software companies to improve their traceability processes and to to adhere to the traceability aspects of the medical device software standards (as detailed in section 3) was developed. Med-Trace is a lightweight software traceability process assessment and improvement method for the medical device industry. Med-Trace is based on traceability best practices emanating from software engineering process models (CMMI_R, ISO/IEC 15504-5), software engineering traceability literature and medical device software standards and guidelines i.e. the traceability practices that are expressed in table 1.

4.2 Med-Trace Implementation and observations

This section discusses how the Med-Trace assessment method was implemented in two medical device organizations and the resulting observations. The objectives of the case studies were to demonstrate how Med-Trace could be used to assess the current status of the software traceability processes within similar organisations and to discover the main problems that medical device software development organizations face in terms of traceability.
Med-Trace was implemented in two Small to Medium Sized (SME) medical device organisations. Both organisations developed electronic based medical devices that require compliance with both the FDA and the MDD. One organisation was based in Ireland while the other was based in the UK.
From the Med-Trace assessment the following observations were made across both organisations:

- A member of management was responsible for implementing traceability and its importance in medical device software development was recognised and understood.
- Tracing requirements and managing risk was recognised as difficult and complex.
- There is a lack of detailed guidance on how to implement traceability.
- Their process for software development with regard to traceability needed to be improved and formulised.
- The requirement for relevant training and the ability to record and leverage best practice with regard to traceability also emerged.
- The need for automated tools to manage traceability was recognised as was the serious limitation of using manual tools.

- Financial constraints needed to be considered when adapting automated tools.

Both organisations considered Med-Trace to be worthwhile and very relevant and appreciated the fact the Med-Trace is lightweight. The findings report addressed key areas where improvements were required and both organisations agreed to adapt the resultant traceability process improvement plan and agreed to be reassessed.

5   Conclusions and Future Work

An effective traceability process is essential when developing medical device software due to its safety critical nature. The requirement for effective traceability is mandated by the medical device standards and guidelines and its importance is evident from the number of times traceability is referred to in these standards and guidelines. However, the implementation of an effective traceability process is recognised as difficult and complex.

While effective traceability is mandated by the standards and its necessity was understood by the two organisations who participated in the Med-Trace assessment, there is a lack of detailed guidance in how to best implement an effective traceability process within the medical device software domain. There currently are a challenging number of standards governing medical device software development and to determine the exact traceability requirement from each of these standards can be time consuming. Med-Trace addresses these challenges by providing a light weight assessment method which may be used to diagnose an organization's strengths and weaknesses in relation to traceability in their software development processes.

The limitations (slow, tedious and prone to error) of using manual traceability tools such as Excel is an issue that needs addressing. Automated tools mitigate these limitations to some extent, however there are concerns around these tools such as cost, missing traces, needless traces, training, and the fact that tools require validation in their own right. Automated tools alone don't provide for accountability and so human intervention in safety critical domains such as medical device software development is necessary. The limitations of existing automated traceability tools imply the need for further development of effective tools for SME organisations operating in the safety critical domain.

To-date, Med-Trace has been applied in two SME organisations and has been well received. It is envisaged that Med-Trace will continue to be refined based on ongoing research and feedback from future assessments. Future plans include a tool to automate Med-Trace with the objective of facilitating its national and international roll out and to encourage its wider use.

Acknowledgement

6   References

1.   Rakitin, S.R., *Coping with Defective Software in Medical Devices.* Computer, 2006. **39**(4): p. 40-45.

2.   Burton, J., F. McCaffery, and I. Richardson, *A risk management capability model for use in medical device companies*, in *Proceedings of the 2006 international workshop on Software quality.* 2006, ACM: Shanghai, China. p. 3-8.

3.   Caffery, F.M., A. Dorling, and V. Casey, *Medi SPICE: An update*, in *International Conference on Software Process Improvement and Capability Determinations (SPICE).* 2010: Pisa, Italy. p. 195-198.

4.   Council, E., *COUNCIL DIRECTIVE 93/42/EEC concerning medical devices*, in *Official Journal of The European Communities.* 1993: Luxembourg

5.   Council, E., *COUNCIL DIRECTIVE 2000/70/EC (Ammendment)*, in *Official Journal of the European Union.* 2000: Luxembourg.

6.   Council, E., *COUNCIL DIRECTIVE 2001/104/EC (Ammendment)*, in *Official Journal of the European Union*. 2001: Luxembourg.
7.   Council, E., *COUNCIL DIRECTIVE 2003/32/EC (Ammendment)*, in *Official Journal of the European Union*. 2003: Luxembourg.
8.   Council, E., *COUNCIL DIRECTIVE 2007/47/EC (Ammendment)*, in *Official Journal of the European Union*. 2007: Luxembourg.
9.   FDA, *Off-The-Shelf Software Use in Medical Devices; Guidance for Industry, FDA Reviewers and Compliance*. 1999, CDRH: Rockville.
10.   FDA, *General Principles of Software Validation; Final Guidance for Industry and FDA Staff*. 2002, CDRH: Rockville.
11.   FDA, *Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices*. 2005, CDRH: Rockville.
12.   ISO/IEC, *12207:1995 Systems and software engineering — Software life cycle processes*. 1995, ISO: Geneva, Switzerland.
13.   AAMI, *AAMI SW68 Medical device software - Software life cycle processes*. 2001.
14.   ANSI/AAMI/IEC, *62304:2006 Medical device software—Software life cycle processes*. 2006, AAMI: Arlington, VA.
15.   Gotel, O. and P. Mader, *Acquiring Tool Support for Traceability*, in *Software and Systems Traceability*. 2012, Springer.
16.   Gotel, O.C.Z. and C.W. Finkelstein. *An analysis of the requirements traceability problem*. in *Requirements Engineering, 1994., Proceedings of the First International Conference on*. 1994.
17.   Gotel, O., J. Cleland-Huang, and A. Zisman, *Software and Systems Traceability*, ed. J. Cleland-Huang, O. Gotel, and A. Zisman. 2012, London Dordrecht Heidelberg New York: Springer.
18.   McCaffery, F., et al., *Medical Device Software Traceability*, in *Software and Systems Traceability*, O.G.e. al, Editor. 2012, Springer.
19.   Espinoza, A. and J. Garbajosa. *A Proposal for Defining a Set of Basic Items for Project-Specific Traceability Methodologies*. in *Software Engineering Workshop, 2008. SEW '08. 32nd Annual IEEE*. 2008. Madrid.
20.   Casey, V. and F.M. Caffery, *Med-Trace: Traceability Assessment Method for Medical Device Software Development*, in *EuroSPI 2011*. 2011: Denmark.
21.   Kannenberg, A. and D.H. Saiedian, *Why Software Requirements Traceability Remains a Challenge*. CrossTalk The Journal of Defense Software Engineering, July/Aug 2009.
22.   Mason, P., *On Traceability for Safety Critical Systems Engineering*, in *Proceedings of the 12th Asia-Pacific Software Engineering Conference*. 2005, IEEE Computer Society. p. 272-282.
23.   FDA. *Medical & Radiation Emitting Device Recalls*. 2012 [cited 2012 07 Feb]; Available from: http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfRES/res.cfm.
24.   Lee, I., et al., *High-Confidence Medical Device Software and Systems*, in *Computer Society*. 2006, IEEE Computer Society. p. 33-38.
25.   ISO, *ISO 14971:2007 Medical devices — Application of risk management to medical devices*. 2007, ISO: Switzerland.
26.   IEC, *IEC/TR 80002-1:2009 Medical Device Software Part 1: Guidance on the application of ISO 14971 to medical device software*. 2009, ISO: Switzerland.

7 Author CVs

Gilbert Regan

Gilbert Regan is a PhD student at Dundalk Institute of Technology. He is a member of the Regulated Software Research Group in Dundalk Institute of Technology and a member of Lero. His main research subject is Software Process Improvement with particular focus on Traceability within Medical Device Software.

Fergal Mc Caffery

Dr Fergal Mc Caffery is a Lecturer with Dundalk Institute of Technology. He is the leader of the Regulated Software Research Group in Dundalk Institute of Technology and a member of Lero. He has been awarded Science Foundation Ireland funding through the Stokes Lectureship, Principal Investigator and CSET funding Programmes to research the area of software process improvement for the medical device domain. Additionally, he has received EU FP7 and Enterprise Ireland Commercialisation research funding to improve the effectiveness of embedded software development environments for the medical device industry

Derek Flood

Dr. Derek Flood is a post doctoral research fellow with the Regulated Software Research Group at Dundalk Institute of Technology (DkIT). Derek has recently worked as a post doctoral research assistant at Oxford Brookes University on the PACMAD project, examining the usability issues associated with mobile applications. Through this work Derek has published at a number of international conferences such as the IADIS IHCI and INTERACT. His current research interests include Software process improvement for Medical Devices, Spreadsheet risk, Usability of mobile applications, and the psychological impact of mobile applications on end users.

Kevin McDaid

Kevin is a lecturer in Computing at Dundalk Institute of Technology. He leads research in the area of Spreadsheet Engineering in the Software Technology Research Centre. His background is in Mathematics and Statistics and his research focus on the application of statistical methods to decision problems in software development.

Increasing software systems quality by changing the development processes

Andressa Ianzen[1], Everson Carlos Mauda[2], Marco Antônio Paludo[3], Sheila Reinehr[4], Andreia Malucelli[5]

[1, 2, 3, 4, 5] Programa de Pós-Graduação em Informática (PPGIa)

Pontifícia Universidade Católica do Paraná (PUCPR)

Curitiba – PR – Brasil

[1,3, 5]{aianzen,paludo, malu}@ppgia.pucpr.br, [2]mauda@mauda.com.br,

[4]sheila.reinehr@pucpr.br

Abstract

In order to increase the quality of systems developed by the IT department of a financial company, the workflow of one development team has changed several times until it has stabilized. This paper presents the steps to get this improvement, some perceptions of the team about the process evolution and some problems that had to be solved. After three attempts, the final process, established as a workflow, is being used in this study team for over a year, with success. In addition to achieve improved quality systems, which was the main objective of this initiative, other gains were noticed considering the employees, now with increased motivation and satisfaction. This experience report concludes that any change that impacts the activities of teams will only be accepted and will achieve its goals if the involved people are comfortable with the changes and realize they will always be able to do their best and have opportunities to grow in their careers.

Keywords

Software Process Improvement, Systems development, Software Engineering, Software Quality Improvement

1 Introduction

The software industry is relatively new if compared to other industries, but is continuously evolving. Kruchten, in [1], states that building and maintaining software is difficult and building quality software in a repeatable and predictable way is even more difficult. So, it is important to define a robust software process that produces quality software. Furthermore, it is important to deal with human interaction to ensure software quality too [2]. Humphrey, as presented in [3], justifies the need to strongly consider this aspect because the most important asset of an organization that produces software is talented people [3].

Therefore this experience report describes incremental changes made in the process of an Information Technology (IT) team of a major finance company, aiming to improve the quality of software systems developed, and also to increase the satisfaction and motivation of the teams.

2 Context and workflow

This experience report takes place within one of the teams of a financial company, which has more than ten thousand employees working directly with IT, placed in several centers around the world. The addressed team, like all others, follows the development method defined by the company, but the way each team works with the method can be tailored to get more adherent to their characteristics.

Up to the year 2010 every software project had one responsible person, called "project focal", for monitoring the whole project, since de beginning up to the end, imposing to the project focal a major responsibility. That was one of the critical aspects of that workflow. The projects at the company uses object-orientation and consist basically of the following phases: elicitation and definition of requirements, analysis, design, construction, unit test, system test, user acceptance test and deployment.

Another critical aspect of this workflow was when the outsourcing approach was used, where activities like analysis, design, construction and tests (unit and system) were executed by temporary employees for specific projects. In these cases, the project focal should orient the execution of each phase following the organizational standards and, mainly, evaluate the product quality at the end of each phase. It is important to mention that besides the previous activities, the project focal was also responsible for managing project schedule, budget, available resources, and effort among other items related to managing a project.

2.1 Most critical activities

Among all activities of the project, there are two that are considered the most critical and important, due to the final product quality: requirements review and system architecture definition.

Requirements review: the requirements of a new system are not elicited by the development team because there is another team responsible to contact the customers and elicit the requirements, as well as business needs, in order to conceive the use cases. The use cases received by the development team are validated by the project focal, who analyzes each document to verify if the scope is correct, and to search for failures. These validations should happen because the use case is considered as a kind of contract with the companies that outsource the development of a system.

One characteristic of the process is that the elaboration of the use cases can be outsourced, where another company can be responsible for transforming the elicited requirements into use cases. In many situations, the use cases received to review have poor quality, with few business rules descriptions and with ambiguous and confusing definitions.

System Technical Solution Definition: this activity defines the required technical solution of a new

project. The project focal have to define the architecture to be used in the project, according to the involved teams, defining software components that can be used and following the organizational guidelines of acceptance and security. Another activity is the analysis of the existing hardware structure, verifying if it will be able to handle the access demands of the new system, and also verifies the security standards that have to be followed to avoid architectural failures.

Considering all these activities, the project focal must have good technical experience with the systems and with the standard architectures in order to execute this phase with quality. It is also required that the project focal has to be always updated, considering the processes, architectures and business rules, in order to get a project with the best solution, adherent to the requirements and with correct budget and schedule.

3 Pilot project – the new process

The new workflow begun to be defined earlier in the year 2010, mainly based on the problems found in the use case documentation due to the lack of quality. At that time, the team had ten employees and five analysts were with the following responsibilities:

- Five analysts with the role of project focal, working in all activities previously described;

- Two person were responsible for activities in the production environment, as the correction of errors found by users, visual changes in the projects and other activities to improve the systems;

- Three people were responsible for activities of migration of outdated technology or that would lose the support contracts and also other projects where the team's participation was punctual and represented little effort;

Considering this scenario, the team has identified some problems that were occurring with some frequency:

- the use case quality was very low;

- the artifacts developed by the outsourcing companies were not fully reviewed, because there was not enough time available for it;

- the defect rate found in the user acceptance test was very high, well above the desirable limit set by the company: the number of defects found in the user acceptance test must be equal to or less than half the number of defects found in integration test to be considered a normal rate.

The defect rate found after the deployment of projects in production was also very high. There is an indicator that measures the effectiveness of the tests. This indicator is the result of the division between the sum of the total number of defects found before deployment and the same sum plus the number of defects found until 30 days after deployment. The result should be equal to or less than 0.5 to be acceptable, but the results were always above this value, reaching 2.5.

Considering the issues identified, the team has decided to split the members to work into "cells", as well as have discussed the composition and the activities of each cell. The first version of the new workflow was conceived at the end of April 2010. In this first version, the team was divided into four cells: Project Cell (2 participants), Analysis Cell (1 participant), Design/Test Cell (2 participants), Build/Production Cell (5 participants). In each of these cells there should be a project focal for each project, responsible for the activities of the cell, as presented in section 3.1 of this report.

3.1   The cells

The definition of who would be assigned to which cell was done considering the skills and qualities constantly perceived by other teammates. These definitions intended to improve the workflow and increase the chance of succeeding, but not all members were satisfied with this division. Some considered that their skills would not be well used and others would like to do different activities, looking

for new opportunities and knowledge to grow in their careers.

The process presented in Figure 1 shows the activities of the Project cell. The artifacts generated by each cell are used as input to the next one, which is responsible for validating and complementing if necessary. Some activities are not in sequence, but should be executed throughout the project development.
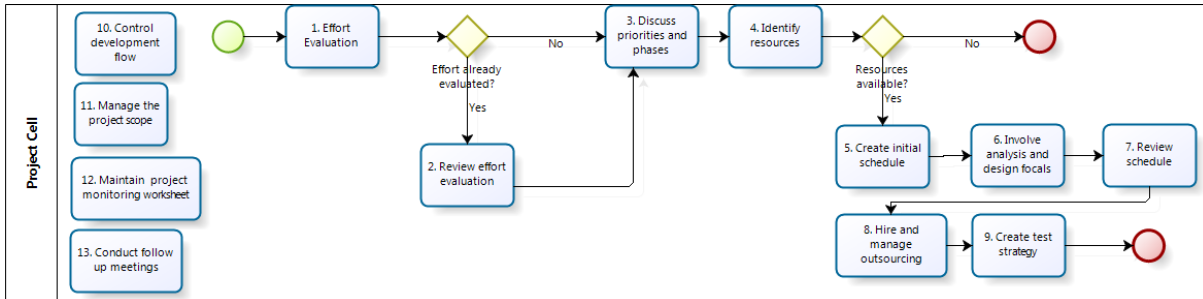
Figure 1: Project cell activities

The process presented in Figure 2 shows the activities of the Analysis cell, which mainly consist of requirements and analysis documentation activities. The main objective of the Analysis cell was to ensure that use cases documents were validated in order to have the highest possible quality.
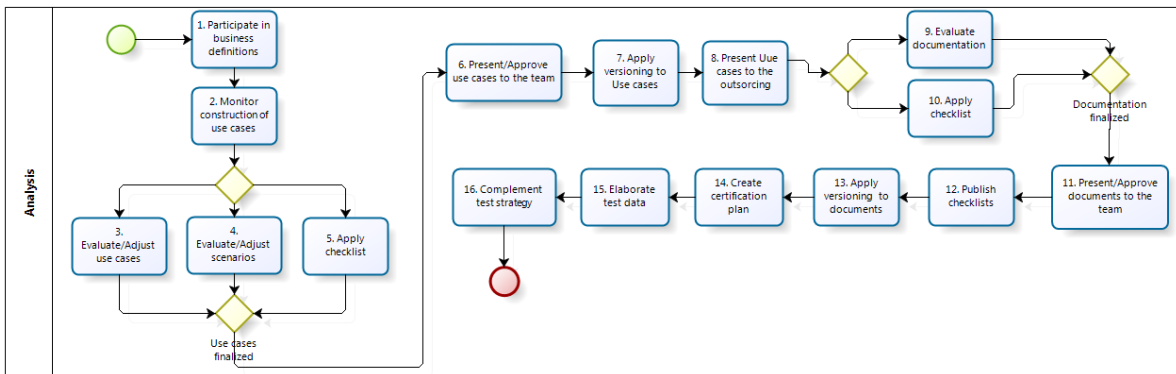
Figure 2: Analysis cell activities

The process presented in Figure 3 shows the activities of the Design/Test cell. One of the main activities of this cell was to provide technical support for Analysis cell during the use cases specification, removing doubts and solving issues related to specification items that could not be constructed or would require adjustments in the specifications.
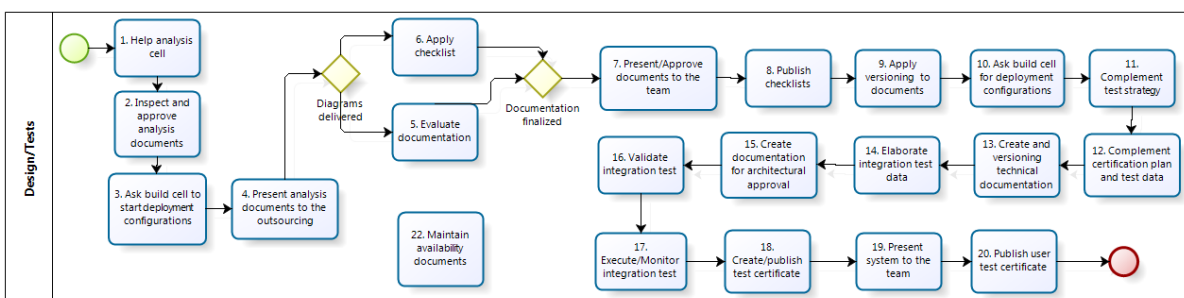
Figure 3: Design/Tests cell activities

The process presented in Figure 4 shows the activities of the Build/Production cell. The main responsibility of this cell was to ensure the quality of the source code produced, verifying its effectiveness and if it conforms to standards. This cell had a particular characteristic, as it was split into two parts. The first part focused on the Build, which takes care of new projects. The second part deals with pro-

jects that are already deployed in production and are already stabled. A project is considered stable just after a period of 30 days with no errors when executing in the production environment.
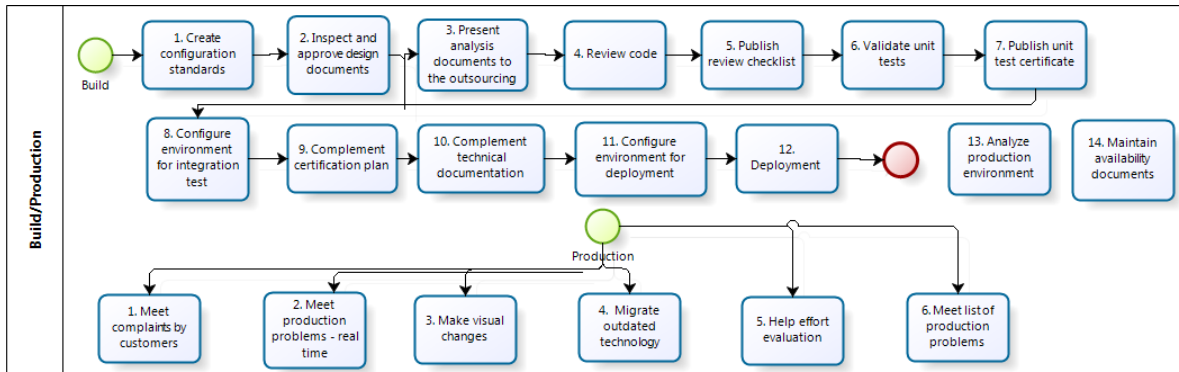
Figure 4: Build/Production cell activities

The cells and their activities were defined in order to create specialized teams to act in each phase of the project lifecycle. Another important issue was the validation of artifacts made by team members from two different cells. For example, the analysis focal validated the use cases. On the other hand, the design focal performed another validation before accepting these artifacts as input to the cell. The same validation should occur in all cells.

3.2   Adaptation for the new workflow

When defining the new workflow, it was necessary to adapt the projects that were under development. The projects that were in build phase or previous phases would be adapted to the new workflow. The projects that were in the test phase would remain with the current project focal until the deployment to production. Besides these projects, all the new ones would be developed using the new workflow. After about two months, all projects were already being developed following this new workflow.

After a few months using the new workflow, some problems began to appear mainly in the execution of Design/Tests and Build/Production cells: the configurations that were required for installing the system in the test environments should be requested by the Design/Tests cell and performed by the Build cell, but the Build cell could not execute all required activities; the members of the Build/Production cell had failed to divide the time between of the Project's demands and of the Production environment.

In September 2010, the activities were reviewed and the Build/Production cell was divided. The Build part was integrated with the Design/Test cell, creating two new cells, the Design/Test/Build cell and the Production cell. At the end of December 2010, the entire team discussed the issues of the new workflow, as following presented:

- Project Cell: Problems of lack of communication of this cell with the others cells, like schedule or scope was changing but not communicated. The Project cell could not get a clear view of the cells allocation to define the schedule of projects correctly;

- Analysis Cell: The members of this cell had all their time reviewing the use cases documents, because the schedule did not allow the execution of all the other activities. They were tired of this repetitive task and not being involved on the business decisions made by the Project cell;

- The new Design/Test/Build cell had many activities and was overloaded;

- The cell members did not feel part of the project. The perception of everyone was that all projects belonged to the Project cell and the other cells worked for them;

- The planned schedule of each cell did not allow the validation of artifacts, which should be performed by project focals from two different cells; Without the validation of the use cases documents by the cell that would guide its construction, some decisions could not be imple-

mented, creating rework;

Analyzing these problems, it became clear that most of them were related to communication and lack of visibility of team availability, due to the number of people involved in the project. Thus, the workflow was changed again, creating a new cell by the merging of the Project and Analysis cells, in a way that each project would have only two project focals. One responsible for the project managing and analyze the business point of view (Project/Analysis cell), and the other responsible for technical activities and system's production deployment (old Design/Test/Build cell).

4 The second workflow

The union of the cells had generated the need to review the proposed activities. This revision was made by removing the activities and actions that were not performed due to lack of time. Other unspecified activities were being performed but were not listed in the previous workflow, and were included in this new revised version. The new cells activities are presented in figure 5 and figure 6. The Production activities remain the same that were presented in Figure 4.
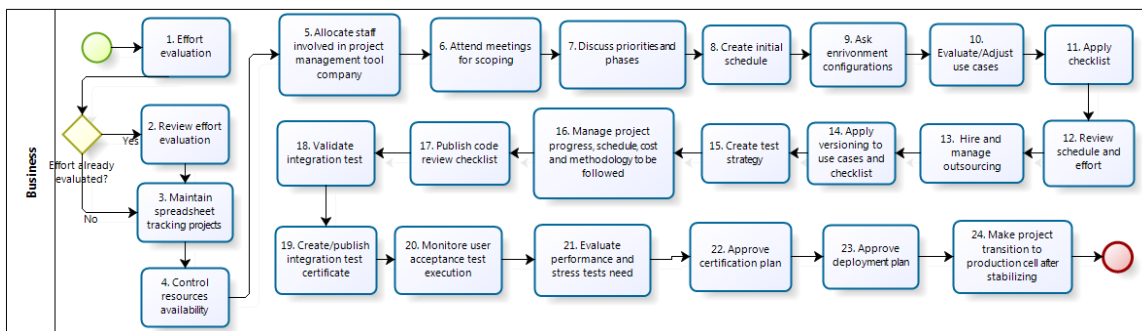
Figure 5: Business activities
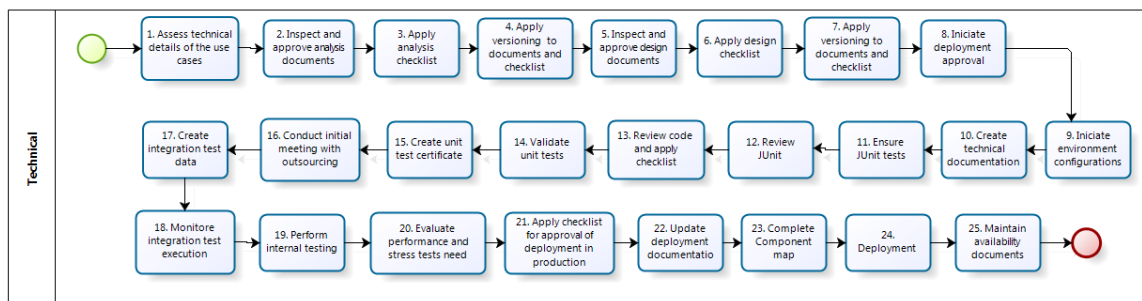
Figure 6: Technical activities

5 Results Discussion and Conclusion

The first attempt to change the workflow caused the impression that each cell was responsible only by their activities, and as soon as the activities were finished, the cell was no more compromised with the project. This was not the original intention, and it changed on the second workflow. Many meetings were made to change this vision and to make clear that all are responsible for the project development, each with their responsibilities, but always together and working to reach a common goal.

This last workflow started to be executed on March 2011 and is used by the team until the present days. The members are satisfied with their responsibilities and opportunities to grow with each new project. The main considerations that can be evidenced are listed below:

- The systems are being deployed in production with superior quality of compared to those deployed using the previous workflow, considering the results of the indicator that measures the effectiveness of the tests and the lower number of new production defects;

- The cell members are allocated to projects according to their expertise, but always with growth opportunities, considering that the teams are not always the same and the challenges to be faced during the execution of the projects are different each time;

- Both cells feel being part of the project, each with their responsibilities;

- The quality improvement in the use cases documents remained, because the previous analysis cell worked with the team responsible for the construction of these documents, and showed the common problems, guiding how to build better use cases;

- The integration tests are more complete, making the acceptance user tests generating less defects;

- The communication problems no longer exist, because the technical project focal is involved from the beginning of the project, even when it is not the moment to start the activities;

- The schedules are defined based on the actual resource availability;

- There are team members with focus exclusively on production environment.

The only difficulty faced nowadays, considering the last workflow, is the adaptation period for new employees to get integrated to the actual teams, considering that this workflow is not known by the other teams of the company. The new workflow has achieved the initial objective of promoting satisfaction and motivation in employees, as well as increased the quality of systems developed. More and more, new standardizations are being created, reducing the time to perform current activities and freeing up time to perform validations that will increase the quality of systems.

This experience report has demonstrated that, in order to propose changes in a workflow of a team, it is necessary to consider that the real objective will only be achieved if the involved people are comfortable with the changes and realize that they will always have opportunities for growing in their careers. Otherwise, the change will not bring positive results and may even frustrate the professionals and decrease productivity.

6 Literature

1. P. Kruchten. The Rational Unified Process - An Introduction. 2 ed. Reading, MA: Addison-Wesley Publishing Company, 2000. 298p.

2. K. Nakamura, M. Sugawara, M. Toyama. Defect prevention activities and tools. Communications, ICC'91, Conference Record. IEEE International Conference on, 1991, v. 1, p. 360-363

3. S. W; Humphrey. A discipline for Software Engineering. Reading, MA Addison-Wesley Publishing Company, 1995. 789p.

7 Author CVs

Andressa Ianzen
Has a Bachelor's degree in Information Systems and is Master's student in Informatics at the Pontifícia Universidade Católica do Paraná. Is a system analyst at HSBC Bank Brasil S.A. Has experience in computer science, software engineering, systems development, and processes.

Everson Carlos Mauda
Has a Bachelor's degree in computer science and is Master's student in Informatics at Pontifícia Universidade Católica do Paraná. Has experience in computer science, software engineering, reuse, component certification, and object-oriented languages.

Marco Antônio Paludo
Has a Bachelor's degree in computer science and Master's in Informatics at Pontifícia Universidade Católica do Paraná. Has experience in computer science, software engineering, components, object-orientation, and processes.

Sheila Reinehr
Has a Bachelor's degree in mechanical engineering, Master's in Informatics and Doctorate in Engineering. Professor at Pontifícia Universidade Católica do Paraná. Has experience in computer science, software engineering, software process improvement, software quality, project management, software product lines, and metrics.

Andreia Malucelli
Has a Bachelor's degree in informatics, Master's in electrical engineering and Doctorate in electrical and computing engineering. Professor at Pontifícia Universidade Católica do Paraná. Has experience in computer science, software engineering, artificial intelligence, organizational learning, ontologies, multiagent systems and healthcare information systems.

Tailoring Software Process With Argument

Masao Ito
Nil Software Corp.
2-17-7, Kinuta, Setagaya, 1570073, Japan
nil@nil.co.jp

Koichi Kishida
Software Research Associates
4-1, Kioi-cho, Chiyoda, 1028578, Japan
k2@sra.co.jp

Abstract

In this paper we propose the method to clarify the relationship between the process and the product. In the activity of Software Process Improvement (SPI) we try to improve our organizational competency to produce the better product effectively by rethinking our process. Here we can consider the relationship between the process and product. In it we define a process with artefacts, as products. But the general process model and in-out product definition cannot afford to the real process correctly, so we usually do the tailoring the model of process. In this case we keep the reasonable relationship between the process and the product, but it is the difficult work to do. When a process model says that we need the review records as the output of 'review' process, we have to decide how we do review or what kind of review process is appropriate for our project.

In this paper, we propose the new way to tailoring of the process and the product for a project with the rationale why those are right. To do this, we use the Goal Structuring Notation (GSN), which is broadly used in the system safety field. By our method we can get the capability to instantiate the process from a process model and validate the process-product relationship.

Keywords

Software Process, Process Tailoring, GSN, Tools, Process-Product Relationship

1 Introduction

The aim of the SPI is the organizational capability improvements to build the product effectively and efficiently that satisfies the requirement. When we start a project, we define a process instance that has many process elements with input and output artefacts. This instantiation of model process is difficult because we have to consider the many parameters like resources (cost, personnel, time) and the characteristics of a project. Usually the macroscopic view of model process is very simple, but we consider many factors in a process instance because it consists of small elements. That is, it is the difference between abstract and concrete. For example, take a look at the relationship between process element "test" and product "test procedure/test results". In case of the embedded system design of automobile, we have to prepare the cross-platform development environment and/or real environment and check the resources (memory, cpu time) carefully. But when we develop the WEB system, it is important to test the transaction execution and UI behaviour. The contents of test process and product (artefacts) are different each other. This is the difficulty of instantiation from a process model.

In our paper, we try to make this instantiation easier by using the goal-oriented approach. And at the same time it supports the record of the rationales to create process instance and product (artefact). Usually it is hard to record the rationales of the process, that is, why this process instance exists or why we need create this artefact. In our approach, we can tailor a model process into the process instance with its rationale, and explicitly show the relationship between the process and the product.

2 Approach

We propose the approach that supports writing the relationship between process and product with rationale by using the argumentation mechanism of Goal Structuring Notation (GSN) [1]. GSN illustrates the structure in which the goal is argued by the evidence (cf. 3.1). We assume that a goal is an aim of a process, the evidence is the product (artefacts) and we can indicate the validity of the process-product relationship by using the argumentation mechanism of GSN.

Usually we tailor the existing process model and define a process instance when we start a project. The process instance consists of many instances of process elements of process model (i.e. activity/task in terms of ISO 12207). We chose one of them and assign it a goal node. For example, in the Configuration Management (CM) the process element 'configuration management planning' is a candidate of a goal. The goal is decomposed into the sub-goal and we give the evidence (product) with argument that indicates how we know the completion of a process element. This is done iteratively till we can establish the tight and concrete relationship between process and product.

It is important for us to write down the instance of process. The process model is merely a starting point of our approach. There are various cases; we may need only a tentative process like creating a prototype, or we may have to consider the way to handle the system that has many variants that have similar process instances. We argue this problem in 4.3.
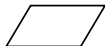
In our approach we associate process and product by using the GSN expression of argument and we can validate this relationship reasonably. So we can easily check the progress of a project by checking the product because product is tightly coupled with the detailed process instance.

3 Argument and GSN

3.1 GSN

To assure that a system is safe, we can use the safety case, and in several standards (e.g. ISO 26262, UK Defence standard 00-56, EN 50129) writing the safety case is mandatory requirement. The GSN is one of the techniques to express the safety case and it is used to clarify the structure of arguments. In the GSN the claims of the argument are documented as goals and items of evidence are described as solutions. It has the graphical representation, and easy to understand the structure of argument. There are some variations in GSN notation; we show the table of notation for our approach in this paper (Table 1)

Table 1: Our GSN symbols

Symbol Name	Semantics	Notation
Goal	It represents a claim, and can be divided into sub-goals.	G
Argument	It shows how an evidence supports a claim (Usually the symbol of argument is same as the goal, but we distinguish them).	A
Evidence	It indicates a solution that supports an argument.	◯
Context	It shows a context of goal or strategy.	⬭
Strategy	It gives the information about relationship between goal and sub-goal or argument.	▱
Assumption	It provides an assumption of a goal/strategy/argument.	⬭ A
Justification	It provides an additional explanation or a rationale.	⬭ J
Undeveloped entity	It indicates that a line of arguments doesn't be fixed.	◇

GSN is a generic concept, so it's used in many fields. We can use the GSN notation in the Assurance Case and the Dependability Case, not only in the Safety Case. It is one of the effective means for the validation of goal structures.

Our notation is a little bit different from the typical notation of GSN [2] because we want to emphasize the argumentation. GSN doesn't have the special symbol for argument. It uses the same symbol as the goal node. That is, in GSN people think that dividing goal into sub-goals means the argument. But we distinguish the argument from dividing a goal, so we have the special notation for the argument. This approach is similar to Claims-Argument-Evidence (CAE) method [3]. In CAE, the claim has different symbol from argument. Figure 1 shows the class diagram that indicates the relation of symbols in our approach.
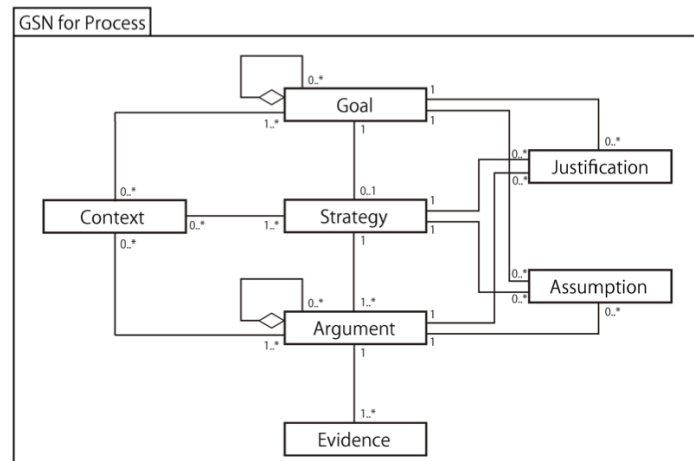
Figure 1: Our GSN meta-model

3.2   Application to software process

We show the steps of our approach when we analyse the new process or tailoring from the model process both in the management process or technical process. We assume that a process element is already chosen before starting the steps.

- A process element is assigned to the top goal node. We describe the context of the process element (or the project) in the context node. The context encircles the process element; the 'envelope' is another name. (Relating node: goal, context and assumption)

- If we can divide the goal, we allocate the subdivided goals to the sub-goal nodes. In this case, the top goal is the whole requirement of the process element, and the sub-goals represent the sectioned parts of the requirement. (Relating node: goal, context and assumption)

- Then we argue the sub-goal with argument node, which is a designed detail process instance. Usually we use the strategy node to represents the strategy of division from a sub-goal to argument nodes. The argument node or this strategy node must have a justification node, which shows the validity of strategy or rationale of the detailed process instance. (Relating node: argument, strategy and justification)

- Finally the leaf nodes of the argument have more than one artefact as the evidence. We carefully have to choose the evidence because it directly underlies the validity of an argument. We think that a whole document is usually inappropriate; we had better designate a chapter or a section of a document.

We have to execute those steps iteratively to stabilize the relationship among goal, argument and evidence. Usually it is hard to accomplish the GSN in initial phase of a project, because we may not be able to define the structure of argument and product. In that case, we can use the 'undeveloped entity' symbol to indicate we have to consider it in the next iteration cycle.

3.3   Examples

We show the two examples from the management process area and technical process area.

The configuration management (CM) process is one of the management processes. Here we think about the CM in a small project as an example (Figure 2). There is no special requirement from the customer and we have to release products several times. We describe that information in the context node. And after the discussion with customer, we know the constraints about cost, we write this in the assumption node. In the initial phase of the project, we may not be able to get the detailed cost

information. Then we give the argument as a detailed process. Here we start defining the procedure of the CM. We cerate the detailed process instance as an argument node, and then we associate them to the evidence.
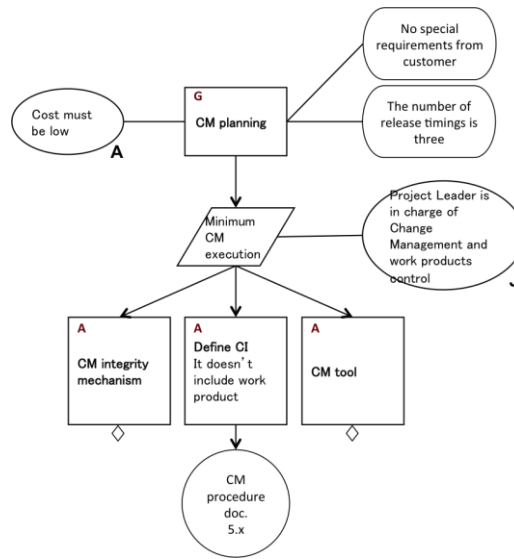
Figure 2: An example of management process

Next we see the example of the technical field. Usually processes of this type aren't considered deeply in the SPI work. Because the granularity of process is small and it changes easily by each projects. But we have to consider this process because the process of technical field (e.g. design) is directly relating to the characteristics of a product.

The automotive cruise control system (ACC) has the millimetre-wave radar, and we choose a non-functional requirement of it for the explanation purpose (Figure 3). When we apply our approach to design phase, we use a (functional or non-functional) requirement as a goal that is a sub-goal of the ACC system. In this case the strategy node insists that we need the hard-real time design and the justification node explains reason why. There are many hard real-time design features. The first argument node of this example explains about the absence of the violation of time frame ("No delay of response" node). And finally the evidence node that refers the architecture design document supports it.
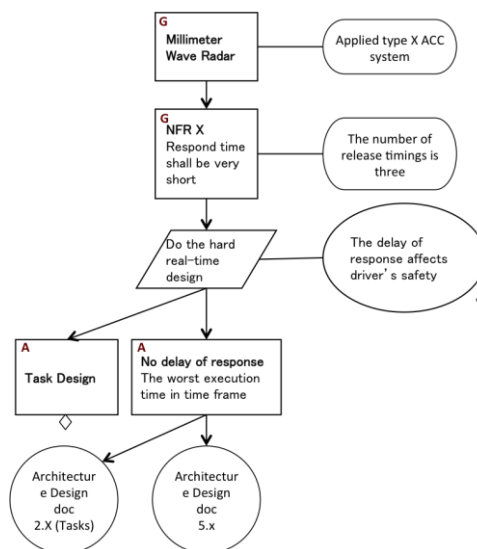
Figure 3: An example of technical process

3.4   Observation: writing the process argument

In case of the management process, the process argument can be written in early phase of a project, and we may re-write later if its situation would change. We show the supplemental meaning of each main node.

- Goal: the target management process element

- Argument: the detailed management process instance, which usually accompanies with the justification node

- Evidence: the work products (e.g. process definition, records of process)

In case of the technical process, it is hard to define all the process argument when we start a project, because the evidences depend on the specification or the design results. We can use the 'undeveloped node' for this purpose. Continuously we have to validate the process argument and we may modify the argument node or add the evidence node if necessary.

- Goal: the target design process element of a system or part of it (a.k.a. component)

- Argument: the detailed design process instance, which usually accompanies with the justification node

- Evidence: work products (e.g. specification, design document)

The meaning of the GSN nodes is almost the same in two examples.

After writing the GSN we can get the detailed process instance with the justification nodes. And those process instances are associated with the work products.

Reading the GSN diagram is not so difficult because the justification nodes help us to understand it. But writing the GSN diagram is hard work, because we also have to design the process usually having many constraints. So, we can use the Software Process Optimization (SPO) method alleviate this difficulty. The SPO provides two ways for problem-solving analysis: The 'close-world' analysis is a deductive way to solve the problem of the process, and the 'open-world' analysis uses the abductive reasoning. To do those analyses we can use the UML diagram. Our tool (chapter 5) can support also this approach because it is implemented as a plug-in on UML editor.

4 Additional Merits

4.1   Impact analysis

After the GSN has been completed, if we need to modify the diagram, we can check the extent of the impact. This is a 'ripple' effect; a single change may affect other consequential nodes [4]. The GSN diagram is the network of nodes, so we can easily know the potential nodes for change. For example, if we change the content of a context node (e.g. 'No special requirements from customer' in Figure 2) that is connecting to the top goal, we have to re-examine the other all nodes below it.

4.2   Justification

In our approach, we explicitly distinguish the goal node and the argument node and we emphasize the justification node. So, it is not difficult to validate the GSN. And if we can introduce some metrics such as the ratio of the number of the argument nodes and the number of the goal nodes, or the ratio of the number of justification nodes and the number of the goal node, those metrics help us to check the GSN in a simple way. For example, low ratio of argument or justification nodes indicates the lack of the rationales of process design.

4.3   Common and difference

The GSN has the generic notation mechanism that supports to create the GSN patterns [5]. The pattern assists in writing the GSN easier and in removing the unnecessary differences between the persons. As we said before we distinguish the goal node and the argument node. The systems in product-family have many common components, but the difference still exists. If we start analysing the two similar systems from a same component, we can easily compare the GSN. After this comparison we can get the commonality (pattern) and the difference between them.

5   Tool

We implemented a GSN editor tool on the Nirvana, which is a Unified Modelling Language (UML) editor and supports the team development with the database. In many cases we write the UML diagrams (e.g. class diagram, state machine diagram) for analysis or design. Our tool is the plugin of the UML editor and it can easily link with those UML diagrams. So, we can use them as the evidence in the GSN diagram. We also can link with the other document created by other tools (e.g. word processor or spread sheet). Namely, we can connect the work products through the GSN diagram uniformly. The Nirvana also has the version control mechanism, so it also can work as a configuration management tool.

Here are other advantages of our tool.

(1) Examine the range of influence

As we see in the chapter 4.1, we can check the range of influence through the network structure of the nodes in the GSN representation. The tool can automatically examine the influence. For example, if we change a top node, the tool shows the all nodes that have to be examined, and if we change the linked evidence the tool shows the evidence has changed.

(2) Find the difference between versions and between similar diagrams

Our tool has the version control mechanism and can extract the difference between versions automatically. It is very helpful to understand what have changed. The tool also has the replay mechanism and we can play back the process of creation and modification of the GSN. If there are many versions, this function is useful to understand the differences in a short time.

To find the difference is, on the contrary, to find the common. And if we find the common part between diagrams, we can make a pattern from them.

6   Related Works

The GSN is widely used to validate a property (safety) or properties (safety, security and so on) of a system. But it is rare to validate the software process using the GSN. In [6] and [7], we can find the 'process argument' and it connects the process and evidence. But there are two differences. We emphasize on the justification and distinguish between the goal node and the argument node. Secondly, we starts from the process instance, but authors [7] mainly use the generic arguments (i.e. pattern).

In a paper [8], author compares many software models and creates the new process model. She uses the GSN to express her process model. This is for process model, not for designing the process instances and it has no tailoring mechanism.

There is a famous approach for goal-based validation: Goal-Question-Metric (GQM)[9], [10], [11]. The GQM is mainly focusing on the design of metrics, but in [12], authors use the GQM for process tailoring. It is the same approach because we also start from the goal and validate the GSN. But in our approach the argument is the main element and we focus on creating the process instances.

7 Summary and Future Research

In order to tailor the process in software development we need the rationales of it. To do this we use the revised GSN, and it is useful to show and record the rationale of tailoring. We emphasize on the argument, so we distinguish sub-goal node and argument node, and the steps of our approach make us write the justification node with support metrics. This metrics and pattern mechanism is very important to evaluate the sufficiency of argument.

Our GSN tool supports not only tailoring of the process, but also connecting the work products that are also the part of evidence. So we think it is another way to manage the work product uniformly. And we can know the rationales why those evidences are valid for this process.

Finally, there is a good side effect in our approach. The process asset library (PAL) is the repository of work products and the process definitions to facilitate project performance. But usually it is hard to reuse the PAL. Because we need the reason why they define this process and write those documents. We think our approach can assist this situation by using our process arguments.

8 Literature

1. Kelly, T.P., *Arguing safety-a systematic approach to managing safety cases.* UNIVERSITY OF YORK DEPARTMENT OF COMPUTER SCIENCE-PUBLICATIONS-YCST, 1999.

2. Origin Consulting Limited. *GSN COMMUNITY STANDARD VERSION 1.* 2011; Available from: http://www.goalstructuringnotation.info/documents/GSN_Standard.pdf.

3. Adelard LLP. *Claims, Arguments and Evidence (CAE).* Available from: http://www.adelard.com/asce/choosing-asce/cae.html.

4. Kelly, T. and J. McDermid, *A systematic approach to safety case maintenance.* Reliability Engineering & System Safety, 2001. **71**(3): p. 271-284.

5. Kelly, T. and J. McDermid. *Safety Case Construction and Reuse Using Patterns.* 1997. Springer Verlag.

6. Habli, I. and T. Kelly, *Process and product certification arguments: getting the balance right.* ACM SIGBED Review, 2006. **3**(4): p. 1-8.

7. Spriggs, J., *GSN-The Goal Structuring Notation: A Structured Approach to Presenting Arguments*2012: Springer Verlag.

8. Kirk, D., *A Flexible Software Process Model*, in *Department of Electrical & Computer Engineering*2007, The University of Auckland: New Zealand.

9. Mashiko, Y. and V.R. Basili, *Using the GQM paradigm to investigate influential factors for software process improvement.* Journal of Systems and Software, 1997. **36**(1): p. 17-32.

10. Basili, V.R. and D.M. Weiss, *A methodology for collecting valid software engineering data.* Software Engineering, IEEE Transactions on, 1984(6): p. 728-738.

11. Van Solingen, R. and E. Berghout, *The Goal/Question/Metric Method: a practical guide for quality improvement of software development*1999: McGraw-Hill.

12. Basili, V.R. and H.D. Rombach. *Tailoring the software process to project goals and environments.* 1987. IEEE Computer Society Press.

9 Author CVs

Masao Ito

Mr Masao Ito is CEO of Nil Software Corp, a Tokyo-based software tool development company. He has long career in IPSE (Integrated Process Support Environment) construction and consulting. He is a board member of SEA (Software Engineers Association of Japan) and a core member of SEA-SPIN. Nowadays, he is also being involved in research activity of Volume CAD application development as a visiting scientist of RIKEN and NIRS (National Institution of Radiological Sciences), and serving as CEO of a venture company VCAD Solutions, Co., LTD.

Experience Report: Implementation of a Multi-Standard Compliant Process Improvement Program

Onur KAYNAK
Quality Assurance Consultant
INNOVA INFORMATION TECHNOLOGIES
okaynak@innova.com.tr

N. Alpay KARAGÖZ
Quality Assurance Manager
INNOVA INFORMATION TECHNOLOGIES
nkaragoz@innova.com.tr

Abstract

This paper explains the experience of an IT Solutions Provider company about the integrated management of the ISO/IEC 15504 based process assets library with other related IT standards which are ISO/IEC 27001, ISO/IEC 20000 and ISO 9001:2008. In order to establish and execute an efficient and effective infrastructure, a companywide process improvement program has been conducted. The lessons that are learned from the execution of this process improvement program, key implementation practices and major challenges are shared in this paper.

Keywords

Process Improvement, Process Management System, ISO/IEC 15504, SPICE, Integrated Process Improvement Program, Experience Report.

1 Introduction

INNOVA Information Technologies Corporation is Turkey's one of the greatest IT solution provider companies which is mainly working on Information Technology Infrastructure, Telecommunication, Enterprise Resource Management, Financial Transactions and Applications, Kiosk, Automation and Public and Health Sector related system, service and software development solutions and products. INNOVA provides solutions and products such as mobile loyalty applications, development and operation of Olympic level sports organizations systems, document management systems, health information systems, designed to fit KIOSK hardware and software for its national and international clients.
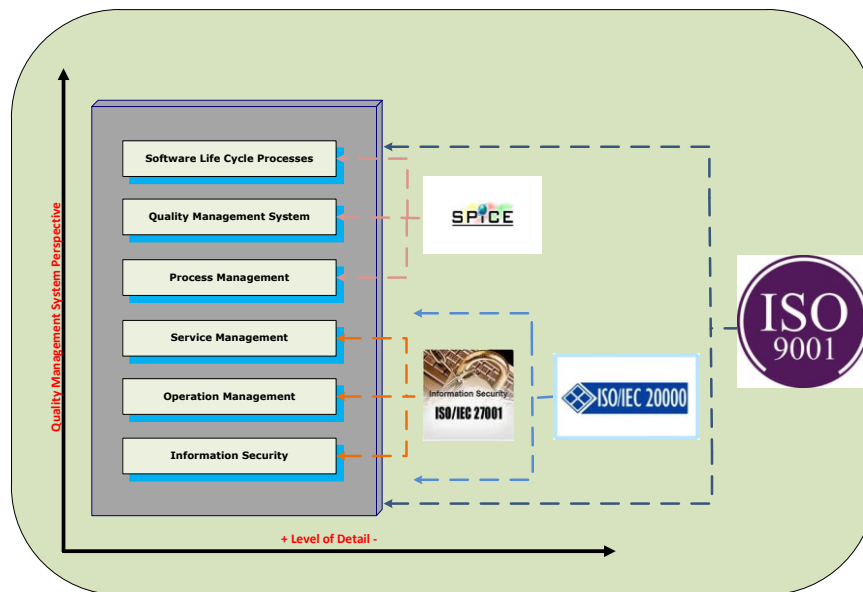
Figure 1: INNOVA Quality Management System Diagram

INNOVA implements various quality management systems regarding customer expectations, legal liabilities and to gain competitive advantage. ISO 9001:2008 forms the base of the quality management system. ISO/IEC 27001 is used to establish the norms for the Information Technology – Information Security related activities, ISO/IEC 20000 is used to define the Information Technology – Service Management processes and ISO/IEC 15504 is used to establish, maintain, appraise and improve system and software life cycle related processes. The main objective of this paper is to share the integration and improvement experiences that were obtained during the ISO/IEC 15504, ISO/IEC 27001, ISO/IEC 20000 and ISO 9001:2008 process improvement activities. The standards point of view to the processes and work areas are shown in Figure 1.

1.1   Motivation

INNOVA first established an ISO 9001 [3] based quality management system which was later enriched by ISO/IEC 15504 [1] Organizational Maturity Level-2 certified software development processes such as requirements elicitation, software requirements analysis, software design, software construction, software integration, software testing and management processes such as; project management, risk management, quality assurance, configuration management, problem resolution management and change request management [1]. Those days the company began dealing with the difficulties of maintaining two different but related management systems. Personnel had a tendency to call the company processes as either 9001 procedures or SPICE processes, despite the fact that those were actually INNOVA processes.

Then ISO/IEC 27001 [2] and ISO/IEC 20000 [4] certification requirements are satisfied due to legal and competitive advantage motivations. Quality assurance group, as the primer responsible of maintaining and establishing these certification initiatives, defined and executed a process improvement program in order to build an integrated process management system.

2 Background Information About the Company

INNOVA performs its activities in mainly two different geographical locations with seven major work branches consisting of more than 700 staff. These branches are TELCO, Public Solutions, ITS, Business Solutions, Financial Transactions Applications, Kiosk - Automation Solutions and Health Solutions. Due to the different natures of the sectors and customers targeted by these branches, various development life-cycles, organization structures and processes are in use.

Telco is the founding branch of the company whose major working area is telecommunication automation software development and maintenance covering both Operations Support Systems (OSS) and Business Support Systems (BSS) applications. The ISO 9001 based quality management system was established based on the processes of the Telco branch and then deployed to other branches with minor modifications.

Public Sector Solutions is one of the youngest branches of the company which deals with developing software automation solutions such as; custom information system solutions, Activity Management System and Document Management System products especially for governmental organizations. Following the establishment of this branch, the company needed a more detailed software and systems management framework and chose the ISO/IEC 15504 standard as the guide for this purpose. All of the software and systems development processes are updated in order to comply with the ISO/IEC 15504 Part-5 Organizational Maturity Level 2 requirements and the company received a SPICE Level-2 certificate in 2010.

The objectives of the company for 2011 included developing complex systems integration projects which revealed the necessity of developing information security management and IT service management systems. As a result, ISO/IEC 27001 and ISO/IEC 20000 based systems are defined and integrated with the existing quality management system. In 2011, the company realized a huge growth both in the number of personnel and the number of systems integration projects which also included service management commitments. The company realized multi-million dollar international projects such as Winter Universiade and European Youth Olympic Festival successfully as a consequence of its integrated process and project management infrastructure.

The following sections explain the challenges faced and key implementation practices deployed in order to overcome these challenges during the process of building an integrated process management system.

3 Challenges

3.1 Challenge 1: "Rapid Growth"

INNOVA was founded as a small company in 1999 and was acquired by Turk Telekom Group in 2007. Every year from then the number of staff doubled as well as the total income. As a result of such a rapid growth, each branch developed its own development environment, development processes and organization structure.

3.2 Challenge 2: "Cultural Distinctions"

As an expected consequence of rapid growth, the branches developed their own culture which in turn made it difficult to build a common approach for implementing inter-related processes.

3.3 Challenge 3: "Inter-process Relations & Dependencies"

Due to the requirement of compliance with 4 different management systems, some of the processes were common to all of these standards and some of them were inter-related with each other. This highly coupled structure increased the probability that a change in a process might break a compliance rule at another process in the process architecture.

3.4 Challenge 4: "Low priority of improvement tasks against software development projects"

Most of the improvement initiatives fail because of low priorities and inadequate resources allocated to the improvement tasks. There was a tendency to lower the priorities of the improvement tasks in our program too.

3.5 Challenge 5: "Process Audits"

Process Audits were performed as planned internal audits which were conducted one time a year. As a result, the corrective actions were limited with the planned internal audits and third party audits. But in order to establish a continuously improving quality management system, the number of internal audits needs to be increased to get sufficient corrective actions.

3.6 Challenge 6: "Low motivation of the personnel"

Personnel motivation decreases when additional tasks are assigned as a result of improvement actions and when the results of the improvement activities are not seen as soon as possible.

3.7 Challenge 7: "Unavailability of a Process Asset Library"

Processes were not collected under a central Process Asset Library (PAL). As a result staff faced with difficulties in accessing the up to date versions of the process assets.

3.8 Challenge 8: "Uncertain Roles & Responsibilities"

Process liabilities, roles and responsibilities were not defined such as; Process Improvement Manager (PIM), Process Analyst (PA), PAL Admin, Process Review Responsible (PRR). As a result, process management became too complex to manage.

3.9   Challenge 9: "AS-IS Analysis"

Processes were improved locally and some of the improvements were not communicated with other work branches, which resulted in a gap between the performed processes and the defined processes.

3.10   Challenge 10: "If you don't measure, you can't improve"

Measurement is the basis of any improvement initiative. Because of the lack of defined process performance metrics and measurement and analysis methods, the objectives for process improvement actions were not identified well.

3.11   Challenge 11: "Lack of a Process Management Process"

Processes were developed without the guidance of a defined process assets management guideline. As a result, the notations which were used for representing process flow diagrams varied among departments. Also, the granularity of the process definitions varied from process to process.

4 Key Practices

4.1   Take the Control in Hand

As the first activity, we formed a Process Asset Library (PAL) and collected all of the process assets together, so that we (Quality Assurance Group) could manage centrally and the staff could access easily. We also defined a Process Architecture diagram which we used as a basis for planning the process improvement actions. The process change requests were directed to the Process Improvement Manager and updates were published through the approvals of the Process Owner, Quality Manager and the Top Management Representative. Then the Process Owner sent an announcement to the staff informing the updated process asset. This practice was implemented in order to overcome Challenge 7: "Unavailability of a Process Asset Library".

4.2   Define Appropriate Roles

We had to reorganize the Quality Assurance Group in order to realize the activities we planned. We defined the following roles to be fulfilled by the Quality Assurance Group members;

- Process Improvement Manager (PIM), who would be in charge of managing the process improvement projects, monitoring and controlling the activities, establishing the coordination of the Process Analysts and reporting progress to the Quality Manager.

- Process Analyst (PA), who would help the Process Owners model, analyse and define their processes and also work in coordination with the PIM.

- PAL Admin, who would make ready and publish the updated process assets in the library and make sure that only the final and approved versions of the process assets are published.

- Process Review Responsible (PRR) is responsible for reviewing the process definitions with respect to the checklists and related company procedures.

Besides these roles, each process had its Process Owner (POW) and when necessary, the POW assigned one or more Process Responsible in order to assist her in modelling or defining the process. In order to provide the top management involvement, one of the Executive Board Members was selected as the Top Management Representative (TMR) who was in charge of approving that the process definition complies with the company rules and the process performance indicators are aligned with the business objectives. TMR also helped effective deployment of the processes throughout the company.

This practice was implemented in order to overcome Challenge 8: "Uncertain Roles & Responsibilities".

4.3   Perform AS-IS Analysis

As Watts S. Humphrey has put forward, "If you don't know where you are, a map won't help" [6]. In order to determine our position, we performed an as-is analysis in accordance with the ISO/IEC 15504 standard. We selected 10 processes and performed an appraisal covering 2 sample projects. Based on our findings we were able to categorize our strengths and prioritize improvement opportunities. This practice was implemented in order to overcome Challenge 9: "AS-IS Analysis".

4.4   Define the Approach and Apply the Method

Initially, in order to model process and related assets consistently, a process assets description guideline was prepared. This guideline defines how to generate process flow diagrams and process definitions.

Process flow diagrams were generated in accordance with a notation derived from ARIS eEPC [8] and UML Activity [7] diagrams. The notation includes; manual activity, resource, outsource, event, decision point, database, parallel flow, control flow, input, output, process, swim lane, form and template drawings. Processes were role based generated. Process definitions give detailed information about the process flow diagrams based on text format. Process definition process includes the following steps; Forming the Process Improvement Team, Planning Process Definition Steps, Process Definition and Modeling, Control of Process Asset, Release of Process Assets, Pilot Application and Improving Process Infrastructure.

Process definition document consist of the following sections; Process Policy and Goals, Definitions and Abbreviations, References, Application Scope and Tailoring Criteria, Roles and Responsibilities, Source and Target Processes, Application and Process Flow, Training, Human Resource and Infrastructure needs, Control and Storage of Records, Process Performance Indicators. This practice was implemented in order to overcome Challenge 11: "Lack of a Process Management Process".

4.5   Get Top Management Commitment

Improvement Program was planned to be handled as a set of projects such that a project exists for each standard. A Project Manager was assigned for each project that was responsible for planning the activities, coordinating the project team, utilizing the resources and reporting the progress periodically to the Quality Manager (QM). Top management was informed about the status of the projects by the Quality Manager via presentations or progress reports. QM also prepared memos to provide detailed information on major issues and a yearly Quality Report that summarizes quality activities. This practice was implemented in order to overcome Challenge 4: "Low priority of improvement tasks against software development projects".

4.6 Define and Execute the Process Improvement Program

This practice was implemented in order to overcome challenges 1,2,3 and 5.

4.6.1 Process Modeling and Definition Guideline

The process modeling and definition guideline describes the notation and the method that is going to be used for process modeling.

Process Context Diagram is used to show the top level view of the process. Interfaces between the source and target processes, inputs and outputs, triggering and resulting events are shown in this diagram.

Process Flow Diagram, which is a derivation of the UML Activity diagram and the ARIS eEPC diagram, is used to depict the details of each process utilizing a swim-lane view.

Process definition includes detailed descriptions which do not exist in the process flow diagrams. This asset also includes the process performance indicators, tailoring criteria, records control mechanism.

4.6.2 Process Improvement Process Definition

Process Improvement Process aims to identify the right processes to be developed/updated by the right responsible, to be released in an efficient way, to be announced for the right persons and to begin execution in the right time.

Process Improvement Process consist of the following steps; Preparation/Revision of Process Assets, Control of Process Assets, Release of Process Assets, Pilot Implementation, Enhancement of Process Infrastructure.

4.6.3 Process Implementation Infrastructure

Infrastructure planning was the first step of the process improvement. There was two major type of infrastructure planning which were; infrastructure that was needed for organization wide process improvement and that was needed to make specific processes more productive.

Team Foundation Server (TFS) and Visual Studio Team System (VSTS) have been used by Analysis, Design, Application Development and Test Groups. TFS combines the all development phases in a single framework.

Time Sheet software has been used by all employees of the company. Project Management uses that tool in order to track the effort that have been spent and calculate Actual Cost of Wok Performed (ACWP) and Budgeted Cost of Work Performed (BCWP) metrics.

MS Project Server integration to MS SharePoint was performed and Project Managers began to monitor and control the status of the project by this way. The tasks that are formed at MS Project Server directly go to related stakeholders form MS SharePoint.

JIRA has been used in order to keep bug reports which are found in application tests.

MS SharePoint has been used widely by all organizational groups in the company. Quality Assurance announces Process Assets Library, Human Management distributes and collects 360 degree self-assessment reports, Project management maintains projects, Administration occurs realize announcements.

4.6.4 Execution of the Process Improvement Plan

The Process Improvement Program consisted of internal projects that were treated as ordinary projects, that is, they were managed by a project manager, who set the objectives, allocated resources, monitored the progress, and managed risks. The coordination of these projects were provided by bi-weekly progress meetings by the involvement of the top management representative.

4.7 Actively Involve People

In accordance with the SPI Manifest [5] we tried to actively involve people in the process improvement activities. We built Process Development Teams (PDT) consisting of the Process Owner, Process Responsible and the Process Analyst. These teams modeled the As-Is process, determined improvement actions, developed the To-Be model and defined all of the necessary process assets. POW was responsible for describing the To-Be model as appropriate and PA was responsible for assuring that the final model was compliant with the existing process architecture and verifying the model against the checklist.
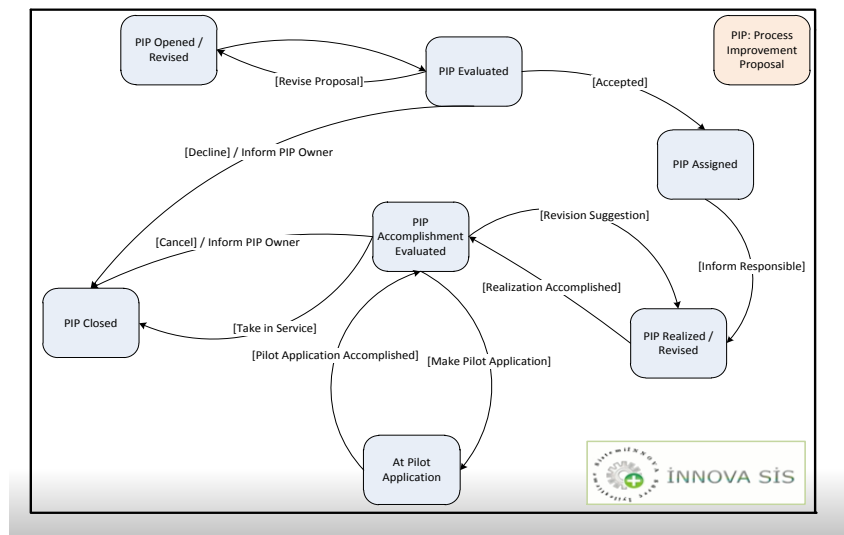
Figure 2: INNOVA Process Improvement System (INNOVA SİS) State Chart Diagram

All of the personnel were introduced to the quality management system through orientation trainings and key personnel attended to the half-day seminars on quality. Through the end of the 2011, we introduced the INNOVA Process Improvement System which enables the personnel to offer process improvement proposals (PIP). The flow of a PIP is depicted in Figure-2 as a state chart diagram.

The Process Improvement Proposals are evaluated by the Process Improvement Team (PIT) and the accepted ones are categorized as Corrective Action, Preventive Action or Process Change Requests. PIPs are assigned to a Quality Group member and followed until closure. Sometimes we needed a project team to realize an approved proposal. This practice was implemented in order to overcome Challenge 6: "Low motivation of the personnel"

4.8 Align with the Business Objectives

In order to realize an effective and sustainable process improvement program, we knew that we required improvement objectives aligned with the business objectives. Following a 2-day lasting high level management meeting, we prepared a mind map describing the business objectives of the company and then we prepared the Process Improvement Plan which includes the processes, priorities, process owners, process analysts and other resources that should take part. The Process Improvement Manager used this plan to monitor and control the progress.

Since we defined the process improvement activities as projects, the performance of the responsible people was taken into account in the yearly performance evaluations. This was another mechanism to guarantee that the priority of the improvement tasks would not decrease against a software development project, since it was also a project task. This practice was implemented in order to overcome Challenge 6: "Low motivation of the personnel".

4.9   Manage Change

Improvement means change, and change means resistance. By actively involving people and getting the involvement of the top management we were able to avoid some of the resistance. However there were some people who did not want to be disturbed in their comfort areas. We tried to handle such people by training seminars, workshops and demonstrating the benefits of our studies in other work branches. In addition, we tried to eliminate paperwork and developed workflows on the intranet as much as possible so that the personnel should not feel the new processes as a burden. This practice was implemented in order to overcome challenges 1, 2, and 6.

4.10 Perform Process Audits

The implementation of the processes was checked through internal or external audits. Internal audits could be planned or unplanned. The planned audits are performed focusing on selected projects once a year. Unplanned audits are performed whenever the quality management identifies a need. In order to close the non-conformances that were identified during the audits, corrective actions are initiated and all corrective actions are planned to be handled until the external audits. The corrective actions are presented to the top management at management review meetings. We performed 10 internal and 4 external audits which resulted in 63 corrective action and 12 preventive action requests. This practice was implemented in order to overcome Challenge 5: "Process Audits".

4.11 Collect Measures and Analyze

The 10th section of our process definition template includes process performance indicators including the definition of the measures, the reporting period, the measurement method and the target value. The Process Owners are responsible for collecting these measures and the Process Improvement Manager is responsible for gathering, analysing and reporting the measurement results.

The Quality Group analyses the measurement results in order to discover if there are any trends that requires the modification of the processes. Process Owners are also responsible to analyze the results to find any improvement opportunities. This practice was implemented in order to overcome Challenge 10: "If you don't measure you can't improve".

5   Conclusions

In this paper we shared our process improvement experience in an IT solutions provider company complying with four different management systems. We listed the challenges we have faced and the practices that we used to overcome those challenges.

We need to see more iteration of these practices in order to conclude that we have built a sustainable and optimized process improvement program.

6   Literature

[1] TS, ISO / IEC 15504:2007 Software Process Improvement and Capability Determination

[2] TS, ISO/IEC 27001:2006 Information technology -- Security techniques -- Information security management systems -- Requirements.

[3] TS, ISO 9001:2008 Quality management systems — Requirements

[4] TS, ISO/IEC 20000:2010 Information technology -- Service management

[5] Pries-Heje Jan, Johansen Jørn, "SPI Manifesto", 2010

[6] Humphrey Watts, "Managing the Software Process", SEI, 1989

[7] Rumbaugh James, Jacobson Ivar, Booch Grady, "UML Reference Manual", Addison Wesley, 1998

[8] Scheer August-Wilhelm, "ARIS: Business Process Modeling", Springer, 2000

7 Author CV's

N. Alpay KARAGÖZ, Ph.D.

N. Alpay Karagöz is currently the Quality Manager at INNOVA IT Solutions. He is leading the company wide process improvement initiatives and quality assurance activities. He is also an adjunct faculty member in the Middle East Technical University, Informatics Institute, Software Management Program. He holds Ph.D. and MSc. degrees both in the Information Systems Department and B.S. degree in the Department of Computer Engineering, Middle East Technical University. He took part in many projects as senior software process improvement consultant. His research interests include software management; process improvement, business process modeling, service management, software quality management and object oriented software development.

Onur KAYNAK, MSc.

Onur KAYNAK graduated from Bilkent University Computer Technology and Information Systems department with B.Sc. degree in 2005. He had M.Sc. degree from Middle East Technical University Software Management program in 2009. He took part at 8 projects as software quality engineer in defense industry. He is interested in Quality Management, Process Improvement and Cost of Quality Measurement. He is still working as a Quality Assurance Consultant at INNOVA IT Solutions.

IDIEF - An Integrated Framework for Software Process and Product Improvement in SMEs

Murtala Adeyemi Ayanwale and Elli Georgiadou

School of Engineering and Information Sciences, Middlesex University, London, United Kingdom
ma1907@live.mdx.ac.uk, e.georgiadou@mdx.ac.uk

Abstract

The essential objective of software development is to understand, manage and control the process activities in order to produce a quality product. Software development organisations have been adopting various methods, models and frameworks to produce a quality product. The focus within the software engineering community is either placed on the product quality or the process quality, often paying less attention to the workforce (People). In this paper we propose an integrated framework (IDIEF) for software process and product improvement in SMEs which face additional problems due to scarce resources. IDIEF was developed through a combination of literature review and case study analysis. It integrates aspects of P-CMM and ISO9126, both of which are widely used within the software engineering community.

Keywords

P-CMM, Quality Models, ISO/IEC 9126, Quality Management, Software Development

1 Introduction

The Software Engineering (SE) community has and still is experiencing what is termed as the "Software Crisis", although the success rate in producing a quality software is increasing with time compare to the last decade. Nevertheless, the practices within the SE community require improvement in order to cope with the rapid evolvement of technologies, the complexity and proliferation of systems. Over the last decades, different models, frameworks and methods have been developed to achieve success in software development and to reduce the failure rate. However, many SMEs have failed to successfully implement them. This could be due to their size or time required to successfully implement them [1]. The implementation of quality in software product is an effort that should be formally managed throughout the software engineering lifecycle [2]. Thus, there is need for the process to be considered. The quality of a product is directly related to the quality of the engineering process; thus emphasising the need for a manufacturing-like process [3]. One could argue that for the product quality to be satisfactory, the process needs to be satisfactory as well.

Software development organisations are knowledge intense organisations. Knowledge has become a crucial and key resource in software development. Knowledge Management (KM) and Knowledge Sharing (KS) are strongly linked to organisational maturity [4]. Knowledge is intangible, difficult to

measure and is acquired over time, without knowledge no organisation can survive. Hence, the need for organisations to capture its employees knowledge and shared throughout the organisation as at when needed.

2 Related Work

For over 40 years the software industry was following corrective approaches mainly concentrating on testing and inspections at the end of the development lifecycle. Product quality models were proposed by McCall [5], Boehm [6] and Dromey [7]. The International Organisation for Standardisation (ISO) and International Electrotechnical Commission (IEC) in 1991 introduced a product quality standard named ISO/IEC 9126; it was prepared by a joint technical committee ISO/IEC JTC 1. The ISO/IEC 9126 standard consists of four parts covering product quality, external metrics, internal metrics and quality in use metrics [8].

ISO/IEC is defined by means of characteristics of the software, which are further broken down into sub-characteristics and further decomposed into attributes. The bottom of the hierarchy consists of measurable software attributes. ISO/IEC is the only model that supports all the perspectives of quality (with the exception of the transcendental perspective). Furthermore, its predictive framework clearly supports both top-down and bottom-up approaches [3].

The ISO/IEC 9126 is widely used in the software engineering community; it was developed to provide a comprehensive specification and evaluation model for software product quality. However, since there have been argument that the software product quality is directly related to the process quality. Thus, there is a need for standardised quality process within the software development organisations.

2.1   A shift towards process quality

2.1.1 Focussing on prevention rather than detection

Burr and Georgiadou [9] observed that in the early years Software Engineering adopted an end of cycle quality inspection just as the early manufacturing where the quality of the product was achieved through inspections at the end of the production line. Inspections in turn resulted in three categories namely the accepted, the rejected and those products requiring rework. The last two 'heaps' namely the rejects and the reworks gave a measure of the losses which every manufacturer needed to reduce for survival and competitive advantage".

Over the last 40 years there have been various models for assessing and improving the software process known collectively as Software Process Improvement (SPI). Software process is the set of activities, methods, and practices that guide people in the production of software [10]. Organisational success and continuous improvement are seen as dependent on the ability of the organisation to see things in a new way, gain new understanding, and produce new patterns of behaviour, on a continuing basis and in a way that engages the organisation as a whole [11].

Software process improvement arose out of the quality principles of Deming, Juran and Crosby in the mid 1980s in response to the belief that better management of software development process would lead to improved software reliability and quality (as cited in [12]). SPI is a structured approach to improve software organisations capability to deliver quality services in a competitive way, inspired by experiences for quality management. SPI initiatives typically use normative models to assess current software practices and provide guidance for how to prioritise improvements. The most widely used model is the capability maturity model (CMM) [13].

The CMM describes five levels of maturity based on a set of key process areas (KPAs) that should be in place for each level. These levels are Initial, Repeatable, Defined, Managed, and Optimising. Each maturity level except the initial consist of three to seven key process areas that must be satisfied before moving to the next level. Each process area organises a set of interrelated practices in a critical area of workforce management.

The maturity levels, key process areas, common features, and key practices have been extensively discussed and reviewed within the software engineering community. However, in the case of the CMM, its structures have a large-scale military orientation that may be unsuitable for smaller-scale commercial products developed by the majority of smaller commercial firms [1]. Some practitioners argue about the relevance of the CMM for SMEs. Although process assessment methods and improvement models used for SPI are available to all enterprises, studies show that these proposals from SEI or ISO are difficult for the vast majority of the SMEs to apply [14]. We consider that this is due to the complexity of the recommendations of the models, the time and resources required to implement.

The People Capability Maturity Model (P-CMM) places priority on considering the employees' maturity within the software development organisation. Every employee has an impact on the quality of the product in the process of software development. The motivation for P-CMM is to improve the ability of software organisations to attract, develop, motivate, organise and retain the talent required to continuously improve software development capability. P-CMM is a tool to help an organisation successfully address the critical people issues. It is a proven set of human capital management practices that provides an organisational change model through an evolutionary framework based on a system of workforce practices [15].

Working P-CMM, like CMM, has five maturity levels; each of the levels represents a different level of organisational capability for managing and developing the workforce. These levels are: - Initial, Managed, Defined, Predictable, and Optimising [15]. These levels establish foundations for continuously improving individual competencies, developing effective teams and avoiding temptations to try skipping levels which is counterproductive because each maturity level in the P-CMM forms a necessary foundation from which to achieve the next level.

Due to the context dependent quality, the people involved in the development process need to be considered based on their role and contribution to the process. Project managers, testers, developers and customers, each can value the same qualities of a product in different ways. People's behavioural competencies or characteristics of professional conduct influence the effectiveness and efficiency with which they perform a predetermined role in the software process [16].

Software development organisations are knowledge intensive organisations. Knowledge has become a crucial and key resource in the software development. Managing the knowledge within and outside the organisation requires organising and managing principles. Intangible knowledge is difficult to measure. Without knowledge no organisation can survive. There is need for knowledge creation, transfer, storage and retrieval, and application so that an organisation can make better use of it human resources. The most valuable asset of the 20th-century company were its production equipments while the most valuable asset of the 21st-century institution, whether business or non-business, will be its knowledge workers and their productivity [17].

Working with process improvement show that things go wrong for SMEs. Firstly, the perspective of the individual is often forgotten and secondly the manager and change agents are frequently abstract in their communication of outcome, business benefit and consequences [18]. Following a review of relevant literature. The main issues reported were identified and the research question was formulated: "*What are the obstacles to Software Process and Product Quality Improvement in organizations?*". This question was further broken down into different sub-questions which formed the basis of structured interviews. The main categories of factors are Organisational and Software Development issues. Organisational issues identified are: Lack of required skills, Goals, Lack of communication, High staff turnover. Software development issues identified are: Adopted method/lifecycle, Requirements and Quality Management Techniques (such as Testing).

2.1.1 Research Methodology and Research Design

The research method selected was that of cases study targeting SMEs. According to Galliers [19] a case study is an attempt at describing the relationships which exist in reality through capturing greater detail and analysing more variables than in any of the other methods such as experiments and surveys. The reason for this choice was the availability of the organizations and time limitations. The instrument used for capturing reality are a questionnaire which was used for guiding structured interviews.

The research design comprised a literature review, formulation of hypothesis, questionnaire design, filed study (data collection), data analysis, drafting of a framework, use of the framework to evaluate the participating case study organisations and refinement of the framework. A mixture of top-down and bottom-up approaches were used for the design of the research instrument and for the analysis of the data.

Following a literature review the hypothesis was formulated.

Hypothesis: People capability and competency can influence the software product quality.

The main hypothesis was further broken down into 2 sub-hypotheses

    (i)   Peoples' capability influences the software process quality

    (ii) High peoples' competency affects positively the software product quality

3 Case Studies

Two case study organisations (SMEs) form the basis of this investigation. The investigation is being carried out in order to reach an informed understanding of the impact of process quality on software product quality in relation to peoples' (employees') capability and competency. This understanding is needed to help determine whether the theories established in the literature review is consistent with the findings from collected data from the two SME software development companies. Neither of the case study organisations has been assessed by SEI or other organisation. From data and information gathered from both organisations, they can be placed at the level 2 (Repeatable) of the Capability Maturity Model. Structured interviews were conducted using a structured questionnaire in order to find out the present status of the two organisations.

The results from the interviews are discussed in the rest of this section. In order to keep the anonymity of the organisation, they will be refer to as 'X' and 'Y'.

3.1 Organisation X

Based in the United Kingdom, X is a small software development organisation that has an understanding of the impact of lack of process quality in software development and the end product quality. X was founded in 2001. The founder set out to create a total software solution for an accident claim specialist to manage its complex business process. The successful implementation of the claims handling software resulted in faster more streamlined administrative processes. Using the core of this software, the X software development team have produced, implement, and managed other systems such as Project Management, Human Resource Management and Point of Sales systems.

Organisational Issues: X constantly seeks ways to manage its organisational issues. Staff retention (high staff turnover), Development process, Communication and Relationship issues amongst employees, and Lack of rewards are the major issues identified at X. Due to its size and the competition within the software development industry, it is clear that X need to retain and acquire the talents required. It is difficult for X to keep its workforce, the high staff turnover within the organisation also leads to knowledge loss, X always relied on technology and the knowledge of its workforce to produce and support its work. Once the staff acquired the experience and knowledge, it is easy for them to leave for a bigger organisation. This means that the company needs to constantly employ and train new staff.

X always seeks ways to improve communications and collaboration amongst its staff. The management is concerned with the effect of lack of communication and relationship issues within the organisation. Although the communication and relationship from the top to the bottom is functional, there is a lack of communication from the bottom to the top which will affect the software process improvement of the organisation. The employees are not satisfied with the reward structure within the organisation;

they feel their efforts are not recognised. This leads to employees not performing at their best and often leaving the company.

Software Development Issues: X relied on technology to produce and support its work and at the core of its vision is the adoption of agile methodology to develop better software product for its clients. The organisation is currently experimenting with agile methodology adoption. This requires frequent delivery of working software, team work, self-organising. However, agile methods require high maturity levels for teams to work smoothly and efficiently.

3.2   Organisation Y

Y is a medium size software development organisation in Nigeria. It specialises in the development of bespoke software, Enterprise data warehouse (EDW), Business Intelligence and Website/Portal design. Y has also suffered from the impact of lack of process quality on software product quality. Y prides itself on its commitment to continually improving its quality to satisfy the customers' expectations. It has 3 branches in different regions of the country where it operates. It was set up and is managed by professionals who have had many years of experience in the IT field gained from Nigeria and from overseas.

Organisational Issues: owing to it size, most Y employees work one software development project at a time i.e. projects are normally developed sequentially. There is a minimal assignation of people to roles because of the small number of employees and also different geographical locations. For example, an employee with speciality in coding may be working within a team on a project at the Lagos office while there is need for a similar role at the other branch. This implies that everyone participates in all roles. The organisation has adopted several methods to support software process improvement. Nevertheless, Y is not immune from problems to software process improvement. Internal politics and cultural differences form the major part of its problem. This has lead to different problems within the organisation and it is affecting the organisational results as a whole.

Communication and relationship issues-: the workforce in Y cut across different regions within the country it operates. Communication within groups is minimal, although they can communicate in a common language. Y has an in-house system and database, this serves as the repository for the organisations process, procedure, and lessons learned from previous projects. The repository content is available to the teams as and when needed. Users can locate and interact with the limited knowledge available. However, there is no system in place within the organisation for workforce knowledge management. Other issues identified are Lack of training, Lack of Knowledge creation, Skills, Documentation, and Quality standards.

Software Development Issues: For bespoke software development, the management are concerned about the testing procedure in place within the organisation. Due to deadlines and time constraints, the scope of software testing is not entirely covered. Thus, products are shipped to customers/users with faults which results in financial losses (because of re-work) but more importantly in customer dissatisfaction which in turn leads to loss of credibility in the market.

3.3   Results: Common areas of concern

The investigation was carried out in two software development organisations based in the UK and Nigeria respectively. Despite the cultural, economic, and political differences the findings reveal some common areas of concern. The focus was to identify patterns within the case study organisation, and understand relationships between different factors. The criteria for comparison are based on the established factors derived from literature and used in the questionnaire.

The study provides insights into the organisational issues and the perception of respondents based on their knowledge and experience in the field. The data and information gathered were later analysed and findings from both organisations point toward the same issues with little variances. With the aid of the data, we are able to establish an informed conclusion on the impact of process quality on product

quality. The common problems identified are people, tools and technology issues. Subsequently, the findings from these comparisons are used to develop a conceptual framework for integrating a process model and a quality model.

4 Integration of P-CMM and ISO9126

Previous studies [14,18] suggest that future work is required to investigate how companies can improve their processes. However, the studies focus primarily on the process improvement. There is little attempt to integrate process and product improvement explicitly. Achieving software quality will require a quality process and most quality models attach less importance to other process elements like human factors and roles. Hence, there is a need for a framework that could be implemented to cater for both process and product quality in software development.

Fig. 1 represents the case study scenario for both organisations. It was developed based on the findings from literature and enriched by insights gained through the case studies. This self assessing model can be applied by any software development organisation to determine the organisational process quality level as well as the required software product quality.
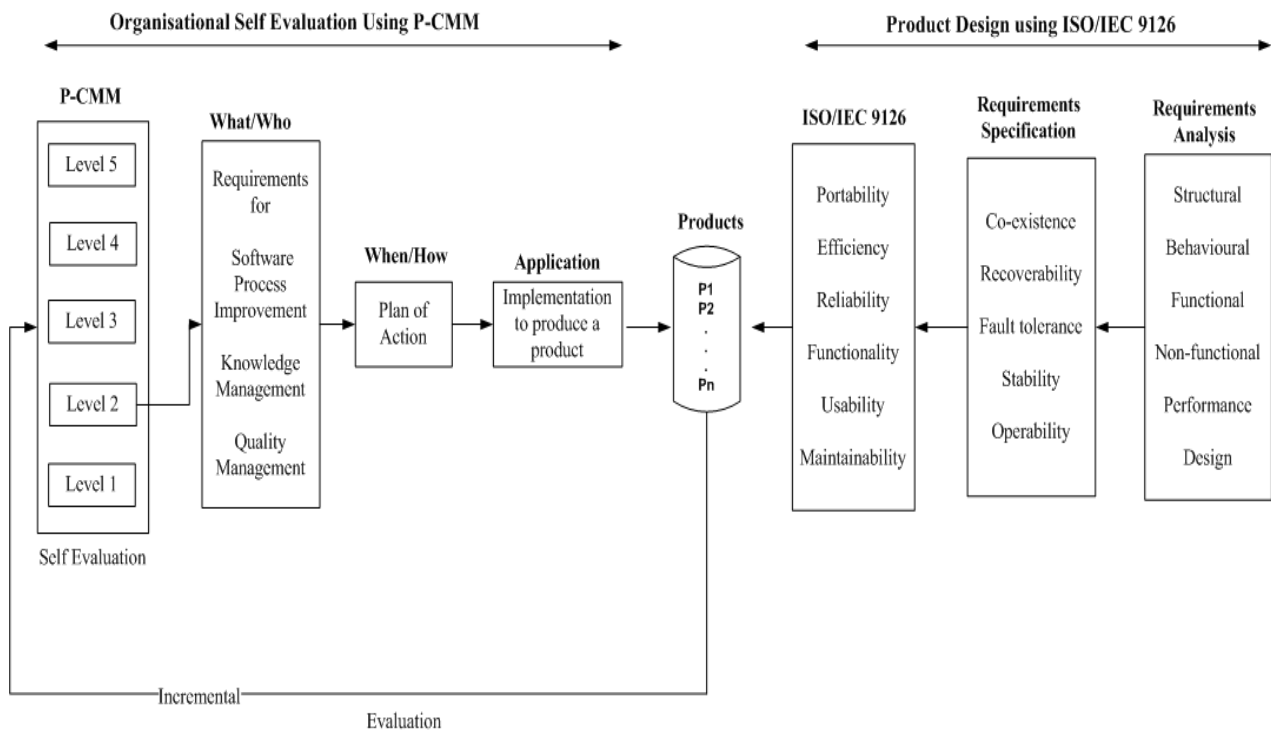
Figure 1: Model Integration and Case study scenario

5 The IDIEF Framework

The proposed framework combines some elements of the People Capability Maturity Model and ISO/IEC 9126 to cater for both process and product quality within an SME software development organisation and ensure continuous process and product improvement. The proposed framework comprises of four phases namely Initialisation, Definition, Implementation and Evaluation (IDIEF) depicted in Fig. 2.

The *Initialisation* phase – focuses on the people (workforce) i.e. what is required and their needs in relation to people capabilities and competencies. The *Definition* phase deals with working out the ideal knowledge strategy and also with establishing a quality assurance system. The *Implementation* phase establishes the software product quality based on the client specification using ISO/IEC 9126 and the final phase *Evaluation/Evolution* enable the collection and analysis of data. These results can then be used to quantify how the software process improvement impacts on organisational performance and on the quality of the resulting products.

The application of this model is organisational dependent. Both process (Organisational self evaluation using P-CMM and Product design using ISO/IEC 9126) can be initiated simultaneously. The organisation should perform self assessment to determine its stage (level) and based on this initiate a process to improve their software process. Findings reveal that normally SMEs develop software in a sequential mode. In a situation where the organisation is in midstream of a software development, the self evaluation can be initiated and the product design phase of the model can be initiated at the start of a new software development.

The self evaluation process should be repeated at a stipulated time (and certainly after completion of a number of projects) in order to determine whether the organisation has improved the required key process areas to ascertain whether the people maturity has indeed moved up to the next level.

Phase 1: *Initialisation*

This is the starting point in the guidelines to process improvement. In this phase, the people (employees) competency and capabilities are identified. Based on this, relevant training can be arranged where necessary. The project manager can fit personnel into roles based on their capabilities and competencies. In this phase, establishing workforce management can run concurrently with other key process area identified such development and training. This is to put in place adequate benefits structure as it may apply based on organisational pay structure.

Fitting people to roles and developing them depend on organisational policy. One form of training is learning on the job. This can be achieved by assigning people with less knowledge to work in a team with someone who has the required knowledge expertise. For job satisfaction as well as for efficiency the knowledge fit is extremely important [20, 21]. Aside this, employees can be given the chance to explore other ways or forms that they feel will improve their knowledge within the context of their job. This will be in agreement with the management.

Phase 2: *Definition*

This phase deals with two crucial aspects: the first is establishing a quality assurance system and the second is management of the knowledge within. The two can be implemented concurrently. This is dependent on the management and the project manager in particular. In order to improve the product quality, there should be a standard or benchmark in place. The quality assurance officer with the approval of the project manager should put a measurement criterion in place informed by the quality model. As the framework was developed based on ISO/IEC 9126, the characteristics and sub-characteristics of this model should be well understood by the project team [22].

There are several knowledge management strategies that could be adopted. Depending on its resources, the organisation may decide to design a hybrid strategy. Upon the design, there is need for the strategy to be reviewed to determine its validity before implementation. Some of the KM strategies an organisation can adopt are codification, personalisation or cognitive (based on views) [23].

Phase 3: *Implementation*

This is a crucial stage where the necessary processes and strategy developed to improve the software process are implemented. At this stage, the Project Manager with the assistance of the Quality Assurance personnel set out an action plan to implement the proposed framework. Some of the processes within the framework can be implemented concurrently.

Phase 4: *Evaluation/Evolution*

Continuous monitoring and formal feedback mechanism need to be in place to ensure participation, communication and learning so that mistakes can be avoided. Once the key process areas have been implemented, periodical collection of data should be establish for continuous assessment. This should

be a continuous exercise to be agreed upon between the project manager and the quality assurance officer. The data collected and analysed can be used to determine the organisation's progress on their software process improvement.
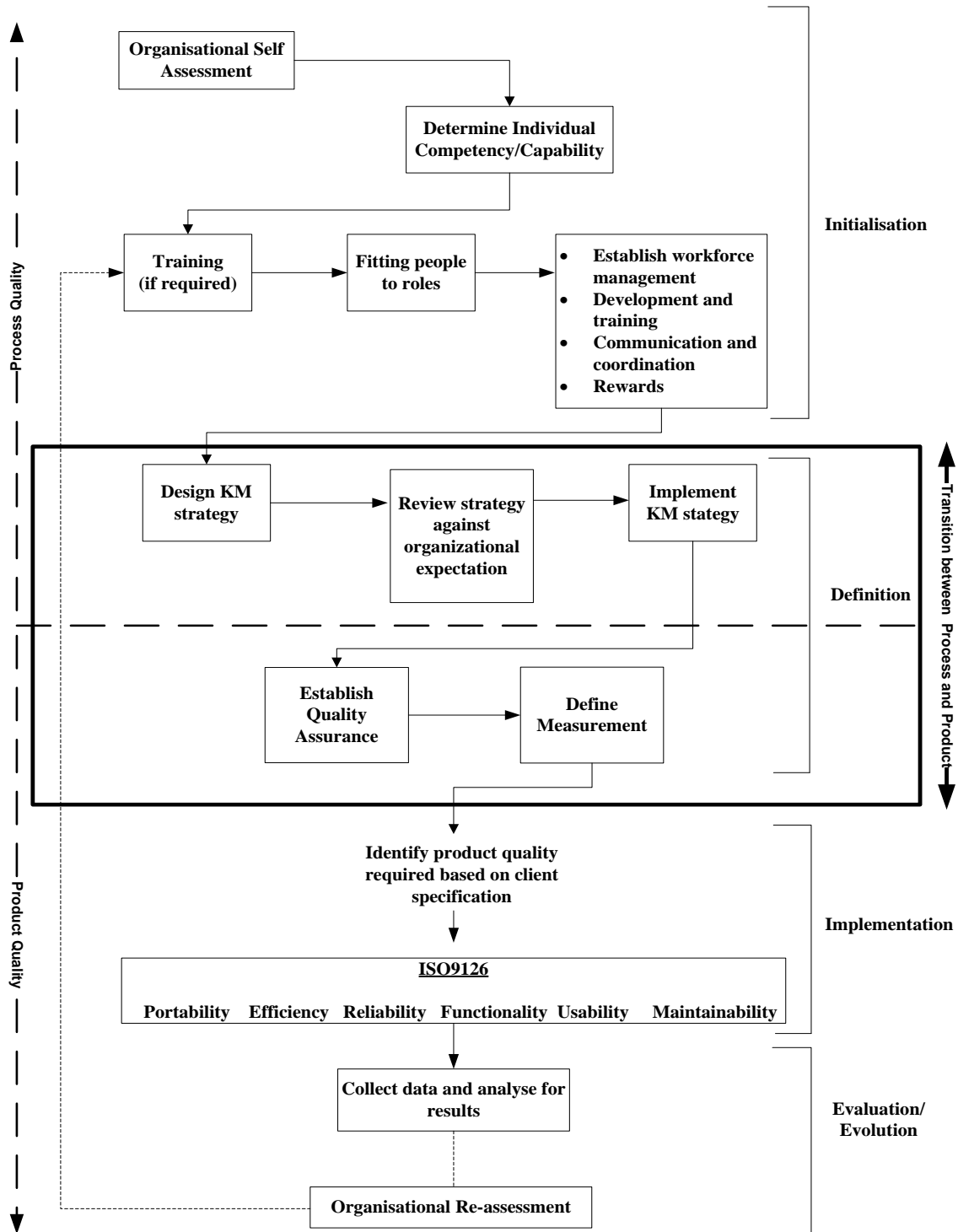
Figure 2: IDIEF Framework

6 Conclusion and Further Study

The findings from this study will help the people involved in SMEs software development organisations to better understand their software improvement process. The results will help to determine where the overlap is between the process and the product quality model adopted by software developing organisations. Process quality and product quality can be achieved with the integration of process quality such as People Capability Maturity Model and software product quality such as ISO/IEC 9126 presented in this study.

The proposed integrated framework could be of great benefit especially to SME software development organisations who cannot afford very expensive third-party evaluation such as those carried out by SEI-CMMI. It is anticipated that the IDIEF framework will help in identifying and in dealing with issues affecting software process improvement in software development in SMEs. In turn product quality improvements are expected as a result of process improvement.

6.1 Limitations of Study

- Sample size

  In this study we presented data collected form 2 SMEs software development organisations. Although the data provides interesting insights into the problems encountered in these companies. The data collected is small and it is not appropriate to generalise from this sample. Data analysis

- Problem generalisation

  Each software development organisation regardless of its size has individual and possibly unique problems. We acknowledge that organisations are likely to vary in where their process problems lie and how they plan to deal with them. However, as the organisations in this study are SME's and are specifically selected for this investigation research work based on their size and maturity level, lesson can be learned from the study findings.

- Data analysis

  Due to the data sample size, the data analysis was limited. Considering the importance of this study to the software engineering community, our future work will involve the use of analysis or statistical modelling tools such as SPSS, S-PLUS to produce a more meaningful data representation.

There are areas where further research is required in order to validate and improve the proposed framework. The next step is to investigate our findings further by acquiring more data from more software development SMEs. With this, the IDIEF framework could be validated. Due to some insights gained through carrying out the field study in two different countries, we aim to also investigate the effects of culture on software process improvement. We anticipate that quantification of improvements will feedback to achieve continuous improvement of both process and product and to attain higher maturity levels of the employees involved.

Acknowledgements

References

1. Bach, J.: Enough about process: what we need are heroes. IEEE Software, Vol. 12, No 2, pp 96–98, (1994)

2. Bourque, P., Dupuis, R., Abran, A., Moore, J.W., Tripp et S, L.L.: Fundamental Principles of Software Engineering - A Journey. Journal of Systems and Software, (2000)

3. Cote, M.-A., Suryn, W., Georgiadou, E.: In search for a widely applicable and accepted Software Quality Model for Software Quality Engineering, Software Quality Journal, Vol. 15, No. 4, pp 401-416 (2007)

4. Georgiadou, E., Siakas, K. V. & Balstrup, B.: The I5P Visualisation Framework for Performance Estimation through the Alignment of Process Maturity and Knowledge Sharing, International Journal of Human Capital and Information Technology Professionals, 2(2), pp. 37-47 (2011)

5. McCall, J. A., Richards, P. K., and Walters, G. F.: Factors in software quality. Griffiths Air Force Base, N.Y.(1977)

6. Boehm, B. W., Brown, J. R., Kaspar, J. R., Lipow, M. L. & MacCleod, G.: Characteristics of Software Quality. New York: American Elsevier (1978)

7. Dromey, R. G. A model for software product quality. IEEE Transactions on Software Engineering 21, 146-162 (1995)

8. International Standard Organization, http://www.iso.org

9. Burr, A., Georgiadou, E.: Software Development Maturity: A Comparison with other Industries, 5th World Congress on Total Quality, India, New Delhi (1995)

10. Humphrey, W.: Introduction to the team software process, Addison Wesley, Reading, MA, (2000)

11. Gasston, J. & Halloran, P.: Continuous Software Process Improvement Requires Organizational Learning: An Australian Case Study, Seventh International Conference on Software Quality Management, Southampton, pp. 19-33 (1999)

12. McGuire, E.G.: Factors affecting the Quality of Software Project Management: an empirical study based on the Capability Maturity Model, Software Quality Journal 5, pp 305-307 (1996)

13. Paulk, M.C., Webber, C.V., Curtis, B., Chrissis, M.B. The Capability Maturity Model: Guidelines for Improving the Software Process, Addisson-Wesley, Reading, MA (1995)

14. Pino, F.J., Garcia, F. & Piattini, M.: An Integrated Framework to Guide Software Process Improvement in Small Organizations, In: European Systems & Software, Process Improvement and Innovation (EuroSPI 2009), Madrid, Spain: CCIS 42, Springer, pp. 213–224 (2009)

15. Curtis, B., Hefley, B. & Miller, S.: The People Capability Maturity Model (P-CMM) Version 2.0, Software Engineering Institute, CMU/SEI-2009-TR-003, July (2009)

16. Acuna, S.T. & Juristo, N.: Assigning people to roles in software projects. Software – Practice and Experience, Vol. 34, No. 7, pp 675-696 (2004)

17. Drucker, P. F.: Knowledge worker productivity: The biggest challenge. California Management Review 41(2): 79–94 (1999)

18. Frederichsen, K.S.: Successful Process Improvement for Small and Medium Sized set ups, In: European Systems & Software, Process Improvement and Innovation (EuroSPI 2009), Industrial Proceedings, Madrid, Spain, pp. 13.9-13.18, September (2009)

19. Galliers, R.D.: Information Systems Research - Issues, Methods and Practical guidelines. Blackwell Scientific Publications, Oxford (1992)

20. Mumford, E.: The ETHICS Approach, Commun. ACM 36(6): 82 (1993)

21. Mumford, E.: "Defining Systems Requirements to meet Business Needs: A Case study example", Computer Journal, 28, 2: 97-104 (1985)

22. Abran, A., Moore, J.W.: IEEE Guide to Software Engineering Body of Knowledge, SWEBOK (2004)

23. Hansen, M. T., Nohria, N. & Tierney, T.: What's Your Strategy for Managing Knowledge ?, Harvard Business Review, 77(2), pp. 106-116 (1999)

7 Author CVs

Murtala Adeyemi Ayanwale

He is a Master in Business Information Systems Management (2011-2012) from Middlesex University, studied Computing from the University of Greenwich (2009). His research field is Software Engineering. His current research interest includes software process and product improvement in SMEs, project management and risk management. His future research interest includes the effect of culture on software process improvement and project management.

Elli Georgiadou

Elli Georgiadou , (BA (Open University), PGCert in Education (London University), Fellow of the BCS, CITP, MACM, MIEEE)) is currently a Visiting Academic at the School of Engineering and Information Sciences, Middlesex University, London, UK, Elli gained her expertise and experience through teaching for over 30 years as a Principal Lecturer (in 2 large London Universities), through on-going consultancy services mainly to SMEs, and through holding many positions of responsibility such as co-ordinator of European Affairs and International exchanges for over 20 years . She has published widely on aspects of Software Quality. She has been managing many European Lifelong Learning Projects funded by the Socrates, Leonardo and Tempus programmes.

The long way to maturity: a road map to success

Antonia Mas[1], Antoni Lluís Mesquida[1] and Bartomeu Fluxà[2]

[1]University of the Balearic Islands, Palma de Mallorca, Spain
{antonia.mas, antoni.mesquida}@uib.es

[2]Brújula Tecnologías de la Información S.A, Palma de Mallorca, Spain
tfluxa@brujula.es

Abstract

This paper describes the experience of Brújula, a Spanish software company which, in order to achieve its business objectives and meet the needs and expectations of its customers, strongly bet on quality as the way to progress towards maturity. We present the continuous evolution suffered by both the management and the production processes through the implementation of quality standards. The knowledge reuse strategy deployed by the company has enabled an important effort reduction when implementing a new standard by taking advantage of the knowledge gained in previous improvement efforts. The most significant results and lessons learned during the improvement path are presented.

1 Introduction

Many people want to have good luck, but few who decide to go for it. Create good luck is to prepare the circumstances for the opportunity but the opportunity is not a matter of luck or chance is always there! Good luck depends on the creation of the necessary circumstances, and creating these circumstances depends only on oneself.

We have used this excerpt from the book Good Luck [1] as a reference to illustrate the behavior of the company whose case is presented in this paper. Brújula began its journey in 2000 and nowadays, even the current economic situation, is an established and recognized company of its economic sector in our region. However, it has not reached this stage by chance, because it has had good luck, but because the company has been able to create the appropriate circumstances. Many factors have influenced the success of the company. In this paper the actions related to the deployment and improvement of both management and production processes that have enabled the company to progress towards maturity are presented.

Brújula is an innovative company that has committed to quality and excellence since its inception. Along the way, it has had to adapt to the constant technological changes occurred in the ICT sector. These changes have greatly influenced the habits of people, forcing companies to develop applications for environments and devices that did not exist a few years ago. Therefore, a continuous adaptation to cover both new user needs and development issues and, also to offer new services to remain on the market has been required during the whole history of the company. The changes resulting from this situation have taken place in the entire structure of the organization, both in top management, by creating new strategies, and in the culture of employees. Making the adjustments to new needs considering the expectations of all employees is never easy, because of people's resistance to change.

The paper is structured as follows: section 2 presents the company background. Section 3 defines the road to maturity, from the beginnings to the current situation. Section 4 describes how the knowledge gained through the implementation of best practices recommended by international standards of different areas has been managed and reused. Finally, section 5 summarizes the lessons learned and outlines the future of the company in terms of maturity.

2 Company background

Brújula is a Spanish company which began its activity in 2000 with a staff of 6 employees. Nowadays, Brújula has 120 employees dedicated to the development of Internet-based marketing and management applications and the implementation of the infrastructure which supports them. The company provides tailored, unique and global solutions that include consultancy, training and the technology necessary for the evolution of customers' businesses and needs. Brújula has four main business lines:

- Software Factory and Applications: Its mission is to provide customers with software development projects based on the latest technologies.

- Systems, Communications and Security: Its mission is to provide customers with system, communications and security closed projects based on technologies from leading manufacturers.

- Incoming & Select4U: Its mission is to provide qualified personnel in key development, systems, communications and security technologies to work directly in the client's office and under his direction.

- IT Infrastructure Management: Its mission is to provide customers with IT infrastructure maintenance services to ensure maximum availability and performance of their systems.

The vision of Brújula is to be a large company of national reference and able of providing a complete solution to the customer needs through continuous improvement of the knowledge and the quality of

their solutions. Quality applied to all the areas is the way that will let the company to achieve its business objectives, meeting the needs and expectations of its customers.

3 *The road to maturity*

Since the creation of Brújula, innovation and continuous improvement are the basis of a culture assumed by all those working in the organization, not only with the aim to grow as a company, but with the clear objective to share the improvements with customers to help them to innovate in their business through knowledge and technology.

This section summarizes the evolution that has taken place in the company after incorporating the good practices of recognised models and standards. The commitment to implement several international standards has impacted different areas of the company, such as integrated management, production processes, information security management and environmental management. This fact has allowed the growth and improvement of the company in all aspects, its consolidation in the software and services development sector in the Balearic Islands, and a competitive positioning thanks to the recognition obtained.

3.1   The beginnings: 2000 - 2004

Brújula started its journey in 2000 with the development of management applications as one of the highlights of its strategic plan. From its inception, the company opted for quality and took part in the first edition of the QuaSAR project [2]. This project was a successful initiative that allowed the analysis of the software development sector in the Balearic Islands and provided guidance for the participant companies to improve their software processes and to be certified according ISO 9001:2000 [3]. At that time, ISO/IEC 15504 (SPICE) [4] had not been still released as an international standard and companies were interested not only in a software process improvement path but also in an ISO 9001 certification.

Thanks to the participation in the QuaSAR project, Brújula established its first Quality Management System (QMS) and obtained the ISO 9001 certification in 2003. Moreover, the company laid the foundations of its software development processes and got motivated for continuous process improvement.

3.2   The evolution: 2005 - 2011

In 2005, as a sample of the evolution towards maturity and continuous improvement, the company still maintains the ISO 9001 certification and incorporates the EFQM Excellence Model [5] to its QMS. Figure 1 shows the nine criteria of the EFQM Excellence Model.
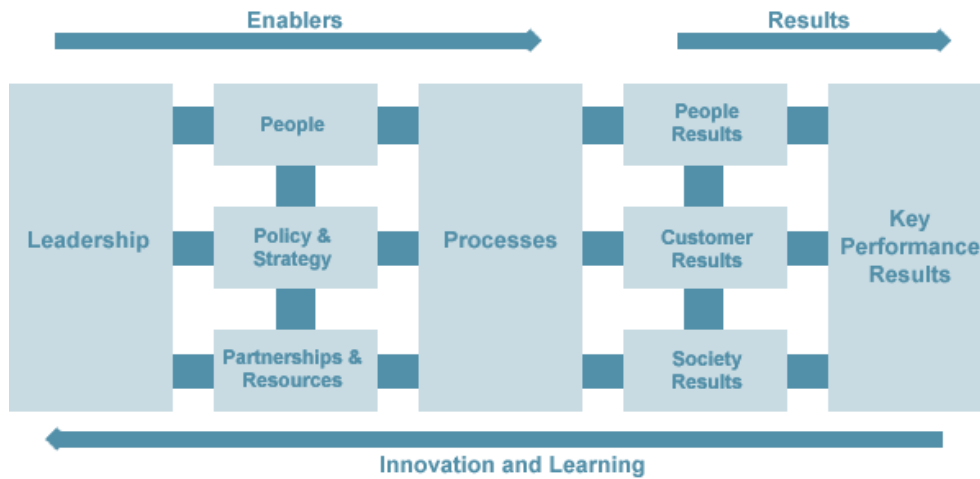
Figure 1: EFQM Excellence Model.

In 2006, Brújula reorganized the company's product offering new solutions and services. During this year the company made an important commitment on environmental management and obtained the ISO/IEC 14001 [6] certification in 2007.

As recognition of all the efforts made, Brújula obtained the Balearic Silver Award of Excellence in Management in 2006 and the Gold Award in 2008.

In 2007, Brújula initiated a software processes improvement programme thanks to public subsidies from the Spanish government, making big internal efforts to adapt some of their processes to the best practices proposed by the ISO/IEC 15504 international standard. After a formal assessment and an improvement phase, the company achieved the capability level 2 in some of its processes. As it was impossible for the company to reach ISO/IEC 15504-7 [7] maturity level 2, the efforts were focused on improving the processes related to customer tendering and project management.

In 2009, as provided in its new Strategic Plan, the company incorporated the information security management in the existent management model and obtained the ISO/IEC 27001 [8] certification.

3.3   The future

The adverse economic situation has made organizations to redirect their efforts to survive in the market. As a consequence, the interest in applying new models or standards has been shifted to the background. In addition, in recent years the Spanish government has reduced or cancelled the majority of public subsidies to improve the enterprise competitiveness through the implementation of process standards.

Software companies have been forced to compete in these recessionary conditions. Brújula is no stranger to this situation. For this reason, the company's commitment is to maintain all the standards it has already implemented and improve the IT service management processes. In order to achieve this new challenge and facilitate the sustained growth of this new business line, Brújula plans to implement ISO/IEC 20000-1 [9] and get a certification according to this standard during 2012.

4 Knowledge management

One of the most important strengths of the evolution of Brújula has been the knowledge management performed. In 2002, with the main goal of supporting and improving the management system of the company, the quality department was established. Without this department, which has served to define, centralize and support the implementation of the processes that have been needed in the rest

of departments, the company may not have reached the current process maturity level.

Given the process management approach that drives the organization, different macroprocesses were identified, defined and deployed: business Management, project management, software engineering, systems engineering, infrastructure management, human resource management and quality management and assurance. Each of these macroprocesses includes a set of processes of the same field at a greater level of detail. For example, the infrastructure management macroprocess includes a set of processes such as user management, internal maintenance, access control and logical security.

The centralization of knowledge together with the skills and competencies of the quality department have enabled knowledge reuse, and therefore the effort reduction of implementing standards in the company. In this section, the evolution of Brújula's internal processes thanks to the reuse of the gained knowledge is presented.

4.1   Establishment of an Integrated Management System

From the ISO 9001 QMS and after consolidating the implemented EFQM Excellence Model, the company realised that many requirements of the environmental management system defined by ISO 14001 [8] and the information security management system defined by ISO/IEC 27001 [6] had wide similarities with the requirements of the QMS implemented and, therefore, could be integrated into a global Integrated Management System (IMS). The IMS of Brújula is shown in Table 1.

EFQM	ISO 9001:2008	ISO 14001:2004	ISO/IEC 27001:2005
1. Leadership	5.1. Management commitment 5.3. Quality policy	4.2. Environmental policy	5.1. Management commitment
	5.6. Management review	4.6. Management review	7. Management review of the ISMS
	5.5.1. Responsibility and authority 5.5.2. Management representative	4.4.1. Resources, roles, responsibility and authority	A6. Organization of information security
	4. Quality management system	-	4.2.1. Establish the ISMS
2. Strategy	5.4.1. Quality objectives 5.6. Management review 8.4. Analysis of data	4.3.3. Objectives, targets and programme(s) 4.6. Management review	4.2.3. Monitor and review the ISMS 5.1. Management commitment 7. Management review of the ISMS
3. People	6.1. Provision of resources	-	5.2.1. Provision of resources
	6.2. Human resources	4.4.2. Competence, training and awareness	5.2.2. Training, awareness and competence
	5.5.3. Internal communication	4.4.3. Communication	-
	6.1. Provision of resources	-	5.2.1. Provision of resources
4. Partnerships & Resources	-	4.4.6. Operational control	-
	6.3. Infrastructure 6.4. Work environment 7.4. Purchasing 7.6. Control of monitoring and measurement equipment	4.3.1. Environmental aspects	4.2.1. Establish the ISMS 4.2.2. Implement and operate the ISMS
	-	4.4.7. Emergency preparedness and response	A14. Business continuity management
5. Processes, Products and Services	4. Quality management system 5.2. Customer focus 5.4.2. Quality management system planning 7.1. Planning of product realization 7.2. Customer-related processes 7.3. Design and development 7.4. Purchasing 7.5. Production and service provision 8.2.1. Customer satisfaction	4.3.1. Environmental aspects 4.4.6. Operational control	4.2.2. Implement and operate the ISMS
	5.6. Management review	4.6. Management review	7. Management review of the ISMS
	8.2.3. Monitoring and measurement of processes 8.2.4. Monitoring and measurement	4.5.1. Monitoring and measurement	4.2.3. Monitor and review the ISMS

EFQM	ISO 9001:2008	ISO 14001:2004	ISO/IEC 27001:2005
	of product 8.4. Analysis of data		
	7.2.1. Determination of requirement related to the product	4.3.2. Legal and other requirements 4.5.2. Evaluation of compliance	A15.1. Compliance with legal requirements
	8.3. Control of nonconforming product 8.5. Improvement	4.5.3. Nonconformity, corrective action and preventive action	4.2.4. Maintain and improve the ISMS 8. ISMS improvement A13. Information security incident management
	8.2.2. Internal audit	4.5.5. Internal audit	6. Internal ISMS audits
	4.2. Documentation requirements 4.2.3. Control of documents 4.2.4. Control of records	4.4.4. Documentation 4.4.5. Control of documents	4.3. Documentation requirements 4.3.2. Control of documents 4.3.3. Control of records

Table 1: Brújula's Integrated Management System

The relationships between the requirements of the three management systems that were integrated are structured in five categories, which correspond to the five 'Enabler' criteria of the EFQM Excellence Model. Next, the most important changes and decisions taken and the lessons learned during the integration process are detailed.

The decisions related to the 'Leadership' criteria were as follows:

- Management commitments regarding environmental aspects and information security were integrated into the existent quality policy of the company, resulting in the new 'Integrated Management System policy'.

- The existence of an IMS allowed the integration of all top management review activities.

- The vast majority of information security responsibilities were integrated into the profile of the quality manager. The other responsibilities were assigned to the information systems manager.

- A committee joining functions of the three management systems was set up as the governing body of the IMS. Some information security responsibilities were assigned to a security subcommittee.

Regarding to the 'Strategy' criteria, the decisions were:

- In order to satisfy existing and future customers demand for higher levels of quality and management, the company decided to continue investing in the implementation of quality standards internationally recognized as ISO/IEC 15504, ISO/IEC 27001 and ISO/IEC 20000-1. This strategic decision is related to the EFQM sub-criterion 2a (Strategy is based on understanding the needs and expectations of both stakeholders and the external environment).

- New strategy and supporting policies were developed, reviewed and updated according to EFQM Sub-criterion 2c (Strategy and supporting policies are developed, reviewed and updated to ensure economic, societal and ecological sustainability).

- A management model including objective allocation to both leadership and middle management was established. A systematic policy to define and formalize personal action plans from the company objectives was implemented. The definition of the objectives, goals and action plans follows EFQM Sub-criterion 2b (Strategy is based on understanding internal performance and capabilities).

Regarding to the 'People' criteria, the decisions were:

- An annual training plan with annual training activities for all staff and specific actions for technical profiles related to security management were established.

- The existing stakeholders communication record was also used for recording information security and environmental issues related to customers and suppliers.

The results related to the 'Partnerships & Resources' criteria were:

- The integration of the ISO/IEC 27001 information security management system facilitated the compliance with EFQM Sub-criterion 4d (Technology is managed to support the delivery of strategy) and 4e (Information and knowledge are managed to support effective decision making and to build the organisational capability) by considering information systems management and maintenance activities, risk analysis, asset identification, backup and user access and responsibility management.

- The integration of the ISO 14000 environmental management system facilitated the compliance with EFQM Sub-criterion 4c (Buildings, equipment, materials and natural resources are managed in a sustainable way).

- The business continuity plan was complemented by the existing emergency plan.

Regarding to the 'Processes, Products and Services' criteria, the decisions were:

- All the clauses related to security and environmental aspects were included in the subcontractor management process. In addition, the procurement process, initially intended for the material purchasing during ISO 9001 implementation, was expanded to systematize part of the subcontractor management process.

- The scope of the evaluation of legal compliance process was expanded with all the applicable security legislation.

- The same channel and the same software tool used to support the registration of non-conformities of the existent QMS were used for recording and managing security incidents.

- IMS internal and external audits were added to the existing audit program. Qualification criteria for security auditors were incorporated into the internal audit process.

- The existing document control and record control processes were improved by including information classification, confidential or public, and adding review and approval responsibilities.

4.2   Evolution of the production processes

As mentioned before, in 2007, the company initiated a SPI programme to adjust its production processes to the ISO/IEC 15504 international standard. The first task consisted on analysing the software development processes defined by ISO/IEC 12207 [10] in order to identify which of the efforts made to implement ISO 9001 and EFQM could be useful when deploying specific technical software development processes. Some of the generic management processes had to evolve to meet ISO/IEC 15504 specific base practices. The evolution of the production processes of Brújula is shown in Table 2. First column shows the processes deployed in the company nowadays. Columns two and three detail the actions and changes that have suffered these processes during the evolution to maturity.

ISO/IEC 12207 Software life cycle processes	ISO 9001 and EFQM implementation (before 2007)	ISO/IEC 15504-5 implementation (2007 onwards)
ACQ.1 Acquisition preparation ACQ.2 Supplier selection ACQ.3 Contract agreement ACQ.4 Supplier monitoring ACQ.5 Customer acceptance	Deployed during ISO 9001 implementation. Focused on material procurement.	The scope of these processes was expanded to cover subcontracting. Supplier monitoring was related to the project management process.
SPL.2 Product release	-	Deployed.
ENG.1 Requirements elicitation ENG.4 Software requirements analysis	Deployed during ISO 9001 implementation.	Existing requirements elicitation and analysis assets were improved by taking into account ISO/IEC 15504 base practices.
ENG.3 System architectural design ENG.5 Software design	Deployed in an unique process during ISO 9001 implementation.	Improved.
ENG.6 Software construction	Deployed during ISO 9001 implementation.	Improved.
ENG.7 Software integration	-	Improved.
ENG.8 Software testing	Deployed during ISO 9001 implementation.	No significant changes.

ISO/IEC 12207 Software life cycle processes	ISO 9001 and EFQM implementation (before 2007)	ISO/IEC 15504-5 implementation (2007 onwards)
ENG.11 Software installation	Deployed during ISO 9001 implementation.	Installation was improved by completing the installation registration procedure. Delivery was fully standardized and consolidated.
ENG.12 Software and system maintenance	Deployed during ISO 9001 implementation.	Improved.
MAN.1 Organizational alignment	Deployed during ISO 9001 implementation and considerably improved with EFQM implementation.	No significant changes.
MAN.2 Organizational management	Deployed during ISO 9001 implementation and considerably improved with EFQM implementation.	No significant changes.
MAN.3 Project management	Deployed during ISO 9001 implementation.	Improved and consolidated through the implementation of project management good practices. Existing project management assets were expanded by including new activities.
MAN.4 Quality management	Deployed during ISO 9001 implementation and considerably improved with EFQM implementation.	No significant changes.
MAN.5 Risk management	-	Deployed. Focused on projects. Security risk analysis was included to project risk management. Having defined a risk management strategy facilitated ISO/IEC 27001 implementation.
MAN.6 Measurement	Deployed during ISO 9001 implementation. Focused on organizational measurement.	No significant changes.
RIN.1 Human resource management	Partially deployed during ISO 9001 implementation and improved with EFQM implementation.	No significant changes.
RIN.2 Training	Deployed during ISO 9001 implementation.	No significant changes.
RIN.4 Infrastructure	Deployed internal maintenance, user management and backup processes during ISO 9001 implementation.	No significant changes.
SUP.1 Quality assurance	-	Partially deployed.
SUP.2 Verification	-	Partially deployed in several processes by defining verification checklists.
SUP.4 Joint review	-	Deployed.
SUP.5 Audit	Deployed during ISO 9001 implementation. Focused on organizational audit.	No significant changes.
SUP.7 Documentation	-	Deployed.
SUP.8 Configuration management	Partially deployed during ISO 9001 implementation.	Improved. Introduction of new tools for managing code.
SUP.9 Problem resolution management	Partially deployed during ISO 9001 implementation.	Improved.
SUP.10 Change request management	Partially deployed during ISO 9001 implementation.	Improved.

Table 2: Evolution of Brújula's production processes

Having implemented a QMS according to ISO 9001 facilitated the company to reach values "L" or "P" for ISO/IEC 15504 capability level 3 process attributes in some processes, because these processes were defined, standardized and measured within the organization. The implementation of the ISO/IEC 15504 standard helped the company to consolidate the processes already deployed, delving into the best practices and aspects not previously detailed.

4.2.1 Adaptation of the production processes to support ISO/IEC 27001

After consolidating the process improvement programme according to ISO/IEC 15504, the company realised that it had already performed some important steps in order to implement some of the ISO/IEC 27002 [11] security controls which are necessary for the implementation of an information security management system, since these controls were considered by ISO/IEC 15504-5 base

practices.

From the analysis of all the existing relations between ISO/IEC 15504-5 base practices and ISO/IEC 27002 security controls presented in [12], the work of the company consisted on modifying or amplifying the deployed ISO/IEC 15504-5 processes in order to make them compliant with the security requirements of the related ISO/IEC 27002 controls. Each change could affect to different process components: process purpose, base practices or/and work products. In order to cover a specific security control, four different types of actions could be performed on the processes:

- Use the process purpose or its base practices to manage the security requirements of the related control, without any process modification.

- Modify or extend one or more base practices.

- Add a new base practice from the related control objective, closely related to the existent base practices.

- Modify or extend the process purpose.

It has to be noted that, although some ISO/IEC 15504-5 processes were easily adapted to cover some requirements of the ISO/IEC 27002 security controls, other security controls that did not have any relation to the deployed ISO/IEC 15504-5 processes had to be implemented as indicated in the ISO/IEC 27002 standard.

5 Conclusion

The case study presented in this paper in an example of a company which firmly bet on quality standards to improve its production processes in a time when few companies in its sector and environment did. One of the keys of the success of the company evolution towards maturity has been the active participation, commitment and motivation of top management. Top management has always provided and trained the necessary human resources to achieve the stated objectives and support the business strategy. Moreover, an important financial investment to facilitate the implementation and standardization of new procedures, establishing new departments or areas and incorporating new support tools has been required. These changes have taken place in all departments and at all levels of the company.

All these efforts have borne fruitful results. Brújula has evolved from a company of 6 people in 2000 to a company of 120 employees. The higher maturity level reached the easier to tackle new projects and gain an advantage in front of the competitors.

Because the road to maturity is long, Brújula will follow its path to quality by continually improving its internal processes. There is not only one way to reach the goal. In this paper, the steps in the way to maturity taken by this company have been presented.

6 Literature

1. Rovira, A. and Trías De Bes, F.: Good Luck: Creating the Conditions for Success in Life and Business. John Wiley & Sons, 2004

2. Amengual, E. and Mas, A.: Software Process Improvement in Small Companies: An Experience. EuroSPI 2007 Industrial Proceedings.

3. ISO: ISO 9001:2008 Quality management systems - Requirements

4. ISO/IEC: ISO/IEC 15504-5: ISO/IEC 15504-5:2006 Information technology - Process Assessment - Part 5: An exemplar Process Assessment Model

5. The EFQM Excellence Model. http://www.efqm.org

6. ISO: ISO 14001:2004/Cor 1:2009 Environmental management systems - Requirements with guidance for use

7. ISO/IEC: ISO/IEC 15504-7: ISO/IEC TR 15504-7:2008 Information technology - Process assessment - Part 7: Assessment of organizational maturity

8. ISO/IEC: ISO/IEC 27001:2005 Information technology - Security techniques - Information security management systems - Requirements

9. ISO/IEC: ISO/IEC 20000-1:2011 Information technology - Service management - Part 1: Service management system requirements

10. ISO:12207:1995/Amd 1:2002/Amd 2:2004 Systems and software engineering - Software life cycle processes

11. ISO/IEC: ISO/IEC 27002:2005 Information technology - Security techniques - Code of practice for information security management

12. Mesquida, A., Mas, A. and Amengual, E.: An ISO/IEC 15504 Security Extension. Software Process Improvement and Capability Determination. Communications in Computer and Information Science, 2011, Volume 155, Part 2, pp. 64-72

7 Author CVs

Antonia Mas

Antonia Mas is a university lecturer of software engineering and project management at the University of the Balearic Islands. Her research interests include software process improvement, project management and service management. She has promoted and coordinated the QuaSAR Project, a software process improvement initiative in small software companies at the Balearic Islands. She received her degree in Computer Science from UAB (Catalonia, Spain) and her PhD in Computer Science from the University of the Balearic Islands. She is a member of the Software Quality Group of ATI Spain from 1998 and she acts as the editorial board member in the REICIS journal. She has also served as program committee chair and program committee member of scientific conferences and workshops related to software quality. She is an ISO/IEC 15504 assessor and is focussed on assessing small companies. Contact her at Campus UIB, Edifici A. Turmeda. Ctra. Valldemossa, Km. 7,5. 07122 Palma de Mallorca, Spain. antonia.mas@uib.es.

Antoni Lluís Mesquida

Antoni Lluís Mesquida is an assistant lecturer in computer science at the University of the Balearic Islands. He is teaching in the area of Software Engineering. His research interests include software process improvement and service management. He has participated in the QuaSAR project, a software process improvement programme in software small companies at the Balearic Islands. Contact him at Campus UIB, Edifici A. Turmeda. Ctra. Valldemossa, Km. 7,5. 07122 Palma de Mallorca, Spain. antoni.mesquida@uib.es.

Bartomeu Fluxà

Bartomeu Fluxà is the Quality Manager of Brújula Tecnologías de la Información. His research interests include software process improvement and TQM models. He has coordinated and participated in different projects to set up and improve a management system, incorporating international standards and models such as EFQM, ISO 9001, ISO 14001, ISO/IEC 15504 or ISO/IEC 27001. He received his degree in Computer Science from the University of the Balearic Islands. Contact him at Brújula Tecnologías de la Información, Edifici U, ParcBit. Ctra. Valldemossa, km 7,4. 07121. Palma de Mallorca, Spain. tfluxa@brujula.es.

Proposal and practice of SPI framework
- Toshiba's SPI History since 2000 -

Hideto Ogasawara, Takumi Kusanagi, Minoru Aizawa

Toshiba Corporation, 1, Komukai Toshiba-cho, Saiwai-ku,, Kawasaki, 212-8582 Japan

{hideto.ogasawara, takumi.kusanagi, minoru.aizawa}@toshiba.co.jp

Abstract

For the effective promotion of software process improvement (SPI) activities in a large organization, it is necessary to establish an organizational structure and a deployment method for promotion and to develop training courses, support tools, and other materials. Even if an organizational promotion system is established, the SPI activities of each development department cannot be promoted effectively without an SPI community. To promote SPI activities throughout Toshiba Group, we organized a Corporate Software Engineering Process Group in April 2000. We also proposed an SPI framework. This framework extracts the features of organizations in which improvement activities were successful in the 1990s, and combines them systematically. For over ten years, we have been constructing Toshiba's SPI framework based on the SPI framework that we proposed and promoting process improvement activities using it. As a result, maturity levels were improved in many development departments, and the applicability of the SPI framework has been confirmed.

Keywords

Software process improvement, SPI framework

1 Introduction

In recent years, software has come to be used increasingly in various domains. Consequently, the scale and complexity of software is increasing, and development organizations are becoming large.

In order to build in higher quality and develop large-scale or complex software efficiently, it is necessary to implement the optimum software development process in a development organization. Indeed, the development process of software has become a focus of attention in recent years.

We have used Capability Maturity Model Integration (CMMI) as a road map for SPI. However, in order to improve a software development process efficiently, it is not necessarily sufficient just to introduce such an existing process improvement technique.

Regarding the improvement of a software development process, the promotion system is an important factor in determining success or failure. In fact, process improvement changes the culture of a software development organization. Development organizations vary greatly in terms of culture and it is rarely effective to apply a uniform process improvement technique as it is. Thus, it is necessary to tailor the existing process improvement technique for various organizations. Accordingly, construction of the organizational promotion system is an important task.

This paper proposes an SPI framework and introduces the results of SPI activities based on Toshiba's SPI framework.

2 Problems related to promotion of SPI activities

Within Toshiba Group, process improvement based on the ISO 9001 quality system has been promoted. Toshiba Group's experience of the promotion of process improvement indicates that the following issues may create problems in the setting up of an effective organizational promotion system.

1. Structure of the promotion organization: Considering the impact on the culture of the organization, an ad hoc promotion system for process improvement never functions effectively. A hierarchical management system is effective for the management of a large organization. In addition, it is necessary to build an organizational promotion system for each level of the hierarchy.

2. Infrastructure for process improvement promotion: Even if an organizational promotion system is established, the SPI activities of each department cannot be promoted effectively without the appropriate infrastructure. This infrastructure can supply common information for SPI through a database. It is also useful to achieve a common recognition of the importance of SPI and, on that basis, guide SPI activities effectively.

3. Establishment of an SPI community: An SPI community is very important for extending the SPI movement. In the 1990s, many organizations in Japan started to get ISO certification with a view to competing more effectively in the global marketplace. Many organizations that lacked a strong technical identity rushed into certification. As a result, SPI activities in field projects slowed. If an SPI community exists, it is possible to avoid such a situation.

4. Improvement of SPI capabilities: To promote process improvement systematically, it is important to achieve a common, accurate recognition of the importance and contents of process improvement. Only after this common understanding is achieved can software process improvement be implemented efficiently. In addition, if the skills of the people involved in SPI are improved, SPI activities can be promoted more effectively. Therefore, it is useful to provide training courses for the people involved in SPI.

3 Proposal of an SPI framework

When in-house deployment of quality control tools was carried out in the 1990s, it was recognized that there were organizations where use of certain tools continued, and organizations where use was discontinued. When considering "the structure for promoting SPI activity", the features of organizations that use tools continuously were investigated and the results of this investigation were reflected in the SPI framework. The analysis results are shown in Fig. 1.
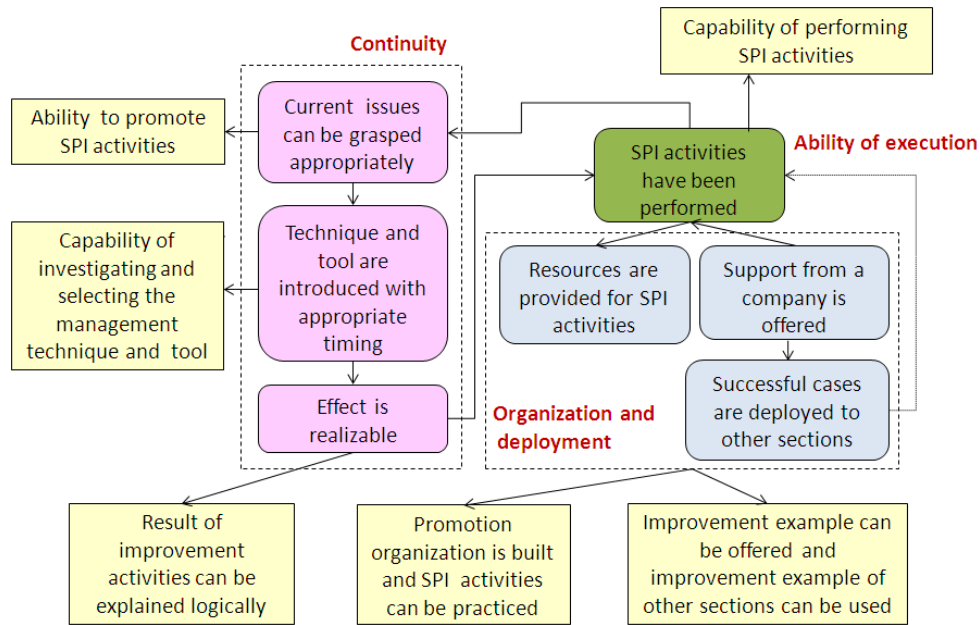
Figure 1: Features of organizations capable of continuing SPI activities

The most important feature of organizations capable of using tools continuously is the ability to continue and carry out improvement activities. Moreover, the performance of improvement activities was found to be rooted in organization and deployment. Furthermore, the prerequisites for carrying out improvement activities continuously were clarified. Six activities that should be strengthened were defined from these analysis results. These six activities were regarded as components and the mechanism for operating those components systematically was defined as the SPI framework. The defined SPI framework is shown in Fig. 2.
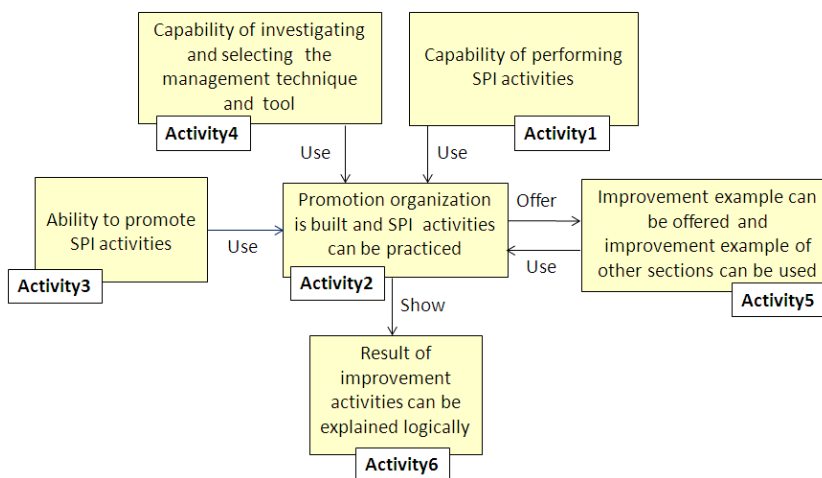
Figure 2: SPI framework

Reflecting consideration of the characteristics of a large organization, the six activities defined in order to establish SPI activity in many organizations are outlined below.

Activity 1 : Improvement model

In order to raise the maturity of an organization, it is important to determine the state where it should be from a mid- to long-term viewpoint. Moreover, in order to reach the state where it should be, it is necessary to continue improvement. Selection of a model that can show the road map from a mid- to long-term viewpoint for SPI activity promotion throughout the company and the methodology for continuing improvement activities effectively and efficiently is required.

Activity 2 : Construction of promotion organization

Human resources are indispensable for supporting the SPI activity in many development departments. Moreover, it is necessary to select project team members in light of the characteristics such as products, markets, and the enterprise, in order to ensure linkage between a support department and development departments for practice of suitable SPI activities.

Activity 3 : Acquisition of improvement technology

Product development by cross-functional teams whose members are drawn from two or more organizations is increasing in response to product diversification. A common concept of a development process and a common understanding of improvement activities are key factors for the success of a project. Moreover, it is necessary to consider the importance of raising the level of the entire company and thus ensuring consistency across the SPI activities of diverse development departments.

Activity 4 : Introductory promotion of management technique and tool

Many resources of an organization are used for product development in a development department. Therefore, even if the validity of a management technique or a management tool is recognized, it is difficult to assign all the necessary resources. Support for customization is required so that the management technique and the management tool can be used in product development and also be developed in the development department.

Activity 5 : Promotion of information sharing

SPI activity in a development department tends to become inward-looking and disconnected from what is happening elsewhere in the organization. Therefore, an environment conducive to the acquisition of information, including the activity situations of other departments, development processes, best practice, etc., is required. Moreover, the results of SPI activity may not be evident. Therefore, it is important to maintain the motivation of SEPG or SQAG members.

Activity 6 : How to show the effect of SPI activity

Investment in SPI activity has an indirect and long-term effect. On the other hand, it is important to show the effect of investment in SPI activity to management in order to maintain its commitment. Therefore, it is necessary to make a logical connection between the result (improvement in maturity) and the effect (improvement in quality, delivery lead time, and cost) of activity, and to visualize them.

4 Toshiba's SPI framework

We have used CMMI as a road map for SPI. The process assessment method is used in order to understand the gap between CMMI and the process used in the organization. The improvement plan is developed and executed based on the detected gap. This improvement cycle is crucially important for the effective implementation of SPI activities. For the improvement cycle, we modified the IDEAL model. IDEAL consists of 5 phases, namely Initiating, Diagnosing, Establishing, Acting, and Learning.

To promote SPI activities in Toshiba Group, we have established Toshiba's SPI framework shown in Fig. 3. The framework has the following features.

1. SPI activities are promoted by SEPG, which has a three-level hierarchical structure.

2. Techniques, such as CMM and IDEAL, and methodologies are not adopted as they are. They are tailored for practical use by each hierarchical level of SEPG and incorporated in the process improvement promotion organization.

3. Corporate-SEPG and Company-SEPG provide consulting services using their knowledge and various materials (CMMI guidebook, improvement solutions, assessment techniques, etc.).

4. The Toshiba Software Forum and the Toshiba SPI Workshop are important events for expanding the SPI community. The forum consists of a keynote speech, presentations, and various special events (SIG, panel discussions, etc.). The aim of the workshop is to facilitate detailed discussion of selected topics.

5. To improve the capabilities of SEPG members, Corporate-SEPG provides training courses. The SEPG leader training course is essential for improving SEPG leaders' skills and cultivating an SPI network.

6. All materials (training texts, event materials, samples of process standards, CMMI guidebook, etc.) are available on the corporate website.

7. Based on the SPI activities data, an SPI activities report is published periodically. This report details the progress of SPI activities in Toshiba Group, the current maturity level, current SEPG activities, and the effects on SPI activities.
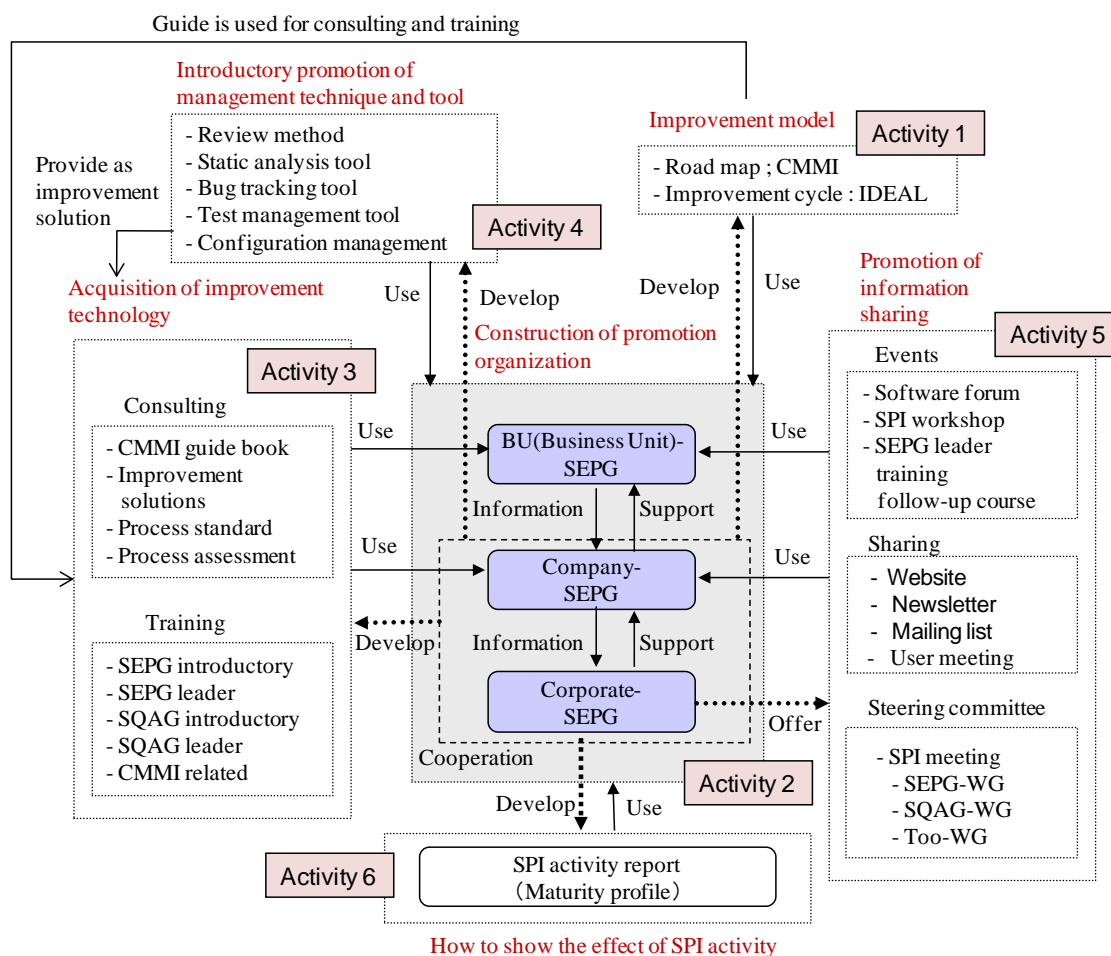
Figure 3: Toshiba's SPI framework

5   SPI activities based on Toshiba's SPI framework

The purpose of software process improvement activities is to achieve the targeted QCD (quality, cost, delivery) in software development and to establish an organizational culture conducive to continuous improvement. Our final goal is to establish a kaizen culture in Toshiba Group. For this purpose, the

Corporate-SEPG, consisting of 6 people, was established in April 2000. Corporate-SEPG consisted of 25 people in 2010. In this section, we explain the SPI activities based on Toshiba's SPI framework.

5.1   Toshiba's SPI promotion policy

SPI promotion does not cover all the development conducted by the software development departments of Toshiba Group. Corporate support is provided for organizations whose management is committed to SPI activity.

The fundamental SPI deployment policy is bottom-up. The major activity in the initial phase (2000 - 2001) was support of the implementation of SW-CMM and training activities. Following completion of that work, the number of BU-SEPGs increased rapidly through the sharing of practical examples without top-down instructions.

To change the culture of a software development organization, a bottom-up approach is more important than a top-down approach. We constructed and established various communities for SPI promotion. In the case of a bottom-up approach, these communities are the key factor for accelerating SPI activities.

5.2   Toshiba's Long-term SPI plan

In order to deploy SPI activities throughout Toshiba Group, Corporate-SEPG needed to earn the trust and cooperation of the people who promote SPI activities in development departments. For this purpose, we explained the long-term SPI plan (Fig. 4) to the entire Toshiba Group. SPI activities initiated from 2000 onward in Toshiba Group have been carried out as planned.

1st Phase: 2000-2003
Establishment of
SPI framework

<Contents>
• Promotion of CMMI, SPI, SEPG, etc.
• Organization of SEPG in development departments
• Development and promotion of SPI-related training
<Organization>
• Establishment of a three-level SEPG hierarchy

2nd Phase: 2004-2007
Deployment of
SPI activities

<Contents>
• Support for high maturity and quality (process aspect and product aspect)
• Establishment of relationship between CMMI and project management
<Organization>
• Establishment of SQA policy and organization

3rd Phase: 2008-2010
Establishment of
SPI culture

<Contents>
• Introduction of cutting-edge techniques for processes and products
• Deployment of information sharing
<Organization>
• Establishment of skill standards for SPI
• Introduction of SPI-related skill titles (SEPG leader, SQA leader, SPI trainer, etc.)

Figure 4: Toshiba's long-term SPI plan

5.3   Three-level hierarchical SEPG structure

SEPG is divided into three classes as shown in Fig. 3 for promoting SPI activities. We have identified the process improvement activities that change the culture of an organization. In order to promote these changes, an adequate grasp of the present culture and flexible action are required. The software products developed by Toshiba Group are numerous and diverse, ranging from large-scale

software such as power plant control software to embedded software such as that for digital TVs. Moreover, within Toshiba Group, culture varies among the many business units developing these products.

How to systematize an organization for promoting process improvement throughout a large organization such as Toshiba Group is an important issue. Our solution was to organize three classes of SEPG. The three classes of SEPG are Corporate-SEPG, Company-SEPG, and BU(Business Unit)-SEPG.

5.4   SEPG and SQAG training courses

SEPG leaders play an important role in determining how effectively and efficiently SPI activities are performed. Corporate-SEPG developed an SEPG leader training course for the SEPG leaders. This training course is spread over 2 - 4 days per month for a period of four months.

The overview of this course is shown in Fig. 5. The purpose of this course, which involves lectures and homework, is to provide the trainees with a detailed understanding of each phase of IDEAL, a basic understanding of CMMI, and knowledge of techniques for promoting SPI. The course provides an opportunity for members of various departments to get to know one another and thereafter to share information related to SPI.

The need for a training course for SQAG members became apparent in line with progress of SPI activity based on the three classes of SEPG. A SQAG introduction training course and a SQAG leader training course were developed in 2006 and offered from 2007 onward. The structure of the SQAG leader course is similar to that of the SEPG leader course shown in Fig. 5.
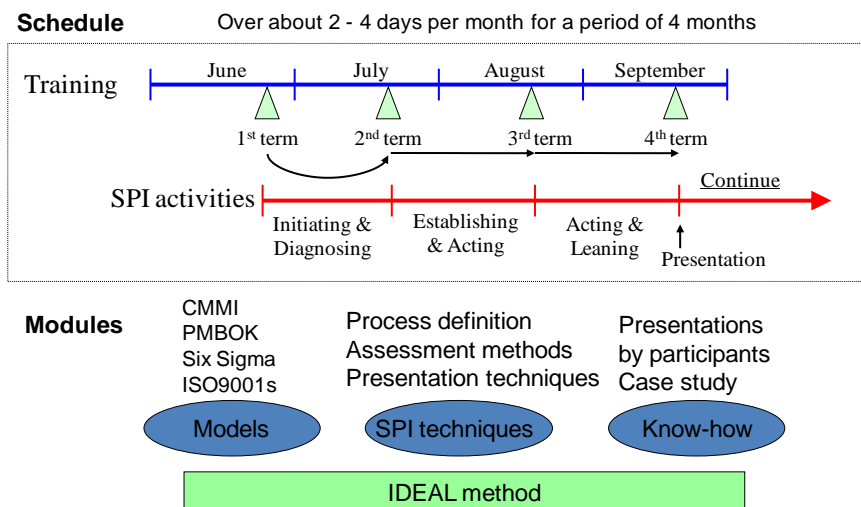
Figure 5: Overview of the SEPG leader training course

5.5   Importance of the SPI community

The people in Toshiba Group have been eager to construct a knowledge-sharing community transcending organizational barriers. There is widespread recognition that community power is the key to solving various problems concerning SPI.

Typically, most of the requirements and functions (user interfaces) of a software system to be developed are provided by the customer. Software developers can only show their creativity in the process of implementing these requirements and functions as a system.

Recently, many organizations have become increasingly sensitive about information disclosure. Of course, we must exercise care in the handling of information provided by customers. But mutual

exchange of process-related in-house information through a mechanism is important and useful for the SPI community.

In Toshiba, organizations are sharing common issues in the software process and jointly trying to solve problems. Rapid growth of the number of SEPGs (Fig. 6) indicates this open-process concept is accepted in the Toshiba software community.

6 Results of SPI activities

From April 2000 onward, we applied SPI activities consisting of consultancy services based on CMMI and the IDEAL model, training courses, improvement solutions, various events, and information sharing. Owing to the positive results obtained, it was judged in 2002 that our SPI activities should be extended to the entire Toshiba Group. In this section, we present the effects of the hierarchical SEPG structure, the maturity profiles of development departments, and the effects of improvement solutions.

6.1   Effectiveness of the hierarchical SEPG structure

Fig. 6 shows the number of BU-SEPGs and Company-SEPGs. Since the establishment of the framework shown in Fig. 3, the number of BU-SEPGs has increased rapidly. This indicates that the organizational infrastructure required for systematic process improvement has been built and that SPI activities have been promoted. The key factor responsible for the success is the establishment of the hierarchical SEPG structure. It is particularly effective for Company-SEPGs to support BU-SEPGs in cooperation with the Corporate-SEPG.
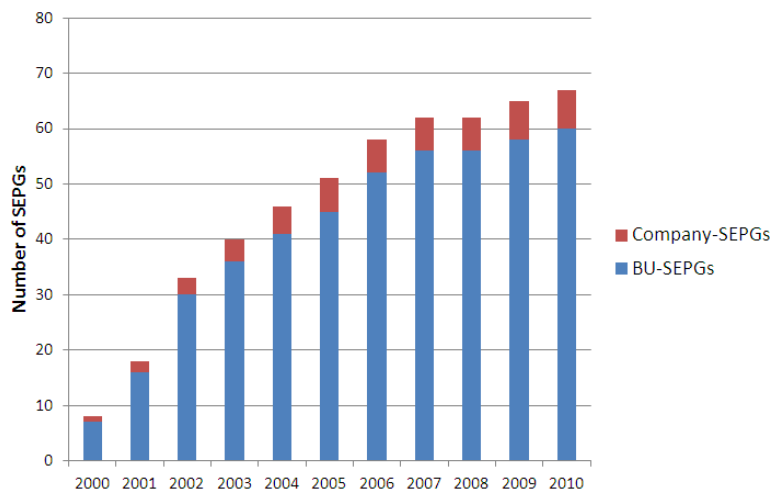
Figure 6: Growth in the number of SEPGs

There were 17 BU-SEPGs in 2001. Corporate-SEPG supported 16 of the 17 BU-SEPGs at that time. That is, the direct support ratio was 94%. However, the direct support ratio fell to 33% by 2010 because:

1. The main companies involved in software development organized Company-SEPGs, which took over support of BU-SEPG activities.

2. The capabilities of SEPG leaders were improved by the SEPG leader training course.

6.2   SPI talented people's training

CMM-related training was provided from 2001 and subsequently the SEPG leader course and the SQAG-related course have been provided. The numbers of people taking CMMI-related, SEPG leader and SQAG-related training course are shown in Fig. 7. Over 1,000 people have taken the CMMI-related training course. Moreover, about 20 people take the SEPG leader course every year, and about 150 people have taken the course so far. There are case of several people from the same department taking the SEPG leader course. This shows that the training of SEPG leaders is progressing purposefully.
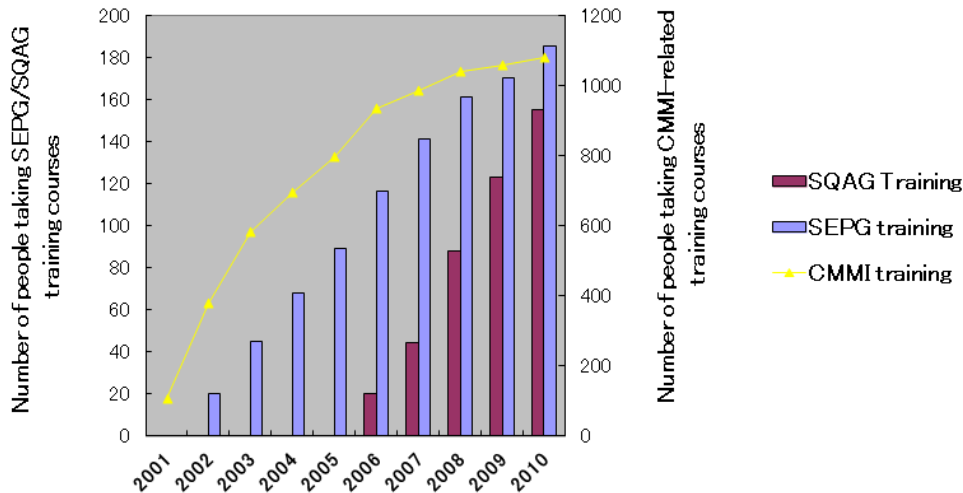
Figure 7: Growth in the number of SPI talented people

6.3   Maturity level

Fig. 6 shows that the three-level hierarchical SEPGs for SPI activity promotion are maintained. Moreover, there are about 60 development department SEPGs. About 80% of the departments consider that SEPG is required at the time an activity starts. Next, the situation of the maturity of an organization is confirmed. Fig. 8 shows the accumulated results of in-house formal assessment since 2000. Half of the departments that practice SPI activity use in-house formal assessment. This figure shows that the number of development departments attaining a maturity level of three or more is increasing every year. From this, it can be judged that the maturity of development departments is improving steadily. Moreover, from the data gathered for SPI activity reports, it can be judged that the development processes of the departments that do not use in-house formal assessment are improving steadily. Additionally, in line with the spread of SPI activity, the methods of various improvement activities such as those reported in [5] and [6] are being proposed.
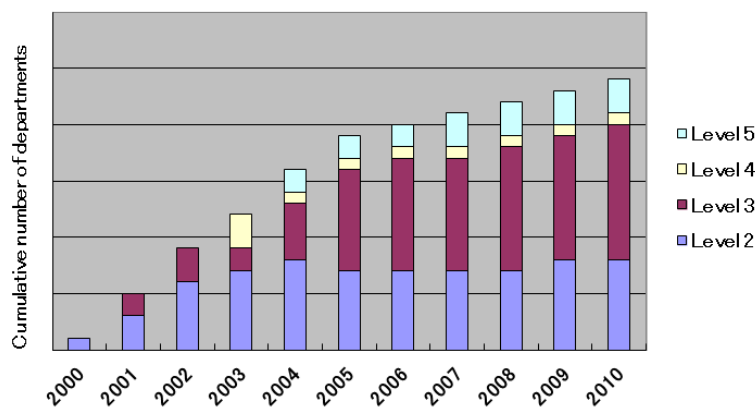
Figure 8: Maturity profile based on the results of in-house assessment

7 Conclusion

This paper shows the results and effectiveness of Toshiba's SPI framework for promoting process improvement activities in a company that includes a large number of development departments. The SPI framework promotes the process improvement activities, and contributes greatly to the improvement in process maturity of developments. For over ten years, we have been constructing Toshiba's SPI framework and promoting process improvement activities using it. As a result, maturity levels have improved in a large number of development departments, and the applicability of the SPI framework has been confirmed.

8 Literature

1. Gargi Keeni, "The Evolution of Quality Processes at Tata Consultancy Services", IEEE Software, vol.17, no.4, July/Aug. 2000, pp.79-88.

2. Hideto Ogasawara et al., "Practical Approach to Evolve SPI Activities in a Large-Scale Organization", Proceedings Volume I, 3rd World Congress for Software Quality, 2005, pp.181-190.

3 Hideto Ogasawara, Takashi Ishikawa, Tetsuro Moriya: Practical approach to development of SPI activities in a large organization: Toshiba's SPI history since 2000, ICSE 2006(pp.595-599), 2006.

4. Robert McFeeley, "IDEAL : A User's Guide for Software Process Improvement", Handbook CMU/SEI-96-HB-001, 1996.

5. Takuro Iida, Hideto Ogasawara, Takumi Kusanagi: Improve to Meet Your Organizational Needs using CMMI Continuous Representation - Good-bye, Level Hunting! -, 4WCSQ (2008).

6. Hironobu Aoki,Tetsuro Moriya,Hideto Ogasawara: Workshop Driven Software Process Improvement, 4WCSQ (The Fourth World Congress for Software Quality) , 2008.

9 Author CVs

Hideto Ogasawara
Mr. Ogasawara has more than10 years of experience in SPI promotion at Toshiba Corporation, where he is managing SPI activities and developing software testing techniques in Toshiba Group as a leader of Corporate-SEPG.

He is the director of the Japanese Software Process Improvement Consortium (JASPIC).

Takumi Kusanagi
Mr. Kusanagi has more than 10 years of experience in SPI promotion at Toshiba Corporation, where he is managing SPI activities in Toshiba Group as a manager of Corporate-SEPG.

Minoru Aizawa
Mr. Aizawa has more than 10 years of experience in SPI promotion at Toshiba Corporation, where he is managing SPI activities and developing software quality management techniques in Toshiba Group as a leader of Corporate-SEPG.

Successfully Transitioning a Research Project to a Commercial Spin-out using an Agile Software Process

Phelim Dowling
TSSG Waterford Institute of Technology
Carriganore, Waterford
pdowling@tssg.org
http://www.tssg.org

Abstract

The ultimate success of any research activity is to see it bear fruit in terms of real life use and commercial success. A key element in driving a good concept or idea through the various research and development stages and into full commercial use is the software process that supports it. In the early days of its evolvement the product will require less in terms of unit test coverage and automated test packages and more in the way of room to research and discover the innovation that will make the product unique and of high value. However, as the project progresses and a horizon appears with capital investors and large customer bases, the supporting software process needs to adapt fluidly to these evolving requirements. Efficient use of resources, shorter release cycles and better levels of quality coverage are a necessity to meet stakeholder demands for new features, better features and all to be delivered more quickly. This paper examines the successful transition of a research project to a fully fledged commercial entity with an emphasis on the software process and quality methodologies used.

Keywords

Software Process, Agile, Verification and Validation, Automated Testing, Release Cycle, Commercialisation.

1 Introduction

Today we live in a world grappling with financial instability, recessions and high unemployment. The value of innovation and research as a contributor to moving society forward and beyond the downturn is well understood [1]. However, the funding streams to drive this activity are not as bountiful as they were previously. Any new research activity has to have a well thought out goal of achieving demonstrable benefits in terms of societal or economic improvements. It will struggle to win the necessary funding without this. To this end, in the area of ICT, a proven software process for delivering real life results from a funded research project is essential. Well established practices from the Agile methodologies can be adopted and applied to a project from its early days and then used to drive the concept through to a commercially viable entity and thus delivering the tangible impact required [2].

1.1   Background

The TSSG (Telecommunications Software and Systems Group) [3] is an internationally recognised centre of excellence for ICT research and innovation. The organisations work is undertaken across core Research Units (RU's) relating to a range of themes such as:

- Experimental Facilities Management

- Bio-Inspired Networking

- Data Mining & Social Computing

- Mobile Platforms, Messaging and Middleware Technologies

- Security

- Financial Information Systems

The activities within these units vary from basic research to applied research to commercialisation projects that deliver innovative and maximum impact products to customers and partners. This paper will focus on a real case example of an idea that was initially funded as a research project within the TSSG and eventually developed to the stage where it attracted significant venture capital funding. Securing this finance created the space for the project team to spin out as a separate commercial entity and create new jobs for the region.

1.2   The project and the company

The project and thereafter the company that journeyed through this process is FeedHenry [4]. The platform put forward by FeedHenry supports the development, deployment, integration and management of secure mobile apps across the entire organisation. Their system allows for the creation of mobile apps using standard web technologies (HTML5, JavaScript, CSS) in a cloud hosted environment followed by deploys to the main mobile platforms (iOS, Android, RIM, WP7, Nokia's WRT) from the one build.

This success story would not have been feasible without a supporting software process behind it. This process is championed by the Experimental and Facilities Management (EFM) group within the TSSG. The EFM team provide services to the other RU's within the organisation in terms of verification and validation (V&V), testbed management and design and usability. With dedicated resources from this team embedded in the FeedHenry project from very early on, the evolution from funded research through to spin out company was skilfully managed and serviced.

Having a process that evolved as the product matured enabled the project to meet its goals and maintain expected quality standards from early research days, to aspiring spin-out and onto fully established commercial unit. We will explain in the next number of sections the strategy employed to achieve this significant transformation.

2 The Research Project Days

The TSSG has a well defined and published agile software development process [5] that has been championed across research and early commercialisation projects by their internal EFM team since 2005. It incorporates key aspects of the agile methodologies and promotes a team based approach where all team members are capable of fulfilling the processes and practices. Some of the key agile tools used within the TSSG include:

- Early, active and regular customer engagement

- Short iterations (2 weeks) with engineering releases every 3 months

- Test driven development

- Continuous integration

- Pair programming

- Code reviews

- Aggressive refactoring

- Daily stand-ups

- Bi weekly Iteration planning meetings

- Usability sessions

- Strong defect management process

All of these ingredients add significant value to the work produced from the TSSG. The FeedHenry team and product benefitted hugely from this early grounding in software process adherence. This solid foundation made it successful as a research project and set it on the right course for future commercial success.

3 Commercial Spin-out

3.1   The need to become lean

As the FeedHenry team expanded and a future picture emerged that demanded more frequent releases of features, a more lean process was required. No longer would the traditional 3 month release cycle of the research project model support the business need to get features to customers on a regular basis. The pursuit of any successful ICT start-up is to stay ahead of the curve in terms of gauging customer trends and then delivering on those expectations before any potential rivals [6]. This drive for a competitive edge inspired a new software development process that made the most efficient use of the resources available and allowed for peace of mind in terms of delivering new features of high quality while also covering regression of existing components. This new process, of course, carried forward all the agile tools and methods that were relevant previously but maybe applied slightly differently.

3.2   A new release cycle methodology
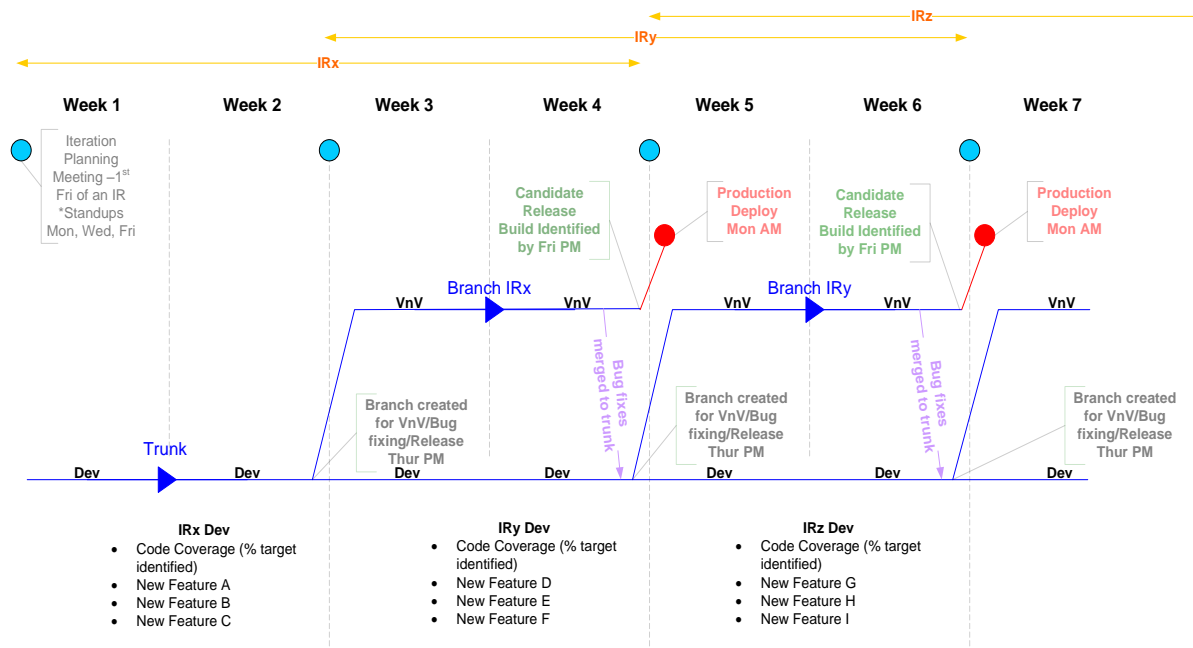
Figure 1: Overlapping 4 week iterations with bi-weekly production releases

It was decided to allocate 4 weeks to each iteration. The first 2 weeks would be focused on the development of new features. The second 2 weeks would be reserved for testing (V&V) and bug fixing. However, the key to this process was to overlap each iteration by 2 weeks. This had numerous advantages in terms of using the available resources efficiently and delivering quality releases on a frequent and regular basis.

The test team were not waiting for features to be released. They could anticipate and plan for the features being delivered and then spend 2 weeks verifying the planned release on a dedicated test branch. When the process was up and running the test team were guaranteed a new iteration release every 2 weeks followed by daily releases with bug fixes.

The engineering team had no down time either. After delivering a set of features for one iteration, they immediately started work on the next set of deliverables. They of course needed to allocate some time to bug fixing within this next 2 week window - for the parallel test phase happening on branch for the previous feature drop.

A new production release with sound process and rigorous testing foundations was deployed and available to customers and stakeholders every 2 weeks.

The idea of overlapping iterations has had some debate within the agile community. Some credible and proven experience has advised against it [7]. The strategy of staggering multiple development iterations can prove problematic to manage. However, in the case of FeedHenry, we are overlapping the development and test iterations in a manner that facilitates easier release planning, establishes a concrete test approach and makes the most efficient use of resources. A similar process has been used successfully by the Google Chrome team [8].

4 The Process Explained

4.1 Trunk and Branch

- The engineering team created features and improved code coverage on trunk in 2 week cycles. Code freeze was on the Thursday @ 2 PM of the 2nd week.

- Branch IRx was then created on Thursday PM.

- The V&V team deployed this new branch to their test environment and proceeded to perform a full test cycle on this branch – also lasting 2 weeks:
    - Functional testing
    - Verify new features
        - and create associated automated test cases for same
    - Verify bugs
        - {a} fixed on trunk in the previous 2 weeks
        - {b} merged from previous branch
        - {c} new bugs raised and fixed on Branch IRx
    - Load testing if required
        - Apache JMeter [9] was the tool of choice for executing load testing on the FeedHenry project
    - Manual testing
        - generally performed for apps on hand held devices
    - Automated testing
        - regressed all existing features across multiple browsers

- At the same time that the V&V team was performing system test on the new branch, the engineering team started new feature development, code coverage improvements, etc. on trunk for IRy. The fact that the majority of V&V resources were focused on the branch meant that a genuine commitment to maintaining high levels of code coverage on the build process was required.

- A candidate release build for branch IRx was identified by the V&V team on Friday @ 1 PM of the 2nd week of test. The final bug fixes on branch should have been completed by EOB on Wednesday.

- Bug fixes were merged from branch to trunk on an ongoing basis. The final merge was performed on Thursday PM before the new branch was created. Late bug fixes implemented on the branch were merged directly to the new branch.

- The production deploy was executed by the Operations team on Monday AM.

- Note: The V&V team maintained 2 test environments. After a candidate release build was identified for a branch, that release was left untouched on environment A. This environment acted as a mirror for production in terms of debugging issues or deploying hot-fixes. Deploys and testing for the next iteration were executed on environment B. The function of each environment switched for every iteration.

4.2   Planning and Execution

Iteration planning meetings were held on the first Friday of every iteration (i.e. every 2nd Friday). This meeting was used to plan the next 2 weeks of development work on the trunk and discuss the test strategy and any known bug fixing to be carried out on the new branch. Targets to increase unit and acceptance test code coverage on the build would also be discussed and agreed here.

Daily standups were run on Mondays, Wednesdays and Fridays – except Fridays where there was an IR planning meeting.

An overall roadmap which looked at the 3 to 6 months plan was also maintained and discussed at staff level.

Bug review meetings were held once per week. Initially the project maintained separate systems for bug tracking (Bugzilla [10]) and for feature planning (Xplanner [11]). As the process was streamlined the disjoint between maintaining these two systems became evident. A migration to a tool that encompassed bug management, feature tracking and time management was required. The tool chosen was Redmine [12] and this proved very successful.

4.3   Features on branch

The design and reasoning behind this 4 week iteration (but 2 week release) process was to facilitate the frequent delivery of new features that would have 2 weeks of development time followed immediately by 2 weeks of full system test. This ticks most boxes for an early stage start-up company in terms of staying ahead of competitors in the race to market while not dropping the ball on quality.

But another benefit inherent in this method of operation is that it incorporates a strategy to release even quicker to market if the need arises. On occasions during the FeedHenry project, a hot requirement would arise that would yield immense benefit to the company if a swift route to a production release could be accommodated. This was achieved by allowing a specific feature to be developed on the current branch in test. It meant that the feature could be live in less than 2 weeks. The use of this fast track to production was used only when critical as it has obvious short cuts in terms of quality. However, in each instance, a risk analysis was performed and the confidence provided by the code coverage stats, automated regression testing and a well managed process meant this was a valid route to a live deploy for very specific requirements.

4.4   Towards continuous delivery

The success of the fast track deploy scenario described in the previous section made the team analyse the requirements that would allow for daily deploys or continuous delivery. Some key improvements to enable this aggressive release strategy were identified. A greater level of code coverage at build time would be necessary. An automated installation procedure to perform rapid deploys to the test environment and also the production system would need to be developed. In conjunction with this rapid deploy system, a reliable rollback capability would be needed to return the system to a known good state should problems arise. Finally, the automated test suite would be required to execute faster with multiple tests running in parallel.

5 Automated Testing

5.1 A fundamental aspect of the release cycle

A key enabler of the new process adopted was a suite of automated test cases that performed functional and integration testing on the core FeedHenry platform. These tests verified > 90% of the features available, across multiple browsers and on a daily basis. They were designed to present a detailed report via email of all the tests executed, their pass/fail status, screenshots of all relevant pages and various other artefacts produced during the test execution.

The benefits reaped by maintaining this automated test suite were crucial to the success of the project and its evolution to spin-out company [13]. It provided early and clear evidence of a broken feature or bug. An accurate sense of product stability and functionality was gleaned as > 90% of functionality was exercised and regressed on a daily basis. It also acted as a safety net when a hot feature needed to be fast tracked into production. The automated tests may not have covered the new feature but they gave a sound analysis of the effect (if any) of the new code on all existing features.

The package of automated tests focused on the FeedHenry platform front end. The tool employed to create and execute the tests was Selenium. It's use on this project is described in the next section.

As mentioned previously, the testing performed for apps on devices was carried out manually on physical hand-sets or on device simulators. A test app that could be deployed to a range of handsets was used to verify an array of API calls and other 'on-device' features. An analysis of certain tools that would provide access to a wide range of devices, along with automated test facilities, was undertaken. Offerings from Perfecto Mobile [14] and Device Anywhere [15] were trialled and examined. However, the costs involved with using these products out-weighed any expected benefits and process improvements that could be achieved for this project.

5.2 Recording and executing the tests
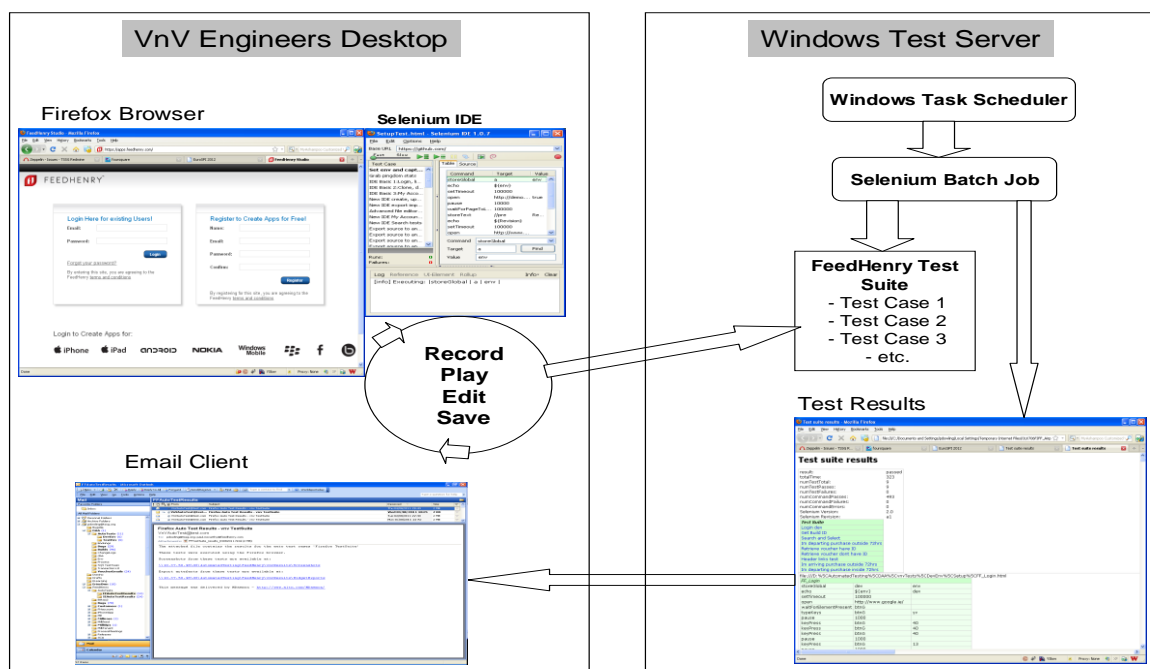
Figure 2: The procedure for recording and executing the automated tests.

- As new features were created, time was allocated during the test cycle to create new automated tests that would verify their functionality and stability through future releases.

- The automated tests were created using the Selenium IDE [16]. This is an add-on available for the Firefox browser. A relatively simple to use tool, the tests were recorded, played back, edited and then saved to the Windows test server.

- A recurring task was created on the test server in the Windows task scheduler. This would kick off the Selenium Remote Control (RC) tests at 5am every day.

  - Selenium RC is a java based server which launches and kills browsers, interprets and runs the Selenese commands passed from the test program, and acts as an *HTTP proxy*, intercepting and verifying HTTP messages passed between the browser and the AUT.

  - It can be used to execute tests using a range of browsers including Firefox, IE and Safari.

- The selenium batch job was created to perform 3 functions:

  - Execute the full suite of test cases across the specified browsers

  - Write the results to disk on the test server

  - Email the results to the appropriate team members

6 IPR Transfer

A very important aspect of creating a commercial company from a research project is the management of the IPR (intellectual property rights). A well governed software process that is maintained throughout the full life cycle of a research project can save enormous amounts of time during the formal IPR transfer of that project to a new company. In particular, the process can help ensure that the IP (i.e. code base) of the new company is maintained in a structured and clear fashion so that the following can quickly and easily be identified:

- Code developed and wholly owned by the new company which may/not be clearly identifiable (i.e. consistent file headers with copyright statements)

- Third party code used by the project, unmodified, which may/not be subject to specific license terms

- Third party code used by the project, modified, where the base code is not owned by the company but the modifications are

If the above items are not easily identifiable, the IPR transfer process could potentially incur very significant delays and elevated costs.

7 Conclusions

The TSSG engages in basic and applied research along with innovative commercial projects. It follows a model that tries to grow research and innovation outputs into tangible benefits for the community, e.g. job creation. The FeedHenry project has proven that this vision of a novel idea or a spark of inspiration being transformed into a viable commercial entity is valid and real.

The case study described in this paper outlines how engaging in software process, specifically agile, from an early stage is a sound strategy. Adapting and evolving the process as the project matures is a necessity to meet the quality and feature delivery expectations at any point in the life of that project.

Having a comprehensive and reliable automated test system is essential for providing the confidence to deploy new features on a regular basis. In conjunction with a V&V team that will run the required manual test spirals and police the agreed process, the combination is a recipe for success.

FeedHenry continues to grow and is now regarded as one of the leading ICT start-ups in Ireland. The award winning company continues to attract new customers globally in a wide range of sectors including telecoms, healthcare, retail, financial services as well as ISV's and developers [17]. The Feed-Henry journey is a story worth studying and one that is destined to continue and grow.

8 Literature

[1] University Alliance; "21st Century universities: engines of an innovation-driven economy", 2010; ISBN: 978-1-908190-01-7

[2] Vijayasarathy L, Turk D; Journal of Information Technology Management; "Agile Software Development: A Survey of Early Adopters"; 2008; ISSN #1042-1319

[3] TSSG {http://www.tssg.org/}

[4] FeedHenry {http://feedhenry.com/}

[5] Cleary F, Dowling P, Kehoe C; "An agile software process successfully imlemented within an expanding research and commercial organisation"; Euro SPI 2008 Industrial Proceedings; ISBN 978-87-7398-150-4

[6] Benihoff M, Adler C; "Behind the Cloud – the untold story of how salesforce.com went from idea to billion-dollar company and revolutioized an industry", 2009

[7] Cohn, M; 'Blog – Succeeding with Agile: Synchronize rather than overlap sprints"; http://blog.mountaingoatsoftware.com/synchronize-rather-than-overlap-sprints, 2009

[8] LaForge A; "Chrome Release Cycle"; https://docs.google.com/present/view?id=dg63dpc6_4d7vkk6ch&pli=1, 2011

[9] Jmeter {http://jakarta.apache.org/jmeter/}

[10] Bugzilla {http://www.bugzilla.org/}

[11] XPlanner {http://xplanner.codehaus.org/}

[12] Redmine {http://www.redmine.org/}

[13] Dustin E, Garrett T, Gauf B; "Implementing Automated Software Testing: How to save time and lower costs while raising quality", 2009

[14] Perfecto Mobile {http://www.perfectomobile.com/}

[15] Device Anywhere {http://www.deviceanywhere.com/}

[16] Selenium {http://seleniumhq.org/}

[17] BloombergBusinessweek {http://investing.businessweek.com/research/stocks/private/snapshot.asp?privcapId==134124470}

Author CVs

Phelim Dowling

Phelim has 15 years experience in the IT industry working in a variety of roles and companies. Starting his career with Intel he rose to the position of technical/project lead at their R&D facility in Oregon. Here he was responsible for coordinating and executing the development, testing and deployment of numerous mission critical products into Intel's manufacturing sites worldwide.

On moving back to Waterford he spent time as a Test Engineer with Waterford Technologies. Following this he worked as a consultant project manager specialising in IT project delivery in the highly regulated pharmaceutical industry. In 2006 Phelim joined the TSSG Verification and Validation team and has facilitated on projects such as iServe, Muzu, FeedHenry and more recently the EU FP7 funded OpenLab initiative.

Barriers to using Agile Software Development Practices within the Medical Device Industry

Martin Mc Hugh, Fergal Mc Caffery and Valentine Casey

Regulated Software Research Group, Department of Computing and Mathematics
Dundalk Institute of Technology, Co. Louth Ireland
{Martin.McHugh, Fergal.McCaffrey, Val.Casey}@dkit.ie

Abstract

Non-safety critical software developers have been reaping the benefits of adopting agile practices for a number of years. However, developers of safety critical software often have concerns about adopting Agile practices. Through a literature review this research identified the perceived barriers to following agile practices when developing medical device software. A questionnaire based survey was also conducted with medical device software developers in Ireland to determine what the actual barriers are to adopting agile practices. In addition a comparison is performed between the perceived and actual barriers and the results are reported.

Keywords

Agile, Methods, Scrum, Medical Device, V-Model, Waterfall, Traditional Models, Plan Driven, XP, Mission Critical Software, Medical Device Software

1 Introduction

The popularity of agile practices is on the rise [1]. Agile practices appear to offer a "silver bullet" [2] for all of the problems associated with traditional plan driven software development lifecycles. A number of surveys have been completed which reinforce this believe [3, 4]. However, a large amount of research has been conducted into the success of adopting agile practices which is broad and does not expressly focus on specific domains within the software development industry i.e. safety critical software development.

Non-safety critical software is developed in accordance with a customer's requirements, but safety critical software must be developed in accordance with both a customer's requirements and national and/or international regulatory constraints. These regulatory constraints are dictated by the region in which it is planned to market the software, be it standalone or part of a hardware device.   For example, if a medical device is to be marketed in the United States (US) it must be developed in accordance with the Food and Drug Administration (FDA) quality regulations, guidance documents and approved standards [5, 6]. Software developed for use within safety critical domains is typically developed in accordance with the Waterfall Model or V-Model software development lifecycles [6, 7]. These lifecycles are defined by upfront design with high importance placed upon the production of documentation [6]. These models produce the necessary deliverables required to achieve regulatory conformance.

Our research is focused on the development of software for use within the medical device domain. Regulatory requirements and development standards such as [8, 9] do not dictate the use of a particular lifecycle when developing medical device software. In fact they state that medical device software can be developed using a traditional, iterative and/or evolutionary approach. Despite this, medical device software developers typically develop software in accordance with the V-Model [7]. Whilst the V-Model produces necessary deliverables such as traceability between requirements and all stages of the software development lifecycle [10] it is seen as being rigid and inflexible in the event of a change once development has begun [11].

This research was initiated by performing a literature review. One of the objectives of undertaking this literature review was the identification of the perceived barriers to adopting Agile practices when developing medical device software. Also as part of this research a questionnaire based survey was conducted amongst medical device software developers in Ireland. The aim of this survey was to evaluate the findings of the literature review and to learn what are the actual barriers to adopting agile practices when developing medical device software.

The remainder of this paper is structured as follows: Section 2 provides information as to our on-going research in this area and how this aspect of our work fits into this research. Section 3 details the perceived barriers to selecting and implementing agile practices when developing medical device software based on the results from our literature review. Section 4 outlines the approach taken by questionnaire based survey conducted amongst medical device manufacturers in Ireland. Section 5 provides the results of the survey and within section 6 a comparison is performed between the perceived barriers and the actual barriers to adopting Agile practices.

2 Research Objectives

As part of this on-going research the following research questions has been identified. This research is:

1. What are the issues associated with developing medical device software?
2. What are the issues with developing medical device software using a traditional software development lifecycle?
3. Which agile practices are suited to developing medical device software?
4. To what extent do existing medical device software development lifecycles need to be tailored to incorporate suitable agile practices?

The results from the research outlined in this paper will be used to help address the third research question. By identifying actual barriers to the adoption of agile practices specific practices can be discounted and the remaining agile practices can be evaluated for suitability. These research questions were formed following the completion of a literature review. This literature review began by broadly looking at generic software development lifecycles. The focus of the literature review moved to the development of safety critical software and then onto the development of software in the medical device industry. Following this phase of the literature review, research was conducted into agile software development. This involved examining mainstream methodologies such as Scrum and XP. Once this was completed we then focused upon the adoption of agile practices in the development of safety critical software. Finally, we considered the adoption of agile practices in the development of medical device software. This literature revealed a number of perceived barriers to adopting agile practices when developing medical device software.

Following the literature review a questionnaire based survey was conducted amongst medical device software developers in Ireland. The objective of this survey was to evaluate the findings of the literature review and to learn what the actual barriers to adopting agile practices are.

3 Perceived Barriers to Agile Adoption

As discussed in section 1, software developed for use in or as a medical device must meet the regulatory requirements of the region where the device is being marketed. As a result many of the perceived

barriers to adopting agile practices in developing medical device software are associated with regulatory controls [12]. The focus of this research is the identification of the perceived barriers that have a direct impact on the development process of medical device software and the implementation of agile methods in this context. Additional barriers do exist, but a number of these are organisational barriers that do not have a direct impact on the development of medical device software. An example of such an organisational barrier is that Human Resource policies and processes do not cater for the requirements of an Agile team [13]. The literature review performed identified the following perceived barriers to the adoption of agile practices when developing medical device software.

The FDA General Principles of Software Validation (GPSV) [9] require manufacturers to explicitly document requirements prior to implementation and test procedures [14]. This would appear to be an apparent barrier to adopting agile practices as one of the fundamental principles of the agile Manifesto [15] is "working software over comprehensive documentation". Combined with this another central principle of agile software development is that requirements are fluid and changes in requirements can be easily accommodated and are even welcomed throughout a development project [16]. Without fully refining requirements prior to the beginning of a project the process of traceability can be difficult and traceability between requirements and all stages of development is required by the FDA [17].

As safety critical software such as medical device software can place patients, clinicians and third parties at potential risk, medical device software developers must perform adequate risk management activities to ensure the software they are developing is safe and reliable. Boehm and Turner [13] suggest that risk management activities can be a barrier to adopting agile practices as agile practices do not provide sufficient guidance as how to perform the necessary risk management activities.

Another perceived barrier to adopting agile practices is that software developed using agile practices is of a lower quality than software developed following traditional plan driven lifecycles [13]. As medical device software is safety critical it must be developed to the highest quality possible.

Agile methodologies such as XP recommend short releases with continuous feedback [18]. When developing medical device software it is not possible to release incomplete software and await feedback as the software must be fully tested and working before it is used in patient treatment [19].

An additional potential barrier to adopting agile practices in the development of both safety critical and non-safety critical software is the loss of management control. Agile methodologies recommend that development teams are self-organising. This process of self-organising teams removes some of the decision making powers from management [20]. This can result in a loss of management control and for agile practices to succeed organisational support is required [21].

4   Survey of Medical Device Software Developers in Ireland

In order to gain an understanding as to what the perceived barriers and actual barriers are to the adoption of agile practices we performed a questionnaire based survey with medical device software development organisations within Ireland. The literature review outlined in section 1 identified the perceived barriers to agile adoption and the survey results have been used to identify actual barriers to adopting agile practices when developing medical device software.

Within Ireland there are approximately 160 medical device manufacturers [22]. No research has been conducted to date to suggest how many of these organisations develop medical device software. As a result when determining sufficient sample size a decision was made to assume all of these organisation develop medical device software. Using sample size equations a sufficient sample size was determined as twenty organisations.

As a result of this the survey was conducted amongst twenty medical device software development organisations in Ireland with multiple responses from each organisation. These organisations ranged from small indigenous manufacturers to large multinational manufacturers. The devices produced by theses manufacturers range from Class I – Low Risk to Class III – High Risk1 products. The primary

---

1 This safety classification is defined by the European Council Medical Device Directive 93/42/EEC. Class I devices are deemed to pose low risk to patients, users and third parties and Class III devices are deemed to pose potentially life threatening risk to patients, users and third parties.

goal of this survey was to gain a deeper insight into the medical device software development industry to further assist with our on-going research. Participants who took part in the survey included all levels of the development team and internal stakeholders such as managers and senior management.

The survey was developed in accordance with "*Introduction to Research Methods: A Practical Guide for Anyone Undertaking a Research Project*" and "*Designing Social Research*" [23, 24]. These books outline effective methods for constructing and undertaking a questionnaire based survey with the objective of achieving the maximum amount of relevant information. The survey began by asking participants what was their role in the organisation and how long they have been working in the medical device industry. The responses from these questions were used to qualify the expertise of the participants and to support the validity of their responses. In addition the fact that we received more than one response from each company enabled us to validate the quality of these responses from each company.

The first piece of significantly relevant information obtained by the survey was which software development lifecycle the organisations are following. As part of this on-going research, recommendations will be made as to how adopting agile practices can resolve problems associated with the current lifecycle being followed.

Following on from this question, participants were provided with a list of activities that are required to be completed in the development of safety critical software. Participants were requested to rate how much importance they place on each of these activities and to rate how effective they deem their organisation to be at performing these activities. The objective of this question was to understand which areas of safety critical software development are being performed most effectively. Again as part of this on-going research, information is being collected that will identify which stages of development pose the most difficulty to medical device software developers. This information will eventually be used to help answer research questions 1 and 2.

Finally participants were asked a series of questions relating to agile software development. Participants were provided with a list of potential agile barriers and asked to select which barriers they perceived would prevent the adoption of agile practices. The goal of this question was to evaluate the findings of the literature review. Participants where then asked what the actual barriers were within their own organisation to the adoption of agile methods. This question was used to establish what these barriers are and if commonality could be identified across organisations as to what actual barriers exist in relation to agile adoption. A barrier being defined as an actual barrier does not imply that the barrier is insurmountable. To maximise the amount of relevant information gathered, space was provided for the respondents to add additional information and/or comments for each question as they deemed necessary.

5 *Results of Survey*

As discussed, one of the objectives of the survey was to establish the actual barriers faced by medical device software developers. The results of the survey have also been used to evaluate the findings of the literature review in this context.

The following results are preliminary as this research is still on-going. The survey revealed that 100% of the respondents who are currently marketing medical device software are developing it for use in Europe. In addition, 79% of these are also developing medical device software for use in the US.

The survey identified that 50% of the organisations are developing software in accordance with the V-Model. An important finding was that another 25% of the organisations are developing medical device software in accordance with agile practices. The remaining 25% of organizations are developing software in accordance with other development lifecycles such as the Waterfall model.

As part of the survey, respondents were asked what they believe to be the perceived barriers to agile adoption. The survey revealed that 25% of respondents reported "Lack of Documentation" as a barrier to agile adoption. In addition 25% of respondents reported "Regulatory Compliance", whilst 16% of respondents reported "Lack of Up-Front planning" and 17% of respondents reported "Insuffi-

cient coverage of risk management activities" as barriers to agile adoption. These results were consistent with the findings from the literature review.

Finally, respondents of the survey were asked what the actual barriers to adopting agile practices are. Of the respondents 50% reported "Lack of Experience", 33% reported that having to change the existing lifecycle as a barrier to agile adoption, 16% reported "Management Opposed to Change "and 16% reported team size as a barrier to agile adoption. A further 17% reported that getting stakeholder buy in as a barrier and 17% reported the level of retraining required as another barrier to agile adoption.

6 *Comparison between Perceived and Actual Barriers*

Table 1 presents a list of the perceived barriers based on our literature review and the actual barriers to agile adoption when developing medical device software based on our survey results.

Table 1 Perceived and Actual Barriers to Agile adoption

Perceived Barriers	Actual Barriers
• Regulatory Control • In-sufficient coverage of Risk Management Activities • Requirements Management • Traceability issues • Loss of management control • Lower quality software	• Regulatory Control • In-sufficient coverage of Risk Management Activities • Lack of up-front planning • Lack of documentation • Management opposed to change • Team size • Modification of existing lifecycle • Lack of Experience using agile • Getting Stakeholder Buy In • Level of Retraining Required

It can be seen in table 1 that a number of the perceived barriers are also actual barriers to using agile practices when developing medical device software. An important point to emerge from our research is how requirements can be identified and successfully managed in the context of utilising agile practises. Regulatory bodies require medical device software developers to document requirements prior to development. These requirements are then used during the development stages to provide traceability. Agile principles dictate that requirements be fluid throughout a development project and this can be seen as a barrier as we have outlined in section 3.

However, regulatory bodies do recognise the acceptability of what can be termed an agile approach to requirements. The FDA GPSV states [9]:

"Most software development models will be iterative. This is likely to result in several versions of both the software requirement specification and the software design specification. All approved versions should be archived and controlled in accordance with established configuration management procedures".

This emerged from our detailed analysis of the relevant regulations, standards and guidance documents. This was not evident from the published academic literature in this area, in fact the opposite was the case as we have stated.

This provides an example of how the perceived barriers to agile adoption can be overcome. Further research will be undertaken to evaluate and determine how each of the other barriers identified can be addressed by employing specific agile practices or by integrating agile practices with a plan driven lifecycle.

7 Conclusions

A number of barriers to adopting agile practices when developing medical device software have been identified through a literature review and questionnaire based survey. Medical device software developers must develop software in accordance with regulatory requirements and as a result a number of the barriers to agile adoption are associated with the process of achieving regulatory conformance.

Regulatory bodies and medical device software development standards do not dictate the usage of a specific software development lifecycle for developing medical device software [8, 9] . Regulatory bodies have a set of deliverables which device manufacturers must deliver. Medical device software developers can develop software in accordance with agile practices once they provide the necessary deliverables. Research conducted by Rasmussen et al and Rottier et al [25, 26], identified that no single agile methodology could be strictly adhered to when developing medical device software as no single agile methodology provides sufficient coverage of all of the areas necessary to achieve regulatory conformance. However, Rasmussen et al and Rottier et al [25, 26] did identify that selectively choosing appropriate agile practices and integrating them with a plan driven lifecycle can reap the benefits of employing agile practices whilst still producing the necessary regulatory deliverables. This is an area we plan to investigate further as part of our on-going research.

8 References

1.   Conboy, K. and B. Fitzgerald, *Method and developer characteristics for effective agile method tailoring: A study of XP expert opinion.* ACM Trans. Softw. Eng. Methodol, 2010. **20**(1): p. 1-30.
2.   Brooks, F.P., *No Silver Bullet Essence and Accidents of Software Engineering.* Computer, 1987. **20**(4): p. 10-19.
3.   VersionOne, *State of Agile Survey - The Stage of Agile Development.* 2010.
4.   http://www.ambysoft.com/surveys/agileFebruary2008.html#Downloads Accessed 04/10/2011
5.   McCaffery, F., J. Burton, V. Casey, and A. Dorling, *Medical Device Industry: Software Process Improvement*, in *Encyclopedia of Software Engineering*, P. Laplanye, Editor. 2010, CRC Press Francis Taylor Group: London. p. 528-540.
6.   Ge, X., R.F. Paige, and J.A. McDermid, *An Iterative Approach for Development of Safety-Critical Software and Safety Arguments*, in *Agile 2010*. 2010.
7.   Bulska, K. and J. Gorski, *Applying Agile Practices to the development of Safety-Critical Software*, in *ICT Young 2011, Scientific Booklets of Faculty of Electronics, Telecommunications and Informatics*. 2011: Gdańsk University of Technology. p. 65-68.
8.   AAMI, *ANSI/AAMI/IEC 62304, Medical device Software - Software life cycle processes*. 2006.
9.   FDA, *General Principles of Software Validation: Final Guidance for Industry and FDA Staff*. 2002, *Centre for Devices and Radiological Health*.
10. McCaffery, F., V. Casey, M. Sivakumar, G. Coleman, P. Donnelly, and J. Burton, *Medical Device Software Traceability*, in *Software and Systems Traceability*. 2011, Springer-Verlag.
11. Munassar, N.M.A. and A. Govardhan, *A Comparison Between Five Models Of Software Engineering.* IJCSI International Journal of Computer Science Issues,, 2010. **7**(5): p. 94-101.
12. Mishra, S. and H.R. Weistroffer, *Issues with Incorporating Regulatory Compliance into Agile Development: A Critical Analysis*, in *Southern Association for Information Systems (SAIS)*. 2008.
13. Boehm, B. and R. Turner, *Management Challenges to Implementing Agile Processes in Traditional Development Organizations.* IEEE Software, 2005. **22**(5): p. 30-39.

14. Vogel, D., *Agile Methods:Most are not ready for prime time in medical device software design and development.* DesignFax Online, 2006(July 2006).
15. Fowler, M. and J. Highsmith, *The Agile Manifesto*, in *Software Development: The Lifecycle starts here*. 2001.
16. Boehm, B. and R. Turner, *Balancing Agility and Discipline: A Guide for the Perplexed.* 2003: Addison-Wesley.
17. Weiguo, L. and F. Xiaomin, *Software Development Practice for FDA-Compliant Medical Devices*, in *International Joint Conference on Computational Sciences and Optimization, 2009. .* 2009: Sanya, Hainan. p. 388 - 390.
18. Abrahamsson, P., O. Salo, J. Ronkainen, and J. Warsta, *Agile Software Development Methods - Review and Analysis*, in *ESPOO 2002.* 2002, VTT Publications.
19. FDA, *Title 21--Food and Drugs Chapter I --Food and Drug Administration Department of Health and Human Services subchapter h--Medical Devices part 820 Quality System Regulation.* 2007, *U.S. Department of Health and Human Services.*
20. Moe, N.B., T. Dingsoyr, and T. Dyba, *Understanding Self-Organizing Teams in Agile Software Development*, in *Conference on Software Engineering, 2008. ASWEC 2008. 19th Australian* 2008: Perth, Australia.
21. Salo, O. and P. Abrahamsson, *Integrating agile software development and software process improvement: a longitudinal case study*, in *International Symposium on Empirical Software Engineering* 2005.
22. McHugh, M., *Medical Device Sector in Ireland.* Network Enterprise Europe, 2010.
23. Dawson, C., *Introduction to Research Methods: A Practical Guide for Anyone Undertaking a Research Project.* 2009: How To Books Ltd; 4th Revised edition edition.
24. Blaikie, N., *Designing Social Research.* 2000, Cambridge: Policy Press.
25. Rasmussen, R., T. Hughes, J.R. Jenks, and J. Skach, *Adopting Agile in an FDA Regulated Environment*, in *Agile Conference, 2009. AGILE '09. .* 2009: Chicago, IL p. 151-155.
26. Rottier, P.A. and V. Rodrigues, *Agile Development in a Medical Device Company*, in *Agile, 2008. AGILE '08. Conference.* 2008.

Author CVs

Martin Mc Hugh

Martin received his B.Sc. (Hons.) in Information Technology Management in 2005 and M.Sc. in Computer Science in 2009, from Dundalk Institute of Technology. He is now undertaking research for his Ph.D. in the area of software process improvement for medical devices with emphasis on the usage of agile practices when developing medical device software, as part of the Regulated Software Research Group in Dundalk Institute of Technology.

Fergal Mc Caffery

Dr Fergal Mc Caffery is the leader of the Regulated Software Research Group in Dundalk Institute of Technology and a member of Lero. He has been awarded Science Foundation Ireland funding through the Stokes Lectureship, Principal Investigator and CSET funding Programmes to research the area of software process improvement for the medical device domain. Additionally, he has received EU FP7 and Enterprise Ireland Commercialisation research funding to improve the effectiveness of embedded software development environments for the medical device industry.

Valentine Casey

Dr Val Casey is a Senior Researcher with the Regulated Software Research Group in Dundalk Institute of Technology. His previous roles include Senior Lecturer and Research Area Leader at Bournemouth University, Researcher with Lero - the Irish Software Engineering Research Centre at the University of Limerick where he also lectured. He has over 20 years' experience in the software industry. He has also provided consultancy services focusing on software process improvement and software testing in the financial and telecom sectors.

Integrating Agile Practices with a Medical Device Software Development Lifecycle

Martin Mc Hugh[1], Fergal Mc Caffery[1], Valentine Casey[1], Minna Pikkarainen[2]

[1] Regulated Software Research Group, Department of Computing and Mathematics
Dundalk Institute of Technology, Co. Louth Ireland
{Martin.McHugh, Fergal.McCaffery, Val.Casey}@dkit.ie

[2] VTT, Technical Research Centre of Finland
Minna.Pikkarainen@vtt.fi

Abstract

The rate at which agile software development practices are being adopted is growing rapidly. Agile software development practices and methodologies appear to offer the silver bullet which can solve the problems associated with following plan driven software development lifecycles. Agile software development practices offer the possibility of achieving lower development costs, increased efficiency and improved software quality. However, there is currently a low rate of publicly available information that suggests there is widespread adoption of agile practices within the medical device software domain. This is largely due to the fact that software developed for medical devices includes challenges not faced when developing non safety critical software. As a result of these challenges, medical device software is typically developed using plan driven software development lifecycles. However, such lifecycles are quite rigid and cannot accommodate changes easily. Previous research has revealed that medical device software development projects can benefit from adopting agile practices whilst still maintaining the discipline associated with following plan driven development lifecycles. This paper outlines the challenges faced by developers when developing medical device software and how shortcomings in both agile and plan driven approaches can be resolved by following a mixed method approach to medical device software development.

Keywords

Medical Device, Agile, Plan Driven, SDLC, XP, Scrum, V-Model, TDD, FDA, MDD, 2007/47/EC, Software Process Improvement

1 Introduction

Software developers are under increasing pressure to develop software faster that not only meets a customer's requirements, but exceeds them [1]. Initially, iterative software development appeared to offer a solution by aiding faster development of systems. Iterative techniques evolved and a subset known as agile appeared [2]. Larmin and Basili [3] identified Dynamic Systems Development Model (DSDM) as being the first agile methodology. Each of the agile methodologies consists of a number of agile practices. These practices are procedures defined as being highly effective and efficient [4] such as sprint planning, an open office space, daily meetings and from Scrum, product backlogs [5]. The use of agile methodologies such as XP and Scrum, in traditional software development projects are becoming increasingly popular [6].

In some case studies, agile practices have been used in the safety critical field. For instance, in Motorola, a select number of XP practices were used during the development of safety critical sys-

tems [7]. In these cases, the use of XP practices was reported to have had a 53% improvement in average quality compared to the plan-driven software development projects. A key challenge in these XP projects was to define how the changes affected the overall end date of the projects. Additionally, Drobka et al.[7] discusses the issue of documentation production. In the case of Grenning [8], the source code was not sufficiently documented for the whole system, and this resulted in the need for a high-level architecture document which provided class diagrams, scenarios and a process view of the system for developers. Furthermore, acceptance tests were not adequate to verify the traceability from end product to customer requirements. Therefore, additional verification review meetings were used to cover this gap in the verification process [7]. Grenning [8] describes how XP practices such as the planning games, small releases, simple design, test first development; refactoring, pair programming, collective ownership, continuous integration and 40 hour week are used in a large company developing safety critical systems. As a result of this analysis, he suggests that some XP practices, such as simple designs integrated with test first development and refactoring work quite well in the safety critical area. In that case, the managers were reported to be happy with the results of the use of XP practices. This was mainly due to the XP team's ability to readily produce working software instead of a high amount of documentation [8]. One of the biggest challenges revealed from this case was that resistance was caused mainly due to the decreased amount documentation. This documentation was needed, for example, to define product requirements, sustain technical reviews, support maintenance and describe interfaces. Based on the experiences of the XP projects it was still understood that the documentation was needed for maintenance and review purposes [8].

In spite of these experiences of adopting agile practices in the safety critical domain, there is still little evidence of agile practices being widely embraced when developing medical device software [9]. The reason for this is still unclear, but one potential reason is that safety critical systems must also meet the appropriate regulatory requirements [10]. As part of the Food and Drug Administration (FDA) regulations most items require formal approval and this is cited as being a reason why many medical device software developers follow a sequential software development model such as the Waterfall or V-Model [11]. Of the limited information available, medical device software projects can benefit from incorporating agile practices, but no single agile methodology can be followed when developing medical device software [12].

It is generally perceived that international medical device regulatory requirements recommend that software developed either as a component of a medical device or as a medical device should be developed in accordance with a plan driven software development lifecycle such as the Waterfall Model or the V-Model. However, the regulatory requirements and development standards do not enforce the usage of any particular software development lifecycle. Whilst no specific software development lifecycle is mandated by the regulations or standards medical device software developers typically develop software in accordance with the V-Model [13]. Following the V-Model produces the necessary deliverables required in order to achieve regulatory approval. The objective of this research is to develop a Software Development Life Cycle (SDLC) that accrues the benefits of utilising agile practices whilst still producing the necessary deliverables provided by following plan driven software development lifecycles. This paper will serve as the foundation for this research. The remainder of this paper will be structured as follows.

In section 2, the latest regulatory requirements which medical device software organisations must adhere to are outlined. In section 3, information is provided as to how regulatory bodies do not require the usage of a specific software development lifecycle and that regulatory bodies are only concerned with the specific requirements such as traceability rather than being concerned with how these requirements are fulfilled. In section 4, a mixed method software development lifecycle is proposed that incorporates agile practices with a plan driven lifecycle and finally in section 5, the conclusions and plans for future work are presented.

2 International Regulations

Medical device software developers must adhere to the regulatory requirements of the region in which a medical device is being marketed for use. Medical devices being marketed for use within the United States (US) must conform to the FDA requirements and medical devices being marketed for use within the European Union (EU) must conform to the Medical Device Directive (MDD) and its latest amendment.

2.1   European Medical Device Directive

On March 21st 2010 the latest amendment to the MDD 2007/47/EC came into effect [14]. As stated, conformance to the latest amendment to the MDD is mandatory for a medical device to be marketed for use in the EU. Competent Authorities and notified bodies within each EU member state are responsible for certifying medical device conformance with the MDD. Once a device is certified within an EU member state the device can be marketed into all of the member states.

The latest amendment to the MDD introduced a number of changes; the most significant with regards to software development is that standalone software can now be classified as an active medical device. Prior to this amendment software was considered a component of a hardware medical device. With the release of this amendment software can now potentially be the only component of a medical device and therefore subject to full regulatory scrutiny. With this change a greater emphasis falls onto the international standards which are followed during the development of medical device software. IEC 62304:2006 [15] is a harmonised standard as part of the MDD [16]. Medical device software developers are recommended to follow IEC 62304 and its aligned standards to receive guidance. However, these aligned standards can be difficult to follow and this is particularly relevant when a medical device only consists of software.

2.2   FDA Regulations

Within the US the FDA is responsible for ensuring medical devices certified for use are safe and reliable. The FDA provides guidance documents which medical device software developers are recommended to follow in order to achieve regulatory conformance. These documents include:

- General Principles of Software Validation (GPSV) [17];
- Medical Device Data Systems Rule [18];
- Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices [19];
- Draft Guidance for Industry and Food and Drug Administration - Mobile Device Applications [20];
- Guidance for Industry, FDA Reviewers and Compliance on Off-The-Shelf Software use in Medical Devices [21].

All medical device manufacturers wishing to achieve FDA approval must conform to FDA 21 CFR Part 820 Quality System Regulations (QSR) [22]. FDA 21 CFR Part 820 requires medical device manufacturers to provide sufficient evidence that a Quality Management System (QMS) was employed when developing a medical device. To accompany, this all medical device manufacturers wishing to achieve FDA approval must follow FDA General Controls [23]. These controls include provisions that relate to device registration, branding and banned devices.

3   Agile Practices in the Development of Medical Device Software

When exploring possible frameworks for developing medical device software, we discovered that developers in this domain often select a plan driven lifecycle such as the V-Model [24], rather than considering following agile practices with iterative or incremental development lifecycle. This may be due to the fact that the agile manifesto places greater emphasis on people rather than processes and also considers documentation as being a secondary output from a software development project [25]. In practice, formal communication, such as source code, test cases, and essential amount of documentation is also used in the agile software development, but not in the same way or to the same extent as in the plan-driven software development process [26]. Additionally, some of the agile principles suggest that business people and developers must work together daily and project information should be shared through informal, face-to-face conversation rather than through documentation [27].

However, regulatory bodies in both the US and Europe require medical device manufacturers to provide documentation which provides information on areas such as risk and hazard management and traceability between the requirements stage and the development stages of a software development project [28, 29]. While this is the case, both the FDA guidelines and the MDD do not mandate the use of a specific life cycle model. The FDA states:

"(the FDA) does not recommend the use of any specific software life cycle model. Software developers should establish a software life cycle model that is appropriate for their product and organization"

To accompany this the FDA also states that the selected life cycle model "*should cover the software from its birth to its retirement",* and also that the lifecycle model should support validation and verification [17]. As discussed medical device software development organisations are recommended to follow IEC 62304 when developing medical device software. IEC 62034 states:

"*The (IEC 62304) standard does not require a particular software development life cycle model*"

To accompany this IEC 62304 also states;

"*Whichever life cycle is chosen it is necessary to maintain the logical dependencies between process outputs*"

IEC 62304 does not mandate the use of a specific lifecycle and is more concerned with the regulatory requirements such as traceability and not with how these requirements are fulfilled. Both the FDA GPSV guidance document and the IEC 62034 standard discuss the use of different life cycle models including the waterfall, incremental and evolutionary models. The FDA GPSV states:

"*Most software development models will be iterative. This is likely to result in several versions of both the software requirement specification and the software design specification*"

Currently, the Association of Advancement of Medical Instrumentation (AAMI) (the publishers of IEC 62304) is in the process of completing a technical information report which will provide recommendations as to how medical device software development organisations can comply with international and FDA regulations by using agile practices [30]. This document will provide a complete mapping between the 12 agile principles [31] and each of the development stages of IEC 62304. All of this information signals that regulatory bodies acknowledge that following a plan driven software development lifecycle is not always feasible and that agile practices can be used in compliance with IEC 62304 without jeopardising the process of achieving regulatory approval.

A number of case studies have been performed in organisations that are actively utilising agile practices in the development of medical device software [12, 32-34]. A common trend emerged out of these case studies that medical device software development organisations recognised it was not possible to fully replace the existing lifecycle with a single agile methodology, but rather they integrated agile practices with the plan driven development lifecycle. Heeager and Nielsen [32] recommend wrapping the traditional lifecycle in a Scrum approach [35]. Likewise, Robres [36] explored the suitability of using Scrum to develop medical device software when attempting to achieve regulatory conformance. In these case, the benefits to be achieved when using agile practices to develop medical device software reflect those achieved when using agile practices in any software development project, they include, reduced development time, reduced costs and increased productivity [37].

4 Proposed Mixed Method SDLC

It is well established that tailoring a SDLC should be done in order to achieve maximum impact [6]. As part of this research we are focusing on integrating agile practices with a plan driven lifecycle. The aim of tailoring a plan driven lifecycle with agile practices is to develop a SDLC which produces the necessary output required for regulatory approval whilst also reaping the benefits associated with utilising agile practices. To perform this tailoring effectively a foundation of a plan driven SDLC was needed. This was required as none of the agile methodologies provide comprehensive coverage of all the necessary development stages required when developing safety critical software [38]. However, Turk [38] does recommended that a safety critical software development project can benefit from combining agile techniques with formal plan driven techniques. The V-Model was chosen as medical device software developers are familiar with the V-Model and it produces the necessary outputs required to achieve regulatory approval and provides guidance for all of the stages of medical device software development. Secondly, research was conducted into which agile practices could be integrated into this plan driven SDLC. The agile practices initially selected were identified through the use of a literature review. The agile practices and SDLC selected must conform to the regulatory requirements whilst still achieving the benefits of tailoring.

4.1   Research Methodology

As part of on-going research into the area of utilising agile practices when developing medical device software a literature review was conducted. This literature was conducted in accordance with Randolph [39]. The literature review began broadly by examining the main software development lifecycles being followed in all software development domains. The literature review then focused on software development in safety critical industries and then concentrated on software development in the medical device domain. Once this part of the literature review was completed, agile methodologies were examined. After this examination was completed the focus moved onto the utilisation of agile practices when developing safety critical software. When conducting the final phase of the literature review a number of seminal papers in the area of using agile practices when developing medical device software were identified [11, 12, 32-34, 40, 41]. In each case study performed, the organisations were initially developing medical device software in accordance with plan driven software development lifecycles. Each of the organisations identified the need to embrace agile software development practices. Consequently upon completion of the implementation of the agile practices each organisation found a noticeable improvement in software quality and increased functionality along with reduced development costs. Table 1 presents summarised findings of the literature review and of the seminal case studies identified. Table 1 shows the case studies identified and which agile practices each of these case studies utilised

4.2   Integrating Agile Practices with a Plan Driven SDLC

As stated when developing a mixed method Software Development Life Cycle a foundation is required. This foundation came in the form of a V-Model for certification proposed by Ge et al. [10]. This V-Model differs from the typical V-Model as it includes development stages for *"Hazard and Risk Analysis"* and *"Regulatory Certification".* These stages are critical which software developed for use in medical devices must complete.

Once the foundation was chosen a method was required to establish which agile practices would be merged with the V-Model. The findings of the literature review were used to establish which agile practices have so far been successfully implemented when developing medical device software. In table 1, a summary of the agile practices suitable for use when developing medical device software are outlined.

	Timeboxed Iterations	Sprint Backlogs	Task Board	Burn Down Chart	Daily Standup Meetings	Self Organising Teams	Continuous Build, Unit Test and Integration	Sprint Retrospspective	On Site Customer / Customer Proxie	Test First Development	Pair Programming	Collocated Teams	User Stories
Rasmussen et al.	x				x		x	x					
Vogel	x				x								
Mehrfard et al.													
Spence	x			x	x	x	x	x	x	x	x		
Rottier et al.		x								x			x
Weiguo et al.		x				x							
Heeagers et al.	x	x	x	x	x	x							

Table 1. Case Studies and Agile Practices utilised

Each of the agile practices used in these case studies were then incorporated into the relevant stage of the V-Model. Figure 1 shows the tailored V-Model. This combines the V-Model proposed by Ge et al. and the appropriate agile practices as identified through the literature review.
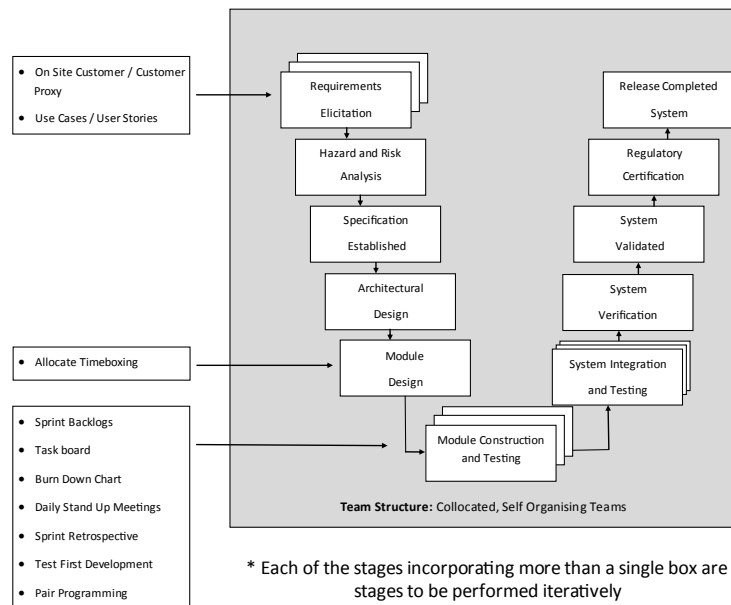
Figure 1: V-Model Incorporating Agile Practices

As part of the literature review, agile practices were identified through the key case studies performed. However, none of the key case studies provide information as to which stage of development the agile practices were incorporated. Despite this, we have performed a mapping which matched the agile practices to the appropriate stage of development, as a number of the practices are more suited to specific development stages e.g. Pair Programming = Module Construction and Testing.

5 Conclusions and Future Work

In conclusion, research into instances of agile adoption when developing medical device software has shown that these projects can benefit from integrating agile practices. Medical device software development organisations are bound by regulatory controls. These regulatory controls prevent medical device organisations from wholly embracing a single agile methodology when developing medical device software. The findings of the literature review revealed that there is no SDLC currently available which medical device software developers can follow which produces the necessary deliverables required to achieve regulatory conformance whilst reaping the benefits of utilising agile practices. Research has been conducted in isolation within organisations such as Medtronic and Cochlear, but no evidence exists that suggest the models used in these organisations can be followed with the same level of success in other medical device organisations. A common trend also appeared within each of these case studies. This trend was that no single agile methodology was appropriate for use when developing medical device software and that medical device software development projects benefit from a mixed method approach combining agile practices with a plan driven software development lifecycle. To this end it becomes apparent that medical device software development organisations would greatly benefit from the development of a SDLC that meets both the organisational needs provided from following a plan driven lifecycle and developmental needs from following agile practices.

Research conducted as part of this paper will be used as the foundation for the development of a tailored V-Model which integrates agile practices which are beneficial to the development of medical device software and facilitate the achievement of regulatory compliance. To establish which agile practices should be included into the tailored V-Model, every agile practices within each of the agile methodologies i.e. Scrum, XP, Test Driven Development, Crystal etc. will be examined to determine suitability for use when developing medical device software. Once suitability is determined the appropriate practices will be integrated into the tailored V-Model.

This tailored V-Model will be developed in collaboration with medical device software organisations in Ireland. Once the tailored V-Model is completed it will be thoroughly tested by industry. Once this model has been finalised the objective is to have a medical device software development project fully developed in accordance with the finalised tailored V-Model.

Acknowledgements

Author CVs

Martin Mc Hugh

Martin received his B.Sc. (Hons.) in Information Technology Management in 2005 and M.Sc. in Computer Science in 2009, from Dundalk Institute of Technology. He is now undertaking research for his Ph.D. in the area of software process improvement for medical devices with emphasis on the usage of agile practices when developing medical device software, as part of the Regulated Software Research Group in Dundalk Institute of Technology.

Fergal Mc Caffery

Dr Fergal Mc Caffery is the leader of the Regulated Software Research Group in Dundalk Institute of Technology and a member of Lero. He has been awarded Science Foundation Ireland funding through the Stokes Lectureship, Principal Investigator and CSET funding Programmes to research the area of software process improvement for the medical device domain. Additionally, he has received EU FP7 and Enterprise Ireland Commercialisation research funding to improve the effectiveness of embedded software development environments for the medical device industry.

Valentine Casey

Dr Val Casey is a Senior Researcher with the Regulated Software Research Group in Dundalk Institute of Technology. His previous roles include Senior Lecturer and Research Area Leader at Bournemouth University, Researcher with Lero - the Irish Software Engineering Research Centre at the University of Limerick where he also lectured. He has over 20 years' experience in the software industry. He has also provided consultancy services focusing on software process improvement, software testing and global software development.

Minna Pikkarainen

Minna Pikkarainen is a Principal Research Scientist in VTT Technical Research Centre of Finland. She has worked in several industrial-driven research projects and project preparations doing close industrial collaboration with large amount of organizations in Europe since 1997. Pikkarainen has been a member of Lero (The Irish Software Engineering Research Centre) during the years 2006-2009. Currently, she is a member of Sirris, The Collective Centre for the Belgian technological industry. Recently, her work and numerous publications have mainly focused on research in the areas of agile development and software innovation.

6 References

1. Cusumano, M. and D. Yoffie, *Software Development on Internet Time.* IEEE Computer, 1999. **32**(10): p. 60-69.
2. Williams, L., *Agile Software Development Methodologies and Practices.* Advances in Computers, 2010. **80**: p. 1-44.
3. Larman, C. and V. Basili, *Iterative an Incremental Development: A Brief History.* Computer, 2003. **36**(6): p. 47-56.
4. http://agilemindstorm.com/agile-glossary/ Accessed 20/02/2012
5. Turk, D. and R. France, *Assumptions Underlying Agile Software Development Processes* Journal of Database Management, 2004. **16**(4): p. 62-87.
6. Conboy, K. and B. Fitzgerald, *Method and developer characteristics for effective agile method tailoring: A study of XP expert opinion.* ACM Trans. Softw. Eng. Methodol, 2010. **20**(1): p. 1-30.
7. Drobka, J., D. Noftz, and R. Raghu, *Piloting XP on Four Mission-Critical Projects.* IEEE Software, 2004. **21**(6): p. 70-75.
8. Greening, J., *Launching XP at a Process-Intensive Company.* IEEE Software, 2001. **18**(6): p. 9.
9. Cawley, O., X. Wang, and I. Richardson, *Lean/Agile Software Development Methodologies in Regulated Environments – State of the Art*, in *Proceedings of Lean Enterprise Software and Systems, .* . 2010, Springer Berlin Heidelberg: Helsinki, Finland. p. 31-36.
10. Ge, X., R.F. Paige, and J.A. McDermid, *An Iterative Approach for Development of Safety-Critical Software and Safety Arguments*, in *Agile 2010*. 2010.
11. Weiguo, L. and F. Xiaomin, *Software Development Practice for FDA-Compliant Medical Devices*, in *International Joint Conference on Computational Sciences and Optimization, 2009. .* 2009: Sanya, Hainan. p. 388 - 390.

12. Vogel, D., *Agile Methods:Most are not ready for prime time in medical device software design and development.* DesignFax Online, 2006(July 2006).

13. Munassar, N.M.A. and A. Govardhan, *A Comparison Between Five Models Of Software Engineering.* IJCSI International Journal of Computer Science Issues,, 2010. **7**(5): p. 94-101.

14. European Council, *Directive 2007/47/EC of the European Parliament and of the Council of 5 September 2007*, European Council, Editor. 2007.

15. AAMI, *ANSI/AAMI/IEC 62304, Medical device Software - Software life cycle processes*. 2006.

16. European Council, *Commission communication in the framework of the implementation of the Council Directive 93/42/EEC of 14 June 1993 concerning medical devices (Publication of titles and references of harmonised standards under the directive)*. 2010.

17. FDA, *General Principles of Software Validation: Final Guidance for Industry and FDA Staff*. 2002, *Centre for Devices and Radiological Health*.

18. FDA, *Federal Register **73** (31), 8637 (15 February 2011)*. 2011.

19. FDA, *Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices*. 2005, *U.S. Department of Health and Human Services*.

20. Thompson, B., *FDA Regulations of Mobile Health*, in *mobi Health News*. 2010: Chester Street Publishing. p. 51.

21. FDA, *Guidance for Industry, FDA Reviewers and Compliance on Off-The-Shelf Software use in Medical Devices*. 1999, *U.S. Department of Health and Human Services*.

22. FDA, *Title 21--Food and Drugs Chapter I --Food and Drug Administration Department of Health and Human Services subchapter h--Medical Devices part 820 Quality System Regulation*. 2007, *U.S. Department of Health and Human Services*.

23. FDA, *General Controls for Medical Devices*. 2009, Food and Drug Administration,.

24. McCaffery, F., D. McFall, P. Donnelly, and F.G. Wilkie, *Risk Management Process Improvement for the medical device industry*, in *Conference on Software Development (SWDC-REK-2005)* 2005: Iceland. p. 92-103.

25. Fowler, M. and J. Highsmith, *The Agile Manifesto*, in *Software Development: The Lifecycle starts here*. 2001.

26. Turner, R. and A. Jain, *Agile Meets CMMI: Culture Clash or Common Cause*, in *1st Agile Universe Conference*. 2002: Chigago.

27. Boehm, B. and R. Turner, *Using Risk to Balance Agile and Plan-Driven Methods.* IEEE Computer Society, 2003.

28. McCaffery, F. and V. Casey, *Med-Trace: Traceability Assessment Method for Medical Device Software Development.*, in *European Systems & Software Process Improvement and Innovation Conference, (EuroSPI).* 2011: Roskilde, Denmark.

29. Bulska, K. and J. Gorski, *Applying Agile Practices to the development of Safety-Critical Software*, in *ICT Young 2011, Scientific Booklets of Faculty of Electronics, Telecommunications and Informatics*. 2011: Gdańsk University of Technology. p. 65-68.

30. AAMI, *AAMI TIR(SW1)/Ed.1 -- Guidance on the use of agile practices in the development of medical device software*. 2010.

31. Martin, R.C., *Agile Software Development - Principles, Patterns and Practices*. 2003: Prentice Hall.

32. Heeager, L.T. and P.A. Nielsen, *Agile Software Development and its Compatibility with a Document-Driven Approach? A Case Study*, in *20th Australasian Conference on Information Systems Compatibility of Agile and Document-Driven Approaches*. 2009: Melbourne.

33. Rasmussen, R., T. Hughes, J.R. Jenks, and J. Skach, *Adopting Agile in an FDA Regulated Environment*, in *Agile Conference, 2009. AGILE '09. .* 2009: Chicago, IL p. 151-155.

34. Spence, J.W., *There has to be a better way! [software development]*, in *Agile Conference, 2005. Proceedings* 2005. p. 272 - 278.

35. Abrahamsson, P., O. Salo, J. Ronkainen, and J. Warsta, *Agile Software Development Methods - Review and Analysis*, in *ESPOO 2002*. 2002, VTT Publications.

36. Robres, A., *SCRUM in the FDA context.* Agile Record, 2011. **April**(6).

37. VersionOne, *State of Agile Survey - The Stage of Agile Development*. 2010.

38. Turk, D., R. France, and B. Rumpe, *Limitations of Agile Software Processes*, in *Third International Conference on Extreme Programming and flexible processes in Software Engineering (XP2002)*. 2002.

39. Randolph, J., *A Guide to Writing the Dissertation Literature Review.* Practical Assessment Research Evaluation, 2009. **14**(13).

40. Mehrfard, H. and A. Hamou-Lhadj, *The Impact of Regulatory Compliance on Agile Software Processes with a Focus on the FDA Guidelines for Medical Device Software.* International Journal of Information System Modeling and Design, 2011. **2**(2): p. 67-81.

41. Rottier, P.A. and V. Rodrigues, *Agile Development in a Medical Device Company*, in *Agile, 2008. AGILE '08. Conference*. 2008.

Extracting Contextual Complexities of Adopting Agile Practice(s)

Hesam Chiniforooshan Esfahani[1], Eric Yu[2], John Carbrey[3]

[1]Department of Computer Science, [2]Faculty of Information
[1,2]University of Toronto, [3]Intrafinity Inc.
hesam@cs.toronto.edu, yu@ischool.utoronto.ca, john.carbrey@intrfinity.com

Abstract

Agile methods are often proposed as a set of practices, from which development teams pick up their selected subset. One of the key challenges in the process of deciding whether to select an agile practice is to know in advance the potential impacts of that practice within the organization. In earlier paper we introduced an evidence-based repository of agile practices. In this paper we explain an experience of using the repository to guide and support the process of decision making over the enactment of a particular agile practice in a medium-scale software company.

Keywords

Agile Methods, Evidence-Based Software Engineering, Software Process Engineering

1 Introduction

Many organizations have been moving to agile methods, by incorporating various agile practices in their development process. Enactment of any agile practice initiates an intense decision-making process, whether to adopt the practice, and if yes, what adaptations are required. This process can be quite complex, as agile practices can show various outcomes depending on context. One of the common risks in such organizations is to start adopting and adapting to Agile practices, without analyzing the organization specific complexities of enacting every practice. This analysis is important as it cab reveal potential points of mismatch, which might void the promises of new process.

This paper introduces a simple method for finding the organization-specific "questions", which must be answered, while deciding upon any agile practice. The very existence of these questions clarifies many hidden issues and potential pitfalls f agile adoption. Furthermore, the answers to these questions help organizations to decide upon the required adaptations, if they decided to adopt an agile practice. The method can be used by project managers and process managers, without the need to engage external consultants.

The method makes use of an *evidence-based repository of agile practices*, reported in [1]. The repository provides information about the objectives, which can be contributed either positively or negatively by each agile practice. As an example, for the Pair Programming (PP) – as an agile practice – the repository supplies the list over 25 objectives (e.g. improved communication, time to market, reduced defect in code, etc.) which can be improved or hurt by PP in different project situations. The content of this repository is evidence-based as it is extracted out of extensive literature review of empirical studies on agile methods. This repository is available online at www.ProcessExperience.Org.

The paper describes a real-word industrial experience of deciding upon an agile practice, where the content of the repository helped the decision-making team to clarify the complexities of the decision. The experience was carried out in the Intrafinity Inc., a small-scale software development company, located in downtown Toronto. The company used to classical (waterfall-based) style of software development, but was going to gradually move to agile. This paper explains the process of deciding upon the agile practice Pair Programming, in a particular context of the organization, which is described in later sections.

2 Method

Figure 1 shows the overall procedure of the proposed method for identifying situation specific questions, which should be answered when considering the adoption of any candidate agile practice. The first step of the method is to extract all of the objectives that the Candidate Practice (CP) contributes to. This step can be easily done by using the relevant page of the agile practice from the repository. Extracted objectives receive either negative or positive contribution(s) from the CP, and in some cases the contribution relations are situational (i.e. the contribution relation is valid only under certain situational conditions).

The second step of the method is where every objective is analyzed to extract its situation-specific questions. The sub-step 2.1 explores the reported contributions of the CP to those objectives. As shown in the repository, it is possible for an agile practice to make different kinds of contributions (strongly positive, positive, negative, and strongly negative) to one objective. These contributions are often reported in various project / organization situations, where the reference and a short description of the situation is provided by the repository.
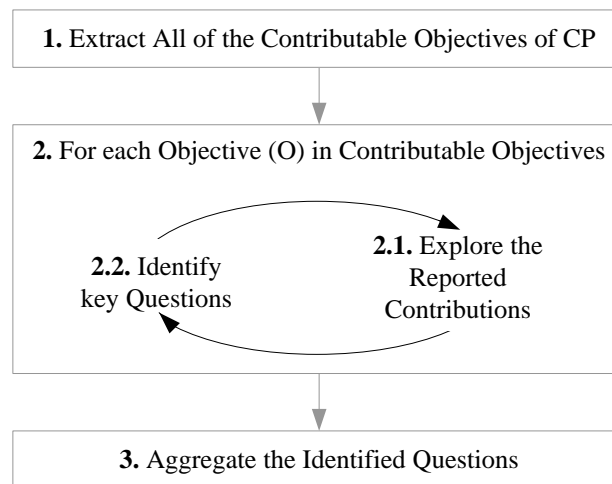
Figure 1: Proposed Method for Identifying Situation-Specific Questions of each Candidate Practice (CP)

The sub-step 2.2 is to identify key questions, which are aimed to clarify the situation-specific contribution of CP to the subject objective. This paper suggests the following template questions for this step of the method:

- Does this objective really matter for the organization?

- Is there any similarity between the current organization context and any of the situations, referenced by the repository?

- Under what situations the reported objective can be achieved or missed? (the requisite dataset of the repository can be used to find this class of questions)

- Which of the situational factors of the project / organization can affect the contribution of the CP to the objective? (situational, or in other term contextual, factors of agile adoption are usually the Size, Criticality, Culture, Skills, etc. – studied in [2].

The repository of agile practices has two major roles in the execution of this method: first, it supplies a list of objectives that are contributable by each agile practice; and second, it facilitates the process of extracting questions which must be answered regarding the achievability of each objective with respect to the specific characteristics of the organization.

3 Explanatory Industrial Experience

The proposed method is experienced in an industrial setting for solving a specific problem (decision making on adopting a particular agile practice). This section describes the context, the problem, and the Solution, which used the proposed method for solving the problem within that particular context.

3.1 Context

Intrafinity Inc. is a software development company, located in downtown Toronto, Canada. The company was established in 2001, and since then has rapidly grown in terms of income and number of staff. The company has around 45 employees, grouped into three departments: Development, Sale, and Support. The development department (which was the target of this research) has 15 members, grouped into three teams. Each team has a lead developer, who is the most experienced member of that team, and one project manager supervises the overall activities.

The dominant software development process of Intrafinity has been a classical, with most of the characteristics of the Waterfall model. However, the company started an initiative to gradually move to agile, by adopting a selected set of agile practices. One of the target agile practices that the organization was going to adopt it was Pair Programming. However, the company was going to use this practice in an special situation.

The company was motivated to hire a number of co-op students for its Development groups. The initiative was motivated by two factors:

- The company was quickly growing, and needed to expand its development staff.

- A considerable number of students has already applied for the co-op positions advertised by the company.

Based on a general understanding of Pair Programming (PP), the initial strategy for handling co-ops was to pair them up with experienced developers, and in this way help them to learn the development framework of the company more quickly. The idea was based on the general understanding of the PP that "Pairing novice programmers with experienced ones, makes the learning process of novices much faster".

3.2 Problem

The initial evaluation of the plan was quite promising. A good portion of the co-ops' salary was coming from tax benefits, a good number of applicants – some with great transcripts – has already applied, and they were expected to become productive members by picking up the required skills through working in pair with professionals. However, since the PP was never practiced in the company, managers were still hesitating about the actual outcomes of the initiative, and their main concerns were:

- Do we actually know the necessary information which is needed to be known about pair programming?

- What would be the drawbacks of this decision?

3.3 Solution

Using the empirical knowledge provided by the repository of agile practices, and following the above mentioned method, the achievability of the objectives of the subject agile practice (Pair Programming) was analyzed, and a list of contextual questions was extracted. Answering this list of questions helped the managers to reveal the complexities of enacting PP, and to come up with a realistic understanding of its benefits and drawbacks. It should be mentioned that the extracted list of questions are not claimed to be sufficient for clarifying all hidden aspects of adopting a particular agile practice. However, our experience showed that the extraction and analysis of these questions can have a strongly positive impact over the process of decision making on agile practices.

Following the first step of the method (Extract all contributable objectives of the candidate practice), and based on the repository of agile practices, the following list of objectives was recognized to be contributable by the agile practice Pair Programming:

- Increased Productivity (--)

- Be on-time to market (+ / -)

- Faster Problem Solving (+)

- Reduced Defects in Code (++)

- Reduced Development Cost (-)

- Improved Design Quality (++)

- Improved Creativity Potential (+)

- Improved knowledge Sharing (++)

- Supportive Environment (++)

- Higher Commitment in Doing a Job (++)

- Improved Learning (++)

- Training New Developers (+/-)

- Improved Job Satisfaction (++/--)

- Reduced Staffing Risk (++)

The positive and negative signs which are within parenthesis in front of every objective, shows the kind of contribution(s), which is reported by the repository from PP towards that objective. The following subsections explain the result of the second step of the method, for a subset of PP objectives that had higher priority to the organization.

Training New Developers

It is a common belief that PP would help the process of training new developers to be faster and more effective. However, the repository information shows that PP can actually make an adverse impact over this objective. The supporting evidence is reported in [1] where the lack of a systematic way of handling new developers caused PP to fail with this objective.

In our experience, the evidence-based knowledge of PP guided the organization managers to come up with the important question of: "What should be considered in a systematic way of enacting PP?". In response to this question, the following list of questions emerged:

- Who should pair with whom?

- How much effort should be invested on the new co-ops?

- How to find out if the co-op student is worth investing upon?

- What do we want them to learn?

Increased Productivity

It is reported that the PP would decrease the productivity of developers, for the sake of improving the correctness of programs [1]. However, in our particular case, the PP was about to be used for a limited-time period, in order to train new developers. This would cause the productivity drop down of professional programmers for two reasons: first, distractions that a new developer who is in learning curve would cause for its trainer; and second, increased congestion, which is due to increasing the number of developers in a limited-space of the development site.

Therefore, the analysis team came up with the following questions, regarding the possibilities of minimizing the negative impacts of PP over the productivity of professional programmers:

- How many interns to hire?

- How many hours should pairs to work together?

- How to monitor the performance variation of professional developers?

- What would be an acceptable range of performance variation for professional developers? (i.e. minimum expected and maximum acceptable percentage of performance degradation)

Reduced Development Cost

Empirical studies have not reported a unified result about the impact of PP on development cost. [3, 4]

reported that PP in general would increase the development cost. However, other studies reported that PP in some cases can have no negative impact on the development cost. [5] mentioned that "when time to market is decisive and paired programmers are much faster than single developers, then the extra cost of PP can be justified.[6] stated that "In case of not very experienced programmers, PP costs as much as peer review". These evidences further suggest that the potential impact of the PP on development cost, in the special circumstance of Intrafinity or any other company, must be analyzed with respect to the specific situation in which the PP is going to be enacted.

The company was eligible to receive tax benefits for hiring co-ops, which would cover a portion of their salary. However, after considering the potential overhead of new junior developers for the professional ones, the analysis team came to this realization that the actual cost of training junior developers is much more than the incentive tax benefits. On the other hand, the reduced productivity of developers would have the following side-effects:

- Change of development plan – since the original plans would be no longer valid. This results in an overhead cost of project management.

- Delay in project completion – since the functionality of developers is reduced. This results in potential costs of being late to market.

Apart from the negative aspects of the PP on the development cost, the analysis team came up with an important positive aspect. If this initiative results in hiring a number of really talented co-ops, it will be a great investment for the company. New developers will be paired up with professional developers for a few months, and then they can start working on their own, which will compensate the training cost very quickly.

At the end of this analysis, the process improvement team came up with the following questions, which their answer would further clarify the impact of PP on development cost: (some of these questions have been originated by the analysis of other objectives, as well)

- What is the maximum acceptable degradation of developers performance?

- How long do we want new developers to work in pair?

- How to evaluate new developers?

- How much delay should be expected for the project completion?

- What would be potential cost of delay in project completion?

- What would be the managerial overhead of preplanning projects (in order to accommodate the new development style)?

Improved Knowledge Sharing

Improving knowledge sharing is one of the common goals of many agile practices, such as pair programming, daily meetings, common code ownership, story board, and on-site customer [1]. Empirical studies [3, 4, 7] suggest that PP can strongly contribute towards this objective. However, the achievement of this object depends to the existence and the maturity of agile values (trust, respect, communication, feedback,…) within the organization context. Since, pair programming was never practiced by the development team of the subject organization, the process improvement team came to these questions regarding the impact of PP on knowledge sharing:

- To what extent our developers would share their knowledge with new trainees?

- How many of them are willing to do so?

- How many of them are reluctant in knowledge sharing for any reason (such as being inpatient, sense of training someone who may become their competitor in future)?

The answer to these questions had a direct impact on the decision of "How many co-ops to hire?". Although pair programming was not practiced previously, but regarding the personal knowledge of managers from developers, and the previous records of every developer in communication and collaboration, it was not hard for the managers to anticipate the willingness of each programmer in handling a new developer. As far as the managers are aware, a particular group of experienced developers will feel that the new members will be a burden for them, and will be a drag on their

productivity. Though there are some developers who are quite receptive in dealing with novice programmers.

Other Objectives of Pair Programming

Among the other objectives of PP, "Reduced Staffing Risk" and "Improved Learning" was evaluated to be most probably reachable. It was well-expected that if the company goes with the plan of hiring co-ops, a number of these students would eventually join the company as full-time developers. The tacit knowledge that they (hopefully) pick up during the pairwise work, would help the company to reduce the negative impacts of staffing loss (i.e. upon the loss of a professional developer, the company can expect the developer was paired with him to have at least a portion of his expertise and knowledge of the code). As of "Improved Learning" it was highly expected that the PP improves the learning speed of new developers (though complexities of Training New Developer).

Regarding the special context of the company, the following PP objectives were evaluated to be unaffected (neither improved nor deproved):

- Reduced Defects in Code

- Improved Design Quality

- Improved Creativity Potential

- Higher Commitment in Doing a Job

4 Summary and Discussion

This paper proposed a method for facilitating the decision-making process of adopting agile practices. The method is intended to self-enable software organizations to decide on the trade-offs of transitioning to a new development process. Using an evidence-based repository of agile practices, the method guides process analysts to extract a set of contextual questions, which their answers would clarifies the complexities of adopting a new agile practice. The reported experience of using this method in a real industrial setting showed that it is functional, and can help process analysts throughout the course of transitioning to agile.

The proposed method has some limitations and strengths. Among the limitations of the method is its sole attention to the "objectives of an agile practice" for extracting its contextual questions. Such a question set can be further complete by considering preconditions of applying an agile practice, which are often provided by the original prescription of the practice. The evidence-based repository of agile practices supplies such information for agile practices, but the requisites datasets are not often as thorough as the objectives datasets. The other limitation of the work, which can also be considered as the strength point of method, is its reliance on the evidence-based repository of agile practice. It is a limitation because the repository might not supply adequate data to the method (for variety of reasons, such as: incomplete literature review, or fundamental lack of empirical evidences for some agile practice. It is a strength point, because the evidence-based nature of the method improves the reliability of its outcomes.

The use of the proposed method in Intrafinity suggests that the plan of deploying Pair Programming was not going to be as promising as it had been envisioned. Achieving the basic intention of organization from deploying PP (training new developers), was subject of a clearly-defined PP-enactment strategy, which was not in-place, and the organization had no prior experience in that regard. Moreover, the list of objectives that the analysis showed that they will remain unaffected, was a clear indication that PP was going to be used out of its intended application context. Typically, the main reason of enabling PP is to "Reduce Defects in Code" and "Improve Design Quality". However, these two objectives are clearly evaluated to remain unaffected by PP, since new developers could barely contribute to the quality of code or design, while they are paired with professional programmers.

The analyzed impacts PP on the productivity of professional programmers, the cost of development, and the projects' time-to-market added more hesitation on the decision of hiring co-ops, and using PP for their training. The other point, which further disvalued this decision, was the perception of

managers that co-op students, often look at the company as a temporary place of work. Come to learn something, improve their resume, and then apply for better paid positions. So, if the company wanted to proceed with the initial plan, managers should have thought of a strategy for keeping its talented co-ops. Otherwise, it would be a failure for the organization.

Acknowledgements

5   References

[1] H. Chiniforooshan Esfahani and E. Yu, *A Repository of Agile Method Fragments*, in New Modeling Concepts for Today's Software Processes (Proceedings of the ICSP'10), J. Münch, Y. Yang, and W. Schäfer, Editors, Springer Berlin / Heidelberg: Germany. p. 163 -174 (2010)

[2] P. Kruchten, *Contextualizing agile software development.*In JOURNAL OF SOFTWARE MAINTENANCE AND EVOLUTION: RESEARCH AND PRACTICE(2011)

[3] A. Cockburn and L. Williams, *The costs and benefits of pair programming*, in Extreme programming examined, Addison-Wesley Longman Publishing Co., Inc. p. 223-243 (2001)

[4] L. Williams. *Integrating pair programming into a software development process*. In Proceedings of 14th Conference on Software Engineering Education and Training Charlotte, NC, USA: IEEE Computer Society (2001)

[5] F. Padberg and M.M. Muller. *Analyzing the cost and benefit of pair programming*. In Proceedings of Ninth International Software Metrics Symposium.: IEEE Press (2003)

[6] M.M. Müller, *Two controlled experiments concerning the comparison of pair programming to peer review.*In Journal of Systems and Software, 78(2) p. 166-179 (2005)

[7] M. Pikkarainen, J. Haikara, O. Salo, P. Abrahamsson, and J. Still, *The impact of agile practices on communication in software development.*In Empirical Software Engineering, 13(3) p. 303-337 (2008)

Agile maturity model - Go back to the start of the cycle

Miklos Biro [1], Morten Korsaa [2] Risto Nevalainen [3] Detlef Vohwinkel [4] and Tomas Schweigert [4]

[1] Software Competence Center Hagenberg GmbH, Hagenberg, Austria, miklos.biro@scch.at
[2] Delta, Copenhagen, Denmark, MKO@delta.dk
[3] FiSMA (Finnish Software Measurement Association), Helsinki, Finland
risto.nevalainen@falconleader.fi
[4] SQS Cologne, Germany
tomas.schweigert@sqs.com, detlef.vohwinkel@sqs.com

Abstract

Long times ago, SPI was based on a bunch of incomparable models. The SPI Community learned in a long run that results of appraisals and assessments need to be comparable. That became true even for the testing community. Discussions between SPI experts and agile evangelists often end up speechless. This Paper tries to help building a bridge between the agile and the SPI world. It also takes into account that there is a critical view on modern software engineering especially argued by some members of the SEMAT group. The bridge building components in this paper are a survey and a collection of recently published Agile Maturity Models. The paper will not present an own Agile Maturity Model. This will be a task for further research but it intends to compile current agile maturity model thinking.

The paper is based on actual industrial surveys whose results are analysed and interpreted in general terms linking them to philosophical issues partly raised in recent initiatives like SPI Manifesto, the ECQA PI Manager Certification and SEMAT.

Keywords

Agile, Maturity, Survey, ECQA, SEMAT, SPI,

1 Introduction

Starting in December 2011 Three surveys were launched to get an idea about what could an agile maturity model deliver and what might be its added value.

The questions of the survey were asked as follows:

1.   Do you think an "Agile Maturity Model" makes sense at all?

2.   What do you think should be the main focus of an agile maturity model?

3.   What do you think could be possible sources for an agile maturity model?

4.   What do you think about the Agile Manifesto?

5.   What is the best approach to manage multiple agile teams?

6.   If an organisation undergoes an audit/appraisal/assessment what should be the main focus?

7.   Please give a first global opinion about Agile Maturity Models

8.   Considering, that each Agile Maturity Model might have a roadmap: Which roadmap to agile maturity would you prefer:

9.   Thinking at a reference model for Agile Maturity: How should it be defined?

10.   What would the best thing to use an Agile Maturity Approach for?

11.   What would be the best frequency of Agile Maturity Assessments?

12.   What would be the best approach for Agile Maturity Improvement?

67 Participants from several agile or/and CMMI® related LinkedIn Groups contributed to the survey.

The following sections discuss the issues underlying the above questions, the results of the survey, as well as their interpretation in general terms linking them to philosophical issues partly raised in recent initiatives like SPI Manifesto, the ECQA PI Manager Certification and SEMAT.

2 The current discussion on Agile Maturity models

Meanwhile it is common opinion, that it is possible that an agile organisation can reach high CMMI® maturity levels or high SPICE Capability levels [Bianco 2011]. But it seems that some agile gurus are not happy with the support offered by CMMI or SPICE. So we have 30 different models that call themselves Agile Maturity Model.
There are several types of agile maturity models published, mainly in the Internet. There are also some principle thoughts about agile maturity published. The discussion about agile maturity is somehow influenced from ideas of the CMMI® Model. See also [Schweigert 2011]
So it seems to be adequate to group the published agile maturity models in those who are close to the level structure of CMMI®, those who have a level structure and those who don't use explicit levels.

2.1 Models with maturity levels

Models which are close to CMMI® (CMMI influenced models):

CMMI® (Reference Model)		Initial	Managed	Defined	Quantitatively managed	Optimising
Patel [Patel 2009]		Initial	Explored	Defined	Improved	Sustained
Yin [YIN 2011], [YIN et al 2011]	Providing also a deep dive mapping to CMMI®	Initial	Managed	Defined	Quantitatively managed	Optimising
Sims[Sims 2009] **(Curt Hibbs** [Hibbs 2005]**)**	Management levels	Initial		Organised	Disciplined	
Anderson [Anderson 2004]		Analysis Ability	End to End Traceability	Stabilize System Metrics	System thinking and a learning organization	Anticipated ROI and the Failure tolerant Organization
Humble and Russell [Humble 2009]		Regressive	Repeatable	Consistent	Quantitatively managed	Optimizing
	The levels are composed on practices in the following sections:	Build Management and continuous integration	Environments and deployments	Release Management and Compliance	Testing	Data management

Proposed agility value driven Maturity Models:

Ambler [Ambler 2010]		Rhetorical stage	Certified stage	Plausible stage	Respectable stage	Measured stage
Proulx [Proulx 2010]		Team Level Maturity	Department Level Maturity	Business Level Maturity	Project Management Level Maturity	Management Level Maturity

Jarajay [Jayaraj 2008]	Regressive (An explicit blame level)	Neutral or Chaotic	Collaborative	Operating (Consistent exhibition of competence)	Adaptive (Expertise to adapt to change)	Innovating (Creative evolution of practice, and spread these practices throughout the organization)
Benefield [Benefield 2010]		Emergent Engineering Best Practices	Continuous Practices at Component Level	Cross Component Continuous Integration	Cross Journey Continuous Integration	On Demand Just in Time Releases
Druckman [Druckman 2011]			Learn	Leverage		Optimise
Minick and Fredrick [Minick 2009]		Introductory	Novice	Intermediate	Advanced	Insane
Sidky and Arthur [Sidky 2007] **(Banerjee** [Banerjee 2011]**)**	This model was also recommended by Banerjee.	Collaborative	Evolutionary	Effective	Adaptive	Ambient
Alhir [Alhir 2009]		Dormant	Speed: Focusing on being expeditious.	Reactive: Focusing on acting relative to change from the perspective of the moment rather than a longer timeframe.	Responsive: Focusing on acting relative to change from the perspective of the moment balanced with a longer timeframe.	
Woods [Woods 2011]		Stage 1 – No Agile BI	Stage 2 – Early Adoption	Stage 3 – Self Service	Stage 4 – The Lake Effect	
Ronen [Ronen 2011]	Ronen is searching for a link to TMM.	Waterfall	Forming	Agile	Performing	Scaling
Sims[Sims 2009] **(Curt Hibbs**[Hibbs 2005]**)**	Technical maturity levels, Link to Hibbs in the article of Sims is already broken.	Non-Agile	Minimum	Consolidated		

These Agile Maturity Models use maturity levels. But even if they use the level names of CMMI® their content is different. To analyse the detailed structure of these models will be a future research task.

2.2   Individually structured models

In this chapter the paper presents models that have individual structures. Some of them might also be considered as collections of requirements for agile software development.

Waters [Waters 2007] Model:

- Active user involvement is imperative

- The team must be empowered to make decisions

- Requirements evolve but the timescale is fixed

- Capture requirements at a high level; lightweight & visual

- Develop small, incremental releases and iterate

- Focus on frequent delivery of products

- Complete each feature before moving on to the next

- Apply the 80/20 rule

- Testing is integrated throughout the project lifecycle – test early and often

- A collaborative & cooperative approach between all stakeholders is essential

Kellys [Kelly 2008] Key Features:

- Knowledge

- Learning

- Technology

- Change

- The relevance of information, people and culture is also mentioned by Rohrbeck.

Amblers [Ambler 2009] agility on scale Model:

- Core Agile Development

- Discipline Agile Delivery

- Agility at Scale

This model is also mentioned in a summary of Heydt [Heydt 2009].

Amblers scaling factors:

- Team size

- Geographical distribution

- Regulatory compliance

- Domain complexity

- Organizational distribution

- Technical complexity

- Organizational complexity

- Enterprise discipline

Ambler and Kroll[Kroll 2011] also point out that a view on governance is needed. They cluster the relevant practices into 6 groups:

- Mission & Principle,

- Organization

- Process

- Measures

- Roles and responsibilities

- Policies and standards.

Refer to Jens Schauders [Schauder 09] critical comment on Scott Amblers Agile Maturity Model, see also Martens [Martens 2009] comments and see also Amblers Discussion with Campbell and Franklin [Franklin 2009].

Walker Royce [Royce 2009] also developed a set of principles that support agility at scale:

Top 10 Management Principles of Agile Software Delivery

- Reduce uncertainties by addressing architecturally significant decisions first.

- Establish an adaptive lifecycle process that accelerates variance

- Reduction.

- Reduce the amount of custom development through asset reuse and

- Middleware.

- Instrument the process to measure cost of change, quality trends, and

- Progress trends.

- Communicate honest progressions and digressions with all stakeholders

- Collaborate regularly with stakeholders to renegotiate priorities,

- Scope, resources, and plans.

- Continuously integrate releases and test usage scenarios with

- Evolving breadth and depth.

- Establish a collaboration platform that enhances teamwork among

- Potentially distributed teams.

- Enhance the freedom to change plans, scope and code releases through automation.

- 10. Establish a governance model that guarantees creative freedoms to practitioners.

Baskarada [Baskarada 2006] names his model an agile model but does not really refer to agile maturity but is more interested in Asset Management

Rudds [Rudd 2011] discussion on the Purpose of an agile maturity model characterizes it a model that characterizes the progressive stages of something (as yet undefined) that gets (measurably) more Agile based on Maslow's need hierarchy. Derby [Derby 2010] thinks what does matter is that your company is satisfying its customers, stakeholders, and employees. So an Agile Maturity Model must be able to measure the degree of satisfaction of these peer groups. Little [Little 2010] strictly opposes all types of agile maturity models. On the other side Colin Doyle [Doyle 2011] explicitly defends the claim of agile purists.

King [King 2011] comes up with a simple pattern:

 If:

  AND(team turns up and really tries; management gives some support);

 Then:

  Return(Success);

  Call Continue;

 Else:

  Return(Error);

  Exit;

 Endif;

Kuruppath [Kuruppath 2009] sets out a set of key questions for tracking agility:

- Was all the agreed on functionality delivered?
- Was a high quality product delivered?
- Was the team responsive to new requirements or changes in requirements?
- Was there open communication during the project?
- Did the stakeholders have proper visibility into progress of the project?
- Was there smooth co-ordination between the Agile project and other projects and activities of the organization?
- Was there a high level of individual satisfaction?
- Was the team productivity high?
- Was there a high growth opportunity for team members?
- Do you think the success of the project is repeatable?

On a more technical level, Usman [Usman 2010] points out that serialization of parallel tasks will lead to serious problems during the build process.

Elssamadisy [Elssamadisy 2007] recommends using a set of implementation patterns but is critical about an agile maturity model, because he does not see agility as a necessary goal of software development.

Pettit [Pettit 2007] stated that an agile maturity model needs to address several aspects of organization and agility:

- Shared Responsibility

- Build

- Requirements

- Testing

- Responsiveness

- Assurance

- Simplicity

- Configuration Management

- Communication

- Governance

Also analyzed by Banarjee [Banerjee 2011].

Malik [Malik 2007] developed an excel based agile maturity assessment that used stories and their analysis to determine the level of agile maturity. Malik also took care of an improvement approach:

- Using this model, the team follows a simple process:

- Write a simple story that describes the process you followed. Examples are included in the spread sheet.

- Rate your process on 12 criteria based on the Agile Alliance principles

- Enter weights and view results

- Create a list of steps to address deficiencies. Follow the normal agile process to estimate these steps and add to the backlog.

This led to a discussion with Selhorst [Sehlhorst 2009] who pointed out that his way to agile maturity is as follows:

- Staffing the engineering team correctly.

- Assuring Quality is in your team's DNA.

- Reducing overhead in the release process.

- Feeding the beast.

- Managing stakeholder expectations.

- Continuously learning from your markets.

Aiello [Aiello 2012] introduces a 7 level scale:

- Adherence to Agile Principles (Purity)

- Repeatable process across the organization

- Scalability – SCRUM of SCRUMS

- Items on the right

- Coexistence with non-Agile

- Harmonization with industry standards and frameworks

- Support for IT governance and compliance

According to Rothman [Rothman 2011], Agile is also a matter of organizational culture. So agile maturity has to deal with cultural issues like fixed mindset vs. growth mindset, power distance and uncertainty avoidance.

All these models have in mind what about 20 Years ago was also one goal of the BOOTSTRAP approach: "Change and reorganize the software development and maintenance activities so that the software production as a whole better complies with business needs.[Kuvaja 94]. The methods changed the goals remains.

But this result also creates a challenge to the SPI community. It like in the beginning of SPICE where lots of models (e.g. ISO 9000, Trillium, TickIT, CMM, BOOTSTRAP, ISO 12207, ISO 15288) were in the market and there was no method to make their results comparable.

The need for a structured modelling is agreed in the whole IT business. (see e.g Test SPICE for the software testing business [Steiner 2010, Blaschke 2010]

So we are back at the start of the cycle.

3 The Results of the Agile Maturity Model Survey

The survey done by Tomas Schweigert (see also [Schweigert 2011] was aimed to analyze the thinking about some base principles of agile maturity. There are some other surveys that address a more technical perspective.

The questions of the survey were asked as follows:

1. Do you think an "Agile Maturity Model" makes sense at all?

2. What do you think should be the main focus of an agile maturity model?

3. What do you think could be possible sources for an agile maturity model?

4. What do you think about the Agile Manifesto?

5. What is the best approach to manage multiple agile teams?

6. If an organisation undergoes an audit/appraisal/assessment what should be the main focus?

7. Please give a first global opinion about Agile Maturity Models

8. Considering, that each Agile Maturity Model might have a roadmap: Which roadmap to agile maturity would you prefer:

9. Thinking at a reference model for Agile Maturity: How should it be defined?

10. What would the best thing to use an Agile Maturity Approach for?

11. What would be the best frequency of Agile Maturity Assessments?

12. What would be the best approach for Agile Maturity Improvement?".

Question 1: Do you think an "Agile Maturity Model" makes sense at all?

Do you think an "Agile Maturity Model" makes sense at all?
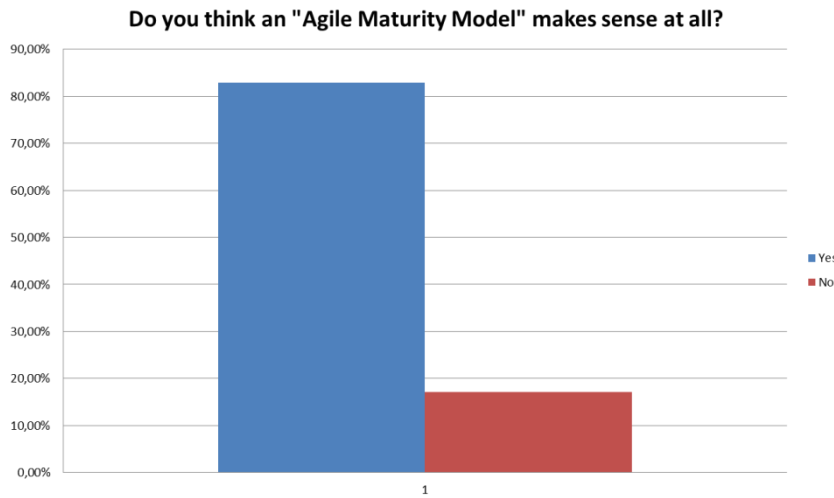
Figure 1: Nearly 80% of survey participants vote for an agile maturity model

The result shows a strong vote for an agile maturity model.

Question 2: What do you think should be the main focus of an agile maturity model?

What do you think should be the main focus of an agile maturity model?
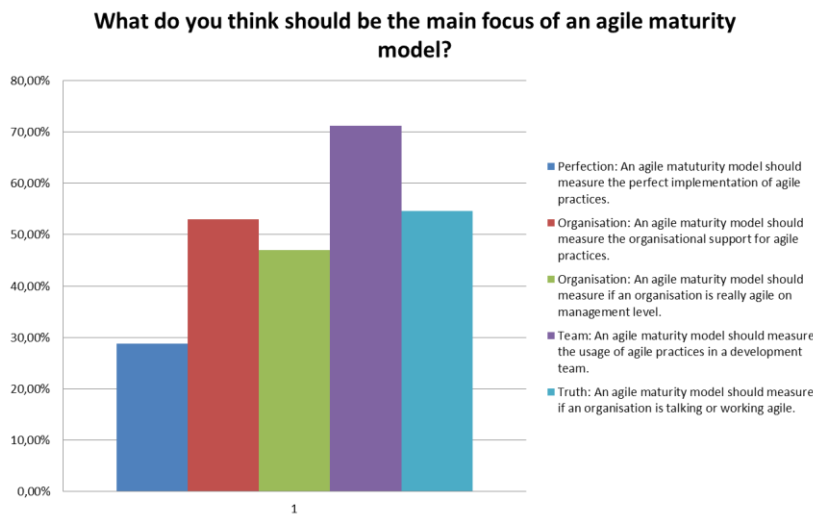
Figure 2: Nearly 70% of survey participants vote for team work as major focus of agile maturity

The result shows that there is only a little support for using the agile maturity approach for measuring agile perfection

Question 3: What do you think could be possible sources for an agile maturity model?
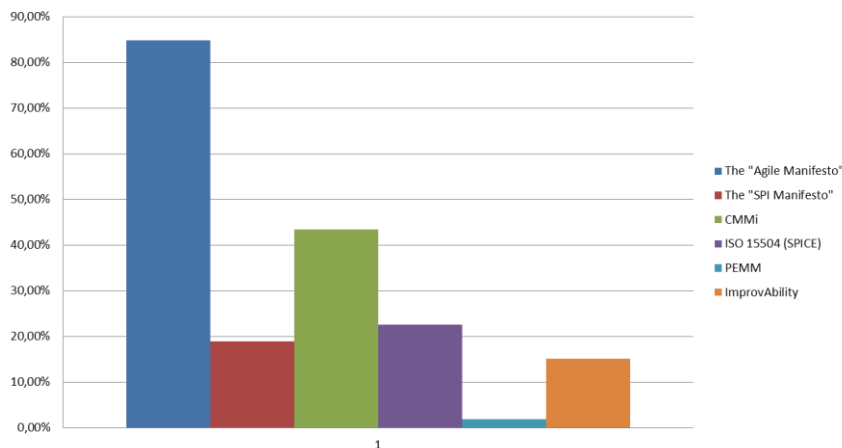
Figure 3: Nearly 80% of survey participants vote for the Agile Manifesto as a major source of an agile maturity model

The result is surprising as PEMM addresses all values of the Agile Manifesto. So the result might be a result of ignorance.

Question 4: What do you think about the Agile Manifesto?
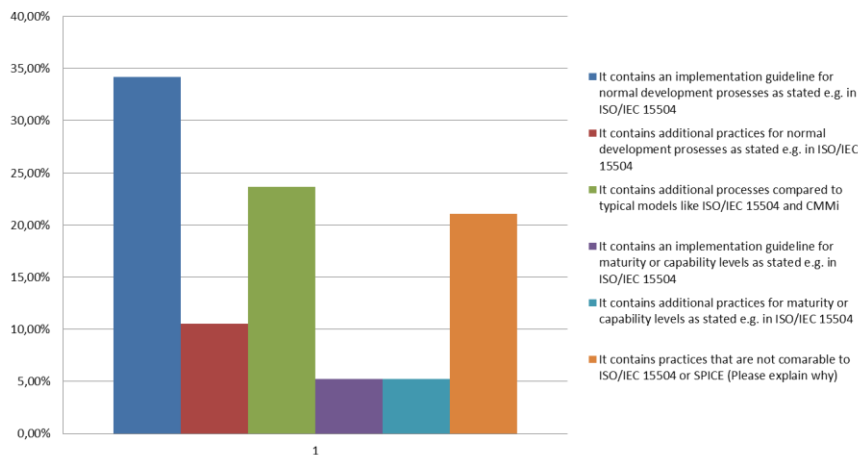
Figure 4: Only 5% of survey participants think that an agile maturity model will contain additional generic practices

All currently described Agile Maturity Models describe new capability or maturity levels. But most of the participants think that the Agile Manifesto contains an implementation guideline for common development processes.

Question 5: What is the best approach to manage multiple agile teams?
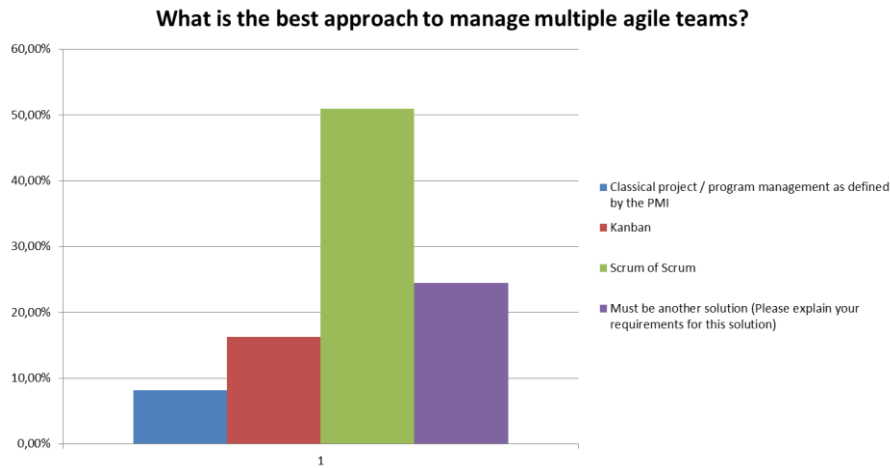
Figure 5: Only 10% of survey participants vote for the traditional project management when asked how to manage multiple agile teams.

There is a substantial need for a new idea how to manage multiple agile teams.

Question 6: If an organisation undergoes an audit/appraisal/assessment what should be the main focus?
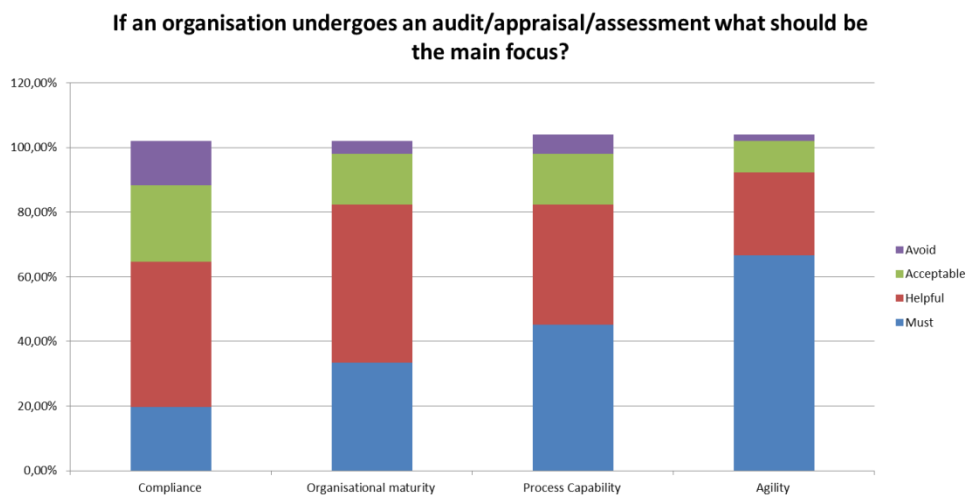
Figure 6: The survey participants rate agility and process capability as most important.

Compliance should not be the major focus of an agile maturity model.

Question 7: Please give a first global opinion about Agile Maturity Models
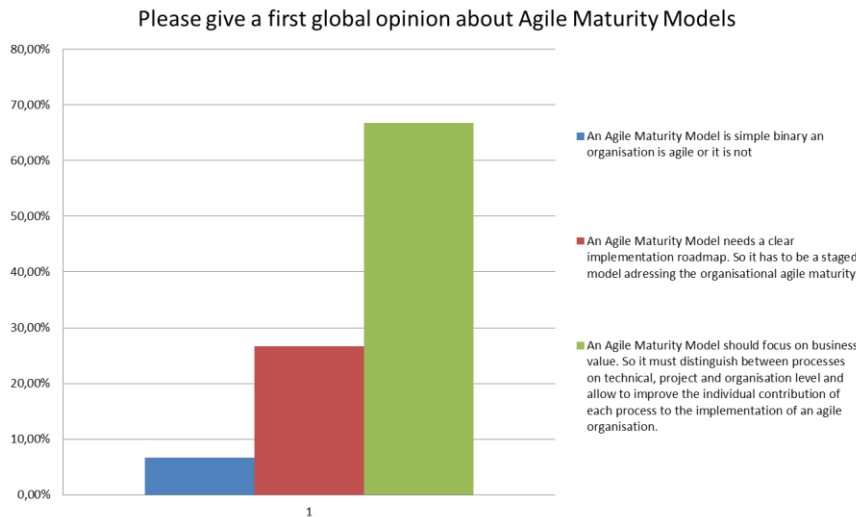
Figure 7: Nearly 70% of survey participants vote for the business value as a major goal of an agile maturity model

As expected most of the participants state that an Agile Maturity Model must focus on delivery of business value. But even is if this is one of the core principles of the SPI Manifesto the SPI Manifesto is not known in the agile community.

Question 8: Considering, that each Agile Maturity Model might have a roadmap: Which roadmap to agile maturity would you prefer:
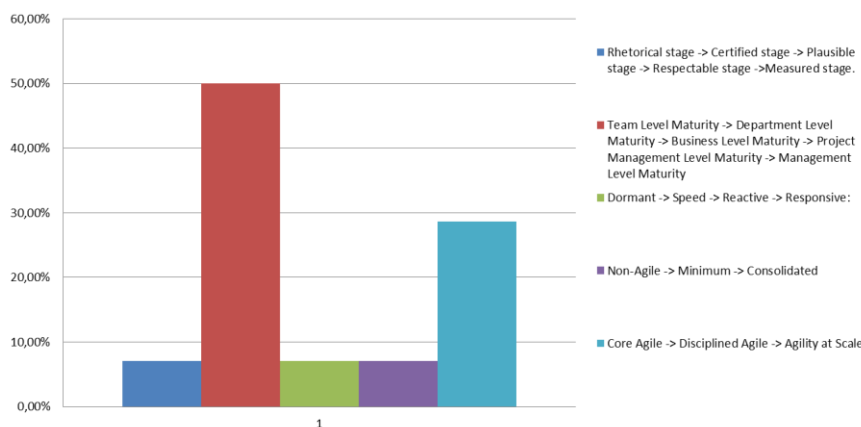
Figure 7: Nearly 50% of survey participants prefer traditional hierarchical thinking about agile maturity

The participants of the survey prefer the model of Proulx, and also the model of Ambler. Both focus on organizational maturity.

Question 8: Thinking at a reference model for Agile Maturity: How should it be defined?
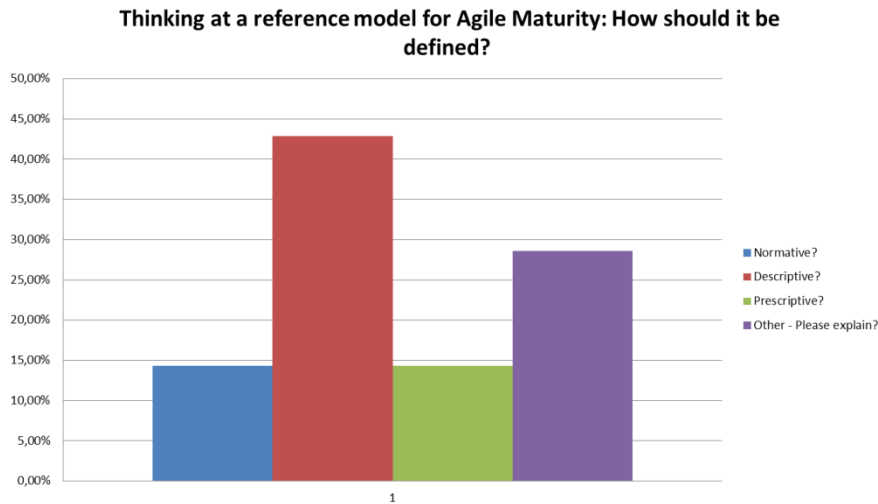
Figure 8: Survey participants think that an Agile Maturity Model should be neither normative nor prescriptive

Most of the participants prefer a descriptive model.

Question 9: What would the best thing to use an Agile Maturity Approach for?
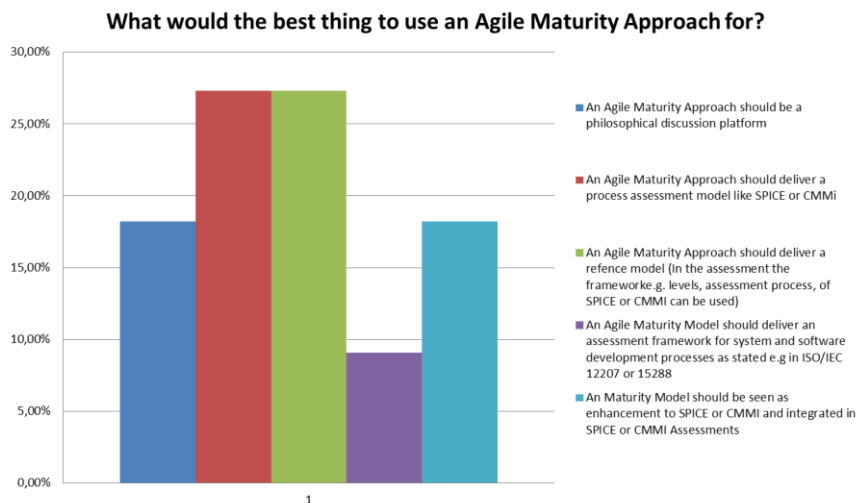
Figure 9: Survey participants agree that there is a need for a PRM and a PAM for agility.

Many participants do not want an Agile Maturity Model to become practical. They prefer a philosophical discussion platform.

Question 10: What would be the best frequency of Agile Maturity Assessments?
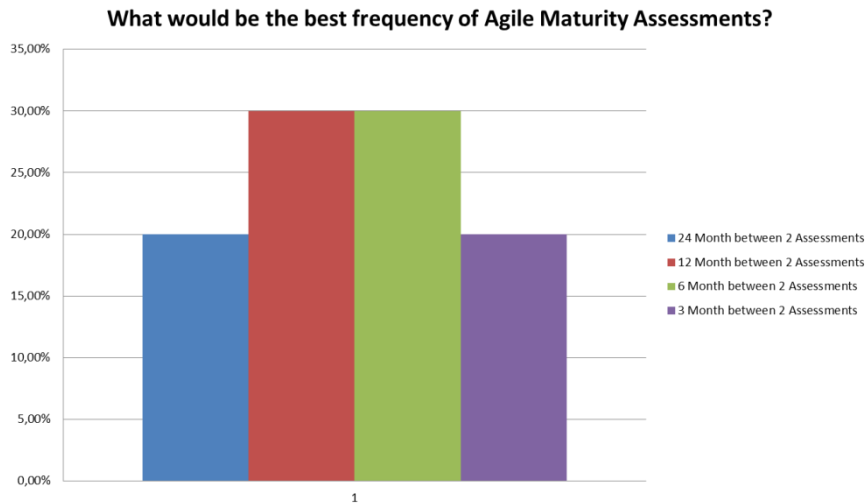
Figure 10: Agile Maturity Assessments will be a business opportunity

A common accepted Agile Maturity Model might be a substantial business case for assessment and improvement service providers. A frequency of 2 Assessments per year seems to create good business opportunities.

Question 11: What would be the best approach for Agile Maturity Improvement?
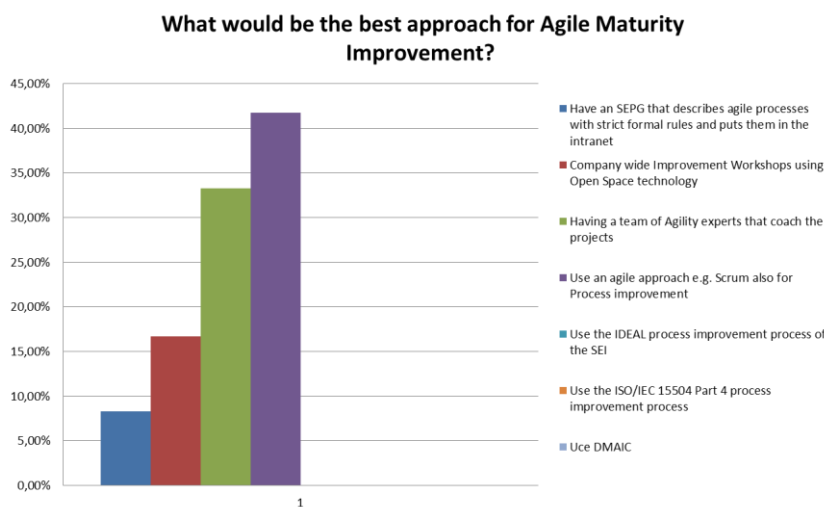
Figure 11: Scrum is rated also as a model for process improvement.

As we can see, there is complete ignorance of all experience of process improvement models in the agile community.

Putting it all together, it seems that a useful agile maturity model might be accepted, as most of the participants agree that an agile maturity model provides value. It is also widely agreed that there is no need for a complete new process assessment model, but the process reference model and the measurement framework of ISO/IEC 15504 (SPICE) need improvements in order to give valuable support to agile.

4 General Interpretation

Considering the results of the survey, it is certainly not surprising that several related initiatives were recently born.

One of these is the [SPI Manifesto 2010] consolidating the values and principles a group of experts within the EuroSPI (http://www.eurospi.net/) community proposed to be of key significance. The three key values of the SPI Manifesto are that SPI

- "Must involve people actively and affect their daily activities

- Is what you do to make business successful

- Is inherently linked with change"

The ECQA Software Systems Services Process Improvement (SPI) Manager Certification Scheme (http://www.ecqa.org/index.php?id=37), developed by the authors of this paper among others, carefully covers these issues.

SEMAT (Software Engineering Method and Theory) (http://www.semat.org/) is an initiative with a high number of supporters who agree that

"Software engineering is gravely hampered today by immature practices. Specific problems include:

- The prevalence of fads more typical of fashion industry than of an engineering discipline.

- The lack of a sound, widely accepted theoretical basis.

- The huge number of methods and method variants, with differences little understood and artificially magnified.

- The lack of credible experimental evaluation and validation.

- The split between industry practice and academic research. "

The survey results reported in this paper are intimately related to the above initiatives and can be considered to validate the general interpreting ideas advanced in the following paragraphs.

It is a general fact that a sound theoretical basis is usually built on mathematical approaches which are clearly used in all widely accepted sciences. Let us use analogies from mathematics to characterize the state-of-the art in process improvement methods.

In mathematics, there can be many interesting statements (usually called theorems, lemmas, etc...) which are deductively proven to be true (valid) starting from axioms whose truth is taken for granted within the particular domain of analysis. Still, there are mostly many ways to get from the axioms to the same theorem. These ways may be very different as far as their elegance (esthetics), effectiveness and usability are concerned, which have a definite impact on their capability to extend, generalize the results.

Abstraction, being essential for model building, is a key process in mathematics just as in software engineering or software process modeling. From the most commonly used mathematical abstraction called "natural number", to the highly different abstractions used in classical analysis and topology, all of mathematics shows how the different models applied in given theories can lead to a more or less elegant and extensible approach to the same or even revolutionarily new concepts.

It is a similar situation we are facing in SPI. We have many different approaches and statements that are claimed to be effective in leading to the achievement of the same business goals. Nevertheless, SPI is not rocket science, so the deductive proof of these claims is mostly impossible, by consequent they must be and are usually proven empirically which means inductive reasoning. The reference to best practices in either agile or heavyweight approaches claimed to be generally effective is clearly inductive reasoning. This is definitely part of the common theoretical foundation of process improvement whether agile or heavyweight.

The key analogy with mathematics is that there are different ways (CMMI, SPICE, Agile maturity models as discussed in the survey,...) to get to the conclusion that a method is effective in leading to the achievement of business objectives. These ways are just as different in their elegance (esthetics) and usability as there are very different approaches in mathematics to the same concepts. It is also true that elegance (esthetics) depends on subjective taste, while effectiveness and usability may be objectively measurable. By consequent, there will always be differences in the opinions regarding the elegance of approaches while at the same time all parties may measurably support their claims regarding the effectiveness or usability of their approach.

At this point, we have to recur to one of the key principles applied by SEMAT as well: Separation of Concerns. Subjective concerns like elegance (esthetics) for example should be separated from objectively measurable concerns like effectiveness and usability, especially if the objective measurements are not differentiating enough. And we are at the heart of the success factors addressed in [Jacobson et al. 2012]: Better, Faster and Happier (BFH).

5   Conclusion

Currently we cannot find a common accepted Agile Maturity Model. Similar to the beginning of SPICE and BOOTSTRAP [Kuvaja 1994], there are lots of incomparable models in the market.

There are estimated about 500 Outcomes in all the models mentioned at chapter 1 and about 50 outcomes that have common relevance. We also see a multidimensional ground model. It will cost some effort to make agile maturity to something that is explainable to the management.

To prepare a common accepted Agile Maturity Model, some intensive research has to be done. It might be taken into consideration to use the Approach of Clarke and O'Connor [Clarke 2011] to extract an acceptable 1st draft out of the available information.

An important question is then what value a specific agile maturity model will bring to the community. The alternative would be to improve existing SPICE and CMMI models to focus on the characteristics of good agile practices, and cover those inside the established process framework.

The question is also who should perform the Agile Maturity Model Assessments? Gurus, understanding only one model that is incomparable to the rest of the world or trained certified and experienced assessors, who used to deal with different reference and assessment models? There was lots of effort invested in proper training of CMMi Appraisers, SPICE Assessors and SPI manager [Korsaa 2010, Korsaa 2011].

6   Literature

[Aiello 2012] Bob Aiello, Agile Process Maturity - Introducing the Framework, Configuration Management Best Practices. Web publishing http://cmbestpractices.com/articles/agile-process-maturity-framework

[Alhir 2009] Sinan Si Alhir, Maturity Models: Leanness, Agility, Competitiveness, and Collaboration, Web Publishing 2009, http://salhir.wordpress.com/2009/06/26/maturity-models-leanness-agility-competitiveness-and-collaboration/

[Ambler 2009] Scott W. Ambler, The Agile Scaling Model (ASM): Adapting Agile Methods for Complex Environments, IBM Rational White Paper, Web Publishing ftp://ftp.software.ibm.com/common/ssi/sa/wh/n/raw14204usen/RAW14204USEN.PDF

[Ambler 2010] Scott Ambler, The Agile Maturity Model (AMM), Web publishing   http://drdobbs.com/architecture-and-design/224201005 see also http://drdobbs.com/tools/224201005

[Ambler 2012] Scott Ambler, Agile Surveys, http://www.ambysoft.com/surveys/

[Anderson 2004] David J. Anderson, Agile Management for Software Engineering, Applying the theory of constraints for business results, Prentice Hall, 2004, web publishing http://books.google.co.uk/books?id=hawMF31KCRsC&pg=PA105&lpg=PA105&dq=anderson+agile+maturity+

model&source=bl&ots=Zj7aDyaLUT&sig=EV6F3Y1PbfuJHY4ijA5blIOY-
dQ&hl=en&ei=TKT5SfqWEouJ_QahuIS2BA&sa=X&oi=book_result&ct=result&resnum=1#v=onepage&q&f=fal
se

[Banerjee 2011] Udayan Banerjee, Agile Maturity Model – Three Different Approaches, web publishing
http://cloudcomputing.ulitzer.com/node/2081511 also on http://ca.sys-con.com/node/2081511 also on
http://setandbma.wordpress.com/tag/agile-maturity-model/ also on
http://setandbma.wordpress.com/2011/11/30/agile-maturity-model/

[Baskarada 2006] Sasa Baskarada, Jing Gao, Andy Koronios, Agile Maturity Model Approach to Assessing and
Enhancing the Quality of Asset Information in Engineering Asset Management Information Systems, in: Witold
Abramowicz (ed.), Business Information Systems, Proceedings of BIS 2006, Poznań, Poland, Web Publishing
http://pdfbrief.com/agile-maturity-model-approach-to-assessing-and-enhancing-the.html

[Benefield 2010] Robert Benefield, Seven Dimensions of Agile Maturity in the Global Enterprise: A Case Study,
Proceedings of the 43rd Hawaii International Conference on System Sciences – 2010, Web Publishing
http://www.agileleantraining.com/download/HICSS43-Benefield.pdf

[Bianco 2011] Celestina Bianco, Agile and SPICE Capability Levels, in O Connor, Rout, Mc Gaffery, Dorling,
Software Process Improvement and Capability Determination, 11th International SPICE Conference
Proceedings, S. 181ff

[Biro 2009] Miklos Biro, The Software Process Improvement Hype Cycle. Invited contribution to the Monograph:
Experiences and Advances in Software Quality (Guest editors: D.Dalcher, L. Fernández-Sanz) CEPIS
UPGRADE Vol. X (5) pp. 14-20.

https://vm.mtmt.hu/download/1290381.pdf (accessed: 21/02/2012)
http://www.cepis.org/files/cepisupgrade/issue%20V-2009-fullissue.pdf (accessed: 21/02/2012)

[Blaschke 2010] Monique Blaschke, Michael Philipp, Tomas Schweigert, The Test SPICE Approach – Test
process assessments follow in the footsteps of software process assessments, Testing Experience, S.56

[Clarke 2011] Paul Clarke, Rory V O'Connor, An Approach for Evaluating Software Process Adoption, in O
Connor, Rout, Mc Gaffery, Dorling, Software Process Improvement and Capability Determination, 11th
International SPICE Conference Proceedings, S. 28ff

[Derby 2010] Esther Derby, Achieving Agility: Means to an End, or End in Itself, Web Publishing
http://www.estherderby.com/2010/06/achieving-agility-means-to-an-end-or-end-in-itself-2.html

[Doyle 2011] Colin Doyle, Agile Development Doesn't Have to Mean Fragile Enterprise Processes, MKS
Whitepaper, Web Publishing
http://www.agilejournal.com/pdf/Agile_Doesn%27t_Have_to_Mean_Fragile_white_paper%5B1%5D.pdf

[Druckman 2011] Angela Druckman, Agile Transformation Strategy White Paper
http://www.agilejournal.com/pdf/CollabNet_Agile_Transformation_Strategy.pdf

[Elssamadisy 2007] Amr Elssamadisy, Does the Agile Community Need a Maturity Model? Web Publishing
http://www.infoq.com/news/2007/10/agile_maturity_model

[Elssamadisy 2008] Amr Elssamadisy, Agile Adoption Patterns A Roadmap to Organizational Success, Addison
Wesley, 2008.

[Gujral 2008] Ravi Gujral, Shaun Jayaraj The Agile Maturity Model,
http://whattodowearelikethatonly.blogspot.com/2008/08/agile-maturity-model.html also at
http://agilephilly.ning.com/page/agile-maturity-model

[Franklin 2009] Carl Franklin, Richard Campbell, Scott Ambler on the Agile Process Maturity Model, Web
Publishing http://s3.amazonaws.com/dnr/dotnetrocks_0460_scott_ambler.pdf

[Heydt 2009] Michael Heydt, Agile Process Maturity Model, Web Publishing 2009, http://42spikes.com/post/Agile-
Process-Maturity-Model.aspx

[Hibbs 2005] Curt Hibbs, Towards an Agile Process Maturity Model, original source lost, last source web citation
(Download) http://www.guj.com.br/java/94510-modelo-de-maturidade-para-metodologias-ageis

[Humble 2009] Jez Humble and Rolf Russell, The Agile Maturity Model Applied to Building and Releasing
Software, ThoughtWorks White Paper, Web Publishing http://www.thoughtworks-
studios.com/sites/default/files/resource/the_agile_maturity_model.pdf

[Jacobson et al. 2012] Ivar Jacobson, Shihong Huang, Mira Kajko-¬-Mattsson, Paul McMahon, Ed Seymour, Semat – Three Year Vision, Web Publishing http://www.semat.org/pub/Main/SematDocuments/Semat_-_Three_Year_Vision13Jan12.pdf (accessed: 21/02/2012)

[Jayaraj 2008] Shaun Jayaraj, The Agile Maturity Model, Web Publishing http://whattodowearelikethatonly.blogspot.com/2008/08/agile-maturity-model.html

[Kelly 2008] Allan Kelly, Changing Software Development: Learning to Become Agile, Wiley 2008

[King 2011] James King, A mathematical formula to make agile work, Web Publishing http://kingsinsight.com/2011/07/14/a-mathematical-formula-to-make-agile-work/#more-529

[Korsaa 2010] Morten Korsaa, Miklos Biro, Richard Messnarz, Jörn Johansen, Detlef Vohwinkel, Risto Nevalainen, Tomas Schweigert , The SPI manifesto and the ECQA SPI manager certification scheme, Journal of Software Maintenance and Evolution: Research and Practice, published online: 25 AUG 2010 DOI: 10.1002/smr.502

[Korsaa2011] Morten Korsaa, Jörn Johansen, Tomas Schweigert, Detlef Vohwinkel, Richard Messnarz, Risto Nevalainen and Miklos Biro, The people aspects in modern process improvement, published online: 2 NOV 2011 | DOI: 10.1002/smr.570

[Kroll 2011] Scott Ambler, Per Kroll, Applying lean thinking to the governance of software development, IBM White Paper, Web publishing http://public.dhe.ibm.com/common/ssi/ecm/en/raw14000usen/RAW14000USEN.PDF

[Kuruppath 2009] Krish Kuruppath, Maturing Agile Processes to Deliver Better Value, Razorfish White Paper 2009.

[Kuvaja 1994] Pasi Kuvaja, Jouni Simila, Lech Krzanik, Adriana Bicego, Samuli Saukkonen, Günter Koch, Software Process Assessment & Improvement, The BOOTSTRAP Approach. Oxford (Blackwell) 1994.

[Little 2010] Jason Little The Only Agile Maturity Model You Need, Web publishing 2010 http://www.agilecoach.ca/2010/07/20/the-only-agile-maturity-model-you-need/

[Malik 2007] Nick Malik Simple Lifecycle Agility Maturity Model Web Publishing http://blogs.msdn.com/b/nickmalik/archive/2007/06/23/simple-lifecycle-agility-maturity-model.aspx

[Martens 2009] Ryan Martens, Does Agile Need Its Own Process Maturity Model? Web Publishing http://www.rallydev.com/agileblog/2009/06/does-agile-need-its-own-process-maturity-model/ see also http://agile-pm.pbworks.com/w/page/1526535/Agile%20Maturity%20Model

[Minick 2009] Eric Minick, Jeffrey Fredrick, Enterprise Continuous Integration Maturity Model, Urbancode Whitepaper, Web publishing http://www.agilejournal.com/pdf/Enterprise_Continuous_Integration_Maturity_Model.pdf

[Patel 2009] Chetankumar Patel, Muthu Ramachandran, Agile Maturity Model (AMM): A Software Process Improvement framework for Agile Software Development Practices, Int.J. of Software Engineering, IJSE Vol.2 No.1 January 2009, S. 3-28, Web Publishing http://www.ijse.org.eg/Content/Vol2/No1/Vol2_No1_1.pdf

[Pettit 2007] Ross Pettit An "Agile Maturity Model?" Web publishing 2007 http://www.agilejournal.com/articles/columns/the-agile-manager/52-an-qagile-maturity-modelq

[Proulx 2010] Martin Proulx, Yet Another Agile Maturity Model (AMM) – The 5 Levels of Maturity Web publishing 2010, http://analytical-mind.com/2010/07/12/yet-another-agile-maturity-model-the-5-levels-of-maturity/

[Rohrbeck 2011] Rohrbeck, René, Corporate Foresight, Heidelberg, Physica Verlag, 2011,

[Ronen 2011] Shirly Ronen, Agile Testing Maturity Model - practical view, Web publishing http://www.slideshare.net/shirlyronen/agile-testing-maturity-model-practical-view

[Rothman 2011] Johanna Rothman, Where is Agile Going? Web publishing 2011, http://www.slideshare.net/johannarothman/where-is-agile-going-withculture

[Royce 2009] Walker Royce, Improving Software Economics, Top 10 Principles of Achieving Agility at Scale, IBM Rational White Paper, Web Publishing http://www.agilejournal.com/pdf/RAW14148USEN.pdf

[Rudd 2011] Charlie Rudd, What is the Purpose of an Agile Maturity Model?, Web Publishing http://www.solutionsiq.com/the-agile-ceo/bid/51477/What-is-the-Purpose-of-an-Agile-Maturity-Model see also http://www.solutionsiq.com/the-agile-ceo/bid/51478/What-is-an-Agile-Maturity-Model

[Schauder 09] Schauder, Jens, Do we need an Agile Maturity Model? Web publishing http://blog.schauderhaft.de/2009/09/06/do-we-need-an-agile-maturity-model/

[Sehlhorst 2009] Scott Sehlhorst, Tyner Blain, Agile Maturity Model – What's Next? Web publishing http://tynerblain.com/blog/2009/06/30/agile-maturity-model/

[Seuffert 2009], Seuffert, Mark, Agile Karlskrona test, web publishing http://www.mayberg.se/archive/Agile_Karlskrona_Test.pdf

[Sidky 2007] Ahmed Sidky, James Arthur, A Disciplined Approach to Adopting Agile Practices: The Agile Adoption Framework, web publishing http://arxiv.org/ftp/arxiv/papers/0704/0704.1294.pdf

[Sims 2009] Chris Sims, Scott Ambler Revisits Agile Process Maturity Models, Web Publishing http://www.infoq.com/news/2009/04/Agile-Maturity-Models

[Schweigert 2012] Tomas Schweigert , Morten Korsaa Risto Nevalainen Detlef Vohwinkel Miklos Biro , Agile maturity model: Oxymoron or the next level of understanding, Proceedings of the SPICE 2012 Conference.

[SPI Manifesto 2010] SPI Manifesto Version A.1.2.2010 http://www.eurospi.net/images/documents/spi_manifesto.pdf (accessed: 21/02/2012)

[Steiner 2010] Michael Steiner, Monique Blaschke, Michael Philipp, Tomas Schweigert, Make test process assessment similar to software process assessment—the Test SPICE approach, Journal of Software Maintenance and Evolution: Research and Practice   Article first published online: 25 AUG 2010 DOI: 10.1002/smr.507

[Usman 2010] Usman Muzaffar, Seven Deadly Sins of Slow Software Builds, Web Publishing http://www.agilejournal.com/pdf/EC-WP_SlowBuilds.pdf

[Waters 2007] Kelly Waters, 10 Key Principles of Agile Software Development, web publishing http://www.agilejournal.com/resources/directory?sobi2Task=sobi2Details&catid=10&sobi2Id=14

[Woods 2011] Dan Woods, An Agile BI Maturity Model, Forbes web publishing 2011, http://www.forbes.com/sites/danwoods/2011/10/26/an-agile-bi-maturity-model/3/

[YIN 2011] Alexandre Paulo Guo Yin, Scrum Maturity Model, Dissertacao para obtencao do Grau de Mestre em Engenharia Inform´atica e de Computadores

[YIN et al 2011] Alexandre Yin, Soraia Figueiredo, Miguel Mira da Silva, Scrum Maturity Model, Proceedings of the ICSEA 2011 : The Sixth International Conference on Software Engineering Advances, S. 20 - 29

7 Author CVs

Dr. Miklos Biro

Key Researcher Process and Quality Engineering at Software Competence Center Hagenberg GmbH (SCCH), Full Professor (Ordentlicher Hochschulprofessor in German) nominated by the prime minister of Hungary, Doctor Habilitatus in the field of Management and Business Administration with software engineering, university teaching (including professorship in the USA), and management experience. Ph.D. in Mathematics, Master of Science in Management (Purdue University). Fluent in Hungarian, English, and French. Initiating and managing role in numerous European projects. Author of Hungarian and English language books and publications. Founding president of the professional division for Software Quality Management of the John von Neumann Computer Society, Hungarian national representative in IFIP TC-2 Software: Theory and Practice. Chair of professional conferences, member of programme committees and journal editorial boards.

Morten Korsaa

Process Specialist at DELTA - a Danish research and consultancy company. Morten has worked as a global SPI manager in a large telecom company and have as a consultant supported companies from 20 people to it departments of 3500. He has performed more than 100 assessments in both vendor and acquire organisations. Mr. Korsaa is SEI certified "Introduction to CMMI Instructor" and was proud to be member of the teams developing the ImprovAbility model, the PI manager education and the SPI Manifesto

Risto Nevalainen (Lic.Techn.)

Senior advisor and acting managing director in FiSMA (Finnish Software Measurement Association). Risto Nevalainen has long experience in software measurement and quality topics. His working experience includes position as managing director of Finnish Information Technology Development Center during 1989-1995. Mr. Nevalainen has participated in ISO15504 (SPICE) standard development since beginning. He is Competent SPICE Assessor and ISO9000 Lead Assessor. Nevalainen is trainer and job committee member in the ECQA SPI Manager profession.

Detlef Vohwinkel

Manager of the Process Intelligence Competence Center of SQS Cologne and responsible for servicing the support of the clients in the process assessment and improvement sectors in the whole of Europe. His main duties and responsibilities include the implementation of assessments and the organization and support of process modifications for the SQS clients. This also includes organizing trainings in this sector. He is a representative of the SQS in the advisory board of intacs. Business Point of Contact of the SEI for the CMMI Product Suite. Member in the Steering Group of the SIG Test SPICE. iNTACS™ certified ISO/IEC 15504 Principal Assessor SEI authorized CMMI® Instructor (V1.3) CMMI® Team Member Qualification
Intacs certified trainer for assessor courses (provisional and competent) Detlef Vohwinkel studied business economics at the University of Cologne and his degree focused on business informatics.

Tomas Schweigert

Senior Principal Consultant at SQS, Software Quality Systems AG, intacs certified SPICE principal consultant, head of ECQA JRC SPI Manager, V-Modell XT PING. Member in the steering Group of the Test SPICE SIG. Tomas Schweigert joined SQS in 1991; he is an experienced process assessor, model developer and process improver.

Transition to agile development - Experiences from a medical device manufacturer

Richard Seidl, GETEMED Medizin- und Informationstechnik AG, seidl@getemed.de

Abstract

This paper describes our experiences with the transition from a classical software and hardware development process to an agile development process. Within one year two major projects and teams switched to this new development process. The major problems and constraints but also the key success factors are our lessons learned from establishing this new development process.

Keywords

Agile development, medical device, software development process, regulatory, soft skills, process improvement

1 From classical to agile development

1.1 Motivation

After 25 years of developing medical devices and software mainly for cardiological diagnostics and ambulatory monitoring of vital signs in high-risk patients we acquired a consistent and complete quality management system for our company. This includes a structured and documented development process for our hard- and software which complies with the regulatory requirements such as DIN EN ISO 13485 [1] or DIN EN IEC 62304 [2]. With the constant growth of the organization and the department for research and development this implemented process reached its limits. So the development process had to be improved to accomplish projects more efficient and to shorten the development cycles. Inspired from industry best practices, experiences, conferences and literature we decided to investigate and evaluate agile development processes.

1.2 Choosing an agile development process

The major issues when choosing an agile development process are the constraints to stay conform to the regulatory requirements and the needed technical documentation. Classical development processes were often linked and referenced by regulatory requirements. Most of the agile development processes do not refer to regulatory requirements, especially not to regulatory requirements for medical device manufacturers. Therefore it was not possible to choose a strict implementation of an agile development process and the decision was made to adapt the methods, ideas and approaches from different agile processes to the development process of the organization with consideration of the regulatory requirements. So we picked agile principles and adapted them to our development process with the condition to stay conform to the relevant standards.

The new process was not rolled out to the whole organization but was implemented in our daily work in one pilot project which covers all relevant aspects of a typical development project in our organization:

- Development of hard-, soft- and firmware

- Medium complexity

- Broad range of involved departments and roles

Sources for the agile ideas, methods and approaches we choose were mainly:

- Manifesto for Agile Software Development [3]

- Kanban – Evolutionäres Change Management für IT-Organisationen [4]

- Agile Testing – A Practical Guide for Testers and Agile Teams [5]

- Implementing Lean Software Development – From Concept to Cash [6]

- Basiswissen Medizinische Software – Aus und Weiterbildung zum Certified Professional for Medical Software [7]

- The Scrum Guide [8]

2 Best practices and constraints

Agile development processes have numerous methods and best practices which can be used. During the implementation and improvement of our new process, we identified some of these practices which have the most benefit for us since our process transition. On the other side some constraints are given, especially from the regulatory requirements which need to be reflected at every time.

2.1 Planning of small iterations

The biggest impact to our project management in terms of controlling the project and to the efficiency of the development team was the split of the project in small iterations. For the big picture and planning a classical project plan was set up. But for the operative work in the development team this plan is divided into small iterations (sprints). A sprint has a duration of 2 to 4 weeks and the planned work has to be finished within this sprint.

Each iteration is planned in a separate planning meeting at the beginning of a sprint where the product owner (product manager) explains his high priority requirements and discusses these requirements with the development team (consisting of developer and tester). So all relevant roles – the product owner, the developer and the tester – have the same understanding of the requirements. Based on this same view the development team decides what they are able to deliver within this sprint. But to deliver a part of the product doesn't mean just finishing the development. It also means that the following tasks are done in this sprint:

- Design specification and review

- Unit and component testing

- Refactoring of existing parts

- Review of the sources, schematics, etc.

- Integration and system test

- Regression test of the existing parts

This ensures that the development artefact at the end of the sprint is not implemented quick & dirty but has an assured quality and a consistent documentation.

At the end of the sprint the results are presented by the development team and reviewed by the product owner, which gives his feedback directly to the team.

There are a lot of best practices how to organize planning meetings and sprints and how to estimate the work for the upcoming sprint. But based on the experience, it is more important to change the project to work with these small iterations as the details of planning and estimating. The methods and approaches used to organize and manage the iterations may change during the project and should be adapted to the needs of the development team.

The big advantage of splitting the work in such iterations is the possibility to control the progress more often, to review the development results during the project and to get one point of view to the requirements between the development team and the product owner.
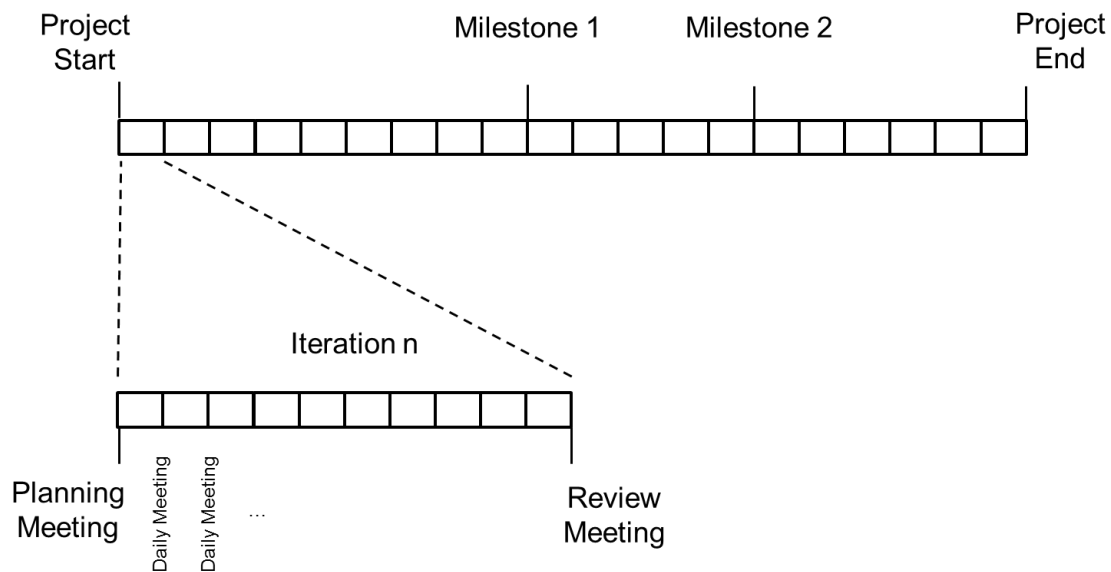
Figure 1: Referencing Project Plan & Iterations

2.2   Communication and collaboration

Within the last years of development the team members got more and more focussed on "their" work which leads the organization to two major problems:

- Some code, schematics, etc. was only understood and maintainable by one or two developers. In case of issues during illness or vacation an analysis of the problem was very hard.

- Developers got sometimes lost in solving their problems which maybe can be solved by other developers of the team in less time. This causes a big loss of efficiency.

To avoid these problems, two approaches were chosen: the daily meeting and continuous reviews.

2.2.1 Daily meeting

In a daily meeting, time boxed e.g. to 15 minutes, each team member tells the others what he had done since the last daily meeting, what he will do until the next meeting and if there are any problems which block his work. With this short meeting, maybe done as a stand-up meeting in the morning, each team member knows what's going on in the project. This reduces the risk, that a team member gets lost in a problem and wastes time with solving it where another team member can solve it very quick. It's essential that this meeting is not understood as "reporting to the project lead" but as know how transfer to the other team members.

2.2.2 Continuous reviews

One part of the „definition of done" defined for one sprint is that all artefacts (design documents, source, schematics, …) are reviewed within the sprint by at least one other team member. Based on the complexity and the risk some documents are reviewed in a formal way, other in an informal way. This leads on the one hand to a continuous quality check of the artefacts and to a broad knowledge about the artefacts in the team. The quality of the artefacts of our development teams increases significantly by implementing continuous reviews. Some developer also took one step forward and started coding with pair programming, where two developers worked on a problem at one workstation.

2.3   Tools

Within the implementation of the new process we evaluated our existing tools to support our new process. Our main rule was that we first want to define a part of our process as it fits to our needs and then decide which tool can support us. The tools for developing stay mainly as they were. As a tool for managing our whole development process we decided to use an application lifecycle management tool: Polarion ALM [9]. Polarion is a very generic tool with the possibility to customize nearly the whole product. It is based on the version control system Subversion [10] and works with work items (e.g. requirement, design, test case, anomaly, etc.) which can be linked together and to other artefacts like source code, etc. The main benefits for us were:

- Traceability over items: We are able to show the implementation path e.g. from a requirement over the design to the test cases, the found bugs and the according source code. So it is possible to do simple impact analysis if one item changes and furthermore this traceability is an essential part of the regulatory requirements for medical device manufacturer.

- Traceability over time: Because Polarion is based on Subversion, it is possible to get back to a set of work items at a specific time point in the past. For example you can view the requirements as they were in an old release and can show up the changes from the old release to the actual state of the development.

- Customization: Polarion gives us the ability to adapt the tool to our process needs. There are a lot of templates, APIs and configurations which can be used to design a workflow that fits the needs.

Another big issue was the task management of the development teams. Different tools were used for the tasks, e.g. Microsoft Outlook and Polarion but both were not usable for fast tracking, changing and working with the tasks. This leads to the situation, that the developer didn't use the tools. After some iterations a really light-weight tool was found to manage the tasks: a simple whiteboard with post-it-notes. Each note is one task which can have the state open, in progress or done. On the notes also the relevant items in Polarion (e.g. defects or requirements) are referenced. The whiteboard is also ideal to use for the daily meeting, because it shows the actual state of the iteration.

Figure 2: Sample of a whiteboard for task management

2.4   Team motivation and mind set

More than in classic development projects soft skills of the team members and the team motivation are essential for the success of the project. All team members have to commit to the new methods

and approaches for a successful implementation of the process improvement. The experience of our development team shows that sceptic team members mostly had not understood the benefits e.g. of the small iterations. But presenting the results iteration by iteration in the review meeting were the best argument.

The agile methods often contain a funny element which motivates the team member. Some ideas which inspired us:

- Daily Stand-Up meeting outside the office building

- Motivation cards for team members

- Bug hunting sessions

- Planning and estimation poker

- Small challenges

But all the change is not only a process change. Also the mind set and the culture of the organization are affected. For a development team who works in a classical development process this is sometimes not easy and cannot happen from one day to another. Especially if the developers worked only at their domain with a poor interaction with the team. So everyone has to be aware that the transition takes time and the efficiency may decrease at first. Within our projects 5 to 7 iterations were needed to be more productive and efficient than before.

2.5   Regulation and documentation

One characteristic of developing medical products is the big amount of regulations that have to be fulfilled. In case the medical device manufacturer wants to join the worldwide market there are also a lot of regional standards that have to be fulfilled to get accredited in these countries. Next to the product standards there are process standards which require a traceable and consistent documentation from the first to the last step of the project. Another aspect is quality and risk control. A medical device has a lot of strong requirements for safety and reliability, which causes a broad range of quality assurance along the whole project from reviews, static analysis to dynamic tests.

These requirements for documentation, quality assurance and risk control have to be part of your process at every iteration of the project. When we began with the definition of our new development process it looked like a no-go for this transition to an agile process. But our experience shows that these regulatory requirements can be fulfilled in an agile process as well. For us, the agile process even has advantages compared to the classic development process in two areas:

- Documentation: According to our definition of done at the end of each iteration also the documentation has to be done. This means the documentation (e.g. requirements, design specification, test cases, etc.) have to be consistent to the development artefacts and have to be reviewed within one sprint. This brings us to a continuous, consistent and quality checked documentation at the end of every iteration.

- Test automation: Just as the documentation also the test automation grows with every iteration. It has to be adapted and executed in every sprint, otherwise the iteration is not finished.

In both cases our experience shows that within a few iterations not only the tester pursues the goal to finish the documentation and test tasks in an iteration but the whole team starts to work intensely at documentation and test automation. Additionally an employee from quality management joins the agile development team to support these activities.

3 Conclusion

In summary the transition to an agile development process worked for us as a medical device manufacturer. The regulatory requirements require to think about every step in changing the process, especially because the agile methods and approaches do not focus on documentation and regulatory as needed in the medical area. So an adapted agile process has to be used. The methods, ideas and approaches which are established in agile development can be taken to improve the existing development process. This takes us to a standard conform and more efficient development process which is now applied to two big projects developing medical hardware and software.

4 Literature

[1] DIN EN ISO 13485:2010 - Medical devices - Quality management systems - Requirements for regulatory purposes

[2] DIN EN IEC 62304:2009 – Medical device software – Software life cycle processes

[3] Manifesto for Agile Software Development, http://www.agilemanifesto.org

[4] Anderson, D. J.: Kanban – Evolutionäres Change Management für IT-Organisationen. d.punkt.verlag, Heidelberg, 2010

[5] Crispin, L.; Gregory, J.: Agile Testing – A Practical Guide for Testers and Agile Teams. Addison-Wesley Longman, Amsterdam, 2008

[6] Poppendieck, M.; Poppendieck, T.: Implementing Lean Software Development – From Concept to Cash, Addison-Wesley, 2007

[7] Johner, C.; Hölzer-Klüpfel, M.; Wittorf, S.: Basiswissen Medizinische Software – Aus und Weiterbildung zum Certified Professional for Medical Software, d.punkt.verlag, Heidelberg, 2011

[8] Schwaber, K.; Sutherland, J.: The Scrum Guide – the official rulebook, http://www.scrum.org, 2011

[9] Polarion ALM, http://www.polarion.com

[10]   Subversion, http://subversion.apache.org

5 Author CVs

Richard Seidl

Since early 2011 Richard Seidl has been test expert and test manager at GETEMED Medizin- und Informationstechnik AG, Teltow (Germany). His work focuses on the planning and implementation of software and hardware tests of medical devices and the implementation of development processes. After graduating in communications engineering, Richard Seidl began working as software engineer and tester in the banking field. From 2005 to 2010 Richard Seidl was test expert and test manager at ANECON GmbH, Vienna (Austria). In 2006 he acquired the Full Advanced Level Certification of the ISTQB®. In the following year he qualified as IREB Certified Professional for Requirements Engineering and as Quality Assurance Management Professional (QAMP). He published the books "Der Systemtest" and "Software in Zahlen" together with Harry M. Sneed and Manfred Baumgartner and "Basiswissen Testautomatisierung" together with Thomas Bucsics and Manfred Baumgartner.

ISaPro®: A Process Model for Safety Applications

Hans Tschürtz and Peter Krebs
Vienna Institute for Safety & Systems Engineering
FH Campus Wien - University of Applied Sciences,
Favoritenstrasse 226, Vienna, Austria

Abstract

In the field of system safety and functional safety a lot of different standards with differing requirements are demanded. Furthermore, the realisation of innovative ideas in software increases the complexity and leads to dangerous system states or even failures that put the safety of a system at risk.

This paper introduces a framework model, the Integrated Safety Process (ISaPro®) that structures the different requirements to a systematic approach. The ISaPro® is an original work from the Vienna Institute of Safety and Systems Engineering. It provides the opportunity to integrate the required activities of different standards into defined processes of the different life cycles of the model. The model consists of the project, engineering and safety life cycle. Furthermore the model can be tailored to different project situations. The essential part of the ISaPro® is a generic process landscape, combining the main processes for system development in the safety-critical area.

An example demonstrates the application of the ISaPro®, in a software development project. It shows how to derive an engineering life cycle and adapt the project and safety life cycle to a project landscape that includes the base practices of all relevant disciplines.

Keywords

system safety, functional safety, SW engineering, project management, IEC 61508

1 Introduction

Some of the most important safety standards – especially in Middle Europe – are the IEC 61508 [1] and the ISO 26262 [2]. Both standards are treated as functional safety standards. Functional safety is achieved by safety functions that are realised by safety-related systems. These functions are primarily implemented in Electrical and/or Electronic and/or Programmable Electronic (E/E/PE) technologies, i.e. E/E/PE safety related systems. Since the philosophy of implementing safety functions is not sufficient any more, it is necessary to consider system safety. The basis of system safety is treated in the MIL-STD-882 [3] on system basis and in the DO-178B [4] on software basis. Clifton Ericson defines system safety in [5] as an 'engineering discipline for developing safe systems and products where safety is intentionally designed into the system or product'. MIL-STD-882 defines system safety as 'the application of engineering and management principles, criteria, and techniques to achieve acceptable mishap risk, within the constraints of operational effectiveness and suitable time and cost, throughout all phases of the system life cycle'.

Providing system safety is a continuously growing challenge since all these safety standards do not show a generic and efficient approach that covers the demanded requirements. A methodical, process-oriented, generic and defined approach that fulfils the requirements of the standards is not yet available. In fact, the industry and especially small and medium-sized companies suffer from a lack of implementation knowledge when it comes to a project in the safety related area. Additionally, standards like ISO/IEC 15504, also known as SPICE (Software Process Improvement and Capability Determination) or CMMI (Capability Maturity Modell Integrated) are demanded from the project owners.

Apart from that, seemingly additional costs that are induced by the development of safe systems pose a great challenge with respect to competition in the marketplace. This paper describes a process model that covers all mentioned aspects. The ISaPro® structures the way of working, so that all the demanded activities (demanded from different standards) will be performed at the right time in the right phase. Since the model can be easily adapted to different project situations, it reduces the organisational and technical complexity.

The next section describes an overview of the ISaPro®. It explains the structure of the model, its life cycles and the main processes of each of them. Section three shows an excerpt of a practical application. It explains how to tailor the model to a specific project situation and how to follow the model. The last section concludes with a summary.

2 Integrated Safety Process (ISaPro®)

The ISaPro® is a framework model that includes all essential disciplines for developing a system for the safety-critical area. As depicted in Figure 1, this model is composed of a generic process landscape. It consists of project, safety and engineering life cycle and all the necessary support processes. The model is split up into 4 sections: the three project-oriented spaces – problem, model and solution space – and the operation space which is not part of this paper. Within each of those sections the different phases of the diverse life cycles can be fit in where suitable and can be synchronised with each other. Each life cycle shows the main processes of the model. All the demanded activities for each process can be derived from the required/different standards (e.g. ISO/IEC 15504 and IEC 61508).

Stage gates (within the project management life cycle) denote the limits of each space section. Stages are specific periods in the project and gates are decision points which precede every stage. Unless specific criteria have been met, as evidenced by certain approved deliverables, the subsequent stage will not be started. Individual milestones can be defined in the different life cycles in between the stage gates. Usually, at milestones, and certainly at stage gates, work products of the activities in the respective concluded phase are submitted. Those work products, even from the different life cycle phases have to be exchanged between them. For the sake of comprehensibility the following sub-

chapters explain the project, safety and engineering life cycle as well as the vertically ordered spaces.
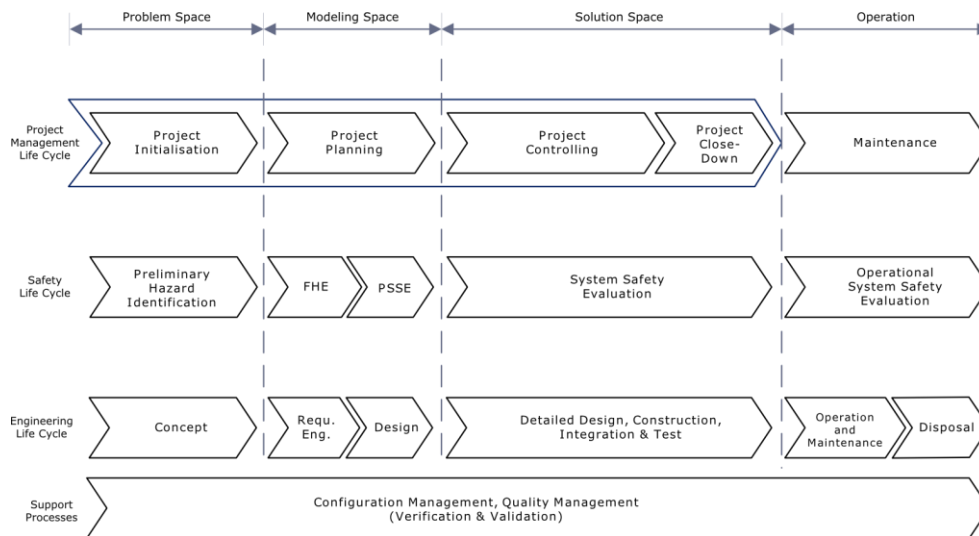
Figure 1: Framework of the Integrated Safety Process (ISaPro®)

The main objective of the problem space - from a business point of view - is that the project costs can be easily estimated. This requires a pre-estimated Safety Integrity Level (SIL)[1], which is derived in the preliminary hazard identification process, on the basis of a technical concept. The modelling space needs well defined requirements and a well-defined system design. These are the sources for a Functional Hazard Evaluation (FHE) and the Preliminary System Safety Evaluation (PSSE). Derived safety goals, based on the safety integrity level, will be incorporated into the relevant processes and considered in the project planning. The solution space ensures that the safety goals will be fulfilled according to the planned activities.

2.1 Project Management Life Cycle

The project management life cycle for development projects is divided into the four sub-processes of *Project initialisation*, *Project planning*, *Project control* and *Project close-down*. Generally, the objectives of the project management process are to manage the project complexity and dynamics, to successfully achieve the project objectives and to manage the relationships of all processes of the process landscape.

The goal of the project initialisation process is to determine the scope of work, to identify the required SIL and to create a preliminary project plan. All these issues are the basis for the major output, the estimated cost of the project.

The project planning process covers project management, engineering and safety activities. In this paper the engineering process is applied to the development of a system in the software development domain. The project management plan is the basis for all controlling activities in the project.

The objectives of the project close-down process are the planning of the post-project phase, completion of the remaining tasks, project evaluation and the dissolving of the project team. An additional important task is the transfer of 'lessons learned', especially safety issues, to the permanent organization and to other projects. Ultimately, the project owner's approval of the project finalizes the project close-down.

---

[1] IEC 61508 specifies four levels of performance for a safety function, termed Safety Integrity Levels. SIL1 is the lowest level and SIL4 the highest.

2.2   Safety Life Cycle

The safety life cycle of the ISaPro® consists of *Preliminary Hazard Identification (PHI)*, *Functional Hazard Evaluation (FHE)*, *Preliminary System Safety Evaluation (PSSE)* and *System Safety Evaluation (SSE)*.

During the PHI, high-level hazards will be identified based on a first technical concept of the whole system. The creation of the technical concept is part of the engineering life cycle and is described in the next chapter. Using checklists and available data such as lessons learned, accident records, known hazards in an existing hazard log or relevant review results, a group of experts performs a brainstorming session to create a preliminary hazard list. Additionally, a Preliminary Hazard Analysis (PHA, as described in [5]) creates safety goals and high level safety requirements which are the basis for the first SIL allocation. The SIL allocation is the major input for the project management plan.

The preliminary hazard list and the customer requirements form the basis for the Functional Hazard Evaluation (FHE). While considering the required functionality and the respective system environment the FHE shall answer the question: How safe does the system need to be? The analysis performed during the FHE provides the safety objectives and a first set of safety requirements which are necessary in order to fulfil the safety goals and to prevent the identified hazards from occurring. The system requirements are amended by the safety requirements.

The Preliminary System Safety Evaluation (PSSE) is performed at each design level. The PSSE shall answer the question: Is tolerable risk achievable with the proposed solution? Therefore, it is verified that the safety objectives can be fulfilled with the available design. Of course, this evaluation step can result in the definition of additional derived safety requirements. If a design modification is required due to, for example, improper implementation of safety requirements or additional identified hazards, the PSSE must be repeated for the altered design. The purpose of the SSE is to check, if the safety requirements are performed in the right way that level of residual risk is reached.

2.3   Engineering Life Cycle

The generic engineering life cycle of the ISaPro® model consists of the concept phase, which lies in the pre-project phase, the requirements engineering, design and the realisation phase. The operation and disposal phases are not part of this paper. Each phase consists of several processes. Their level of detail depends on the project size and complexity of the system.

A concept is a first draft or a raw version of the system design. It is derived from the project objectives and considers the system environment. In the broader sense, it is a combination of the customer's idea with technical solutions. This is the foundation for a first hazard identification (PHI, as mentioned before) from which the safety integrity level (SIL) can be derived. The actual SIL allocation is used as input for the project management plan and treated as an objective of the project.

The Requirements Phase is a thorough exploration of the intended system with the intention of discovering the functionality and the behaviour of the system. The safety requirements that have been established during the PHA are incorporated into the system requirements specification but specially treated due to their importance.

The requirements specification is input for the design phase. Of utmost importance is the traceability to the defined high level requirements, especially to the safety requirements. Correct realisation of the design and all design components is highly dependent on the design structure that includes the right safety solution. When the concept confirmation phase has been completed, the requirements specification and the design document need to be approved in order to enable commitment from all relevant parties for the implementation. All documents have to ensure that an adequate safety integrity level has been achieved.

The necessary processes for the realisation phase depend on the objectives, size and complexity of the project. Projects of large size and complexity necessitate the creation of a detailed design or requirements specification by breaking up the design into smaller elements – in such cases FHE and

PSSE have to be repeated. Considering a software development project, the next step could be a software requirements specification or software architecture.

Since this paper discusses a software development project only, the realisation phase consists of a software implementation process and the defined test processes. The test plan is part of the overall project management plan. For large projects the test plan should be a dedicated document which must be synchronised with the overall project management plan.

The essential support processes for safety-critical development projects are verification, validation, configuration management, change management and quality assurance..

3 An Example Application of the ISaPro®

The following sections demonstrate the application of the ISaPro® in the development of a safety-critical software. The intention is to derive a project landscape according to the model's framework and to clarify the activities carried out in each of the phases.

3.1 Derived Project Landscape

This section demonstrates how to derive a project landscape for a software development project. IEC 61508 and ISO 26262 require the selection of a software life cycle model and suggest the V-Model. The selected model has to be elaborated at the latest during the project planning process and has to be specified in the project management plan. The derived V-model, adapted to the business requirements of the actual project, is the baseline for the whole process model. It shows the needed engineering and support processes, in that case verification and validation process. According to the size and complexity of the project additional support activities or processes, like configuration management and quality assurance or a certification liaison process, have to be adapted.
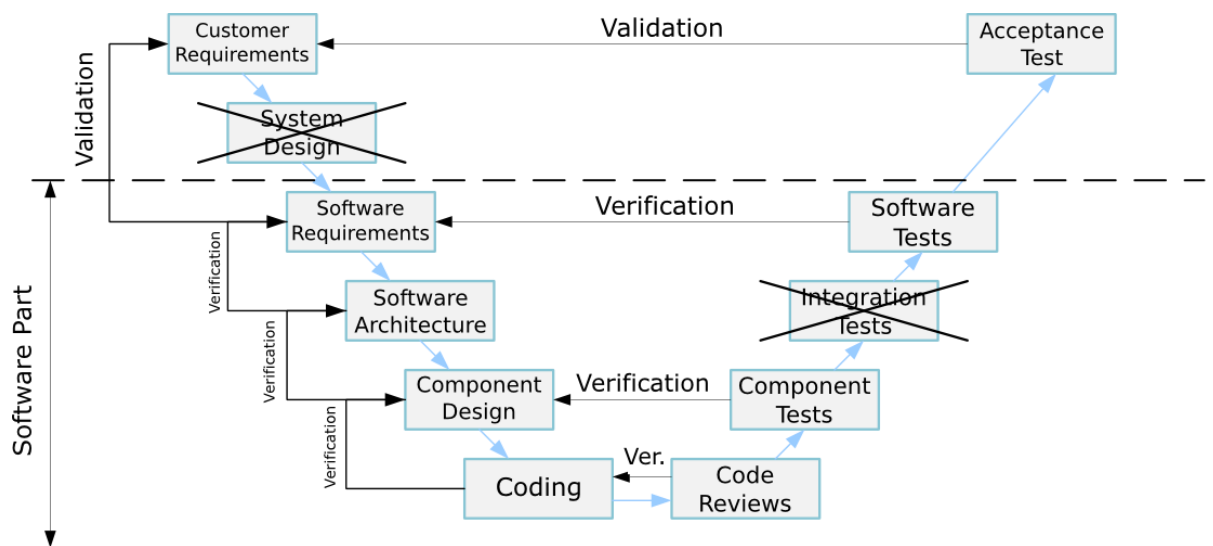
Figure 2: Adapted V-Model according to project requirements. Processes pertaining to customer requirements and system development are only shown for the sake of completeness and not considered in the example.

The example described demonstrates the flexibility of the ISaPro® model by customising it to a SW

development project only. The first step is to adapt the V-model for the specific conditions of a project as shown in Figure 2.The described V-Model shows the necessary steps of the demanded development. Hereby, the right transition into the particular steps is of utmost importance. The essential goal lies in the error-free, complete and traceable transfer of the individual requirements starting from the customer requirements down to the software code.

The first steps concern the customer requirements on the left side of the V-model. Since only software is developed and hardware is not clearly defined, the customer requirements will be directly transferred into the software requirements. System requirements and system design steps are not necessary. Because of unclear hardware conditions, safety application conditions will be worked out with the customer. Next, the software architecture is developed according to the defined software requirements. In order to reduce the complexity of the software part the whole software is broken down into several software components which are described in the component design specification. The latter is the basis for the coding of the software. The transitions from one phase to the next are of particular importance. Reviews and inspections are performed to ensure the effective transfer of the requirements during those transitions.

On the right side, the model depicts the individual testing phases during which the outputs of the corresponding phases on the left side are tested and verified, respectively. Code reviews are carried out during the coding phase and verification tests are conducted from the component phase onwards to the system phase. Finally, the validation of the whole system is carried out at the customer level by means of acceptance testing. The basic prerequisites for an effective and efficient procedure are the clean definition and clear communication of the transitions between the particular phases. The quality of the transitions has to be ensured by reviews performed by the relevant project roles. Of utmost importance is the traceability starting from the customer requirements via the software requirements and software architecture, the component design and code to all testing activities on the right side of the V-Model and backwards.

The derived V-Model is needed to identify the coherence between the development steps on the left branch and the testing activities on the right branch of the V-Model. In the following step the V-Model is transferred into a software development life cycle, to display the processes on the time axis. Each step of the V-Model is created in a way that fits to existing processes of the ISaPro®. Additionally, the above mentioned support processes, configuration management, validation & verification and quality assurance are also displayed in figure 3.

As a next step, the safety life cycle and the project management life cycle will be adapted to the development life cycle.
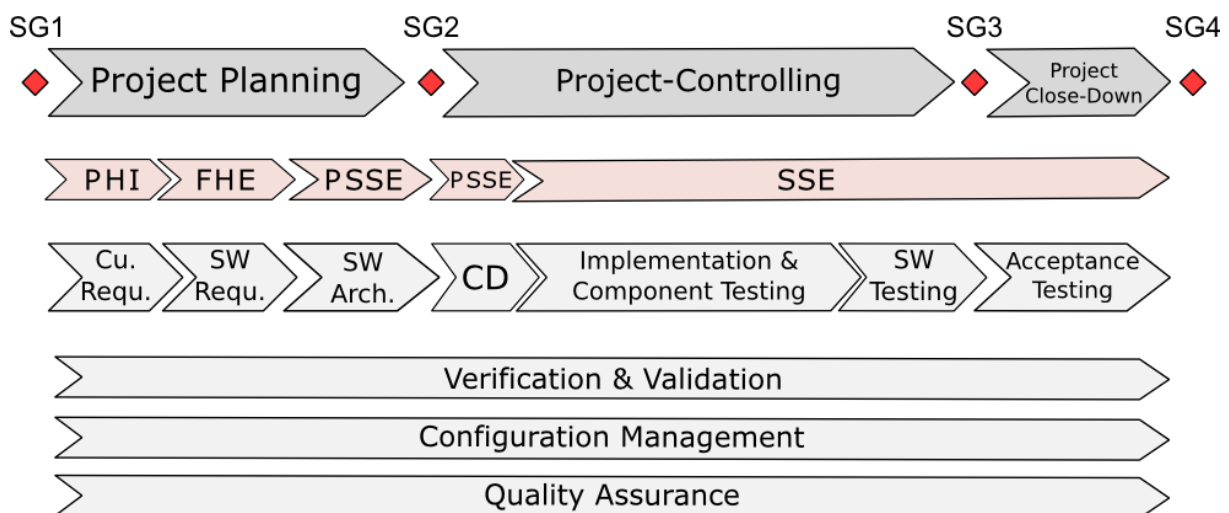
Figure 3: SW project landscape for the example. The phases are derived from the generic landscape of ISaPro® described in the prior sections.

As shown in Figure 3, all the life cycles of the generic landscape are present but only the processes necessary for the SW development remain in the engineering life cycle. The project management and safety processes are adapted accordingly to form a complete process landscape again. Together they fulfil the aspects of system safety as defined in [5] by considering both engineering and management principles.

In the following sections, the derived landscape will be applied to the development of an example SW with safety aspects to explain the processes and their activities in detail. Since this practical example shows a software development project only, it is purposefully constrained to the SW development life cycle with its relationship to the derived software safety life cycle – i. e. the processes of the project management life cycle and the support processes are not considered.

3.2   Definition of the Example Software

In order to give concrete examples of activities and outputs, a sample application is defined on which the subsequent analyses and methods are performed. This application, termed the Safety-related SW (SR-SW) has the following characteristics:

- The main function of the SR-SW is to read a configuration file from the file system and check its content for validity.

- To read the file, the SR-SW directly or indirectly uses the standardised interface provided by the Operating System (OS) – e. g. the functions defined by POSIX:2001 (see [6] for details).

- How the configuration file is stored on the file system is not constrained – it could be written locally or transferred over the network.

- Other functionality of the SR-SW is irrelevant, i. e. only the part that deals with reading and checking the configuration file is considered.

Figure 4 shows an illustration of the SR-SW in relation to the OS and the configuration file on an arbitrary host platform. Furthermore, it is assumed that the configuration file is transferred over an unspecified network (e. g via FTP or SSH) to the host and stored by the OS.
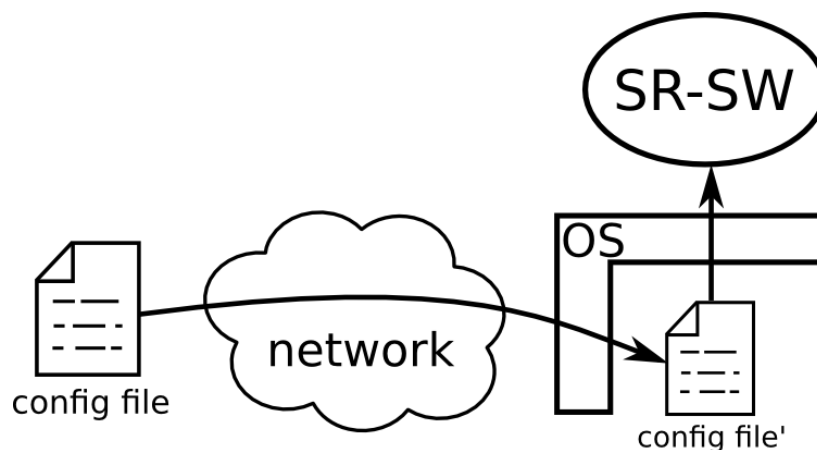
**Figure 4: Schematic illustration of the SR-SW sample application
and its environment as basis for the hazard analysis.**

Although this may seem like a contrived example, this exact scenario was part of a real-world project

in which the ISaPro® was applied to the development of a complex safety-critical SW.[2] It is therefore deemed suitable to demonstrate the principle approach of the ISaPro®.

3.3   Step 1: Functional Hazard Evaluation (FHE)

The objective of the FHE is to determine the risks of all hazards of a safety-critical SW, specify risk-reducing countermeasures, if necessary, and to ultimately formulate safety requirements. Although part of the safety life cycle, this nonetheless concerns the SW engineer, as specific know how on the utilised technologies is necessary to determine potential problems and to assign risk measures in a meaningful way.

The input to this phase is the list of hazards identified in the prior PHI. For the SR-SW the following hazard is chosen: 'The SR-SW uses a wrong configuration and does not perform its intended function'. Note that "wrong" can either mean that a valid configuration file was changed to an invalid one (i. e. the syntax is no longer correct) or that it was changed to a different but still valid one (e. g. a syntactically correct but unintended change in a parameter value). Considering the environment of the SR-SW the following possible causes for this hazard can be identified:

- The configuration file is changed during the transfer over the network (e. g. bug in FTP/SSH application).

- The configuration file is changed when written to the file system (OS error)

- The configuration file is changed while stored on the file system (e. g. bit flip in solid-state storage)

- The configuration file is changed when read by the SR-SW (OS error)

Once identified, the possible causes are qualified with a risk based on the probability of occurrence and the severity of the effects. If this risk is deemed too high, countermeasures must be formulated. For the identified causes, this could be:

- The configuration file must include a hash value[3] calculated from its contents. The hash is computed before the configuration file is transferred to/stored on the host of the SR-SW. The hash of the contents (excluding the hash in the file) is calculated by the SR-SW when the configuration file is read and compared to the hash in the file itself. If the two values are different, the file was corrupted and is rejected as invalid.

This countermeasure is a safety function and must perform with a sufficient degree of safety integrity. Assuming that the development should comply to IEC 61508, the required integrity is arbitrarily determined to be SIL2.[4] Note that the safety function is actually not considered a distinct system (as in: separated from the safety-critical SW) but rather an inherent part of the application in conformance to the paradigm of system safety.

3.4   Step 2: SW Requirements Engineering

This phase has the objective to formulate the functional and non-functional requirements of the SR-SW, based on the top level requirements negotiated with the customer and the results of the prior step. Only the last aspect is considered for the example, as the other (non-safety) functionality of the

---

[2] The ISaPro® was initially applied to the development of a safety-critical Voice-over-IP (VoIP) client SW according to IEC 61508. Part of this client was a component which parses and checks a configuration file – in essence the functionality provided by the SR-SW.

[3] A hash function takes a data set of arbitrary size as input and calculates a hash value with a fixed length which is representative of the input data. A good hash function is sensitive to minor changes in the input data so the hash value can be used to quickly assert if a data set has changed.

[4] Although only IEC 61508 is considered here, the requirements of many safety standards pertaining to SW development are sufficiently similar, as demonstrated in [7] to not constrain the example too much.

SR-SW is out of scope.

The formulation of safety requirements follows directly from the countermeasures defined in the FHE and must comply to the constraints imposed by the safety standard. This means, the methods used to formulate requirements must be chosen according to the SIL of the safety function from the choices listed in the standard. For SIL2 the IEC 61508 recommends[5] one of the following methods ([1], Part 3, Table A. 1):

- Semi-formal methods (e. g. UML)

- Formal methods (e. g. Z notation)

The actual choice is, of course, dependent on the know how of the SW engineer and the complexity of the requirements. Independent of the choice, it is assumed that a set of atomic safety requirements for the safety function will result. Examples of such requirements, formulated in an informal way, are:

- Usage of the MD5 algorithm for calculating the hash as defined by RFC 1321.

- Exact format for the string containing the MD5 value in the configuration file (e. g. delimited by the characters '#').

- Syntactic rules for the occurrence of the MD5 string (e. g. only at the end of the file).

In addition, the safety standard imposes requirements on how the safety function is to be implemented – i. e. the SW engineer must adhere to a number of constraints both on the utilised technology (such as the programming language), the tools (such as the compiler version) and the methods (e. g. white box testing with specific minimum code coverage). These "implicit" safety requirements must also be formulated as they impact the subsequent development phases. Again, the exact set of constraints depends on the allocated SIL per safety function. Concrete examples of implicit safety requirements for the SW-SR according to [1], Part 3 could be:

- Implementation of the SR-SW in C.

- Adherence to a coding guideline and coding style guide.

- Utilisation of static analysis and code reviews.

- Execution of white box tests with the goal of 100\% statement coverage.

- Enforcement of metric limits for source code units (e. g. maximum value for Lines of Code (LOC) per function).

To demonstrate how these implicit safety requirements are fulfilled by the subsequent phases, the requirement for the adherence to a coding guideline is chosen as it tends to impacts the design and implementation more than other requirements. In particular, it is required to follow the well-known MISRA-C coding guidelines.[6]

In some cases, the implementation of a safety function is not sufficient to guarantee the allocated safety integrity. This occurs, when the safety function depends on properties of the environment which it can not compensate. In this case, the safety requirements must be complemented by a set of "Safety Application Conditions" (SACs) – basically requirements for the customer (!) which must be fulfilled to guarantee the allocated SIL of the safety function. For the SR-SW, a possible SAC could be:

- The MD5 hash must be computed on a system distinct from the host system of the SR-SW to eliminate the possibility that faults of the OS can compensate themselves – i. e. an error occurs in the MD5 calculation that masks a subsequent change of the configuration file's content.

---

[5] "Recommended" in the context of the IEC 61508 means, that a justification must be provided in the case that the method is not applied or an alternative method is chosen.
[6] A coding guideline, also called a "subset" restricts the functionality of a programming language to exclude unsafe construct/functions. The MISRA-C guidelines are widely used in the automotive and in other safety-relevant sectors.

3.5   Step 3: SW Design and SW Component Design

The aim of this step is to define dedicated *components* which will implement one or more requirements by way of specifying their functionality and interfaces. For the example, a component named "CONF" is defined and an informal specification of its functionality could be:

- Open and read configuration file.

- Check syntax of configuration file.

- Calculate MD5 hash of configuration file content.

- Check calculated hash against hash in file.

- Return parsed configuration parameters or error.

Both external (to the OS or other processes) and internal (to the rest of the SR-SW) interfaces must be specified for CONF along with the data that is transported over them, e. g.:

- CONF to file system (external interface)

    o   interface: open() and read() functions as defined by POSIX

    o   input data: file path of configuration file

    o   output data: a byte stream terminated by EOF OR error

- SR-SW to CONF (internal interface)

    o   interface: the function read_config_file()

    o   input data: file path of configuration file, buffer for parsed configuration parameters

    o   output data: parsed configuration parameters OR error

Note that one of the inputs of the internal interface is a buffer for the returned data, assumed to be statically allocated by the caller in advance. This is necessary because the MISRA-C guidelines prohibit dynamic allocation of memory – an example of an implicit safety requirement influencing the design of the SW.

Depending on the complexity it may be necessary to further refine a component in a SW component design by splitting it up into finer-grained *modules.* For CONF, two modules are defined as follows:

- **CONF_PARSE** - Opens, reads and parses configuration file content into configuration parameters.

- **CONF_HASH** - Calculates MD5 hash of configuration file content and checks against hash in file.

The interfaces of CONF_PARSE would be the same as for the CONF component. The CONF_HASH module is only used internally and has a single interface with the configuration file content and hash as input and a return value indicating a valid or invalid hash as output.

The component and modules are described informally here only for the sake of simplicity. As with the requirements, the actual SW design and component design must utilise methods from the safety standard appropriately chosen for the SIL. For SIL2, [1], Part 3, Tables A.2 and A.4 recommend one of the following:

- Structured diagrammatic methods (e. g. Real-Time Yourdon)

- Semi-formal methods

- Formal methods

In addition, every design decision (e. g. specific algorithms, interfaces and assumptions) which may

impact the safety of the SW must be thoroughly documented[7].

After completion the design is analysed by the safety engineer in the PSSE phase to assure that the safety requirements are correctly incorporated into the design and that the latter does not introduce new safety problems. For example, it could be determined that the collision probability of MD5 is too high for the typical size of a configuration file and that its successor SHA-1 must be used instead.[8]

3.6   Step 4: SW Implementation

The components and modules are now implemented in the chosen programming language(s) according to the constraints laid out in the design. During implementation, two separate verification paths are followed:

- **Static analysis** - The source code is analysed for suspicious and wrong constructs without being actually executed. This includes the application of a standalone analyser (also known as a lint-like tool), various compiler flags (e. g. the -Wall flag of GCC), manual or tool-assisted code reviews and source code metric reports. The goal of this approach is to verify conformance to the coding guidelines and coding style guide as well as detecting overly complex and error-prone code parts.

- **Dynamic testing** - The traditional approach of executing the source code with known test input and verifying that the expected output results. Tests on the individual modules and components are carried out in a separate phase discussed in the subsequent section.

Additionally, the safety standards demand a rigorous documentation of the source code in terms of interfaces ([1], Part 3, Table B.9). Inline documentation is recommended since it is easier to keep it up-to-date compared with documentation separate from the code. Furthermore, all source code artefacts must be subjected to a configuration management (e. g. via CVS, Subversion or Git).

Applied to the SR-SW, the following activities, besides the actual implementation, could be part of this phase:

- Using a static source code analyser to check against the MISRA-C coding guidelines.

- Checking compliance of the source code to the coding style guide via manual review or tool.

- Performing a manual code review for all MISRA-C rules that can not be checked by the analyser.

- Generating metric reports and refactoring code parts which exceed the metric limits.

- Writing inline function documentation (e. g. with Doxygen markup).

- Checking in all source code and header files into a Subversion repository, using the Id tag to automatically include data such as time of last modification or the last person who modified the code.

From the SW construction phase onwards, the safety engineer periodically assesses the SW life cycle in the SSE phase. The primary objective is to assure that the safety requirements are correctly implemented and tested with enough rigour to ensure the allocated safety integrity. Furthermore, the outputs resulting from the verification activities are checked to ensure their adequacy as evidence for the safety case and, if necessary, additional data is demanded. For example, the MD5 implementation might be subjected to probabilistic tests in addition to simple test vectors as recommended in [1], Part 3, Table A.7.

---

[7] The IEC 61508 requires this documentation under the rather obscurely named point of 'Data recording and analysis', explained in [1], Part 7, section C.5.2.
[8] A hash collision happens when two different input data sets map to the same hash value. In this case a changed configuration file could be mistakenly accepted as unchanged because the same hash is calculated.

3.7   Step 5: SW Component Testing

The SW components and modules are tested in isolation by substituting caller and callee with a test framework. Two distinct test strategies are employed:

- **Black box testing** - The classical testing approach where the test data is based on the type of the input parameters – i. e. the range of the input parameters is divided into valid and invalid partitions whose boundaries are then chosen as test input (e. g. highest and lowest possible value of an integer type parameter).

- **White box testing** - The test data is constructed so that the tests execute enough of the code under test to satisfy a specific coverage criterion (e. g. all statements, all branches) – the minimum required coverage depends on the SIL and is defined [1], Part 3, Table B.2.

Because the code under test must be thoroughly understood to effectively construct white box test cases, the component/module tests are carried out by the implementer rather than a separate tester. The tests itself are not discarded but kept under configuration management too, as they are a viable documentation and necessary for regression tests (in case of modifications). This phase is also the first opportunity to execute the production code under the supervision of a dynamic analysis tool (e. g. Valgrind) as a supplement to the static analysis.

3.8   Step 6: SW Testing

During this phase, the whole SR-SW is tested against its requirements.[9] The SW test focuses on the external interfaces by subjecting them to test input and by injecting faults into the interfaces themselves. Considering the CONF components external interface, this could be achieved by the following:

- Using test configuration files with valid and invalid MD5 hashes.

- Using a configuration file with a valid MD5 hash and causing errors in the POSIX open() and read() functions.

The SW tests are executed on a dedicated test hardware which must be specified in detail for regression testing. In addition, a test environment must be created which should be as close as possible to the environment used in the risk analysis – with the added possibility to inject errors in a controlled manner (e. g. corruption of the configuration file during network transfer). The tests itself should be fully automated and log as much data as possible to facilitate defect analysis.

4   Conclusion

The ISaPro® as a generic process model shows a holistic project picture from the system safety point of view. On the one hand it fulfils the system safety aspects in applying the engineering and management principles in a complete model and on the other hand it is structured in a way that safety becomes an inherent aspect of a system. Due to its generic nature it can be easily adapted to different project conditions, while still providing an increase in efficiency and effectiveness. Furthermore, the work products of the model can be considered as significant inputs to the safety case.

As demonstrated by using a simple safety function as subject, the individual processes and their activities were explored in detail. The main intention of this approach was to show the impact of safety demands on the SW engineering activities due to the integration of the safety life cycle by the derived process landscape.

---

[9] Depending on the complexity of the SW a separate integration testing phase could be necessary prior to the SW testing. Since the example focuses on a single component, integration is not considered. It is assumed, that the CONF component was successfully integrated with the remaining SR-SW before.

Literature

[1] International Electrotechnical Comission (IEC), IEC 61508 Edition 2.0 - Functional safety of electrical/electronic/programmable electronic safety-related systems, Geneva (2010)

[2] ISO International Organisation for Standardisation, ISO 26262: Road vehicles - Functional safety, International Standard, Geneva (2011)

[3] Department of Defense (DoD), MIL-STD-882E: Military Standard, Standard Practice for System Safety, 2005

[4] RTCA, Inc., RTCA/DO-178B: Software Considerations in Airborne Systems and Equipment Certification, Washington, D.C. (1992)

[5] Ericson, C. A.: Hazard Analysis Techniques for System Safety, John Wiley & Sons, Hoboken (2005)

[6] IEEE, IEEE Std 1003.1-2001 Standard for Information Technology - Portable Operating System Interface (POSIX) System Interfaces, Issue 6, New York (2001)

[7] Krebs, P., Platschek, A., Tschürtz, H.: A FLOSS Library for the Safety Domain. In: Proceedings of the 13th Real Time Linux Workshop, pp. 279-288. Open Source Automation Development Lab (OSADL) eG, Schramberg (2011)

5 Author CVs

Hans Tschürtz, M.Sc, M.Sc, Head of R&D

University of Applied Sciences FH Campus Wien, 1100 Vienna, Favoritenstrasse 226, Austria, telephone (+43) 1 606 68 77 8401,
e-mail: hans.tschuertz@fh-campuswien.ac.at

Hans Tschürtz has gained technical experience during more than 30 years working in the private industry. This included activities as a software engineer as well as project leader in multi-site projects. He is head of the Vienna Institute for Safety and Systems Engineering (VISSE) of the University of Applied Sciences FH Campus Wien. One of his major responsibilities is project and process management with the focus on safety, at present with focus on ISO 26262. He is graduated engineer in the field of technical science and holds two Masters of Science. Moreover he is certified as a senior project manager (IPMA level B).

DI(FH) Peter Krebs, Researcher

University of Applied Sciences FH Campus Wien, 1100 Vienna,
Favoritenstraße 226, Austria, telephone: (+43) 1 606 68 77 8407,
e-mail: peter.krebs@fh-campuswien.ac.at

Peter Krebs has been working as researcher at the Vienna Institute for Safety and Systems Engineering (VISSE) for 3 years after graduating in Information and Communication Technologies from the University of Applied Sciences FH Campus Wien. Since 2011 he is head of a project team His main fields of working include the implementation of complex software under safety constraints (mainly IEC 61508) and Voice-over-IP (VoIP) technology.

Experiences from the Development of a Safety Operating System for AUTOSAR

Alexander Mattausch *alexander.mattausch@elektrobit.com*
Alexander Much *alexander.much@elektrobit.com*

Abstract

The automotive standard AUTOSAR[1] provides a standardized basis for electronic control unit (ECU) software development consisting of a layered software architecture with over 80 software modules and libraries accompanied by an associated development methodology.

With the finalization of safety standards for the automotive domain more automotive ECUs[2] are classified as "safety-relevant". This is also motivated by the fact that increasingly more than one function is now allocated to one specific ECU and these functions are often developed by different parties: the basic software vendor, the Tier-1 and the OEM. An ECU is therefore more likely to be classified as safety-relevant, because only one function needs to be safety-relevant.

This paper details the experience gained from the development of an operating system that supports the AUTOSAR architecture, provides freedom from interference and is developed up to ASIL-D[3]/SIL[4]-3.

From a process and methods point of view the challenge was to extend an Automotive SPICE compliant development process to cover recent safety standards such as ISO/IEC 61508:2010 or ISO 26262:2011. Especially the verification of work products needed significant changes.

This paper also details the decisions that have been made and presents some solutions that have been introduced for the development of the operating system.

Keywords

AUTOSAR, Memory Partitioning, ISO 26262, ISO/IEC 61508, Automotive SPICE

---

[1] AUTOSAR: AUTomotive Open System ARchitecture, http://www.autosar.org.
[2] ECU: Electronic Control Unit.
[3] ASIL: Automotive Safety Integrity Level, see [1], Part 1.
[4] SIL: Safety Integrity Level, see [2], Part 1.

1 The Challenge

Automotive ECUs are becoming more and more complex. Additionally, the OEMs increasingly request that more than one specific function has to be provided by a single ECU. Allocating multiple functions to a single ECU makes it therefore more likely that the ECU becomes safety-relevant. This is the case if only one of these additional functions is safety-relevant. Besides the commonly used (A)SIL decomposition, this effect also leads to an increasing allocation of different (A)SIL levels to different parts of the ECU software.

The AUTOSAR architecture is meant to reduce complexity and increase reuse across ECU projects. In an AUTOSAR world, different functions are usually realized as different software components (SW-Cs), which access the AUTOSAR stack via the Runtime Environment (RTE), or complex device drivers (CDDs), that are "below" the RTE, see Figure 1 for a simple view of the architecture.
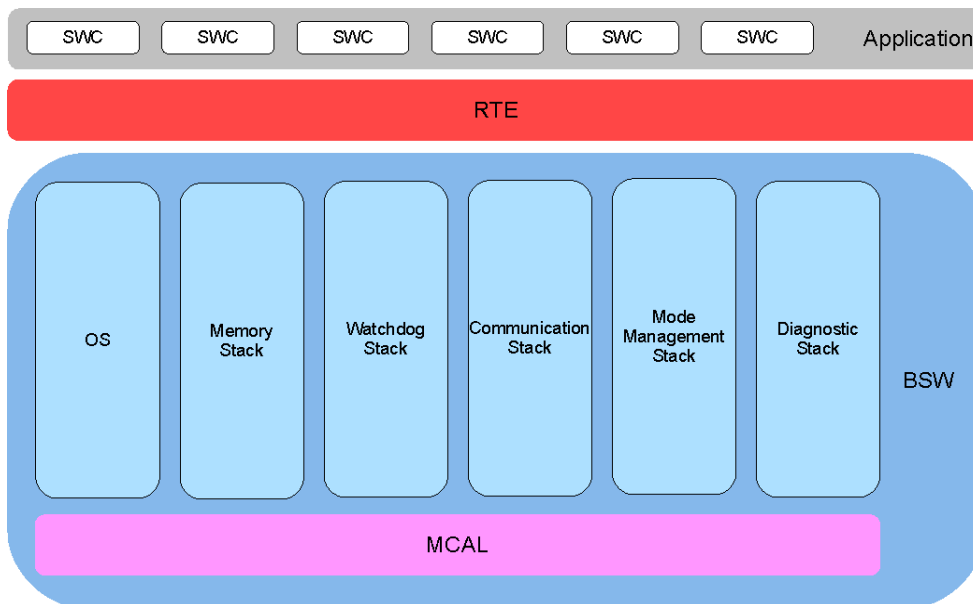
Figure 1 Basic view on the AUTOSAR architecture

Non-interference between software components is a well-established technique in safety projects. The avionics industry applies this concept successfully in a similar environment with the IMA[5] architecture, which has been formalized by certification authorities; see [7] or [8].

Non-interference usually has two domains: the spatial independence, which is often realized by hardware supported memory partitioning, and the temporal independence, which is often realized by using some form of control flow, time and execution protection mechanism which is coupled to a hardware watchdog, see e.g. [2] for clear definitions.

If non-interference is not given, mixed (A)SIL allocations for different functions lead to the fact that all software units of the ECU inherit the highest (A)SIL level, which in turn leads to a large codebase that needs to be developed according to safety standards. Non-interference becomes even more important for ECUs where functions are implemented by different suppliers. Suppliers usually develop each function for their targeted (A)SIL level, independent of the highest (A)SIL allocation of the specific ECU software.

Using an architecture that is strongly based on non-interference increases the confidence in the safety mechanisms and also dramatically simplifies safety analyses which are demanded by safety standards. Without such mechanisms the analyses are not "boxed-in" and need to take all other ECU

---

[5] IMA: Integrated Modular Avionics, see

software into account. In reality this can lead to the extension of such analyses to software units provided by different suppliers, which is usually not feasible as the software integrator doesn't have adequate knowledge or documentation for all software units. In some projects object files are integrated, which makes an analysis or verification very complicated.
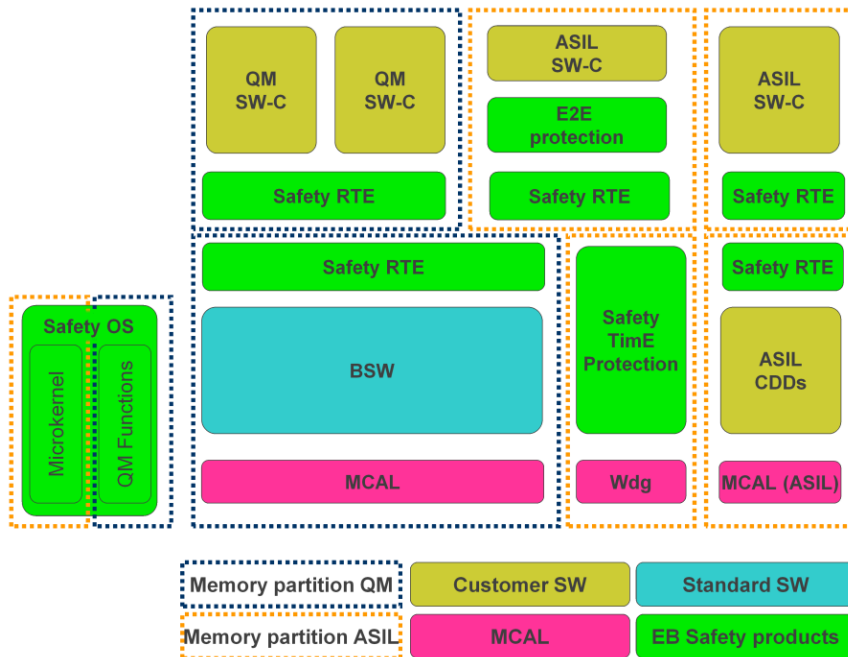
Figure 2 Safety architecture of an AUTOSAR-based ECU using memory and execution protection

The current automotive ECU hardware is also evolving and multi-core processors are now being introduced. For projects with a high (A)SIL allocation directly in software the latest trend is to use dual-core processors in lock-step mode combined with ECC-secured memory. Figure 2 shows how the basic AUTOSAR architecture evolves with added safety mechanisms and different (A)SIL allocations to software components that use the non-interference provided by the safety operating system.

With an operating system designed for these processors, it is possible to combine the implementation of safety requirements with high performance. The lock-step mode eliminates the necessity of redundant calculations and control-flow monitoring in software on a detailed level. The ECC-secured memory[6] provides protection against different kinds of sporadic data errors and minimizes the effort for redundant data storage and verification. The operating system uses the memory protection units and the different processor modes of such processors to implement the memory partitioning mechanism which provides the non-interference for the spatial domain.

The challenge is now to create an operating system that is capable of:

- Supporting current dual-core processors in lock-step mode together with a memory protection mechanism

- "Blending" seamlessly into the AUTOSAR architecture

- Being usable in a safety context of up to ASIL-D or SIL-3

- And the development starts with using an Automotive SPICE level 2 conforming process and method framework.

---

[6] ECC memory: Error-Correcting Code memory. Such memory can detect or correct "common kinds" of internal data corruption.

1.1 Automotive SPICE and Safety

The process environment that was in place when the project started was strongly focused on Automotive SPICE in the HIS scope (see [3]). There have been multiple assessments performed together with OEMs that rated the process capability at level 2.

All safety relevant projects at EB[7] need to comply with Automotive SPICE. The decision was taken that the existing continuous process improvement framework needs to be extended to safety development. This was necessary to keep safety and non-safety projects "compatible" as it often happens that specific ECU projects start being non-safety and need to be gradually converted into safety projects.

Two main lessons have been learnt:

1. The HIS scope is not sufficient to support safety development

2. Internal SPICE assessments have been extended with additional questions and checklists to check method definitions and their usage in addition to pure process related questions. Such assessments can then be used as functional safety audits according to ISO 26262.

In order to support safety development the process scope has been extended from the HIS scope to cover more processes. Figure 3 shows the processes from Automotive SPICE that support safety development and which have been added to the existing process framework.
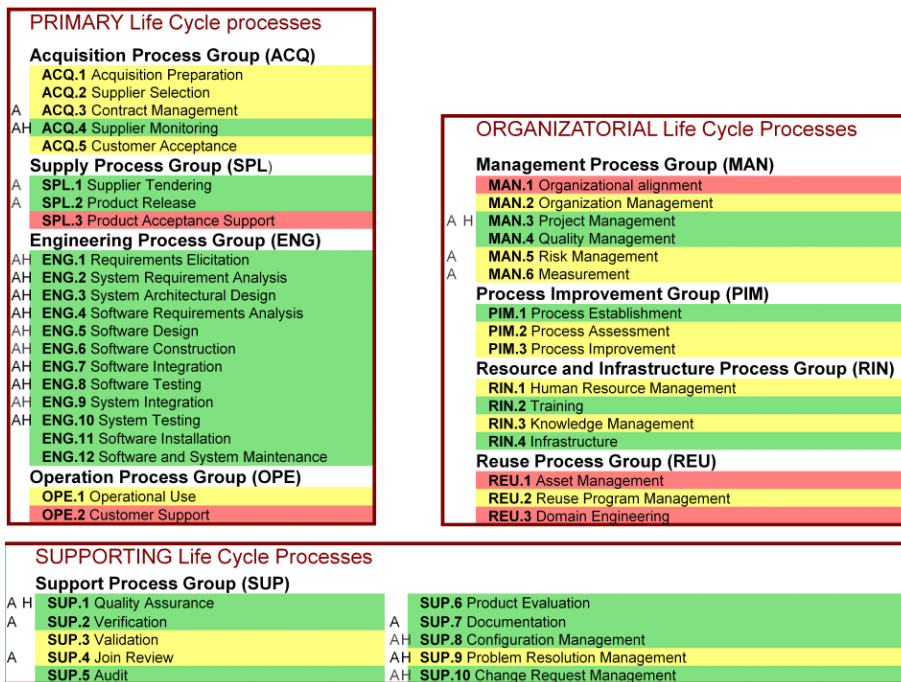
Figure 3 Automotive SPICE processes supporting safety development (green: strongly, yellow: partially, red: slightly/none).

2 Continuous Integration

Continuous integration is commonly used in software engineering to apply a continuous quality control to small changes of the software and a frequent measurement of the current state. A widely used technique is "daily builds", where the complete software release is built, measured and automatically tested every day. The development teams focus on fixing deviations first before continuing with

---

[7] EB: Elektrobit Automotive GmbH

feature development. Continuous integration is often applied in more "agile" projects that are based on small changes to the work products that are part of the development.

Safety projects seem to contradict with an agile approach as work products are expensive to change after reviews for compliance to a safety standard have been performed. The effects of freezing documents in this way can be really far-reaching and can have a similar effect as "big bang integration" for the integration of software.

The goal of continuous integration is that a software product is always close to a state where the software product can go into the release process. In safety projects the continuous integration approach needs to be extended to include the consistency and completeness of safety work products and – as far as possible in an automated way – the verification and testing activities of safety requirements.

2.1 Work Product Dependencies

The first step is a defined document management which is part of every mature approach to configuration management. It needs to be defined which work products exist, how they relate to each other and what state a work product is allowed to be in.

This basically creates a work-flow for changes:

- Document A is changed => status of document A is set to "draft"

- Document B depends on document A => status of document B is also updated

The dependencies between work products and their states can be automatically evaluated and the validity of each document can thus be derived. A change is only complete and acceptable if all related work products have been updated and are in the correct and verified state.

The approach taken is similar to the "qualifying machine" from Open-DO (see [6]).

2.2 Extending Tracing

A wide-spread technique to ensure consistency between the content of work products is to extend the dependencies between work products to tracing between the content of work products. E.g. to ensure that the correct functionality has been implemented, bi-lateral tracing from requirements through the software architecture and the software unit designs to the source code is already part of Automotive SPICE.

In safety-relevant projects, the number of documents that are required to fully specify and document the implementation as well as its verification can increase significantly: e.g. a safety-case providing the safety argument, a hardware-software interface specification and a safety manual that details the correct and safe use of the software product are the most prominent work products.

The base of the content of these documents is the software safety requirements specification and each of the following documents can contain the fulfilment of a requirement:

- Safety case: by an argument, e.g. a result of a safety analysis (see [9]).

- Hardware-software specification: e.g. if the hardware behaves as documented

- Safety manual: e.g. if the software is correctly configured and integrated and all assumptions on the external software or system architecture are satisfied
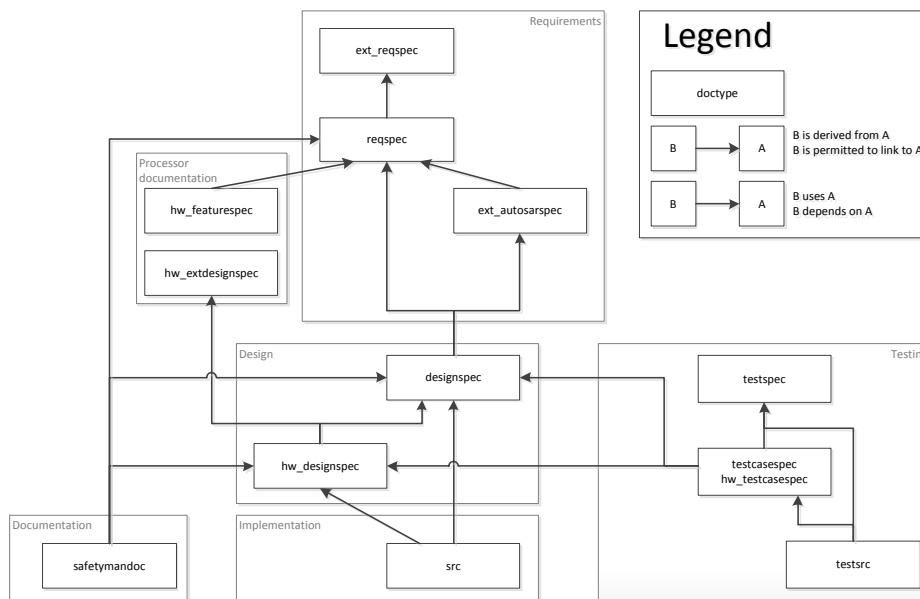
Figure 4 Tracing model used for safety operating system

As these documents can become an end-point for software safety requirements, from a tracing point of view they need to be treated just like a software unit design or the implementation itself. The tracing does not have to be bi-lateral as e.g. a safety manual also contains information that is not directly motivated by a software safety requirement.

The tracing model of Automotive SPICE needs to be extended to include such safety work products in order to construct an argument for the completeness of the fulfilment of software safety requirements. Figure 4 depicts the extension of the Automotive SPICE tracing model to include safety work products and takes external requirements like AUTOSAR requirements into account.

The extension of the tracing model to include safety work products is especially important in a SEooC[8] development where the final context the software is expected to be used in is assumed and documented.

Automated tracing reports that include safety work products should be part of the continuous integration environment and deviations should be treated similarly: they are resolved before the change can be accepted and feature implementation continues.

2.3   Methods and Tools

Maintainability of work products and their verification is an important topic in safety development and some important method and tool decisions have been made for the development of the safety operating system:

- All work products are in version control, which is based on subversion (see [4]). This system is open source and is in use at EB for more than 6 years holding millions of lines of code without any data corruption or consistency issues so far.

- In order to be able to work on iterations incrementally there are no work products or tools used for the specification, implementation or testing that are based on binary formats as these complicate building differences and in turn working incrementally. If possible, XML-based work products are used which have a defined schema and can be validated.

---

[8] SEooC: Safety Element out of Context. This is defined in [1] and allows the development of safety-relevant software independent of the ECU, taking into account the assumptions that are demanded from the final system and software architecture.

- No databases are used in the project as these are external to the repository and complicate baselining in the project. This is especially important for requirements management tooling.

- All data is stored exactly once, and work products using data are created using the project build environment. This greatly reduces inconsistencies in work products.

For the project all documentation has been produced using DocBook (see [5]). In order to include tracing information, the DocBook schema definition has been extended by meta-information that allows the description of requirements, design elements or even test cases. Such meta-information can request coverage from a derived object and provide coverage to an object from the upper layer. For example, a design element can provide coverage to a requirement and request coverage from a test case.

The input documents for the tracing tooling cover requirements, design models, source code and test cases. All such meta-information is then subject to a validation step against pre-defined schemas, which minimizes incorrect or inconsistent documentation data.

No commercial tooling is currently available to perform such flexible and verifiable tracing and EB needed to implement its own tracing solution. With extensible importers and exporters, the tool allows developing importers for nearly every other requirement syntax and exporters for the analysis and presentation of the requirements and the tracing data.

Testing is also performed using the previously existing test frameworks for product development and additional verification activities are based on established and commercially available tooling.

2.4   Change Control

The tools and methods described above become especially powerful for managing change requests. In safety-relevant projects there is a strong tendency to keep a code base fixed and not to touch it anymore as soon as a stable and verified state is reached. However, for a SEooC, which is used in various different projects, there is strong market pressure to adapt the software to certain needs, to extend its functionality or to support new standards to stay competitive.

In order to ensure the once reached high integrity level, a strict change management process has been defined. The project uses different mechanisms to prevent uncontrolled changes to any work product. One powerful method is coupling software configuration management tooling with a change management system on a detailed level: a check-in without change request in the change management system is automatically prevented. A second mechanism is a workflow that has been implemented in the change management system, which splits the lifecycle of a change request into three different phases:

- Analysis phase

- Implementation and testing phase

- Review phase

During the analysis phase, technical and safety experts analyse the impact of a change request. The available tracing data allows the identification of all elements that are affected by the change of a requirement or design element. The dependencies between the work products are known to the tracing tool and all relevant work products and tracing items are automatically marked as invalid when one of the objects in the tracing chain changes its version. The analysis is reviewed by a change control board (CCB) involving technical and safety experts, who accept or reject the change request.
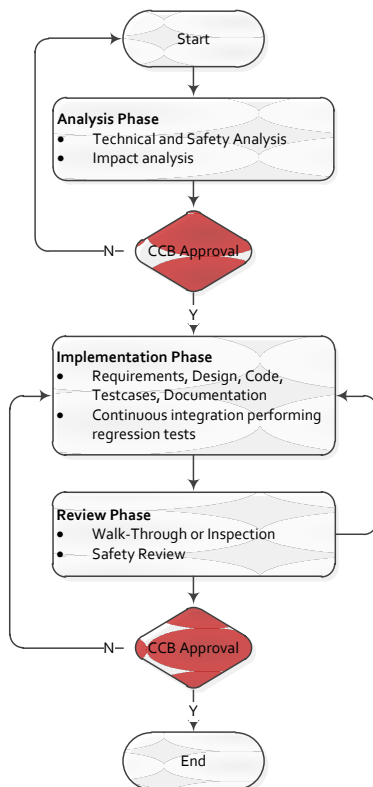
Figure 5 change management workflow with different phases

In the implementation and testing phase the change request is actually implemented. The change comprises all affected work products, i.e. requirements, design, source code, test implementation, documentation, etc. Using the ideas from continuous integration a change is completed only when all work products are in a consistent state and have been verified.

The review phase is a further verification step following the implementation and the testing. Depending on the required integrity level, the corresponding review activities are performed, which could be an inspection or a walk-through. Due to the limited nature of the change request, the review must consider two different aspects: the correct implementation of the change request as well as the integration into the existing environment. The change-request based reviews usually take less effort than a full inspection after a collection of change requests. The overall control of such changes is controlled by the CCB.

This approach combines agile methods and safety development. A change request starts from a verified state and the goal is that the project regains its verified state again as soon as possible.

3 Summary and Outlook

In this paper we presented experience from the development of a safety operating system for the AUTOSAR architecture. The operating system provides freedom from interference in the spatial domain for projects up to ASIL-D or SIL-3.

For the argument of complete fulfilment of the software safety requirements the tracing model inherent in Automotive SPICE has been extended to cover the additional safety work products, which has been achieved by the implementation of a new tracing solution.

Quality attributes like changeability and maintainability are key factors for long-term product development and these could be achieved by using a very limited set of text-based tooling that enables incremental changes and simplifies baselining of work products.

The "leap" into safety development and the invest in processes and methods has been very challenging, but as methods from safety development are expected to be standard for automotive ECUs in the future, the new processes and methods will be rolled out to standard software unit development at EB.

We strongly expect AUTOSAR to be used in a large amount of automotive ECUs in the future and the development of standardized safety mechanisms based on a standardized software architecture will lead to more reuse of software, safety cases and documentation. The reuse of "patterns" in safety architecture and parts of the safety case simplifies development and leads to a reduction of problems as well as of costs.

4 Literature

[1] ISO 26262, Road vehicles - Functional safety, ISO copyright office, 2011-11-15.

[2] ISO/IEC 61508, Functional safety of electrical/electronic/programmable electronic safety-related Systems, IEC central office, 2010

[3] Hersteller Initiative Software, http://www.automotive-his.de/

[4] Apache subversion, http://subversion.apache.org/

[5] DocBook, http://www.docbook.org

[6] Open-DO, http://www.open-do.org/

[7] ARINC. ARINC 653-1 Avionics Application Software Standard Interface. October 2003.

[8] RTCA/EUROCAE. Integrated Modular Avionics (IMA) Development Guidance and Certification Considerations DO-297, August 2005.

[9] T. P. Kelly, Arguing Safety – A Systematic Approach to Safety Case Management, DPhil Thesis, Department of Computer Science, University of York, UK, 1998

5 Author CVs

Dr. Alexander Mattausch

Alexander Mattausch studied physics at the University of Erlangen-Nuremberg and received a Dipl.-Phys. degree. He worked as research assistant at the same university and received a PhD in theoretical solid state physics in 2005. In 2007, he joined Elektrobit Automotive GmbH, where he is responsible for the operating system development for automotive ECUs. He is the author of several publications in renowned physics journals and German computer magazines.

Alexander Much

Alexander Much studied mathematics at the University of Erlangen-Nuremberg and Cambridge (UK). He received a Dipl.-Math and M.Sc degree. In 2000 he joined SuSE Linux and in 2004 Elektrobit Automotive GmbH (formerly 3SOFT GmbH), where he is responsible for the quality management of the development for automotive ECUs. He is also an Automotive SPICE assessor and actively participating in industry groups focused on software processes and safety.

Experiences with Trail Assessments Combining Automotive SPICE and Functional Safety Standards

R. Messnarz [1], F. König [2], O. Bachmann [3]

[1] *I.S.C.N. GesmbH, Austria & Ireland,* rmess@iscn.com
[2] *Friedrichshafen AG, Friedrichshafen, Germany*
[3] *SIBAC GmbH, Germany*

Abstract

In 2009 and 2010 [6], [7] papers were published at EuroSPI explaining how a task force of leading suppliers extended Automotive SPICE with additional practices to cover aspects of IEC 61508 and ISO 26262 as well. In 2011 [8] the partnership published at EuroSPI an example of how Automotive SPICE compliant engineering processes have been extended to cover functional safety architectures as well. This integrated assessment model has been used in 2011 in trial assessments at Tier 1 (leading Automotive) suppliers and in this paper we describe the lessons learned and the next steps the working group is taking in 2012.

Keywords

Automotive SPICE, Functional Safety Standards, Safety Requirements, Technical Safety Concept and Safety Architecture

Towards an Ideation Process applied to the Automotive Supplier Industry

Martin Neumann[1], Andreas Riel[2], Serhan Ili[3], Daniel Brissaud[4]

[1] KSPG AG, 41460 Neuss, Germany
martin.neumann@de.kspg.com
[2] EMIRAcle AISBL c/o Grenoble INP, G-SCOP UMR5272, 38031 Grenoble, France
andreas.riel@grenoble-inp.fr
[3] ILI CONSULTING, Karlsruhe, Germany
ili@ili-consulting.de
[4] Grenoble INP, G-SCOP UMR5272, 38031 Grenoble, France
daniel.brissaud@grenoble-inp.fr

Abstract

Under the continuously rising innovation pressure, companies from many industry sectors find themselves confronted with the need to introduce or reinforce systematic innovation management methods and processes. It is widely recognized that ideas coming from both within and outside the organisation can be considered as the very source of innovations. Nevertheless, ideation - the generation and management of ideas - happens in the so-called "fuzzy front-end" of the otherwise well-structured process landscape of numerous modern organisations. This makes it difficult to assess and control idea generation in a way that the organisation can capitalize on the creativity of internal and external stakeholders to maximum. This article investigates the need of a structured approach towards idea generation and management within a company, and tries to give essential elements of an answer on the basis of the analysis of key success factors and a practical case study carried out at a German automotive supplier.

Keywords

Fuzzy Front-End of Innovations, Idea Generation, Idea Management, Process Development, Innovation Management, Automotive Supplier Industry

Approach to evaluate process performance focused on minimizing resistance to change*

Mirna Muñoz[1], Jezreel Mejia[1], Jose A. Calvo-Manzano[2], Gonzalo Cuevas[2], Tomás San Feliu[2]
[1] Centro de Investigación en Matemáticas, Av. Universidad no 222, 98068 Zacatecas, México
{mirna.munoz, jmejia}@cimat.mx
[2]Facultad de Informática, Universidad Politécnica de Madrid Campus de Montegancedo, Boadilla del Monte, 28660 Madrid, Spain
{joseantonio.calvomanzano, gonzalo.cuevas, tomas.sanfeliu}@upm.es
* Authors are ordered by organization

Abstract

Current approaches to evaluate processes performance in software organizations are based on applying questionnaires that most of the times increase the people resistance to perform software process improvement initiatives because people perceive them as an assessment of their job. Moreover, the questionnaires are adaptations of models and standards mainly ISO 15504 and CMMI, so that, the processes gaps are identified by comparing organization practices with those practices contained in reference models. Later, these gaps will be targeted in the implementation of software process improvements, as a result, the obtained process do not reflect the organizations needs. This paper presents an approach in order to carry out an organization processes internal fast assessment without using questionnaires so that, change resistance is minimized. The proposed approach makes a traceability between the organization internal best practices and the organization business goals in order to identify process improvements opportunities based on the organization business goals needs. This is considered a key element in order to address a successful software process improvement towards the achievement of high priority business goals. Therefore, during the software process improvement it will be possible to focus on those practices contained in models and standards but depending on the organization's business goals needs.

Keywords

Process assessment; software process improvement, software organizations, business goals, best practices, resistance to change

1 Introduction

In order to be a successful organization, it is necessary to have the skills to create strategic advantages with respect to its competitors and software process improvement is an obvious and logical way to be competitive in the software industry [1][2][3]. Besides, it is well known that the quality of software products is largely dependent on the processes that are used to create it [5]. Therefore, software industry is more and more concerned about software process improvement (SPI) [4].

However, although many organizations are motivated to improve their software processes, very few know how best to do so. As a result, most improvement efforts fail, stakeholders feel frustrated, organizations are more convinced than ever that they must continue doing their work as before and the resistance to change in software process improvement increases [6].

In this context, many authors have identified the process assessment as a key element to achieve a successful software process improvement [7][8][9].Unfortunately, the personnel perceive most of the time process assessments as an assessment of their job. Therefore, answering the questionnaires that are often used in the assessments increase the resistance to change of personnel.

This situation results in two cases: 1) the resistance to the implementation of software process improvement increases and 2) the process improvement does not have the expected [10].

The goal of this paper is to present an approach that allows a fast internal assessment of organization software processes performance focused on two key elements: *internal best practices and business goals*.

The assessment approach will allow organizations to carry out fast and frequent assessments of their software processes performance using few resources and carrying out in a short time. By this way the organization will enable to have a solid baseline related to its business goals needs when developing a software process improvement initiative.

This paper is structured as follows: section two shows the background; section three introduces the assessment approach background; section four presents the assessment approach; section five shows how the experimentation of the approach was performed and the obtained results, and finally, section six presents the conclusions.

2 Background

In this section the key concepts in which the assessment approach is based are described.

2.1 Best practices

A best practice could be a management or technical practice that has consistently demonstrated to improve one or more aspects such as productivity, cost, schedule, and quality or user satisfaction [11].

Due to the importance of best practices, relevant institutions such as the Software Engineering Institute (SEI), the Project Management Institute (PMI), the Institute of Electrical and Electronics Engineers (IEEE), and the International Organization for Standardization (ISO) have been focused on the study of best practices and have developed best practices reference models and standards [12].

The most widespread models and standards developed are: Capability Maturity Model and Integration for Development (CMMI-DEV) [13]; Team Software Process (TSP) [14]; Project Management Body of Knowledge (PMBOK)[15]; ISO/IEC15504 Information technology–Process assessment [16]; ISO 9001:2000-Quality Management System [17] and ISO/IEC 12207-2008 [18].

2.2 Business goals and business indicators

When implementing a software process initiative, organizations should focus on their business goals as the main reference in order to have success. But what are business goals? How an organization can measure a business goal? Next, the concepts business goal and business indicator are briefly described.

- Business goal is a statement or condition of the organization established by senior management to be brought in order to ensure their continued existence and enhance its profitability, market and other factors that should be achieved to the organization's success [19].

  Examples of business goals that an organization could establish may include: reducing developed cycle time, reducing the number of change request during an integration phase, increasing the number of errors found after the second phase; increasing customer satisfaction, reducing the number of project with any kind of deviation, and so on [13].

- Business indicators are words that help organizations to be specific about the measures they need to get information about business goals [20].

  Examples of business indicators are licenses renewal rate, number of customers, quoted lead times, number of common processes, reductions in product development or service cost, management rules, time to accommodate design changes, ration of development time to product life [20].

3 Assessment approach background

The proposed assessment approach was developed as a part of a methodology for a gradual and continuous software process improvement focusing on minimizing change resistance called MIGME-RRC (by its Spanish acronym) [6].

To develop MIGME-RRC methodology, topics such as knowledge management, change management and multi-model environment were used. Besides, the definition of the methodology was supported and supervised by three experts in process improvements of the Universidad Politécnica de Madrid and one senior management of everis consulting.

The assessment approach was developed taking into account topics such as knowledge management, change management and multi-model environment too. The assessment approach purpose to determine the performance of software processes focusing on both the actual best practices carried out in the organization and the business goals defined in the organization.

By this way, the proposed approach makes a fast assessment in as follows: 1) a bottom-up analysis of coverage among business goal, business indicators and internal best practices is performed; 2) an analysis of business achievement is performed; 3) an analysis to prioritize business indicators is performed and 4) the results are communicated to the personnel.

As a result, the need of implementing improvements and where to address them is known by personnel. Therefore, the resistance to change that grows by applying questionnaires has been reduced.

4 Assessment approach

The purpose of the assessment approach is assessing and measuring an organization software processes performance in order to establish the efficiency ranges of "*how well is software process performance with the internal best practices carried out?*"

The assessment approach proposes a different way to assess an organization processes performance as follows: instead of making a traditional assessment with the application of questionnaires that are adapted from international models and standards, it proposes to make a coverage analysis between business goal, business indicators and organization best internal practices

By this way, it enables an organization to address their improvement effort towards the achievement of their business goals, which have a high priority and a low coverage with the internal best practices carried out in the organization.

Then, it is possible to select those external best practices proposed by international models and standards according to their business goals needs. Figure 1 shows the three main analyses contained in the assessment approach: *coverage, achievement and prioritize* and their seven activities.
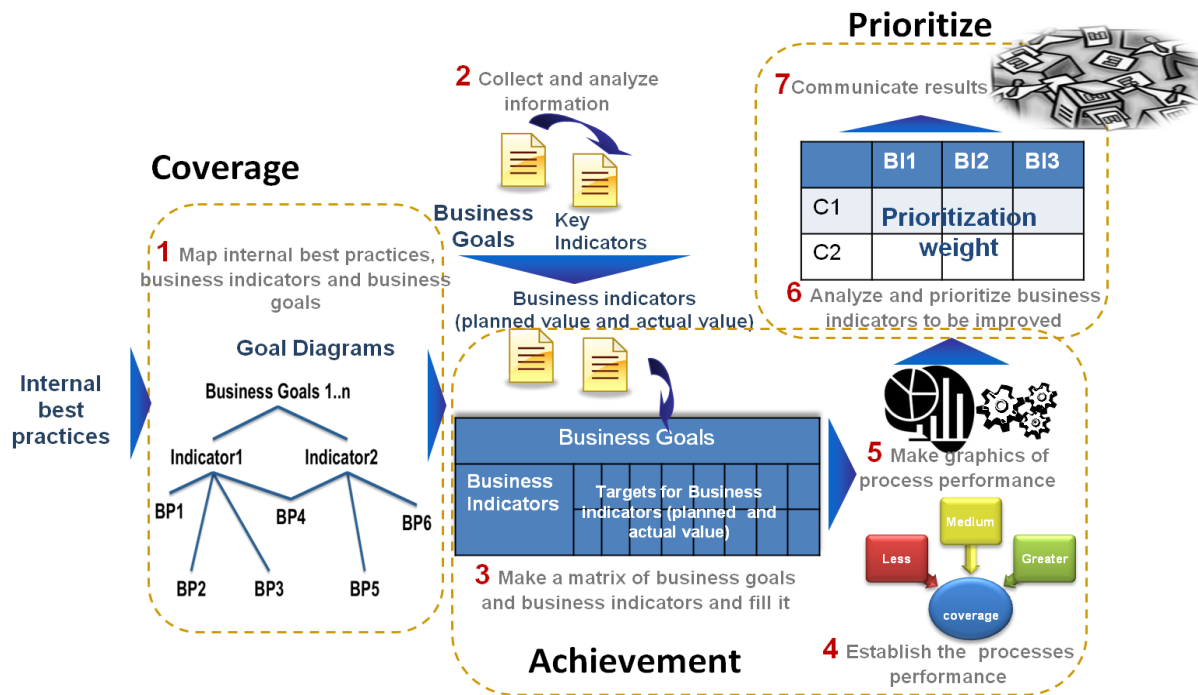
Figure1. Assessment approach

As Figure 1 shows, the assessment approach includes 3 main analyses: *coverage analysis, achievement analysis and prioritize analysis*. These analysis are composed of 7 activities as follows: 1) relations among internal best practices, business indicators and business goals are identified in order to establish the coverage among them; 2) documentation that contains the targeted planned and current values of the business indicators are collected and analysed; 3) matrix of business indicators and business goals is made and filled with the value resulting of calculating the difference between planned and current values targeted for business indicator; 4) values of the matrix are classified and a specific colour is assigned to them in order to establish the process performance; 5) graphics of performance are made, 6) indicators are analysed in order to establish priorities, so that, it is possible to focus on those processes that need to be improved in order to achieve the business goals; and 7) results are communicated to personnel, so that the need of implementing process improvements and where to address the improvement effort is highlighted.

It is important to mention that before performing the assessment approach, the organization internal best practices must be identified. The identification of internal best practices is out of the scope of this paper. More information about how to identify best practices is showed in [21].

Nevertheless, Figure 2 schematizes both approaches: identify internal best practices approach and the assessment approach. Besides, the implication of stakeholders with them is included.

As Figure 2 shows during the performance of the identifying best practices approach, the middle management and the process users have an important involvement because they are the source of the organization tacit knowledge.

Besides, they have an important involvement in the best practices validations. Best practices validations are considered key activities in order to formalize the organization knowledge. As a result, the organizational knowledge is formalized through the organizational best practices. Those practices are used as an input of the assessment approach.

Then, performing the assessment approach, senior management has an important involvement because they: 1) are who establish the business goals and set target values to them; 2) have access to the internal audits data and 3) are able to decide about the criteria and prioritization of business indicators.

Once the coverage and achievement analysis was done and the business indicators have been prioritized, an adequate material must be done to communicate both middle management and process users about the obtained results in order to increase the need of implementing a software process

improvement in order to achieve its business goals. Besides, the results indicate where to address their improvement effort.
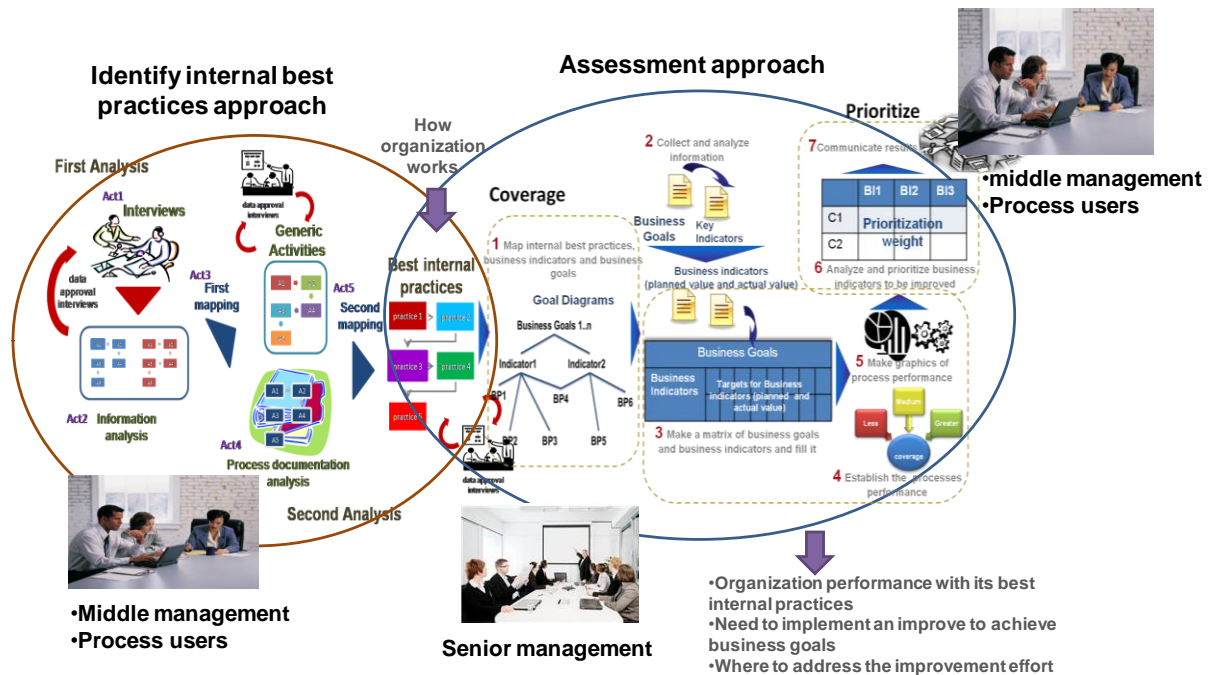
Figure 2. Involvement of stakeholders by identifying internal best practices and assessment approaches

4.1 Coverage analysis

This analysis is focused on identifying the coverage among organization best practices, business indicators and business goals. Performing this analysis enables organization to identify what business goals are really covered by the identified organization best practices. Then, there should be collected any documentation that contains target values established to business indicators. Next, the activities contained in this analysis are described.

4.1.1 Map internal best practices, business indicators and business goals.

The purpose of this activity is to make a bottom-up mapping in order to identify how business goals are covered by the indicators and how the indicators are covered by the internal best practices. As result of performing the analysis two situations may arise: 1) indicators can have one or more related best practices and 2) business goals often have one related indicator. To perform this activity goal diagrams are used.

After performing this activity, business goals with high and a low level of coverage are identified.

4.1.2 Collect and analyze information

The purpose of this activity is to collect information of targets and actual values for business indicators.

- *Planed values*: senior management has established business goals and their business indicator with a planned target. As mentioned above, each business goal has its associated business indicators, which help organizations to be specific about the measures they need in order to get information about business goals.

- *Actual values*: most of the time the source of this information is the internal audits carried out in the organization.

4.2   Achievement analysis

Once the coverage of the indicators has been identified, the second analysis is focused on establishing the percentage or level of achievement that business goals have with the internal best practices.

The achievement analysis uses the information collected in the activity 4.1.2. Besides, to achieve this analysis, a matrix is used. Next, the activities contained in this analysis are described.

4.2.1 Make a matrix of business goals and business indicators

The purpose of this activity is to make an achievement matrix. The achievement matrix records the business goals in the columns and the indicators in the rows. Then, the matrix is filled with the target and actual values of business indicators, so that, each business goal will have two indicators values.

4.2.2 Establish the process performance

The purpose of this activity is to establish the performance of business goals. To achieve it, the achievement of the indicators is analysed by comparing established and actual values of targets that have been defined for indicators as follows: *Achievement = actual value – planed value*.

Then, the matrix is analysed focusing on the resulting value in order to classify indicators. To make an adequate classification of indicators, it is proposed to use a scale of colours as follows:

- Achievement value which has obtained *high percentage* is highlighted with *red colour*. This colour means that the business goal has a *low achievement*.

- Achievement value which has obtained a *medium percentage* should be highlighted with *yellow colour*. This means that the business goal has a *medium achievement*.

- Achievement value which has obtained a *low percentage* are highlighted with *green colour*. This means that the business goal has a *high achievement*.

4.2.3 Make graphics of process performance

The purpose of this activity is to make graphics that allow stakeholders to have a better comprehension of the processes performance. Besides, the graphics highlight the need of implementing process improvement in order to achieve the organization business goals.

4.3   Prioritize analysis

The third analysis is focused on prioritizing the business goals throughout the assignment of weights to the indicators related to them, according to their impact in the achievement of business goals.

The assignment of weight to business indicators should be as follows*: indicators which have a high impact in achieving the business goal according to established criteria must have a high weight*.

It is important to highlight that prioritize business indicators is critical because it enables the organization to know where to address the improvement effort in the implementation of an improvement initiative. Next, the activities included in this analysis are described.

4.3.1 Analyze and prioritize business indicators to be improved

The purpose of this activity is to analyze the indicators and to choose those indicators in which an improvement opportunity is detected. Then, a weight is assigned for each indicator according to a set of criteria.

The criteria can be from criteria established by the organization or criteria such as: *cost savings, quality problems, time problems, answer problems, personal frustration, customers complains, viability*. The criteria should be chosen according to the organization improvement needs. The criteria weight should be set in a scale of 5 (maximum weight) to 0 (minimum weight).

4.3.2 Communicate the results

Communicate the results of both the process performance and the prioritized indicators to be improved to the stakeholders. This activity is very important due to stakeholders could see in a quantitative and graphic way the need of implementing a process improvement. Besides, they could see what processes need to be improved, so that, it is possible to address the improvement effort, and most important, they do not feel these results as a threat to their job.

4.4   Support activities

As mentioned above the approach has been defined by using knowledge management and change management principles. Next, Table 1 shows the activities focused on knowledge management and change management carried out.

Table 1. Knowledge management and change management activities

Knowledge management activities	Change management activities
The knowledge management is carried out by analyzing process performance results, understanding and selecting those data that will be stored as historical data. These data will form the organization process assets to be used in future analysis.	All activities of the approach are focused on highlighting the need of implementing an improvement in order to achieve the established business goals. Moreover, the stakeholders have the opportunity to analyze the assessment results and to understand the improvement needs. The activities focused on change management carried out are: • Observe senior management behavior. Then, describe and classify it in order to identify risk related to resistance to change or improvement commitments. • Make presentations of the results: make material presentations of the results helps to highlight the need of improving the processes in order to achieve the established business goals. Furthermore, a better commitment of stakeholders is obtained.

5 Assessment approach experimentation

This section presents the experimentation of the assessment approach carried out at everis in order to experiment the assessment approach.

everis is a multinational consulting firm with factories in Europe and Latin America. It offers services which provide solutions to large companies in any sector and it is based on three pillars: innovation, methodologies and efficiency. Since its creation in 1996, it has grown both in revenue and staff in a steady and organic way. Turnover in 2009 was over 404M€ and the company employs more than 7,000 people. They have over 1,000 projects opened every month.

Before the approach was tested, it was validated by the delivery management group of everis. This validation was done through meetings with the delivery management group. These meetings allowed feedback that was used for improving the activities described in the approach.

Once the approach was validated, the scope of the experimentation was focused in everis' project management processes because it has a broad impact on the organization business goals. Next the analysis carried out at everis is shown.

Activity1. The inputs to perform the assessment approach are the internal best practices, business goals and business indicators. Once the information is collected, the correspondences among internal best practices, business indicators and business goals were analyzed and identified in a bottom-up way by using goal diagrams. Figure 3 shows the analysis carried out.

As Figure 3 shows, the correspondence among best practices, business indicators and business goals is analyzed in order to identify the business goals coverage. To achieve it, the correspondence should be identified as follows:

a) *Correspondence best practice-indicator*: best practice related to the indicator which contributes towards the fulfillment of the business objectives.

b) *Correspondence indicator-business goal*: indicator which provides information related to the business goal and contributes to establish its fulfillment degree.
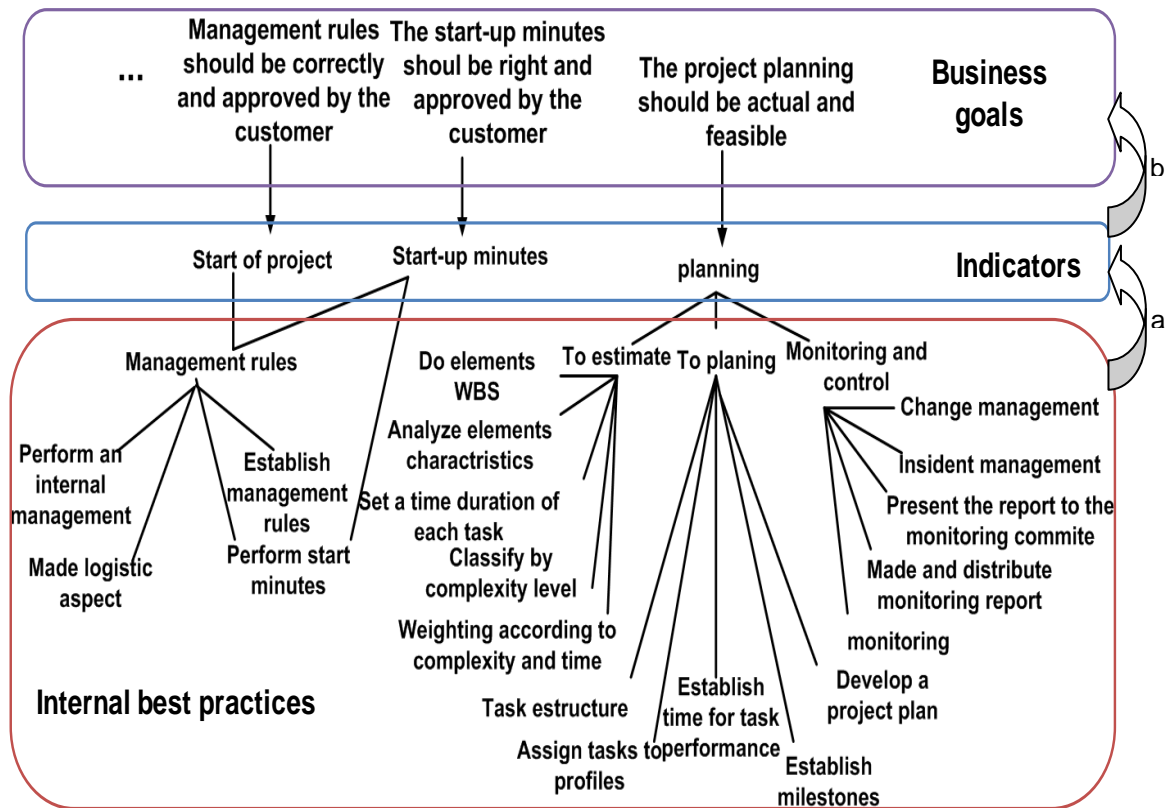
Figure 3. Correspondence among best practices, business indicators and business goals

Activity 2. Information about targets values established to indicators related to business goals was requested to quality manager. Quality manager provided information of internal audits carried out at everis every month in order to monitor their projects*.*

Therefore, established values and actual values of targets for business indicators were collected. Table 2 shows the collected values.

Table 2. Business indicators and their planed and actual target values

#	Indicator	Planned	Actual
1	The percentage of management rules which are wrong and were not approved by the customer	≤5%	15.30%
2	Project planning that were not actual and feasible	≤ 5%	9.30%
3	Indicator 3*	≤ 10%	15,30%
4	Indicator 4*	≤ 10%	17,30%
5	Indicator 5*	≤ 10%	8%
6	Start-up minutes that are not right and not approved by the customers	≤5%	15.30

* For confidentiality others indicators names are omitted

Activities 3 and 4. A coverage matrix was made and analyzed to establish the achievement of software process with the actual best practices. Figure 4 shows the matrix which shows the analysis carried out.

Indicator	Business Objectives					
	Management rules should be correctly and approved by the customer	The Project planning should be actual and feasible	Indicator 3	Indicator 4	Indicator 5	Management rules should be correctly and approved by the customer
Mangement rules	10,3%					
Project plannig		4,3%				
Indicador 3			5,3%			
Indicador 4				7,3%		
Indicador 5					-2%	
Start-up minutes						10,3%

Figure 4. Coverage matrix

Activity 5. A graphic to show the achievement of indicators was made. Figure 5 shows the process performance graphics. This graphic was selected because this kind of graphics are often used in the organization to report results.
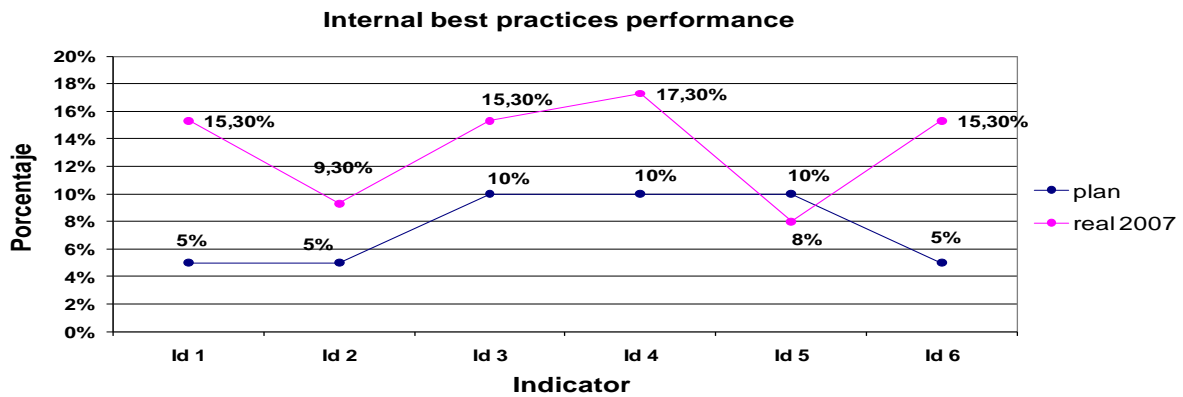
Figure 5. everis process performance with the internal best practices

Activity 6. A matrix of prioritization related to business indicators was done to assign weights according to the established criteria. The selected criteria were: *1) The management rules are a key document in everis because it defines the project framework; 2) It is very important that the documentation of project planning must be accurate and properly updated, and 3) management rules are important since it is important in order to establish project operatives' issues and customer agreements.*

Table 3. Business indicators prioritization

Indicators	Weight
Management rules	5
Project planning	5
Indicator 3	3
Indicator 4	2
Indicator 5	3
Start-up minutes	5

Table 3 shows the prioritizations of indicators.The selected indicators were: *planning tools, management rules and start-up minutes.*

Activity 7. There were established minutes to show stakeholders the results of the assessment approach. During the minutes it was perceived the interest and commitment of stakeholders related to the implementation of the process improvement.

Software improvement Results: because the assessment approach has been implemented as the second phase of MIGME-RRC methodology, next, the results obtained are showed.

The results have showed that performing the assess approach has allowed to address the improvement toward those processes that needed to be improved in order to fulfilment their business goals.

Besides, it has allowed to carry out a successful implementation of software process improvement in project management processes at everis [6]. Figure 6 shows the process graphics of business indicators fulfilment after the improvement.

As Figure 6 shows the three graphics has shown an improvement in the use of those practices contained in new processes which has allowed to achieve three key indicators related to business goals established by everis senior management. All control charts show a decrease in the average and in the upper and lower limits. Besides all of them indicate a better control because the values are closer to the average.
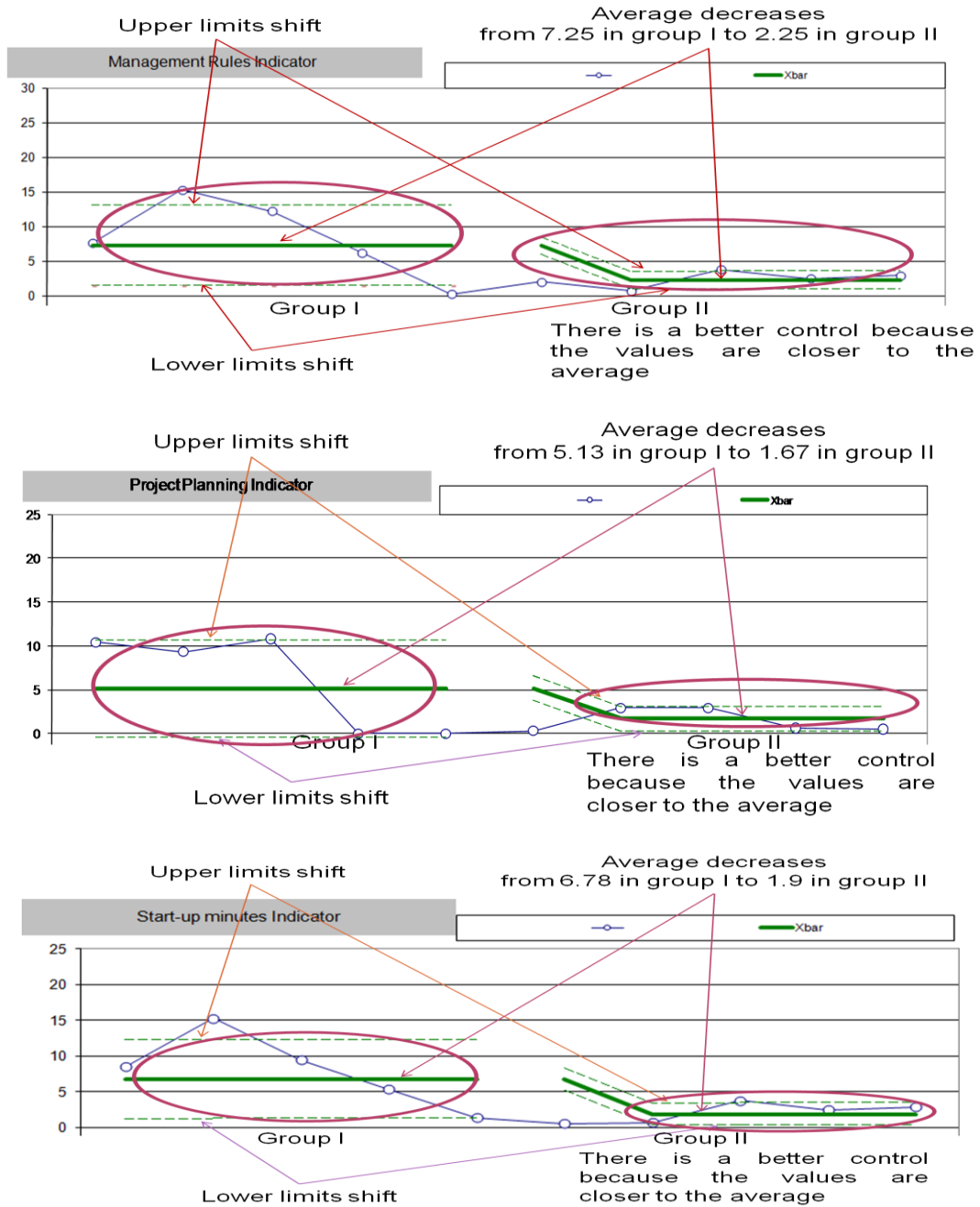
Figure 6. Indicators fulfillment result

6 Conclusions

The proposed assessment approach has allowed assessing the software process performance and addressing the improvement efforts towards those processes that should be improved according to the organization business goals needs. Moreover, performing the assessment approach has allowed to increase the improvement need and to get a high level of commitment of senior management, middle management and processes user.

Three lessons learned of how to minimizing resistance by applying the assessment approach are: (1) using internal audits carried out by the organization as source have allowed to establish a real organization process performance; (2) mapping internal best practices with business indicators have allowed the identification of process strengths and process improvements opportunities without being perceived by stakeholders as a threatened of their work, (3) showing a state of current and desire process performance report to stakeholders has allowed to get a better stakeholders commitment.

It is important to highlight that due the nature of the assessment approach it is easily implemented in large or SME organizations which want to know how to address its improvement effort.

7 Literature

1. Gupta, J., Sharma, S., Hsu, J. An overview of knowledge management. Idea Group Inc.Ch. 1. (2004)

2. Molina, J.L., Marsal, M.: La gestión del conocimiento en las organizaciones, pp. 60-68, 87-94. (2002)

3. Turban, E., Aronson, J.E., Liang, T-P. Knowledge Management. Decision Support Systems and Intelligent Systems, Pearson (Prentice Hall), ch. 9, pp. 487. (2005)

4. Williams, T. How Do Organizations Learn Lessons from Projects—And Do They? IEEE Transactions on Engineering Management, vol. 55, pp. 248-266. (2008)

5. Mishra, D., Mishra, A. Software Process Improvement in SMEs: A Comparative View. Computer Science and Information Systems, Vol. 6, No. 1, 111-140. (2009)

6. Calvo-Manzano, J.A., Cuevas, G., Gómez, G., Mejia, J., Muñoz, M., San Feliu, T. Methodology for process improvement through basic components and focusing on the resistance to change. Journal of Software Maintenance and Evolution: Research and Practice. (2010)

7. Pino, F.P., García, F., Piattini, M., Herramienta de Soporte a la Valoración Rápida de Procesos Software, IEEE Latin America Transactions, Vol. 5, No. 4, July 2007, pp 218-223. (2007)

8. Garcia, I., Pacheco, C., Cruz, D. Adopting an RIA-based tool for supporting assessment, implementation and learning in software in software process improvement under the NMX-I-059/02-NYCE-2005 standard in small software enterprises. Eighth ACIS International Conference on Software Engineering Research, Management and Applications. (2010)

9. Pino, F.J., Pardo, C., García, F., Piattini, M.. Assessment methodology for software process improvement in small organizations" Inf.Softw.Technol. vol. 52, pp.1044-1061. (2010)

10. CMMI Working group: The Economics of CMMI®, NDIA Systems Engineering Division,Version 1.0. (2009)

11. Withers, D.H.: Software engineering best practices applied to the modeling process. In: Simulation Conference Proceedings, 2000. Winter, pp. 432-439 vol.1. (2000)

12. Brotbeck, G., Miller T., Statz, J.: A survey of Current Best Practices and Utilization of Standards In the Public and Private Sectors. Department of Information Resources. (2006)

13. Chrissis, M.B., Konrad, M., Shrum, S.: CMMI Second Edition Guidelines for Process Integration and Product Improvement, United States, Massachusetts, (eds): Addison Wesley. (2007)

14. Humphrey, W.: Introduction to the Team Software Process, Massachusetts, (eds.) Addison-Wesley, pp. 459. (2006)

15. IEEE Computer Society: IEEE Guide Adoption of PMI Standard A Guide to the Project Management Body of Knowledge; IEEE Std 1490-2003 (Revision of IEEE Std 1490-1998), pp. 0_1-216. (2004)

16. International Organization for Standardization: ISO/IEC 15504: 2004 Information technology – Process assessment. (2004)

17. AENOR: Sistemas de Gestión de Calidad (ISO 9001:2000), UNE-EN ISO 9001, Norma Española. (2000)

18. Standard Comitee IEEE 2008: Systems and Software Engineering - Software Life Cycle Processes, IEEE STD 12207-2008," Ieeexplorer, pp. cl-138. (2008)

19. Object Management Group, Business Motivation Model, v1.1, available at: http://www.omg.org/spec/BMM/1.1/, 2010

20. Goethert, W., Hayes, W., Experiences in Implementing Measurement Programs, Software Engineering Institute, Technical Note CMU/SEI-2001-TN-026. (2001)

21. Calvo-Manzano Jose A., Cuevas Gonzalo, Mejia Jezreel, Muñoz Mirna, San Feliu Tomás, Sánchez Angel, Rocha Alvaro, "Approach to Identify Internal Best Practices in a Software Organization" System, Software and Services Process Improvement, 17th European Conference proceedings, ISBN 978-3-642-15665-6 EuroSPI 2010, Grenoble, France. (2010)

8 Author CVs

Mirna Muñoz

She holds a PhD in Computer Science, a Master in Computer Science (2002-2004) and Computer Science Engineer (1996-2000) from the Institute Technological of Orizaba University of México. Her research field is Software Engineering. Her current research interest consists of how to implement software process improvements focused on minimizing change resistance and the use of multi-model environments in software process improvements. She has collaborated with everis consultants in research projects. She is a member of MPSEI research group.

Jezreel Mejia

He holds a PhD in Computer Science, a Master in Computer Science (2003-2005) and Computer Science Engineer (1996-2000) from the Institute Technological of Orizaba University of México. His current research interest consists on outsourcing process improvement. He has collaborated with an everis Foundation in research projects. She has collaborated with everis consultants in research projects. She is a member of MPSEI research group.

Jose A. Calvo-Manzano

He holds a PhD in Computer Science. He is assistant professor in the Facultad de Informática at Universidad Politécnica de Madrid. He is teaching in the area of Software Engineering, specifically in the domain of software process management and improvement. He has participated in more than 20 research projects (European and Spanish Public Administration). He is author of more than 50 international papers. He is author of books related to software process improvement and software engineering topics also. He is a member of SOMEPRO research group, where his main research field is process improvement, focusing in the topics of requirements management, project management, process deployment, and solicitation and supplier agreement development. He is the head of the "Research Chair in Software Process Improvement for Spain and Latin American Region", where he has been a member of the team that has translated CMMI-DEV v1.2 and v1.3 to Spanish.

Gonzalo Cuevas

He received an Engineering degree in Telecommunications in 1965 and a PhD in Telecommunications in 1974. He also received an MS in Computer Science from the Polytechnic University of Madrid in 1972. He has been vice dean of the Computer Science faculty at the Polytechnic University of Madrid, where he is a full professor since 1970. He worked for Iberia Airlines as Software Development Director, supervising over 200 technicians, Data Processing Centre Director, Data Transmission Software Development Director, and the person in charge of strategic planning and software process improvement. He led European projects on software best practices from 1991 to 1995. His main research field is software engineering, including both technology (methods, techniques, and formalisms) and management. His current research interest is process models and methods, and transition packages. He is member of ACM, senior member of IEEE, member of the Telecommunication Engineering Association and member of the Computer Sciences Association.

Tomás San Feliu

He is PhD in Computer Science. He has been working 15 years to Software Engineering field as programmer, associate and assistant professor and consultant. He has been involved in CMM appraisals in Spain and Latin America. As Software Engineering assistant professor at the Polytechnic University of Madrid, his main research field is software process and software process improvement. His current research interest includes project management and risk management. He has published several books on process improvement and he has published technical papers in software engineering and software process improvement.

Social Responsibility Management: a Preparatory Study in Higher Education with Suggestions for Process Reference Models

Kerstin Siakas[1], Miguel-Angel Sicilia[2], Miklos Biro[3], Konstantinos Triantafillou[1]

[1] Alexander Technological Educational Institution of Thessaloniki,
Department of Informatics, P.O. Box, 141, GR-57400 Thessaloniki, Greece
{siaka, kostrian}@it.teithe.gr

[2] University of Alcalá, Computer Science Department,
Pza. San Diego, s/n - 28801
Alcalá de Henares,Madrid,Spain
{msicilia}@uah.es

[3] Software Competence Center Hagenberg (SCCH)
Process and Quality Engineering
Softwarepark 21, A-4232 Hagenberg, Austria
{miklos.biro}@scch.at

Abstract

Social responsibility (SR) concepts and practice have evolved into a heterogeneous account of the field, in which cultural and disciplinary aspects can be hypothesised to have an impact. The Information Systems (IS) community has dedicated significant recent attention to concrete SR issues, such as "Green IT". However, curricular models do not explicitly address SR and there is a lack of integration of a comprehensive view of SR in IS education, which raises the question on the way IS students are being exposed to SR issues. As a preparatory study for the suggested consideration of Social Responsibility on its own right in Higher Education, the aim of this paper is to investigate differences in perceptions of ethical issues and SR practices among Information Systems (IS) students in Greece, Hungary and Spain. In order to obtain a deeper understanding of the phenomenon a survey instrument incorporating a detailed structured questionnaire was designed. Data was collected from three European countries in order to capture opinions of students regarding the seven areas of the recently published ISO-26000 standard. Based on the results of the survey, totalling 205 questionnaires, differences in opinion were statistically tested and analysed. The results of the statistical analysis reveal that there is a strong positive correlation between perceptions of students regarding SR depending on country of origin, gender and age. In addition to spread in higher education, we suggest Social Responsibility Management (SRM) can be considered as a new crosscutting area in process reference models as for example ISO/IEC 15288 and ISO/IEC 12207, and in maturity models as CMMI. Examples of such integration were explored in the context of existing curricula. The results point out to the need for further research including multiple institutions in more countries and comparing opinions by management, administrative staff, educators and students.

1   Introduction

The field of Corporate Social Responsibility (CSR) or Social Responsibility (SR) has grown significantly during the last decade, both in importance and in diffusion. Proofs of this are the publication of the ISO-26000 guide and the SR Training and Certification Schema (SOCIRES) project1. The ISO 26000:2010 standard was presented in November 2010 aiming to provide guidance to both business and public sector organisations regarding SR. The new standard contains consensual definitions on what SR means and the SR issues that organisations need to address. It is expected to add value to existing initiatives regarding SR by providing harmonized and globally relevant guidance based on international consensus among expert representatives of main stakeholder groups. The ultimate aims of the new ISO are to increase the awareness of the SR of organisations and to help them to apply best practice. The SOCIRES two-year project started in January 2011 within the frame of the European Certification and Qualification Association2 and with funding from the EU Lifelong Learning Programme. The project aims at developing a new skill set and a job role qualification study program, where competencies in SR in line with the ISO 26000:2010 standard are customised for the European industry into an online study program complemented with an on-line examination and certification training and certification schema for SR managers. This anticipates creating the grounds for an understanding of ethics and SR by management, as well as by educational and political decision makers, thus assuring the growth and sustainability of the European industry. Recently also numerous SR consultants and special product suppliers have emerged to assist organisations to become more socially and ecologically responsible (Fenwick, 2011).

The potential benefits an organisation can gain by applying SR practices are amongst others, increased corporate reputation and minimized conflicts with primary stakeholder groups. The performance of an organisation in relation to the society it operates within and the impact it has on the environment has lately become an important indicator of its overall performance and its sustainability (Koinig et al., 2011). From a stakeholders' point of view it is more likely to profit from understanding the social and ethical responsibility of an organisation. As a result the cooperation in both directions between organisation and all stakeholders can be improved, with subsequent impact on sustainability, growth and success.

In spite of its degree of widespread adoption, SR concepts, practices and tools conform nowadays a quite a heterogeneous collection. This makes it difficult approaching SR education or training from a general perspective. However, there is an increasing concern for the development of a comprehensive framework for SR that could be used as a point of departure for curriculum design. The discipline of Information Systems (IS) has adopted some of the aspects of SR as important research and practice topics. A prominent example is that of "Green IT" which has become the subject of special interest groups and workshops. IS education has also started to incorporate SR aspects as legitimate, explicit topics. The final draft of the Curriculum Guidelines for Undergraduate Degree Programs in Information Systems (IS2010) includes "Green computing" as one of the topics of the core course "Enterprise Architecture", thus reflecting an SR concern linked to IT infrastructure. However, IS curricula and study programs are still not incorporating SR aspects in a systematic view. This raises the concern about the different viewpoints students form from their IS degrees as a first step towards a better integration of SR topics. As it is known that cultural issues affect SR perception, the study has been conducted in three different countries. The main aims of this study are to investigate the perception of IS students in different countries and to analyse the cultural influences on the perceptions, such as country of study, level of studies (BSc, MSc and PhD), gender, age and parallel working with studies. While the study is limited in geographic coverage, it brings evidence of differences that can be used to derive new hypotheses in future studies.

As IS degrees include process models as an integral part of their curricula, including SR issues in process models appears as a promising option. We suggest SRM to be considered as a new process area in models like CMMI or a new process or process aspect in reference models as for example ISO/IEC 15288 Systems Engineering - System life cycle processes and ISO/IEC 12207 Systems and software engineering – Software life cycle processes.

---

1 http://www.adam-europe.eu/adam/project/view.htm?prj=6730 and http://www.socires.eu/
2 www.ecqa.org

The rest of this paper is structured as follows. In Section 2, the relation of SR and the mission of Higher Education are described. Then, Section 3 reflects on the need of SR integration in curricula. The aims of the present study along with its methodology are presented in Section 4. Results, discussion and implications are provided in Section 5. Finally, conclusions are presented in Section 6.

2   SR of HEIs

The public authorities have exclusive responsibility for the framework of Higher Education (HE), including the institutional framework, the degree structure, the framework for quality assurance, and the provision of authoritative information regarding the HE framework (Nyborg, 2003). Equal access to HE for all qualified candidates is an important element of HE policies in Europe.

The Bologna Process is promoting academic autonomy and freedom, as the new forms of SR (Vasilescu, et al., 2009). "Mass higher education" is a fact and lifelong learning is required in the new society. European universities are committed to raise the students' awareness to the needs of society, as fully involved and dedicated social personalities. This accountability to the whole society involves personal improvement to the benefit of the society and to its main concerns: climate change, global inequities, recycling and environmental protection. The HEIs have both societal dimensions and individual components, such as preparing individuals for the labour market, preparing young people for life as active citizens in a democratic society, contributing to personal growth and development, developing and maintaining an advanced knowledge and skills base. The public authorities can use SR practices to ensure that qualified candidates are treated equally. Public responsibility also includes strategies regarding the improvement of educational opportunities for underprivileged groups. HEIs must ensure that the rights of the students are protected by appropriate policies governing conduct and student life (Lowerly, 1998). This involves the creation of communities of justice and principle that seek higher levels of human potential. Explicit discussion of SR is important, but so is the way in which rules and regulations are formulated and enforced. In order to inspire and motivate students for social responsible behaviours the HEIs must first examine their own institutional cultures and values and subsequently develop programs and policies that honour and reflect them. After considering the values at the core of each HEI they need to be communicated openly not only to the students, but to the entire community. Only then, training programs and curricula will have an impact on students.

3   Should SR be Included in the Curriculum of HEIs?

Our concern is to what degree educational institutions make an impact on the values of their students. Do we need to teach SR explicitly to our students or is it more important to communicate SR to our students by the rules, norms and actions our educational institutions practice and perform.

Matten and Moon (2006) state that the new imperatives for SR raise the challenge for organisations to acquire and develop appropriate competencies and skills. They argue that this raises the question of the role played by universities and business schools, in terms of (a) provision of graduates with SR skills, (b) supply of SR education for practitioners, (c) specialist SR education for industries, (d) research to advance knowledge in SR. We extend this argumentation by questioning if this is only a business school issue or should also other disciplines including engineering disciplines be key players in raising SR awareness of their students.

Reinhard et al. (2010) explored the nature of SR and its relevance to industry and education by studying selected organisations to determine their current practice related to SR. Their findings reveal that due to the fact that more and more organisations both internationally and nationally are showing concern for adherence to the ethical dimension of business, it is important to include SR into the curriculum of HEIs, either as a separate subject or within other subjects. In the United States for example a national consortium of colleges and universities committed to preparing students to become engaged and responsible citizens has been providing grants since 1999 through the Civic Engagement Course (CEC) Programme to faculty members to create or enhance their courses in

ways that enhance civic education (Liss and Liazosa, 2010). The results have shown that projects that serve a real purpose in the community deepen students' enthusiasm and learning.

There is broad agreement that engineering education, like other areas of education, should prepare its graduates for SR (Zandvoort, 2008). In some countries, such as the Netherlands, this task has been formally regulated by educational law by stating in the HE and Research Act that universities and establishments for higher professional education should "pay attention to the students' personal development and to the development of their sense of social responsibility" (Article 1.3 point 3). Other countries have also formulated requirements for preparing students for SR, even though the term itself is not always mentioned. Teaching character is not an easy task and should be done in myriad of ways. Some researchers even have doubts if curriculums of HEIs provide students with learning opportunities for their moral and civic development. Annette (2010; 2005) for example, considers that service learning and community based learning linked to character education provides experiential learning opportunities for students in HE to develop civic virtue and SR through "public engagement". In the United Kingdom (UK) the term public engagement has become to mean a number of new areas of academic activity within HEIs. One dimension is regional engagement including "Knowledge Exchange Partnerships". Another dimension is to provide opportunities for students to develop their values of SR through volunteering. This requires that community-based learning and service learning in partnership with local and regional communities are accredited. As a long-term outcome it enables graduates to become active citizens and play a leading role in civil society (Annette, 2010).

In a qualitative study Einfeld and Collins (2008) also noticed that service learning improves students' SR attitudes. They detected that several students participating in a university-sponsored service-learning program increased their awareness of inequality. Participants also developed several multicultural skills while interacting with their clients, such as empathy, patience, attachment, reciprocity, trust, and respect. Cetindamara and Hopkins (2008) argue that there is more than one solution in bringing SR into HE. Combining pedagogy of critique, possibility and engagement students gain a better understanding of the complexity of issues, and learn to see that they can make a difference through their direct involvement in civic projects. They exhibit a quote from a student who participated in one of the civic summer projects in South-East Turkey: "*for the first time in my life, I was 100% sure that I was doing something right, and something good, and for the first time in my life, I was proud of what I was doing*".

Thornton and Jaeger (2006) identified five dimensions of civic responsibility in HEIs, which they consider to benefit the greater society on the whole through the individual civic development. These five dimensions include:

   i) Knowledge and support of democratic values, systems, and processes;

   ii) Desire to act beneficially in community for its members;

   iii) Use of knowledge and skills for societal benefit;

   iv) Appreciation for as well as interest in those unlike self;

   v) Personal accountability.

In the discipline of IS the development of Free and Open Source Software can be considered as an example of individual civic development (Draganidis and Siakas, 2008), as well as participation of students in competition of innovative entrepreneurial ideas. Professional training in the field of SR, when included in the curriculum of IS education or provided voluntary in order to train IS students and graduates has proved to increase their professional readiness and employability, as well as their entrepreneurial development (Siakas, 2010).

Zandvoort (2008) states that the teaching of ethics and SR to engineers and to researchers tends to focus on the behaviour of the individual and ethical norms that are considered internal to the practice of science and engineering. Much less attention is paid to the SR towards the larger society. Graduates need to be provided with skills in the social, political, legal, and organisational context of their future professional environment in order to be prepared to become capable of socially responsible conduct. Another approach is to concentrate on the attitudes that are required to strive for the realisation of values or the respect for norms. Socially responsible engineering include striving for values that are in harmony with public welfare, safety and health. An important rule for moulding IS

students' SR is to create a socially responsible educational environment. If the environment is perceived by the students as negative regarding the SR practiced by their department of study or university, then it is very likely that any kind of SR education will also fail.

Brown (1997) articulates that in order to fully understand the impact of educational institutions on students, we may have to rethink and expand our notion of the desirable outcomes. She argues that the focus of our inquiry for what educational institutions should provide to students has largely been based on the outcomes valued by an academic community committed to the idea of liberal education in a residential setting (intellectual values, tolerance and liberalization of social attitudes). She proposes that educational institutions need to become learning organisations adopting five disciplines, central to organisational learning:

     i.   Personal mastery: looking at single individuals: ourselves;

    ii.   Mental models: theories we hold in our heads about how things work;

   iii.   Shared vision: the process of sharing, calling to community and action;

   iv.   Team learning: consideration of data and feedback;

    v.   Systems thinking: understanding wholes rather than parts and complex feedback loops.

Solbrecke (2008) studied psychology students' perceptions of professional responsibility by the end of HE and after a year at work. His findings indicate that, although concepts of professional responsibility do not change profoundly in the transition from education to work, they are renegotiated in work contexts. The results trigger educators in all disciplines of HEIs to reflect upon how students may be encouraged to develop moral awareness of professional and SR robust enough to meet the complex challenges of working life. In order to prepare prospective professionals for diverse responsibilities and conflicts of interest characteristic of current professional lives, HEIs should encourage an open attitude that will enable professionals to adjust to the multiple discourses of working life and the needs of individuals while simultaneously serving societal interests.

An earlier study (Voutsa et al., 2006) within the same IS department in Greece regarding the level of the students' awareness of ethical issues was considered relatively low. However, there are no standards or rules of behaviour that can reasonably be applied at all times or in all places. Instead, the question of whether or not an action is morally admissible must be answered by considering the time and culture in which it takes place. Support for this theory comes from the observation that what is considered moral can apparently vary over time within one culture, as well as across different cultures at the same point in time (Voutsa et al. 2005).

4 Aims of the Study

Our motivation for this study is the fact that SR involves an understanding of the broader expectations of society. Our intention is to explore to what degree HEIs adopt such viewpoints, how effectively do they communicate this to their students and implications for curricula, especially considering structured action in the form of process models.

In order to understand the dynamics regarding attitudes and perceptions of SR in general, and in education in particular, we tried to investigate the phenomenon first through a thorough literature review and subsequently through a survey in three HEIs in three different countries. The aim of this paper is to investigate into what degree SR practices are applied in HEIs and to examine and discuss the benefits students and the society on a whole can gain from the intentional (including SR issues in the curricula) or unintentional (promoting SR by institutions own practices) promotion of SR practices. In order to obtain deeper understanding of the phenomenon a detailed structured questionnaire was used to collect rigorous data from three European countries (Greece, Hungary and Spain). The level of current use, benefits and challenges related to learning experienced and viewpoints regarding how SR could provide added value to learning was mapped and analysed. Based on the results of the survey potential strategies for increasing students' perceptions regarding SR in general and SR practices and actions are provided.

4.1   Research Methodology

In this study the statistical analysis of the responses to the questionnaire was used as a tool to explore the hypotheses and to correlate the variables. One important issue in surveys is the size of the sample. Fowler (1993) argues that the size of the population from which a particular sample is drawn has virtually no impact on how well that sample is likely to describe the population. He states that most sample size decisions concentrate on the minimum sample sizes that can be tolerated for the smallest subgroups of importance. For this study we consider that the sample of 205 respondents is adequate for reliable results. However, in order to generalise the results from a cultural viewpoint a more homogeneous distribution of the respondents from the different countries would bring added value to the study, but as a first indication the results can be considered to present a prototype for further improvement of the research instrument and the conduction of a broader study.

The research methodology of this study was a survey, including a quantitative analysis using the SPSS tool. The data was collected by designing a 34-item survey instrument, including both an Internet and a paper based version, based on the ISO 26000:2010 seven principal areas of SR. The research instrument was translated into the local languages (Greek, Hungarian and Spanish) from English, which was the common language for designing the questionnaire.

4.2   Research Population

The survey was conducted in the spring of 2011 aiming to capture perceptions of Information System (IS) students at Higher Educational Institutions (HEIs) in Greece, Hungary and Spain regarding SR practices in the respective institutions of the authors.

In Greece, the research was conducted to a population of undergraduate students at the department of Informatics of Alexander Technological Educational Institute of Thessaloniki. At that time there was a separate course in SR (called Deontology of the Professions). SR was also partly taught within other courses, such as the Information Society course, where SR issues like Organisational Governance, Labour Practices and Green Computing were included. The department has designed a new curriculum to be adopted in the autumn semester 2011-2012. Paradoxically, all business oriented courses; including the above mentioned courses are not included in the new curriculum.

On the contrary in Hungary several major HEIs have started to include courses in the field of SR in their curricula primarily in Bachelor level programmes, and normally as part of a given subject area, such as Business Ethics, Social Work, Business and Management, Communication, Philosophy/Ethics, Leadership. These SR-related courses are in most of the cases optional courses. In the broad field of social sciences several subjects include SR topics in their content, SR constitutes an integral part of a number of courses in Economics, Environment, even Finance, but the presence of SR in these subject areas are neither compulsory nor strategically planned, and highly dependent on the actual module leaders and tutors; therefore sporadic.

In Spain, the respondents addressed were second year students of the Bachelor of Science (BSc) in Information Systems offered by the Technical School of Informatics. All of them had taken courses on business fundamentals, IS fundamentals along with technical Information Technology (IT) courses. The course on fundamentals of IS included Green IT topics during two weeks, focusing on data centre design for energy efficiency.

The questionnaire aimed to the collection of students' perceptions regarding 21 SR variables (three questions for each of the seven SR areas). The SR Questions were graded on a 5-point Likert scale according to their personal views and opinions concerning ethics and SR practices acted upon in the department of study / university.

In total 205 responses were collected. The distribution of responses per country was as follows: Greece 114 responses, Hungary 56 responses, and Spain 35 responses. The Greek population and the Spanish population were undergraduates students, while the Hungarian population also consisted of 19,6 % postgraduate students. The distribution according to gender was 23,9 % female, 74,6 % male and 1,5 % missing data. The age intervals were 18-20 years old 19,5%, 21-23 years old 45,4%, 24-26 years old 22,9% and older than 26 years old 12,2 %. In total 35 students declare that they study

full-time and work full-time, 63 students that they study part-time and work full-time, 15 students that they study full-time and work part-time as well as 13 students that they study part-time and work part-time. An interesting issue is that all Hungarian students work in parallel with their studies. The corresponding percentage for the Spanish Students is 66 % and when it comes to the Greek students the percentage falls down to 42 %.

5   Results, discussion and implications

5.1   Analysis of the main variables

The Chi-square test was used as a test of significance appropriate for nominal values. It predicts the probability that the association values between variables is a result of random chance or sampling error by comparing actual observed distribution of responses, with the distribution we would expect if no association exists (Norusus, 1994). If the null hypothesis is true, the observed and the expected values should be similar. When the significance level is small (under 0,5) the null hypothesis can be rejected.

5.1.1 Country of Origin

The results confirm the hypothesis that the country of origin is decisive for the students' perceptions of SR issues. The Chi-squared test for all categories of SR (questions 00 A, B, C to 07 A, B, C) returned the value 0.000. The null-hypothesis, declare that all the observed and the expected values should be similar for all countries. Since the significance of the Chi-square value is 0.000 the null-hypothesis can be rejected, meaning that there are differences in perceptions of SR depending of the country of origin. In Table 1 the mean values for all 21 SR items are shown per country. The grading scale was from 1 (not at all) to 5 (very much so). From Table 1 we can see clearly what the chi-squared analysis proved, that the results are different depending of the country of origin. The mean grading values for all SR questions were low (2,5) for Greece, medium (3,1) for Spain and high (4,0) for Hungary.

Table 1: SR mean values per country

	QUESTION	HU	GR	ES
00	**Student issues.**			
	University/department of studies support SR in studies:			
a	Respect for law, equal rights, equal opportunities for all students	2,5	4,7	3,6
b	Student support (disabled students, career counselling etc.)	2,9	3,5	3,7
c	Equal & fair access to services provided by university/department	2,8	4,5	3,6
01	**Organisational Governance.**			
	Promotion of norms and practices in profession:			
a	Code of behaviour / Ethical behaviour	2,6	3,5	3,2
b	Fair opportunity underrepresented groups (women, disabled etc.)	2,8	4,5	3,7
c	Respect for the law	2,6	4,5	3,8
02	**Respect for human rights.**			
	Promotion of protection of human rights in profession:			
a	Personal data (address, phone-numbers, financial issues etc)	3,2	4,5	3,8
b	Safety issues, security and data protection	3,3	4,4	3,7
c	Transparency and fair rules (available rules and regulations etc.)	2,8	4,2	3,4
03	**Labour Practices:**			
	Preparation for SR issues in future profession:			
a	Laws and regulations (copyright, licensed software etc.)	2,7	4,5	2,9
b	Employment relationships and conditions of work	2,6	4,5	3,1
c	Social issues (digital divide, social dialogue, social protection)	2,6	4,7	3,0
04	**Environment.**			
	Preparation of SR environmental issues for future profession:			
a	Green computing (environmentally sustainable computing)	2,1	4,1	2,7
b	Global (ecological/carbon) foot-print (Eco-friendly resources)	2,0	3,9	2,4
c	Material recycling	2,2	3,5	2,7
05	**Fair operating practices.**			
	Information of fair operation practices in profession:			
a	Fair competition	2,4	3,9	2,5
b	Anti-Corruption	2,2	4,2	2,3
c	Responsible political involvement	2,2	3,8	2,3
06	**Environment.**			
	Preparation for SR environmental issues in profession:			
a	Provision of helpful, accurate, fair and transparent information	2,6	4,3	3,0
b	Protection of consumers/customers rights, interests and health	2,5	3,9	2,8
c	Ensuring access to services for all without discrimination	2,6	4,3	3,3
07	Preparation of **Community involvement and development:**			
a	Overcoming poverty and disadvantage	2,1	3,9	2,6
b	Organisation of events for community involvement (talks, visits)	2,4	4,0	3,2
c	Promotion of professional memberships (prof. associations)	2,3	3,5	2,7
	Average for all questions together:	**2,5**	**4,0**	**3,1**

In order to understand the differences, we used Hofstede' values regarding the work-related cultural dimensions for the three countries (Table 2). Hofstede distinguished four key elements, or "*dimensions",* of culture (Hofstede 1994; 2001) as described below:

- **Power Distance (PD)** describes the extent to which hierarchies and unequal distribution of power is accepted;

- **Uncertainty Avoidance (UA)** indicates the extent to which a society feels threatened by ambiguous situations and tries to avoid them by providing rules, believing in absolute truths, and refusing to tolerate deviance;

- **Masculinity versus Femininity** describes the relationship between the masculine assertiveness, competitiveness and materialism opposed to the feminine concern for quality of relationships, nurturing and social well being;

- **Individualism versus Collectivism** describes the relationship between the individual independence and the collective interdependence of a group.

All the four dimensions are a continuum between two extremes (0 and 100) and only very few national cultures, if any, are wholly at one or the other extreme.

Table 2: Hofstede's work-related values

	Power Distance PDI	Individualism IDV	Uncertainty Avoidance UAI	Masculinity MAS
Greece	60	35	112	57
Hungary	46	80	82	88
Spain	57	51	86	42

From table 2 we conclude that all three countries are relatively close on the PDI scale (60, 46 and 57). Greece is a collectivistic country (35) compared to Hungary that is an individualistic country (82) and Spain that is in the middle (51). Greece has a very high Uncertainty Avoidance Index (UAI) (112); much higher than Spain (86) and Hungary (82), that also have relatively high UAI values. On the Femininity Masculinity Scale Hungary is the most masculine country (82) followed by Greece (57) and Spain (42).

Scholars claim that masculinity has a significant negative effect on corporate social and environmental performance (Ringov and Zollo, 2007) and is negatively related to ethical policies of a firm (Scholtens and Dam 2007). Kolodinsky et al. (2010) found that students holding highly materialistic values were not comfortable with business having a socially responsible role beyond profits and wealth maximization. These results indicate higher masculine values, where materialism, assertiveness and profit are considered success factors. Kolodinsky et al. (2010) also found that ethical idealism has a positive relationship with SR attitudes, while ethical relativism a negative relationship. Spirituality among business students on the contrary did not significantly predict SR attitudes. Our findings do not support these conclusions. The phenomenon is complex and there are many aspects to this, such as the influence of SR training, the SR practices of each of the HE institutions and influences from the other work-related dimensions defined by Hofstede.

In Greece, Skouloudis et al. (2011) did a first attempt to provide a systematic overview of SR practices. They used a study in line with Midttun et al. (2006), who developed four metrics as SR indicators based on (a) scores on sustainability indices and developments in the capital market, (b) the adoption of management standards (c) the promotion of corporate non-financial SR reporting practices (d) the participation in globally acknowledged voluntary SR initiatives. Their findings show that "*only a few large companies have articulated a sound strategy to promote SR activities. Correspondingly, the adoption of externally developed SR initiatives, guidelines and standards is limited and governance mechanisms towards a more socially responsible business conduct are scarce*". They conclude that the majority of Greek firms have limited awareness of SR and minimal engagement in relevant activities and much work remains to be carried out in extending and deepening our knowledge regarding the perceptions, developments and barriers to overcome.

The most compelling approach for cross-national comparisons of SR engagement seems to be the study by Midttun et al. (2006), which revealed that all Mediterranean countries score low on SR indices. Portugal and Spain were ranked considerably higher than Italy and Greece in the industrial memberships in SR communities.

Our results compared to previous findings suggest that there is a huge potential for further research regarding cultural influences on SR awareness and practices.

5.1.2 Age

Chi-squared tests for age also showed that depending on age the responses were different. The only exception was the question 00B "*To what degree does your university/department of studies support you regarding social responsibility issues being important in your studies - Student support (support of disabled students, career counselling etc.)*" that returned a Chi-squared value 0.601. Our interpretation to the similarities in responses only for question 00B can be attributed to the fact that the Greek students, which in general showed a low estimation compared to the other two countries, seem to be aware of the fact that there is a separate Career office in the university providing services to all students regarding their future professional career. There is also a special office counselling disabled students. In particular in the department of Informatics all students are aware of the fact that an elevator was build to serve some disabled students to have access to the second floor. Also all

students with dyslexia are by law allowed to give oral exams and for students with sight or hearing problems there is a specialised teacher accompanying the students in all their classes providing special interpretation of the lecture.

The younger students showed relatively lower values. This can be attributed to the distribution of the sample. In Greece there were 38 students 18-20 years old, while in the other countries this age group was missing in the sample.

5.1.3 Gender

Chi-squared tests for gender similarly showed that depending on gender the responses were different except for the following questions that both genders seemed to agree on:

- 01a: Organisational governance: Does your department of studies explicitly promote ethics and SR norms and practices in your future profession regarding Code of behaviour / Ethical behaviour

- 02: Respect for human rights. To what degree does your department of studies promote protection of human rights in your future profession regarding:

  a. Personal data (address, phone-numbers, financial issues etc.)

  b. Safety issues, security and data protection (confidentiality of applications etc.)

- 07a: Community involvement and development: To what degree does your department of studies prepare you for community involvement and development, such as: Overcoming poverty and disadvantage (e.g. community capacity building of information society, technology development and access etc.)

The IS field is generally considered a male dominated field. In our sample the male respondents were around three times more than the female respondents. We expected, based on the literature review, the mean value for the female respondents to be higher than for the male respondents, but the sample showed 3,2 mean value for male respondents compared to 2,7 for the female respondents. Since the SR Questions measure the perceptions of students regarding SR Issues practices in the university/department of studies our interpretation is that the female students may be more critical to the SR practices applied in their respective departments/universities.. In order to generalise the results and to draw confident conclusions this issue however, needs to be further in-depth investigated.

5.2 Limitations of the study

The interpretation of the findings of this study and their generalisation should be handled with care due to the fact that the study has several limitations and potential biases. The first limitation of this study is its cross-national nature. The convenience sampling is likely to give rise to some unintended biases. Further, the relatively small sample for testing cultural differences place limitation of the findings (205 responses from three countries). However, the results show a first indication about the status in IS students perceptions regarding ethics and SR. More institutions are needed in each country and also more countries need to be investigated for statistically robust results. Also a longitudinal study is needed to determine changes in perception over time and the perceptions of educational staff, managers and administrative staff could also provide interesting results.

5.3 Implications

5.3.1 Curriculum implications

The results of this study show that IS students in general assess the SR practices in their HEIs to be insufficient (as evidenced in Table 1). This includes some implications as to the inclusion of SR issues in the IS curriculum.

The results of a similar study by Wong et al. (2010) regarding Business students' perception in United States, China, and India also show that there are implications for public policy and multinational corporations. They found that in the choice of business goals, the two main goals considered by the business students are to take care of consumers' needs and owners' interests; both goals mainly economic goals without much consideration of SR. A United Nations report proposes organisations to redesign their business goals acknowledging the need of social citizenship and sustainability when setting profitable goals (Prinsloo et al., 2006). Wong et al. (2010) propose that in promoting acceptance of such redefined goals, business schools need to be more insistent in presenting to their students that sustainability and social citizenship should be terms of risk management, such as climate change, loss of benevolence, and higher employee turnover. Furthermore they consider that ethical and SR values can best be embraced by experiential approaches.

Depending on the results of this study and the increasingly emergence of SR awareness, we postulate that in order to form ethical values and sensitivity for SR, it is important to include SR in the Engineering, as well as in the IS curriculum. In addition we consider that optional professional training in SR issues should be provided within the lifelong learning context both to students and to graduates of HEIs. If this can be carried out in practical partnerships with local and regional communities it will enable graduates to become active citizens and will create potentials for long-term added value to involved stakeholder groups.

5.3.2 Public Policy Implications

Age seems to play an important role in the ethical and SR orientation of students (Arlow, 1991). Also our study confirmed that the majority of the responses had a positive relation to age differences. As a result governments that are concerned about the ethical and civic development of their citizens should introduce such civic education during students' formative years, before they reach university education levels.

5.4   Solutions and Recommendations

The overall task of educators is to raise awareness of SR in the context of education for professional practice. It seems that HEIs have started to show signs in this direction, and some offer courses in this field as part of study programmes in certain fields. Two major tasks would be: a) to enlarge the number of subject areas in which SR could be incorporated into the curriculum, b) to include the subject of SR at strategic level when curriculum development takes place at a HEIs.

By now it has become clear that the concept of SR needs to be part of general education already at the secondary education level (see the history and example of business education, which has already crept into some high school curricula and spreading). It is important that SR as a concept is introduced with a practice-oriented approach, so that students (both at secondary and tertiary level) do not get lost in theoretical concepts, but get acquainted with the idea through real projects. HEIs can also play a key role in the local and regional economy by promoting active citizenship of students and graduates, by creating partnership with industry and governmental organisations aiming to improve SR practices, and finally by making international connections enabling innovation and growth for local and regional partners.

There are national, sex and age differences in the perception of SR. This together with the fact that IS course design at different degrees differ significantly has lead us to propose the transversal adoption of SR practices. As the ISO 26000 guide has made clear, SR is a matter of process that involves a variety of practices across the organization. In consequence, it makes sense to align SR education with process issues instead of presenting it as an isolated topic. Particularly, practice-oriented process reference models (Fettke, Loos and Zwicker, 2006) provide the framework to present SR-aware practices as part of usual business that extends the different processes into a consideration of societal needs. This idea fits with existing proposals. For example, in the CMMI context, Corporate Social Responsibility (CSR) has been considered as an approach similar to CMMI itself which "helps companies to better address reputation risks, attract investors, improve relations with stakeholders, and become more competitive in mature markets" (Udovicki, 2007). As author of one of the essays in the book (Forrester et al, 2010), Barbara Neeb-Bruckner suggests CSR to be included into various

generic practices of CMMI. Going beyond these ideas, we suggest SRM itself can be considered a new process area with its own goals, practices and capability levels that would of course enable the practices of many other process areas. In ISO/IEC 15288 social responsibilities are currently considered as a factor influencing the use of other life cycle processes. In ISO/IEC 15288 and particularly in ISO/IEC 12207, we suggest the new SRM process to become part of the group of Organizational Project-Enabling processes which among others "provide resources and infrastructure necessary to support projects and ensure the satisfaction of organizational objectives and established agreements".

Integrating SR education in process reference models entails several benefits, including the following: (a) emphasizing SR as an extension and integral part of current practice rather than a separate one in line with current SR thinking, (b) easing curriculum integration by maximum flexibility – concrete SR issues are introduced at the points in which process models are presented, and (c) SR is described in a context, which would help students apply SR notions to real world situations.

As an example, an analysis of how ISO 26000 aspects could be introduced was carried out in the Spanish context of the study. Concretely, it was introduced in the context of IT Service discussion focusing on the ITIL process model. The main change to existing teaching practice was presenting the quality improvement approach of ITIL (*Continuous Service Improvement*) aligned with the improvement practices presented in section 7.7 of the ISO 26000 guidance regarding "reviewing and improving an organization's actions and practices related to SR". This introduction can be made effective by including SR-related performance indicators as part of several practices. The following are examples used:

Supply management

In the *ITIL Supplier Management* process, the aim is "to ensure that all contracts with suppliers support the needs of the business, and that all suppliers meet their contractual commitments". The consideration of SR should be presented in this case as a part of the "Supplier Strategy" document, introducing explicitly elements in several ISO 26000 core subjects as labour practices or environmental issues. This in turn may reflect in concrete requirements in supplier's terms and conditions and it will finally determine how the process of "Evaluation of new Suppliers and Contracts" (and also renewal and termination) is carried out. How this is materialized can be presented in different forms. A restrictive option is requiring some kind of commitment as for example, participation in the UN Global Compact Initiative[3], or scoring in Greenpeace's GGE indexes[4]. More particular cases would be that of considering carbon emission indicators in providers of data services for example. Even cases for human rights-related practices can be introduced, for suppliers in developing countries in which these are a key concern. In any case, as SR is very specific to the business context and local stakeholders, the variety of cases that can be presented to students is also high.

Facilities management

The aim of the *ITIL Facilities Management* process is "to manage the physical environment where the IT infrastructure is located". It includes all aspects of managing the physical environment, for example power and cooling, building access management, and it explicitly addresses environmental monitoring. In this case, the alignment of SR environmental issues as described in the ISO 26000 guide is straightforward and carbon footprint is considered as an indicator in actual practice. However, the trace back of these environmental issues to an overall IT Service Strategy that is considering SR more broadly is not evident, and providing students with a coherent rationale for these environmental indicators enhances the understanding of SR as a transversal and integral part of process design.

6   Conclusions

As a preparatory study for the suggested consideration of Social Responsibility Management on its own right in higher education and process reference models, this study investigated the perceptive differences in ethical issues and SR practices among Information Systems (IS) students in Greece, Hungary and Spain by using a survey comprising 205 questionnaires. The results of the statistical

---

[3] http://www.unglobalcompact.org/
[4] http://www.greenpeace.org/international/en/campaigns/climate-change/cool-it/Guide-to-Greener-Electronics/

analysis revealed that there is a strong positive correlation between perceptions of students regarding SR depending on country of origin, gender and age.

The students' perceptions of SR issues as grouped in ISO 2600:2010 was lowest in Greece (2,5), medium in Spain (3,1) and highest in Hungary (4,0).

The implication of our study calls for policy makers to involve sound SR practices in their strategies. The adoption of governance mechanisms towards a more socially responsible business conduct is imperative for organisations and institutions to include a similar approach in their practices. Only by ethical and social governmental conduct and by promoting values of ethical and social responsible behaviour we can assure growth and sustainability. In countries with low SR awareness this cultural change will take longer than in countries that have a high social responsible awareness. However, this seems to be the only way to go if we want to sustain and provide future generations the prosperity our generation, at least in Europe, has experienced. Due to national differences in the perception of SR, a practice-oriented introduction of SR issues appears as the more appropriate curricular approach. Reference process models can be used for that purpose, as they appear at some point of any IS curriculum worldwide and provide a contextualized presentation of SR supporting its consideration as integral part of organizational behaviour manifesting in concrete actions that can be monitored and measured.

Further work will include a similar study involving a much bigger sample. Also the opinions and viewpoints by management, administrative staff, and educators will be captured and compared to those results of the students. A longitudinal study will also provide insight about the changes in perception over time. In order to analyse the cultural influences on the perceptions of SR practices more institutions and countries need to be involved in order to limit biased results.

Acknowledgements

7 References

Annette, J. (2010). "The Challenge of Developing Civic Engagement in Higher Education in England", *British Journal of Educational Studies*, Vol. 58, No. 4, pp. 451–463

Annette, J. (2005). "Character, Civic Renewal, and Service Learning for Democratic Zitizenship in Higher Education", *British Journal of Educational Studies*, Vol.53, No.3, pp. 326 –340

Arlow P. (1991). "Personal characteristics in college students' evaluations of business ethics and corporate social responsibility" *Journal of Business Ethics* 10(1) pp. 63–69

Biro M. (2009) The Software Process Improvement Hype Cycle. Invited contribution to the Monograph: Experiences and Advances in Software Quality (Guest editors: D.Dalcher, L. Fernández-Sanz) Cepis Upgrade Vol. X (5) pp. 14-20.

Brown, J. S. (1997). "Becoming a Learning Organisation", in P.M. King and C. C. Schroeder (eds.) *About Campus*, Vol. 1No.6, pp. 14-16

Cetindamara, D. and Hopkins, T. (2008). "Enhancing students' responsibility towards society through civic involvement projects", *Innovations in Education and Teaching International*, Vol. 45, No. 4, pp. 401–410

Clark, C. R. (1993). "Social Responsibility Ethics: Doing Right, doing Good, doing Well.", *Ethics & Behaviour*, 3(3 & 4), pp. 303-327

Fowler Floyd J. Jr. (1993). "Survey Research Methods (2nd Ed.): Applied Social Research Methods Series", Volume 1, Sage Publications, London

Kolodinsky, R. W., Madden, T. M., Zink, D. S. and Henkel, E. T (2009). "Attitudes about Corporate Social Responsibility: Business Student Predictors." *Journal of Business Ethics*, 91, pp.167–181

Draganidis, S. and Siakas, K. (2007). "Revealing Semantic Quality Characteristics of Free and Open Source Software.", in E. Berki, J, Nummenmaa, I Sundly, M. Ross, G. Staples, *Proceeding of the 15th Software Quality Management Conference* (SQM 2007), August, Tampere, Finland, The British Computer Society, pp.109-116

Einfeld, A. and Collins, D. (2008). "The Relationships between Service-Learning, Social Justice, Multicultural Competence, and Civic Engagement", *Journal of College Student Development*, Vol. 49, No 2, pp. 95-109

Fenwick, T. (2011). "Learning 'social responsibility' in the workplace: conjuring, unsettling, and folding boundaries", *Pedagogy, Culture & Society*, Vol. 19, No. 1, pp. 41–60

Forrester, E.C., Brandon L.B., Shrum S. (2010). CMMI for Services: guidelines for superior service (Pearson Education) ISBN: 978-0-321-63589-1.

Fettke, P., Loos, P., & Zwicker, J. (2006). Business Process Reference Models: Survey and Classification. (C. Bussler & A. Haller, Eds.)*Business Process Management Workshops*, *3812*(Bprm), 469-483.

Fuchs, V.R. (1994) (ed.). "Introduction and Overview", *Individual and Social Responsibility: Child Care, Education, Medical Care, and Long-Term Care in America*, pp 3-6

Guy, N. (1997). "Markets, Higher Education and Social Responsibility" *Higher Education Polk*: Vol. 10, No. 3/4, pp. 161-162

Hofstede, G. (2001). "Culture's consequences: comparing values, behaviours, institutions, and organisations" - 2nd Ed. - Thousand Oaks, Calif.; London: Sage Publications

Hofstede, G. (1994). "Cultures and Organisations, Intercultural co-operation and its importance for survival, Software of the mind", McGraw-Hill, UK

Jones, S. and Abes, E.S. (2004). "Enduring Influences of Service-Learning on College Students' Identity Development", *Journal of College Student Development*, Vol.45, No.2, pp. 149-166

Koinig, S., Sicilia, M-A, Messnarz, R., Barriocanal, E.G, Garre-Rubio, M., Siakas, K. and Clarke, A. (2011). "Understanding the relation of SPI and SR: a proposed mapping of the SPI Manifesto to ISO 26000:2010" *18th European Software Process Improvement (EuroSPI 2011) Conference*, 27-29 June, Roskilde University, Denmark, (in print)

Liss, J. R. and Liazosa, A. (2010). "Incorporating Education for Civic and Social Responsibility into the Undergraduate Curriculum", *Change: The Magazine of Higher Learning*, January/February 2010, pp. 45-50

Lowery, J. M. (1998). "Institutional Policy and Individual Responsibility: Communities of Justice and Principle," in J. M. Lowery, (ed.) *Beyond Law And Policy: New Directions of Student Service*, no. 82, Jossey-Bass Publishers, 15-27

Maignan, I and Ralston, D. (2002). Corporate Social Responsibility in Europe and the U.S.: Insights from Businesses self-presentations. *Journal of International Business Studies*, 33(3), pp. 497-514

Mason, C. and Simmons, J. (2011). "Forward looking or looking unaffordable? Utilising academic perspectives on corporate social responsibility to assess the factors influencingits adoption by business", Business Ethics: A European Review, Vol. 20 No. 2, pp. 159-176

Matten, D. and Moon, D. (2004). "Corporate Social Responsibility Education in Europe.", *Journal of Business Ethics*, 54, pp. 323–337

Midttun, A., Gautesen, K. and Gjølberg, M. (2006). "The political economy of CSR in Western Europe". *Corporate Governance*, Vol. 6, No 4, pp. 369–385.

Norusis, M. J. (1994): *SPSS advanced statistics 6.1*, Prentice Hall, Chicago

Nyborg, P. (2003). "Higher Education as a Public Good and a Public Responsibility." Higher Education in Europe, Vol. 28, No. 3, pp. 355 - 359

Prinsloo, P., Buekes, C. and De Jongh D. (2006). "Corporate citizenship education for responsible business leaders". *Development Southern Africa* 23:197–211.

Reinhard, K., Osburg, T. and Townsend, R (2010). "Introducing corporate social responsibility as component in cooperative education: Results from a student research project in Germany and the United States supported by Intel Corporation", *Asia-Pacific Journal of Cooperative Education*, 11(1), pp. 1-12

Ringov, D. and Zollo, M. (2007). "Corporate responsibility from a socio-institutional perspective: the impact of national culture on corporate social performance". *Corporate Governance: The International Journal of Effective Board Performance*, 7:4, pp. 476–485.

Scholtens, B. and Dam, L. (2007). "Cultural values and international differences in business ethics", *Journal of Business Ethics*, 75:3, pp. 273–284.

Siakas, K. (2010). "Contribution of Professional Training to Graduate Entrepreneurship Development", *ECQA Conference*, 30.9. – 1.10.2011

Siakas, K. V., Georgiadou E. (2003a). "Learning in a Changing Society and the Importance of Cultural Awareness" IADIS 2003 (International Association for development of the Information Society) *International Conference WWW/Internet, ICWI 2003*, Algarve, Portugal, 5-8 Nov, Vol. I, pp.696-702

Skouloudis, A., Evangelinos, K., Nikolaou, I and Filho, W. L. (2011). "An overview of corporate social responsibility in Greece: perceptions, developments and barriers to overcome", Business Ethics: A European Review, Vol. 20 No 2, pp. 205-226

Solbrekke, T. D. (2008). "Professional responsibility as legitimate compromises - from communities of education to communities of work", *Studies in Higher Education* Vol. 33, No. 4, 485–500

Thornton, C. H. and Jaeger, A. J. (2006). "Institutional Culture and Civic Responsibility: An Ethnographic Study" *Journal of College Student Development*, Volume 47, Number 1, January/February, pp. 52-68

Udovicki K. (2007). Baseline study on CSR practices in the New EU Member States and Candidate Countries, UNDP. http://www.acceleratingcsr.eu/uploads/docs/BASELINE_STUDY_ON.pdf. accessed on 15/03/2012

Vasilescua, R., Barnab, C., Epurec, M. and Baicud, C. (2009). "Developing university social responsibility: A model for the challenges of the new civil society", *Procedia Social and Behavioral Sciences*, 2, pp.4177–4182

Voutsa, E., Siakas, K.V., Nisioti K. and Ross M. (2006). "A survey of the student's awareness on Ethical issues at the Technological Educational Institute of Thessaloniki", in R. Dawson, E. Georgiadou, P. Linecar, M. Ross. G. Staples (eds) *Learning and Teaching Issues in Software Quality*, *Proceedings of the 11th INternational Conference on Software Process Improvement - Research into Education and Training, (INSPIRE 2006)*, April, Southampton, UK, The British Computer Society, pp.139-150

Wong, A., Long, F. and Elankumaran S., (2010). "Business Students' Perception of Corporate Social Responsibility: The United States, China, and India*" Corporate Social Responsibility and Environmental Management* Corp 17, 299–310

Zandvoort, H (2008). "Preparing engineers for social responsibility", *European Journal of Engineering Education,* Vol. 33, No. 2, pp. 133–1

Enhancing ISO/IEC 15288 with Reuse and Product-orientation: Key Outcomes of an Add-on Process Reference Model Proposal

Fritz Stallinger, Robert Neumann
Process and Quality Engineering
Software Competence Center Hagenberg, Softwarepark 21, 4232 Hagenberg, Austria
{fritz.stallinger, robert.neumann}@scch.at

Abstract

Systems are typically developed within traditionally project-oriented contexts, generally facing high cost and productivity pressure, increasing complexity, and ever demanding customers. The transformation of this systems engineering business from highly customer-specific development projects to the customization and reuse of 'system products' is generally judged highly rewarding. In order to tackle this challenge, we build on the results of developing an ISO/IEC 15504 conformant process reference model for the industrial engineering that is based on the integration of the key concepts of reuse and product-oriented engineering of industrial systems and its comparison against ISO/IEC 15288 on systems life cycle processes.

The objective of the research is to synthesize the key process elements for enhancing the international standard ISO/IEC 15288 on system life cycle processes with product- and reuse-oriented engineering as well as basic product management practices in order to serve as an integrated, consolidated, and standard conformant reference framework for process assessment and improvement in systems engineering contexts in which systems are developed and evolved as products.

Keywords

systems engineering, product engineering, product management, reuse, system family, process reference model, ISO/IEC 15288, ISO/IEC 15504

1 Introduction and Background

Industrial engineering (IE) as an engineering discipline and specialization of systems engineering is concerned with the engineering of industrial solutions, e.g. power plants, rail systems, chemical plants, or substantial parts thereof. It integrates a series of specialized disciplines, e.g. mechanics, electrics, or software, and is typically carried out in different phases from acquisition, requirements analysis, to operation and maintenance. IE has to enable the parallel and coordinated development work of different engineering disciplines, including ensuring the integrity of the various discipline-specific solution components. It typically develops highly customer-specific and complex systems and is confronted with increasing cost and productivity pressure, rising complexity, and ever demanding customers.

The characteristics of IE and of the related business can be compiled from similar domains like industrial and investment goods [1] or complex products and systems [2], [3]. They are identified in detail in [4] and – beyond the criteria mentioned above – include, e.g. the provision of ready-to-use solutions and supporting services by a single provider; the carrying out of the development according to individual customer requirements; the high acquisition effort and a lack of economies of scale; a long life-span of industrial solutions and the respective customer relationships; the lack of testability of the solutions as a whole prior to construction; and the operation of the providers on global markets, constrained by various regional conditions and differences.

Due to the individuality of industrial solutions, their low degree of standardization, and the customer- and project-driven engineering approach, key success factors in the industrial solutions business (ISB) are the controlling of project risks and complexity, the improvement of quality, and the adherence to schedules, while at the same time reducing costs and building and maintaining solution expertise. Nevertheless, industrial solutions may share a sufficient amount of similarities which provide significant improvement potential through systematic reuse. Transforming the ISB from highly customer-specific development projects to the customization of 'system products' and fostering reuse in general are judged highly rewarding to tackle these challenges and exploit reuse opportunities.

The work presented in this paper builds on the results of developing a process reference model for IE, integrating an ISO/IEC 15504 [5] conformant process reference model for reuse and a process model for product-oriented engineering of industrial systems, and their comparison against ISO/IEC 15288 [6] on system life cycle processes (cf. [4], [7]). We synthesize the key processes and outcomes for enhancing the international standard ISO/IEC 15288 with product- and reuse-oriented engineering practices in order to serve as an integrated, consolidated, and standard conformant reference framework for contexts where systems or parts thereof are developed and evolved as products.

1.1 Reuse and Product-Orientation in Industrial Engineering

Enabling reuse across all engineering activities, levels of the solution structure, and engineering disciplines is regarded an effective but nevertheless challenging means for reducing costs and development time and for increasing IE and systems engineering quality (cf. [8]). While reuse is well understood in software engineering (cf. e.g. [9], [10], [11]), reuse in IE is hardly systematized, and often only applied in an ad-hoc manner, and its organization-wide potential is often not even known. The basic reuse approaches applicable in IE (cf. [12]) comprise component-oriented reuse, systematic copy-and-modify at solution and project level, use of prefabricates, application of platforms, and the product line and system family approach (cf. e.g. [13]).

Fostering product-orientation by transforming IE towards the customization of an underlying 'solution product' or 'system product' that is developed and maintained independently from specific customers or customer projects, is judged a promising approach to tackle the challenges IE is faced with. Such product-oriented engineering enables the advantages of product development within solution and systems engineering. It is characterized by an order-independent development process providing a set of products with pre-defined variants. Ideally, resulting from such an approach, a customer order implies no further customization effort and cost-coverage and profit margins can be secured at lower risks.

1.2 Goals, Approach, and Overview

The overall goal of the work presented here is to support the transition of system development organizations towards product-oriented and reuse-based development approaches. The overall approach is to synthesize key processes and outcomes for enhancing the international standard ISO/IEC 15288 with product- and reuse-oriented engineering practices in order to serve as an integrated, consolidated, and standard conformant reference framework for process assessment and improvement in such contexts. ISO/IEC 15288 is an established systems engineering standard, covers the entire life cycle of a system and provides a suitable foundation for the envisioned integrated framework.

In order to reach the above goal, we build on and extend previous research: an ISO/IEC 15504 conformant process reference model for the industrial solutions business (ISB-PRM) has been proposed in [4]. This model regards IE as a specialization of systems engineering and is based on the integration of the key concepts of reuse and product-oriented engineering and thus, is considered suitable for integration into the envisioned framework. The resulting ISB-PRM has further been compared against ISO/IEC 15288 with the objective to get an understanding of the relationship and to ensure the adherence of the ISB-PRM to the international standard. The results of this research are detailed in [7].

In this paper we analyze the results of the comparison of the ISB-PRM against ISO/IEC 15288 with respect to their transferability and integration into ISO/IEC 15288 and propose the key processes and outcomes for an extension of ISO/IEC 15288 with reuse and product-oriented practices.

The remainder of the paper is structured as follows: section 2 presents an overview and background information on the ISB-PRM and its relationship to ISO/IEC 15288; section 3 details the proposed extensions to ISO/IEC 15288 and comprises a description of the intended targeted application scenario, the analysis of previously proposed enhancement suggestions for ISO/IEC 15288, and the provision of process purpose and outcome definitions for the intended extension of ISO/IEC 15288; section 4 finally summarizes the paper, draws conclusions, and provides an outlook on future work.

2 ISB-PRM vs. ISO/IEC 15288

The work on developing an ISO/IEC 15504 conformant process reference model for the industrial solutions business (ISB-PRM) is based on the integration - in systems engineering contexts - of the paradigms of reuse and product-orientation represented by a process reference model for reuse in industrial engineering and a product-oriented process model for the engineering of industrial solutions, both developed in cooperation with Siemens AG Corporate Technology.

Based on the understanding of the content and relationship between both models, the ISB-PRM has been developed and proposed in [4], depicting and integrating the key contents of both underlying models. The ISB-PRM was constructed by performing a bidirectional mapping of the two base models and analyzing the respective mapping results. Table 1 shows the structure of the resulting ISB-PRM.

The analysis of the ISB-PRM against the international standard ISO/IEC 15288 was performed in a bidirectional way, i.e. for each process of a model it was identified through analysis of purpose definitions and outcomes to which degree it is thematically covered by processes of the other model using an NPLF-rating scale (None, Partially, Largely, Fully). The detailed results of this analysis are provided in [7]. The major results are as follows:

- As project management was not in the focus of the ISB-PRM's base models, it does not address the majority of the ISO/IEC 15288 project and project-enabling processes very well, although ISO/IEC 15288's *Risk Management Process* and *Measurement Process* are already part of the ISB-PRM support processes.

- While in ISO/IEC 15288 'the project has been chosen as the context for describing processes concerned with planning, execution, and assessment and control' [6], in the ISB-PRM, especially

Table 1: Structure of the ISB-PRM.

STRATEGIC PRODUCT PLANNING	ACQUISITION
Product Strategy	Acquisition
Commercialization Planning	Customer Requirements Analysis
Domain Analysis	Bid Preparation
Reuse Program Management	**PRODUCT APPLICATION**
PRODUCT DESIGN (DOMAIN ENGINEERING)	Design Planning & Scope Clarification
Product Development Planning	Basic Engineering (Solution)
Product Concept	Detail Engineering (Solution)
Basic Engineering (Product)	Realization & Operational Test
Detail Engineering (Product)	Start of Operation & Acceptance
PRODUCT UPDATE	Maintenance & Service
Collect Feedback	
Analyze and Evaluate	
SUPPORT	
Risk Management	Improvement
Change Management	Measurement
Non-Conformance Costs Management	Asset Management
Product Data Management	Quality Assurance
Tool Management	Knowledge Management
Quality Management	

with product-orientation, there are continuous activities like the development or enhancement of reusable products, which are not necessarily part of a project.

- ISO/IEC 15288 processes *Acquisition Process*, *Operation Process*, and *Disposal Process* are not or just partially addressed in the ISB-PRM.

- ISO/IEC 15288 does not explicitly address processes for reuse and product-orientation, e.g. processes of the ISB-PRM's *Strategic Product Planning* category or the *Asset Management Process.*

- The ISB-PRM does not provide further details on the realization of a solution, although the technical support of these activities is addressed, while ISO/IEC 15288 specifically distinguishes the *Implementation*, *Integration*, *Verification*, and *Transition Process*. The reasons therefor lie in the engineering focused context in which the ISB-PRM and its base models were developed.

An analysis of the detailed results of the comparison with respect to potential enhancements of ISO/IEC 15288 highlights the following issues (cf. [7] for details):

- While the majority of ISB-PRM processes, which address typical customer acquisition, solution development, and support topics, are well covered in ISO/IEC 15288, there is a clear lack of coverage and provision of detail of processes related to reuse and product-orientation.

- Currently unaddressed processes could be added to ISO/IEC 15288 by means of two additional process groups – a '*Reuse Processes'* group comprising the ISB-PRM's *Domain Analysis Process*, *Asset Management Process*, and *Reuse Program Management Process*, and a '*Product Planning'* group comprising the *Product Strategy Process*, *Commercialization Planning Process*, and the *Product Development Planning Process*, addressing the development of reusable system elements or products, independently of an actual customer request or order.

- Non-conformances are addressed in ISO/IEC 15288 mostly on the task level, while in the ISB-context they imply high financial and project risks and require comprehensive management. Therefore, the *Non-Conformance-Costs Management Process* could either be added to the ISO/IEC 15288 *'Project Processes'* group, or integrated with the *Risk Management Process*.

Generally, the underlying process model of ISO/IEC 15288 aids the consistent integration of additional processes. From a structure and granularity perspective, the ISB-PRM in some cases provides more detailed processes that might serve as lower-level processes, similar to the concept applied in ISO/IEC 12207 [14] where specific process activities are replaced by lower-level processes.

3 Key Processes and Outcomes for Enhancing ISO/IEC 15288

The following sections outline the intended application scenario for the enhanced process reference model, discuss initial observations from the comparison of the ISB-PRM and ISO/IEC 15288 regarding their integration, and provide in detail the proposed enhancements to ISO/IEC 15288 in terms of processes, process purposes, and outcomes, and present the final structure of the enhanced model.

3.1 Target Scenario for Application of Enhanced ISO/IEC 15288

Customer-specific systems or solutions are typically developed in a project context, due to the customer demand for fulfilling individual requirements, general system complexity, and long development times. Such projects are executed under high pressure for meeting cost, time, and quality goals. Reuse of existing engineering artifacts from similar past solutions and respective projects is typically performed in an ad-hoc manner and thus projects are vulnerable to errors, risks, etc. and strongly depend on the knowledge and experience of the involved project stakeholders and participants.

There is potential for considerably increasing reuse, given the similarities the developed systems typically share e.g. with respect to functionality, features, or structure, but also with respect to engineering artifacts like requirements specification, architecture description, test specification, etc. These similarities are to some extent introduced by the application domain of the systems.

By transforming the organization from highly project- and customer-driven individual development to product-orientation and reuse, this potential can be realized while still satisfying customer-specific needs. On the one hand, product-orientation means that pre-defined products - provided by internal or external suppliers - are made available for reuse in the development of a given family of systems. These products can be configured and adapted to customer-specific requirements, thus minimizing customer-specific effort in a project. On the other hand, if the overall system is built from customizable products, the system itself can be developed and managed as a product. The customer may choose from a set of such basic 'system products' which then are further customized. The characteristics of such a 'system product' are not determined individually and 'from scratch' as in a traditional project-oriented development approach, but nevertheless are pre-defined to a certain degree. However, the overall customer-specific system will be developed in a project context.

The organization may also choose to offer the pre-defined products, which are reused to provide customer-specific solutions, separately on a more or less anonymous market, typically with a reduced or standardized set of features. In this case, the reusable product itself may become a system product.

3.2 Preliminary Observations for Enhancing ISO/IEC 15288

The ISB-PRM shares the concept of the target scenario that customer-specific solutions are composed of products, that are developed independently of specific customers and that are reusable and adaptable for a variety of such solutions. It is independent of discipline-specific aspects in the development of such products, i.e. similar to ISO/IEC 15288 there are no processes specific to the development of e.g. mechanical, electrical, or software products or system elements, respectively.

In terms of ISO/IEC 15288 a system is composed of general system elements [6]. In the ISB-PRM, the term 'product' refers to these general system elements. The system meta-model of ISO/IEC 15288 features a system-of-interest and the decomposition into system elements, which again can be systems and may be further decomposed accordingly [6]. Similarly, the system-of-interest may be developed as a product, i.e. a system product, which is built from system elements some of which may also be developed as products. Such products can also be regarded as system products if they are also built from a set of products, accordingly.

The life cycle processes provided by ISO/IEC 15288 can be applied recursively for all the system

elements, if appropriate [6]. A similar recursion is envisioned in the enhanced ISO/IEC 15288 if the overall system is developed and managed as a system product, and at the same time is built from a managed set of products. The product- and reuse-oriented processes provided in this paper may thus also be applied recursively to system elements that are developed as products, if appropriate.

3.3 Enhancement of ISO/IEC 15288 towards Reuse and Product-orientation

In [7] the addition of two process groups was envisioned. The following paragraphs shortly characterize the respective key processes, which were not covered by ISO/IEC 15288, and provide – in table form – a detailed proposal for the definition of the respective process purposes and outcomes.

Firstly, a *'System Reuse Processes'* group (denoted *'Reuse Processes'* in [7]) is suggested, based on the ISB-PRM's *Domain Analysis Process*, *Reuse Program Management Process*, and the *Asset Management Process* (cf. Table 2).

The *Domain Analysis Process* aims to describe and characterize an application area or domain for which one or more system products are provided in order to satisfy the requirements of the stakeholders in this domain. This includes the identification of domain-typical terms, objects, concepts, problems, and solutions that are understood by practitioners in the specific area and the knowledge how to build systems in this domain. Further it comprises the analysis of the identified elements, their interrelationships, and the relationships to elements outside the domain as well as the definition of the domain borders. It is essential for the identification of appropriate system products and reusable assets to capture the domain commonalities and domain variability they have to realize.

The *Reuse Program Management Process* aims to plan, initiate, and jointly coordinate various reuse related efforts and projects throughout the organization in order to efficiently exploit reuse opportunities, establish reuse as a key concept and as a business driver in the organization, and support the achievement of the organization's business goals. An organization-wide reuse program is vital for ensuring focused, coordinated, effective, and efficient reuse activities.

The *Asset Management Process* aims to establish the infrastructure for efficient administration and

Table 2: System Reuse Processes.

Domain Analysis Process
The purpose of the *Domain Analysis Process* is to identify and describe a domain, the problems, requirements, commonalities, and variabilities within this domain, and thus support the identification of one or more suitable system products to be provided for the domain and the reusable products these systems are built from.
1. The boundaries of the domain and its stakeholders and their requirements are identified.
2. The terminology used in the domain is defined in a domain dictionary.
3. The elements of the domain, their relationships and dependencies to each other as well as to elements outside the domain are identified and coordinated between the involved disciplines.
4. Domain information is captured in a domain model and made available to relevant stakeholders.
5. Reusable assets that realize commonalities and variabilities in the domain are identified.
6. Templates and artifacts that support the engineering of system products are defined, kept up-to-date, and made available to relevant stakeholders.
Reuse Program Management Process
The purpose of the *Reuse Program Management Process* is to plan, establish, manage, monitor, and control a reuse program and to systematically exploit reuse possibilities within the organization.
1. A strategy for establishing reuse and different reuse approaches throughout the organization is defined and established.
2. A reuse program is defined, coordinated between affected stakeholders, and established.
3. Implementation of the reuse program is monitored and controlled.
Asset Management Process
The purpose of the *Asset Management Process* is to manage the life cycle of assets from conception to retirement and ensure their availability for reuse.
1. Assets are managed, fulfill defined quality requirements, are systematically stored, and are maintained over their entire life cycle.
2. Assets can be effectively searched and retrieved from the asset catalog of the organization.
3. Criteria for asset incorporation into or elimination from the asset catalog are defined.
4. The life cycle of assets is defined and planned.

maintenance of an organization's catalog or library of reusable assets and to manage these assets over their entire life cycle. Making reusable assets and information about them available and easily accessible for potential users is a prerequisite for the acceptance and successful utilization of assets.

Secondly, a *'System Product Processes'* group (denoted *'Product Planning Processes'* group in [7]) is suggested, based on the ISB-PRM's *Product Strategy Process*, *Product Development Planning Process,* and *Commercialization Planning Process*. These processes operate independently of specific projects and therefore, are added as a separate group (cf. Table 3).

The *Product Strategy Process* aims to define the business goals and major characteristics and features of the overall system product as well as those kinds of system elements that are intended to be developed project-independently as reusable products. The respective basic reuse approaches to be applied for these products are also determined. The product strategy has to be in line with the business strategy and the requirements of the markets that are addressed with the system product. Thus, the process fosters long-term strategic thinking about products which is vital for efficient product development and business success.

The *Product Development Planning Process* aims to roughly define the system product and which products it is built from and to determine a plan with defined milestones and the major features to be implemented in the long-term in form of a roadmap for the system product. Due to potentially manifold interdependencies between products, foresighted planning is required to coordinate development and enhancement of the individual products.

The *Commercialization Planning Process* determines how the system product will be introduced into the market by means of market positioning, pricing models, advertisements, etc. These aspects have to be analyzed early in the system product's life cycle because they may impact realization-related processes, activities, or early design decisions (e.g. by setting a maximum limit for production costs, or setting time constraints based on market or competitor information). Therefore, this process aims to prevent a misalignment of products with market needs and to ensure product success.

Further, the addition of the ISB-PRM's *Non-Conformance Costs Management Process* was suggested in [7]. The *Non-Conformance Costs Management Process* aims to manage the risks and reduce tangible and intangible costs induced by non-conformances of the product to internal and external requirements. In the ISB-PRM the process served as a support process due to the potentially high impact such costs might have in the ISB. Nevertheless, this may not be the general case in a systems engineering context. Therefore, the process is not included in the extension of ISO/IEC 15288 as proposed here, but if required may be used as part of the ISO/IEC 15288 *Risk Management Process*, since non-conformances basically are a special kind of risks and thus well covered by this process.
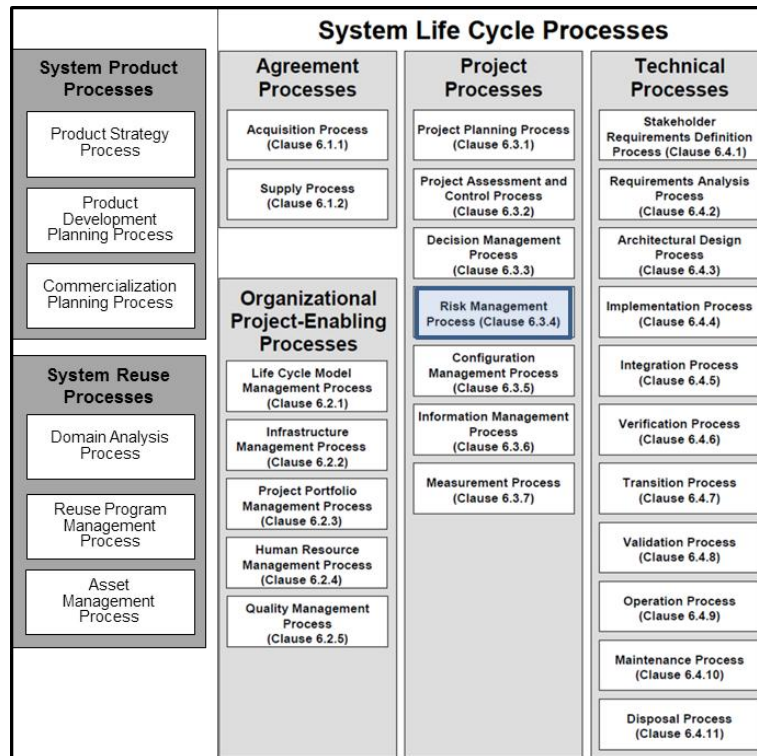
Figure 1: ISO/IEC 15288 Structure Enhanced with Product- and Reuse-Orientation.

Fig. 1 shows the resulting enhanced ISO/IEC 15288 structure. It emphasizes the project-independent character of processes in the *System Product Processes* and *System Reuse Processes* groups. Non-conformance according to the ISB-PRM *Non-Conformance Cost Management Process* may be regarded as special risks in the ISO/IEC 15288 *Risk Management Process* if required.

4 Summary, Conclusions, and Further Work

Table 3: System Product Processes.

Product Strategy Process
The purpose of the *Product Strategy Process* is to define the business goals and major features of the system products and of those products that will be developed order-independently for reuse in future solution projects.
1. The goals, basic characteristics, unique features, and potential applications of the system products and the products they are built of are defined in a product strategy.
2. The product strategy supports the business strategy and regards the requirements of addressed markets within the identified domain.
3. The system product portfolio of the organization is outlined and business opportunities are identified.
4. The general reuse approaches to be applied are determined.
Product Development Planning Process
The purpose of the *Product Development Planning Process* is to roughly define a system product for the system product portfolio according to the market requirements and to determine a plan with defined milestones for the development of the system product.
1. The system product's features, its basic structure, and the products and assets it is built of are defined.
2. The target degree of standardization for the system product is determined.
3. Domain standards or regulations relevant for the system product are identified.
4. A long-term plan for the major system product features, milestones for their implementation, and themes for future developments is defined and maintained in a system product roadmap.
Commercialization Planning Process
The purpose of the *Commercialization Planning Process* is to plan the introduction of the system product into to the market.
1. A strategy for marketing of the system product is defined.
2. The required activities for market introduction and internal promotion of the system product and major milestones are defined and coordinated between relevant stakeholders.
3. Concepts for system product pricing, advertisement, and distribution channels and partners are identified.

The paper presented the background, underlying models and preceding work that provided the foundation for synthesizing the key processes and outcomes for enhancing the international standard ISO/IEC 15288 on system life cycle processes with product- and reuse-oriented practices.

Based on the analysis of the results of the comparison of the ISB-PRM against ISO/IEC 15288 an outline for the extension of ISO/IEC 15288 was proposed that includes the addition of a *'System Reuse Processes'* and a *'System Product Processes'* group in order to support product-orientation and reuse on the overall system level.

The addition of product- and reuse-oriented activities that inherently cross the life cycle of systems and that are independent of specific projects was appropriately supported by the continuous architecture of ISO/IEC 15288 and the suggested enhancements appear considerably suitable. Nevertheless, further refinement, validation, and probably consolidation of the purposes and outcomes of the additional processes has to be performed, in particular with respect to their interfaces and interplay with the existing ISO/IEC 15288 system life cycle processes.

Some topics of the suggested enhancements to ISO/IEC 15288, e.g. product strategy or commercialization, are out of the standard's scope and thus widen the scope for the enhanced model. With progressing product-orientation the need for systematically managing the evolving product landscape increases. Such more sophisticated product management activities (cf. e.g. [15], [16] with respect to software) are currently out of scope of both the ISB-PRM and ISO/IEC 15288 and a candidate subject for future work.

In a related research strand, a software product management process reference model was proposed together with its integration with ISO/IEC 12207 on software life cycle processes [17]. Considering the ongoing harmonization of ISO/IEC 12207 and ISO/IEC 15288 and the resulting alignment of their system context processes, it seems reasonable to examine the possibilities for integrating the enhancements to both standards in order to provide an integrated framework for product management and reuse on the overall system level but also on the level of discipline-specific system elements.

Since industrial systems often are comprised of mechanical, electrical, or other discipline-specific products or are integrated mechatronic products, investigations on the applicability of the proposed product- and reuse-oriented and software product management related processes to such discipline-specific contexts are considered useful and promising.

When aiming towards customer-specific but at the same time product-oriented development, variability management and the traceability of requirements to the different customers and product variants are key issues [18]. It remains to be analyzed whether and how these issues have to be addressed on a process level within the context of the enhancements suggested in this paper.

Further it should be evaluated how the proposed enhancements to ISO/IEC 15288 support the handling of multiple system families or system product lines that share common assets.

Acknowledgment

5 Literature

1. Backhaus, K., Voeth, M.: Industriegütermarketing (in german). 8th edition. Vahlen, München (2007)
2. Davies, A., Hobday, M.: The Business of Projects: Managing Innovation in Complex Products and Systems. University Press, Cambridge (2005)
3. Schaible, J., Hönig, A.: High-Tech Marketing in der Praxis (in German). Vahlen, München (1996)
4. Stallinger, F., Neumann, R., Vollmar, J., Plösch, R.: Towards a Process Reference Model for the Industrial Solutions Business: Integrating Reuse and Product-orientation in the Context of Systems Engineering. In: T. Rout, G. Lami, F. Fabbrini (Eds.): Process Improvement and Capability Determination in Software, Systems

Engineering and Service Management. Proceedings of: 10th International SPICE Conference, Edizioni ETS, Pisa, 2010, pp.129-139. (2010)

5. ISO/IEC 15504:2003. Information Technology – Process Assessment. International Standards Organization (2003)

6. ISO/IEC 15288:2008. Systems and software engineering - System life cycle processes. International Standards Organization (2008)

7. Stallinger, F., Neumann, R., Vollmar, J., Plösch, R.: Reuse and Product-orientation as Key Elements for Systems Engineering: Aligning a Reference Model for the Industrial Solutions Business with ISO/IEC 15288. In: Proceedings of the International Conference on Software and Systems Process (ICSSP 2011); David Raffo, Dietmar Pfahl and Li Zhang (eds.); May 21–22, 2011, Waikiki, Honolulu, HI, USA; ACM ISBN: 978-1-4503-0730-7; pp. 120-128 (2011)

8. Löwen, U., Bertsch, R., Böhm, B., Prummer, S., and Tetzner, T.: Systematisierung des Engineerings von Industrieanlagen (in German). atp –Automatisierungs-technische Praxis. pp. 54-61. Oldenbourg Industrieverlag (2005)

9. Jacobson, I., Griss, M., Jonsson, P.: Software Reuse: Architecture, Process and Organization for Business Success. Addison-Wesley Professional (1997)

10. Mili, H., Mili, A., Yacoub, S., Addy, E.: Reuse-Based Software Engineering: Techniques, Organizations, and Controls. Wiley-Interscience (2001)

11. Sametinger, J.: Software Engineering with Reusable Components. Springer (2001)

12. Stallinger, F., Plösch, R., Pomberger, G., Vollmar, J.: Integrating ISO/IEC 15504 Conformant Process Assessment and Organizational Reuse Enhancement. Journal of Software Maintenance and Evolution: Research and Practice. Volume 22. Number 4. pages 307-324 (2010)

13. Clements, P., Northrop, L. N.: Software Product Lines: Practices and Patterns. Addison Wesley Professional Series: The SEI Series in Software Engineering (2002)

14. ISO/IEC 12207:2008. Systems and software engineering - Software life cycle processes. International Standards Organization (2008)

15. Kittlaus, H.-B., Clough, P.N.: Software Product Management and Pricing: Key Success Factors for Software Organizations. Springer, Berlin (2009)

16. Ebert, C.: The impacts of software product management. The Journal of Systems and Software 80, pp. 850–861 (2007)

17. Stallinger, F., Neumann, R.: Extending ISO/IEC 12207 with Software Product Management: A Process Reference Model Proposal. Accepted at SPICE 2012 Conference. May 29–31. Palma de Mallorca, Spain (2012)

18. Stallinger, F., Neumann, R., Schossleitner, R., Kriener, S.: Migrating Towards Evolving Software Product Lines: Challenges of an SME in a Core Customer-driven Industrial Systems Engineering Context. Proceedings of: 2nd International Workshop on Product Line Approaches in Software Engineering (PLEASE 2011), Waikiki, Honolulu, pp. 15-19. (2011)

6 Author CVs

Fritz Stallinger

Fritz Stallinger is Key Researcher and Project Developer in the Process and Quality Engineering Group (PQE) at the Software Competence Center Hagenberg, Austria (SCCH). His research interests include process and quality management, process modeling, and process assessment and improvement in the fields of software and systems engineering, component-based software engineering, software architecture engineering, reuse, requirements engineering and product management, and strategic and business aspects in software and systems engineering. He studied computer science at the Johannes Kepler University Linz, Austria (JKU). Before joining the SCCH in 2004 he has been a consultant to the European automobile industry in the areas of product and strategic planning and the co-ordination of new car development projects and a research assistant at the Department of Systems Engineering and Automation at the Johannes Kepler University Linz, where he – among others – severed as the project coordinator of the OOSPICE Project, a European Union funded international research project targeting at the development of a process improvement and development methodology for component-based software engineering.

Robert Neumann

Dipl.-Inf. Robert Neumann studied computer science at the 'Brandenburgische Technische Universität' (BTU) in Cottbus, Germany. In his diploma thesis he worked on identifying software quality problems based on the analysis of software change history. In 2007 he joined the Software Competence Center Hagenberg (SCCH) as an Industrial Researcher in the Process and Quality Engineering Group (PQE). There he is concerned with the research and development of new concepts and methodologies for quality and process management as well as the application of customized solutions for the improvement of software and software-intensive systems engineering organizations.

The LEGO Strategy: Guidelines for a Profitable Deployment

Luigi Buglione, Christiane Gresse von Wangenheim, Fergal Mc Caffery, Jean Carlo Rossa Hauck

Abstract

When dealing with improvements, organizations seek to find a break-even point for their applications as early as possible in order to maximize the return from their investment. However, in some cases such a strategy can lead to a long term failure by not realizing the full benefits, when focusing only on a short term. The LEGO (**L**iving **E**n**G**ineering pr**O**cess) approach – a method for building your own process meta-model based on multiple inputs – is a way to make an organization more efficient and effective, optimizing resources, as well as time and costs through looking at its entire Business Process Model. This paper introduces the elements for designing a strategy for a more valuable deployment of a process improvement initiative, in order to optimize the choice of the models and elements to be considered as an input to the LEGO approach.

1 Introduction

In the IT domain there are some periodical reports that [1][1][1] reveal issues regarding organizational and project management styles and the results achieved, such as the Standish Group CHAOS report [2][2][2] or the Gartner Magic Quadrants [3][3][3]. Consequently, there much literature exists that analyzes the possible 'top 10' or 'top 5' main causes for project failure. However, these reports do not focus on the core problem, concentrating instead on short-term objectives, and not on the wider mid to long-term ones [4][4][4][5][5][5][6][6][6][7][7][7][8][8][8]. When trying to understand why this happens so frequently and (possibly) to propose ideas for reducing such phenomenon, it is important to analyze an organizations structure. This can be done by starting from the strategic levels an organization typically formalizes: strategic, tactical and operational, respectively looking at long, mid and short term objectives. Following a Balanced Scorecard (BSC) deployment [9][9][9], all the organizational levels (not only the processes within each defined perspectives) must be aligned and properly communicate to each other, providing enough information to enable management at each level to make the organization effective and proficient.

Unfortunately, there is often a misunderstanding concerning coordinating the different purposes, goals and time-targets, provoking an improper distribution of resources across the different organizational areas [10][10][10] (aka 'organizational conflict'). One of these areas is the monitoring & control of the organization by periodical audits and appraisals. Due to the 'inner quality' costs, many managers feel that minimizing the cost of the 'monitoring & control' process as well as the associated improvement actions could lead to a reduction of profits on the short-term, but do not consider investing this saving in other 'productive' actions. This lack of vision can be easily represented in business terms through the 'cost of non-quality' (CONQ), as in most, well-known Total Quality Management (TQM) studies. Thus, one of the leverages for minimizing the CONQ is to increase the COQ (Cost of Quality), to help prevent post-release defects and problems from occurring. Looking at the list of possible cost attributes that contribute to the COQ, prevention costs are mostly based on appraisals [11][11][11] at different levels, with the aim to detect potential problems or inefficiencies before they happen, removing them earlier at a lower cost, than if detected later after the validation and post-production phases. Thus, the main question is: how much should a company invest in performing appraisals in order to optimize the balancing between COQ and CONQ?

The aim of this paper is to provide at least a partial description of the logical boundary for process appraisals in an organization, not necessarily an IT one, when using Maturity & Capability Models (MCMs). The inclusion of certain questions in management and budget discussions may give the quality department an opportunity to obtain real commitment for long-term objectives [12][12][12][13][13][13]. Such questions could be e.g.: how many processes should be considered to obtain sufficient information for determining an effective process improvement? Which processes should be included in the initial set of processes to be analyzed in order to stimulate an effective and efficient improvement, when investing a certain amount of budget, without having (or willing) to start with a 'big bang' approach?

The paper is organized as follows: Section 2 discusses the main reasons why a process improvement program fails and what should be focused upon to increase the probability of success. Section 3, introduces the LEGO approach and processes and the need for a strategy. Section 4, describes the first LEGO phase, introducing a simple but effective way to derive your own implementation strategy based upon your historical data and objective evidence. Finally, Section 5 provides some conclusions and the next steps for this work.

2 Why Process Improvement Programs Fail?

2.1 The Three Waves and Most recurrent Problems

Traditionally, process improvement has been used to obtain a path towards achieving certification in a certain model/framework, typically to achieve recognition within respective particular market and/or customer base. For instance, in the mid '90s, the *first wave* was to get certified with ISO 9001:1994 (and ISO 9002:1994 for services), the main standard for quality management, allowing for

market recognition as being the best in class. The *next wave* in early Y2K was to gain compliance with CMMI [14][14][14], ISO/IEC 15504 [15][15][15] (for ICT companies) or other maturity models, with a focus on the *staged* representation more than the *continuous* one, because of the possibility to achieve a benchmark to enable them to compete against direct competitors. The *last wave* was in the mid Y2K, searching for multi-models approaches, but not having a defined way to create a meta-model. SEI's PRIME initiative [16][16][16], like other proposals [17][17][17][18][18][18][19][19][19] seek a proficient way to integrate multiple models into a single model for representing the final 'process reference model' (PRM) that may be used to compare with the organizational BPM. Existing literature [20][20][20][21][21][21][22][22][22] highlights the common problems that occur in an improvement project. For example, lack of resources, time pressure, staff turnover, lack of support, lack of sponsorship, etc.

2.2   Further Attention Points

There are some misconceptions and issues that require particular attention:

- An ISO management system standard such as ISO 9001 or ISO 27001 is a list of *requirements*, not a *process* model. Thus, also mappings and comparisons between requirement and process models (e.g. CMMI) need to be carefully considered, but not treated as complete substitutes as it often happens;

- In MCMs, a *staged* representation proposes a predefined list of processes for an evolutionary implementation using blocks of processes. But few people carefully consider, if such predefined progression is valid for them both from a technical and business viewpoint. For instance, even if many studies (and common-sense) propose and demonstrate that an ISO 9001:2000 certified company is approximately equivalent to a company with a maturity level (ML) between CMMI ML2 and ML3 [23][23][23][24][24][24][25][25][25], a basic and core process such as Root-Cause Analysis (RCA), CAR (Causal Analysis & Resolution) is a CMMI ML5 process. This means that using the staged representation, an organization that is ISO 9001:2000+ certified cannot demonstrate directly an equivalent value from this inner capability (RCA is part of the ISO 9001 requirements). Therefore, creating an impression that's less than its real value. On the opposite side, adopting the *continuous* representation would overcome this issue, by instead measuring capability levels (CLs) for the set of processes – whatever the established ML reference – a company intends to evaluate.

- Well-established SPI models such as CMMI or SPICE (ISO/IEC 15504) include a set of processes covering a large part of the project lifecycle in a timely manner and can therefore be defined as 'horizontal' (following a timeline, from the beginning to the end of a project). Even if an organization adopts a multi-model integration approach, there are some questions that should be answered e.g. which risks could arise if the organization didn't perform a preliminary analysis on critical success factors (CFS) for a proper deployment?

Thus, an RCA should be performed at the strategic level to establish which should be the main list of issues whose fixing would represent the starting point for a sustainable, mid-long term improvement program.

3 Looking for a Solution: Back to the Strategy

3.1   Reactive vs Proactive Moods

Instead of many organizations working in a proactive manner to determine the yearly budget for process appraisals and improvement programs (e.g. ISO 9001, ISO/IEC 20000-1, CMMI-DEV, ISO/IEC 15504, etc.), they instead work in a 'reactive' way. Certification and compliance to standards should be achieved through simply following a logical sequence to accomplish business objectives using common-sense rules and principles. Such a common-sense approach should be based on:

- **People**: even if properly designed, processes will only succeed if executed by competent and skilled people. Most of the core processes in any organization (e.g. requirements elicitation, CMMI-DEV RD process area) cannot be automated. Consequently, attention to 'soft skills' is required. Furthermore, as in many performance management models and frameworks (e.g. BSC [26][26][26][27][27][27], MBQA [28][28][28], EFQM [29][29][29], etc.), people are an 'enabler', coming first in terms of timing in the value chain [9][9][9].

- **Data-Information-Knowledge-Wisdom (DIKW)** [30][30][30]: is an acronym from the ITIL v3 Knowledge Management process, describing what is required for increasing the organizational knowledge, from historical *data* (less data equates to lower quality estimates and higher discrepancies between estimates and final values) to real wisdom, providing guidance as to 'why' specific actions are or are not taken. The business questions to be answered include: are we assessing/appraising the right things? And are we assessing/appraising the things right?

3.2   The LEGO Approach

In an attempt to encourage proactive process improvement, we have proposed a common-sense approach, called **LEGO** (**L**iving **E**n**G**ineering pr**O**cess) [17][17][17] for stimulating organizations to improve their own processes, by taking components (such as the real LEGO bricks) from multiple, potential information sources and integrated them to form a unique, reinforced picture for a particular process or set of processes. The starting point – for this paper – is that any model/framework typically represents only a part of the real story. Thus, through handling similar elements from different sources, we can hopefully find more 'fresh blood' for improving the organizational processes. This becomes necessary as a frequent misconception of organizations when dealing with certification programs is to shift the real target (improving the process to better satisfy the business objectives) with the supporting tool (e.g., achieving a certain maturity level). Therefore, in order to achieve the real target, we need to pragmatically improve organizational processes by introducing best practices from a selection of models/frameworks and experiences. Therefore, after establishing the business goals there is a need to search for and identify which 'supporting tools' are most applicable for the current situation. Unfortunately, it is often the case that organizations prioritise what is required in order to compliance against a particular model rather than striving for the best solution in terms of their processes . And in doing it, they risk achieving the opposite effect to what they intended, i.e. to lose and not gain 'value'. Thus, the 'fresh blood' we need are ideas and practices to be tailored, integrated and re-adjusted in the way that they will work in a specific organization, as opposed to a generic one. Thus, the LEGO approach enables little bricks to be used for building a concrete organizational value. LEGO has four main elements:

1. a 'Maturity & Capability Models' (MCM) repository [31][31][31] allowing a systematic search for and identification of relevant processes or MMs from existing models;
2. knowledge about the process architecture of each model, as a basis for understanding how to transform desired elements from a certain model into the target format, especially when considering that the source models may have different architectures that may need to be harmonized into a single model;
3. mapping(s) & comparisons between relevant models, in order to understand the real differences or the deeper level of detail from 'model A' to import into 'model B';
4. a process appraisal method (PAM) to be applied on the target BPM (Business Process Model).

LEGO has also a related four-step process for determining which elements to consider when improving your current BPM:

1. **Identify your informative/business goals**: clearly identify your needs, moving from the current BPM version and content.
2. **Query the MCM repository**: browse and/or search the MCM repository, setting up the proper filters in order to obtain the desired elements (processes; practices; etc.) to be inserted into the target BPM.
3. **Include the selected element(s) into the target BPM**: include the new element(s) in the proper position in the target BPM (e.g. process group, maturity level, etc.).
4. **Adapt & Adopt the selected element(s)**: according to the process architecture of both process models (the target and the source one), the selected elements may need to be adapted, through tailoring such elements as needed.

The LEGO approach and its basic elements have been presented in more detail in [17][17][17][31][31][31]. The next step is to provide tips and common-sense rules concerning how to proficiently apply it from the beginning, providing details about the first step from a strategic viewpoint, but not necessarily from a tactical one.

3.3   Looking for a Strategy

The first step of the LEGO process is to clearly identify your needs and being a technique, it assumes you can choose the preferred way for an organization, according to the amount and quality of data and information available. The interesting question is: how can we do it? This is the goal of this paper. We also attempt to provide an answer this question in a common-sense manner using practically applicable solutions. It is important to have a strategy, and for not to have only a tactical or operational short-mid term focus. Looking at the Webster-Merriam dictionary, one of the possible definitions is "*an adaptation or complex of adaptations (as of behavior, metabolism, or structure) that serves or appears to serve an important function in achieving evolutionary success*". Thus, a strategy should consider the long term ('evolutionary success') and shouldn't be confused with the tactical and operational levels. A possible formalization of such common-sense concepts is the STO model [32][32][32], associating different actors, time-frame and business questions to each level.
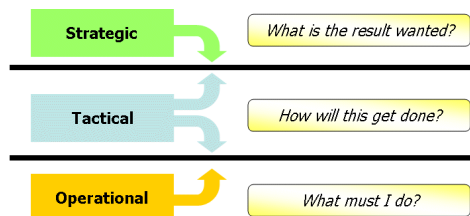
Figure 1: The STO Model [32][32][32]

From the viewpoint of appraisers and auditors, they check that the performance of the strategic decisions that were previously established by their management. But is there any consideration as to whether what we are performing is in fact correct or even are we working in the best way possible to achieve our goals? Table 1 below, illustrates a simple example of using STO goals, from the strategic to the operational level:

Table 1. STO Examples

Level	Scenario #1 – Goals	Scenario #2 – Goals
S – Strategic	• Provide quality product/services	• Be the best ICT provider in a certain market
T – Tactical	• Become ISO 9001 certified within 2 years	• Become compliant with main best practice models/frameworks after taking the best of them using the LEGO approach
O – Operational	• Run quarterly audits based on ISO 19011:2011 guidance	• Run periodical appraisals using the resulting (LEGO) meta-model mapped on the organizational Business Process Model (BPM)

The final choice should be made by considering all elements in such a scenario and calculating (even approximately) the ROI looking at different moments in time, not only for determining the BEP (Break-Even Point).

4   Establishing a Strategy from Historical Data

4.1   Positioning the LEGO Approach in the PDCA Cycle

In this section we outline our proposal for the design of a strategy. Using the well-known PDCA (Plan-Do-Check-Act) phases, Figure 2 illustrates the main steps within each of the four phases and the potential added value (also stressed with the '+' or '-' signs) for an organization adopting this approach to perform process management for the mid-long term. The coloured text shows the additional steps to be run for implementing a strategy against the typical steps for a usual PDCA-based improvement.

Plan	Do	Check	Act
(++)	(=)	(++)	(+)

| • Establish the strategy
• Balance s/term and m-l/term obj
• Consider previous appraisals results
• Determine the best appraisal boundary | • Planned Audit/Apprisals | • Run Gap Analysis
• Apply the LEGO approach
• Output: improved processes
• Determine the Action plan for corrective/improvement actions | • BPR (revising QMS's processes as needed)
• Deploying new processes
• Improve data gathering into organizational PALs (DIKW) |

Figure 2: The LEGO Strategy across the PDCA cycle

- **Plan**: there are three additional steps: (a) establish the strategy; (b) balance short-term and mid-term objectives; (c) determine the best appraisal boundary. The first two relate to the establishment of strategic and tactical goals. The third one concerns the final decision for determining the technical boundary for performing the audit/appraisal. A recent proposal for this last step is described and augmented in [33][33][33].
- **Do**: no additional steps in this phase.
- **Check**: two additional steps: (a) apply the LEGO approach; (b) output: improved processes. These two steps are extensively explained in [1][1][1].
- **Act**: just a final, additional step: (a) improve the data gathering into the organizational PALs (Process Asset Libraries), but introducing something more than solely D-I levels (Data-Information) from the DIKW path previously introduced (typical for a PAL, as described in CMMI OPD SP1.5) from the full DIKW (as described in ITIL v3, Service Transition book [30][30][30]). Some examples and tips are also proposed in (http://en.wikipedia.org/wiki/DIKW).

4.2   Focusing on Strategy: Making it Work

Focusing our attention on the 'Plan' phase, and on the first step (*Establish the strategy*), we can refer to Total Quality Management (TQM) well-known techniques. TQM tools contain many possible answers whether applying old and new quality tools with a simple common-sense mood: e.g. RCA (Root-Cause Analysis), Affinity diagrams, Pareto diagrams, Control Charts, etc. [34][34][34]. Simulations based on historical data could help in designing a program looking at a larger timeframe than is currently used in organizations. Figure 3, presents an example of taking a RCA analysis for determining why a project has many defects more than expected [35][35][35], and then redrawing it using mind maps, as suggested in [36][36][36]. Figure 3a shows the same elements from the original paper, while Figure 3b proposes a refined analysis using the "5why's game", showing 3-4 levels of depth.

When determining the final leaf for each branch, it is possible to create a match with the process element from the improvement model(s)/framework(s) from which useful support can come for re-designing processes. We now take a deeper look at some leafs, from the top of Figure 3b. Time pressure could be due to underestimations which may have arisen for several reasons such as: little historical data was available to assist the estimating process, or estimates were provided by inexperienced people. In the first case, the root-cause is due to the unavailability of a 'measurement repository' (using CMMI-DEV, would be stated in OPD SP1.4) or to missing definitions for some values in the project data (related in such case to MA SG1). In the second case (low experience), the related CMMI-DEV element would be Project Planning (PP) GP 2.5, this relates to the need for people to be trained. Of course, here a single well-known model has been considered, but suppose we wish to include all the potential elements that could be useful when following the LEGO approach. In other

words, the mapping with one/more model elements is a way to name the areas where gaps to be filled need to be reworked.

(a)

(b)
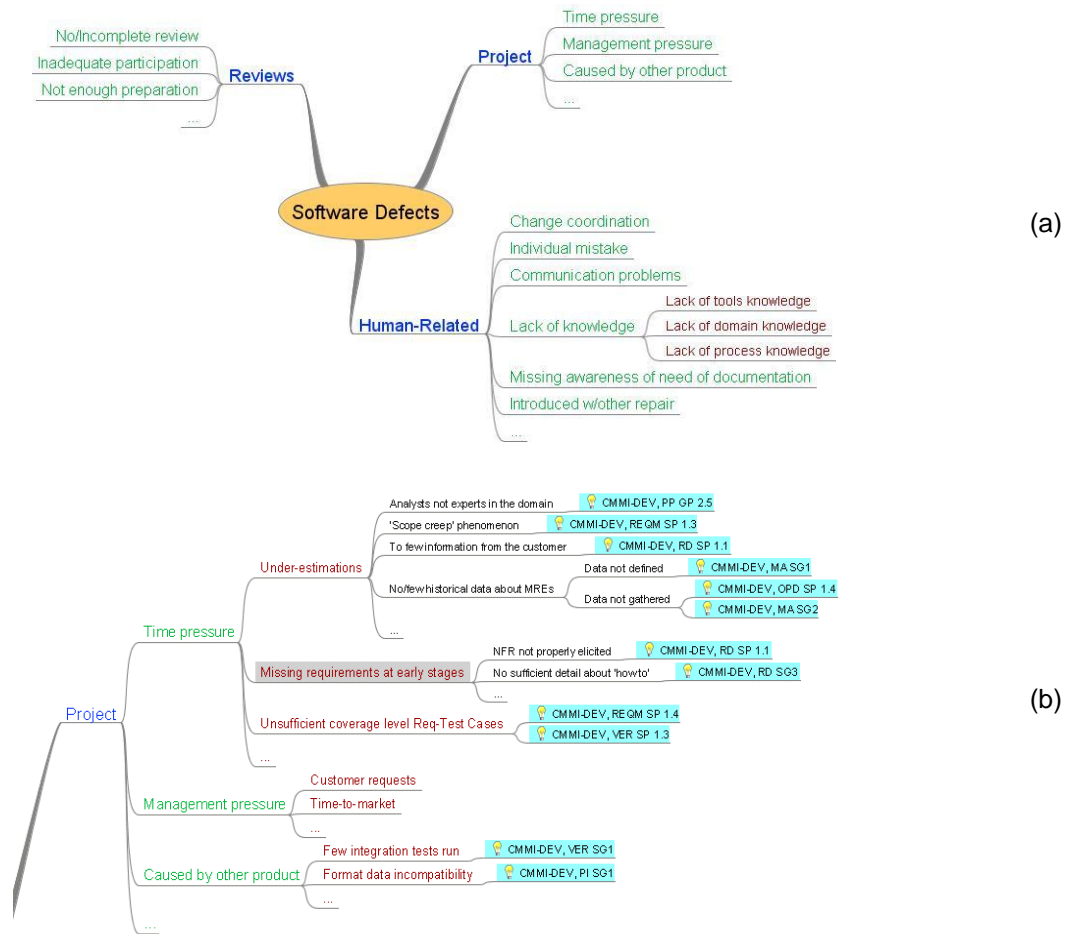
Figure 3: A root-cause analysis (RCA) for the effect '(too many) software defects'

Another fundamental concept in TQM is the classification – based on the frequency a certain fact occurs – or problems with *special causes* (happen with low frequencies and with no seasonality) or *common causes* (happen with seasonality, repeating patterns of activities). A strategic goal should therefore focus upon repeated patterns (in this case with a negative meaning) for determining stable, mid-long term actions reducing (or at least minimizing) potential negative impacts and stressing as much as possible the positive effects for an organization, whatever the perspective. Thus, consider running several RCAs within an organization in a certain timeframe, and think about the frequency of the 'models' elements in order to provide an interesting analysis. We refer to the analysis of the 'Project' main leaf from Figure 3b. Table 2, summarizes how many times that specific element was mentioned and establishing an implementation priority (with 'A' being the highest priority etc.), and based on the causal relationships among processes. Such information is contained into the 'Related Process Areas' section at the beginning of each process area description in the CMMI. Thus, since missing requirements could be the root-cause for having less formalized requirements and therefore fewer test cases than expected (with a higher potential number of final defects at the release phase), working on a better and deeper requirement elicitation (RD – Requirement Development) should be implemented first.

Table 2. Frequency and Implementation priorities from the RCA 'Project' leaf

	Model	Version	Process	Goal	Practice	Frequency	Impl. Priority
			PA	*SG/GG*	*SP/GP*		*Cause-Effect*
1	CMMI-DEV	1.3	**MA**	SG 1		1	**B**
2	CMMI-DEV	1.3	**MA**	SG 2		1	**B**

3	CMMI-DEV	1.3	**OPD**		SP 1.4	1	**B**
4	CMMI-DEV	1.3	**PI**	SG 1		1	**E**
5	CMMI-DEV	1.3	**PP**		GP 2.5	1	**C**
6	CMMI-DEV	1.3	**RD**		SP 1.1	2	**A**
7	CMMI-DEV	1.3	**RD**	SG 3		1	**A**
8	CMMI-DEV	1.3	**REQM**		SP 1.3	1	**D**
9	CMMI-DEV	1.3	**REQM**		SP 1.4	1	**D**
10	CMMI-DEV	1.3	**VER**		SP 1.3	1	**E**
11	CMMI-DEV	1.3	**VER**	SG 1		1	**E**

In order to determine which areas should be given more priority for reinforcing organizational processes, a Pareto analysis can be performed. Such an analysis lists process areas in descending order of the potential gaps from several RCA run across the organization in a certain time frame, see Figure 4.

Process	Frequency	Priority
RD	3	A
MA	2	B
REQM	2	D
VER	2	E
OPD	1	B
PP	1	C
PI	1	E
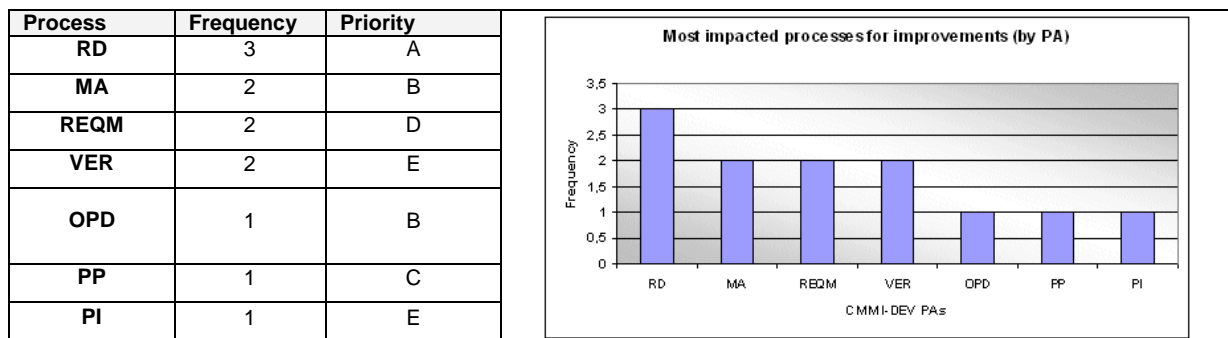
Figure 4: Most impacted processes for improvements (by process area)

Of course, an improvement plan must consider actions grouped by a certain criterion to be run at the same time, because of the causal link between them. In our proposal, this criterion is included in the 'implementation priority' field.

Priority	Process	Frequency
A	RD	3
B	MA, OPD	3
C	PP	1
D	REQM	2
E	VER, PI	3
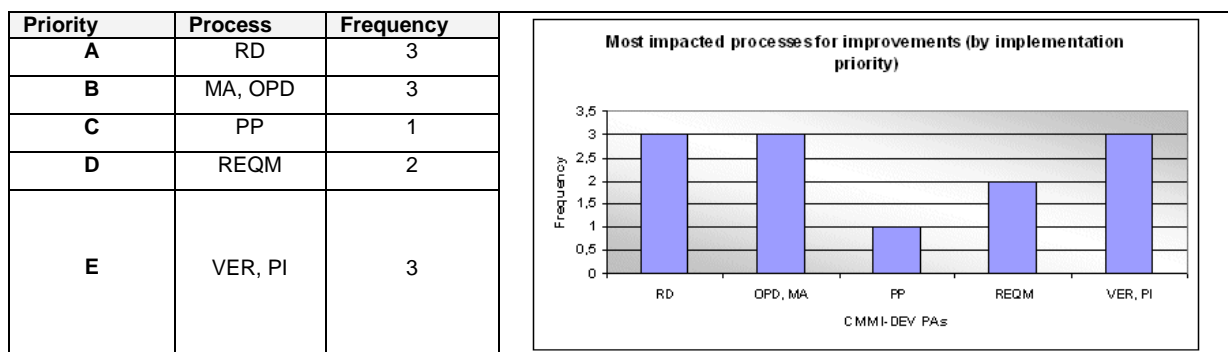
Figure 5: Most impacted processes for improvements (by implementation priority)

Figure 5 groups process areas by implementation priority level (from A to E). Thus, if the main problem for an organization was to have too many defects at the release stage, from such an analysis (assuming it has been validated), the improvement plan should start with refining how requirements are captured, making all requirements visible and no longer implicit (priority A), and then store historical data for improving future estimates (priority B), etc. If work is started on the priority A chunk, LEGO will aim to reinforce the organizational BPM through analyzing all possible maturity models/frameworks in relation e.g. to Requirement Engineering, or Project Management. The substantial difference from this structured analysis as opposed to simply adopting the thoughts from the management of an organization is that such decision will be augmented by the historical data of the organization, therefore adding more strength to such a decision. Finally, Figure 6 summarizes the operational flux for satisfying the 'Establish the strategy' step within the Plan phase.

Thus, the business value from such a preliminary activity would provide a more objective way for deciding which improvement areas should be included when planning an improvement project through

using your own historical data as a starting point. Such data can be retrieved from any type of objective evidence (e.g. audits, appraisals) and it is useful as it provides a better understand issues etc. within previous projects. Therefore, such approach would minimize from the outset the risk of adopting a costly and unfruitful process improvement program.
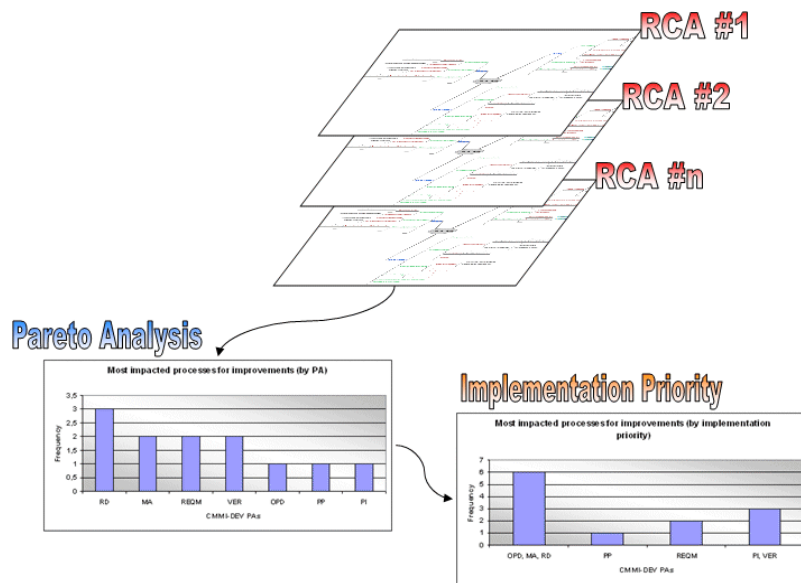
Figure 6: Root-Cause Analysis for determining CSF as pre-filters for adopting LEGO

5 Conclusions & Next Steps

Whatever the organization size, a strategy is always needed: applying a 'flavour of the month' approach cannot allow an organization to achieve mid-long term results, with seeds for a continual improvement over time. Thus, a strategy should be provided that is appropriate to the size and main attributes of an organisation, as also stated both in ISO management system standards (e.g. ISO 9001:2008) and main maturity models (e.g. CMMI-DEV with the quest of introducing a *tailoring* guideline, see OPD SP 1.2). On the contrary, there is a lack of clear organizational strategy that can be easily observed by the absence (or not clear presence) of MVV (Mission-Vision-Values) elements. Such an absence can easily reveal a weak or absent strategy, that would lead an organization to focus mostly on tactical goals, increasing the risk of not achieving its long-term business goals. A BPR (Business Process Re-engineering) initiative applying a multi-model approach such as LEGO should fit with a certain organizational size and characteristics, as often an ideal model is applied without due consideration as to what really happens.

This paper introduced and discussed how a strategy can be established for applying LEGO, through building upon an organization's historical data and objective evidences, using well-known TQM tools. Such a preliminary filter allows an organization to focus resources on its technical priorities but keeps in mind that the reference model is the management system of an organization and that any external model must be a potential input for strengthening it and not the ideal target for modifying the processes. Furthermore, even if many valid models/frameworks could be used for carrying out the LEGO approach, from observing ICT organizations they appear to continue to look for and apply only a few common models, while enlarging the analysis to a wider scope could provide richer sources. E.g. performance management models such as the Malcolm Baldrige Quality Award (MBQA) and the European Foundation for Quality Management (EFQM) could provide greater assistance in relation to Leadership (their first 'enabling' criteria) supporting and sustaining improvement initiatives and programs, as well as when using LEGO, which is not particularly developed in ISO 9001 [37][37][37] requirements and in CMMI or ISO/IEC 15504 models. The more the potential sources to be used, the higher the probability to redesign a set of valuable, improved processes for your own organization.

The next steps of this research will be to formally apply the LEGO strategy on real case studies, in order to validate it by quantitative figures comparing the initial different working hypothesis for an improvement program with and without such an approach.

6 References

[1] Eveelens J.L., Verhoef C., The Rise and Fall of the CHAOS Report Figures IEEE Software, Jan/Feb 2010, pp,30-36

[2] Standish Group, CHAOS Summary 2009. The 10 Laws of CHAOS, 2009

[3] Gartner, Magic Quadrants, URL: www.gartner.com/technology/research/methodologies/research_mq.jsp

[4] PM Solutions, Strategies for Project Recovery, Research Report, 2011

[5] Kappelman L.A., McKeenan R., Zhang L., Early Warning Signs of IT Project Failure: The Dominant Dozen, Journal of Information System Management, Fall 2008, pp.31-36

[6] Al-Ahmad W., Al-Fagih K., Khanfar K., Alsamara K., Abuleil S., Abu-Salem H., A Taxonomy of an IT Project Failure: Root Causes, International Management Review, Vol. 5 No. 1 2009, pp. 93-104

[7] Benjamin R., Project Success as a function of Project Management Methodology: an emergent system approach, MBA thesis, University of Hull/UK, April 2006

[8] Whittaker B., What went wrong? Unsuccessful information technology projects, Information Management & Computer Security, 7/, 1999, pp 23-28

[9] Kaplan R., Norton D., The Balanced Scorecard: Translating Strategy into Action, Harvard Business Review Press, 1996, ISBN 9780875846514

[10] Arvid Johnson R., Management, systems, and society: an introduction. Pacific Palisades, Calif.: Goodyear Pub. Co, 1976, pp. 148–142. ISBN 9780876205402

[11] Jagannathan S.R., Bhattacharya S., Matawie K., Value Based Quality Engineering, TickIT International, 1Q05, January 2005, pp. 3-9 C:\AppData\Local\Temp\Temp1_LEGO Strategy.zip\www.tickit.org\TI1Q05.pdf

[12] Abrahamsson P., The role of commitment in software process improvement, Dissertation, University of Oulu/Finland, 2002

[13] Ahmad A., Making Management Commitment Happen in SPI, Master Thesis in Software Engineering and Management, IT University of Goteborg (Sweden), 2007

[14] CMMI Product Team, CMMI-DEV (CMMI for Development) v1.3, Technical Report, CMU/SEI-2010-TR-033, Software Engineering Institute, November 2010

[15] ISO/IEC IS 15504-x, Information technology -- Process assessment, Parts 1-7, International Organization for Standardization, 2001-2007

[16] SEI, Process Improvement in Multimodel Environments (PrIME), webpage, URL: http://goo.gl/p2GX3

[17] Buglione L., Gresse von Wangenheim C., Hauck J.C.R., McCaffery F., The LEGO Maturity & Capability Model Approach, Proceedings of the 5th World Congress on Software Quality (5WCSQ)

[18] Pricope S., Lichter H., A Model Based Integration Approach for Reference Model, Proceedings of PROFES 2011, Torre Canne (Bari, Italy), June 2011

[19] ARMONÍAS: A Process for Driving Multi-models Harmonization, ARMONÍAS Project, University of Castilla-La Mancha (Spain), 2009

[20] Niazi M., Wilson D., Zowghi D., A maturity model for the implementation of software process improvement: an empirical study, Journal of Systems and Software (JSS); Elsevier, 74(2), January 2005, pp. 155 - 172

[21] CSE, Critical Success Factors for Successful Process Improvement, Centre for Software Engineering, URL: http://goo.gl/lgROc

[22] Niessink F., van Vliet H., Measurement Program Success Factors Revisited, Information and Software Technology, Vol 43 no 10 (August 2001), pp 617-628

[23] Paulk M.C., A Comparison of ISO 9001 and the Capability Maturity Model for Software, Technical Report, CMU/SEI-94-TR-012, Software Engineering Institute, July 1994

[24] Mutafelija B., Stromberg H., Process Improvement with CMMI v1.2 and ISO Standards, Auerbach, 2008, ISBN 978-1420052831

[25] Kitson D., Vickroy R., Walz J., Wynn D., An Initial Comparative Analysis of the CMMI Version 1.2 Development Constellation and the ISO 9000 Family, Special Report, CMU/SEI-2009-SR-005, Software Engineering Institute, March 2009

[26] Kaplan R., Norton D., Strategy Maps: Converting Intangible Assets into Tangible Outcomes, Harvard Business Review Press, 2004, ISBN 9781591391340

[27] Reo D., Quintano N. & Buglione L., ESI Balanced IT Scorecard Infrastructure & Innovation Perspective, ESI-1999-TR-043, December 1999

[28] NIST, Malcolm Baldrige, 2011-12 Criteria for Performance Excellence, URL: www.quality.nist.gov

[29] EFQM, EFQM Excellence Model, 2009, ISBN 9789052365015

[30] Cabinet Office, ITIL v3 Service Transition book – Refresh 2011, ISBN 9780113313068

[31] Gresse von Wangenheim C., Hauck J.C.R., Buglione L., Mc Caffery F., Cardoso Lacerda T., Viera da Cruz R.F., Building a Maturity & Capability Model Repository, Proceedings of PROFES 2011, 12th International Conference Product Focused Software Development and Process Improvement, Torre Canne - Bari (Italy), June 20-22 2011, 2nd Proceedings: Short Papers, Doctoral Symposium and Workshops, ACM, ISBN 978-1-4503-0783-3, pp. 2-5

[32] Callahan K., Brooks L., The Essentials of Strategic Project Management, John Wiley & Sons, 2004, ISBN 9780471649854

[33] Buglione L., Mc Caffery F., Hauck J.C.R, Gresse von Wangenheim C., FIRST: Common-sense Process Scopes for Starting a Process Improvement Program, Proceedings of SPICE 2012, Palma de Mallorca (Spain), May 2012

[34] Ishikawa K., Introduction to Quality Control, Productivity Press, 1990, ISBN 9784906224616

[35] Leszak M., Perry D.E. & Stoll D., A Case Study in Root Cause Defect Analysis, Proceedings of the 22nd International Conference on Software Engineering (ICSE 2000), Limerick (Ireland), June 4-11 2000, ISBN 1581132069, pp.428-437

[36] Buglione L., Strengthening CMMI Maturity Levels with a Quantitative Approach to Root-Cause Analysis, Proceedings of the 5th Software Measurement European Forum (SMEF 2008), Milan (Italy), 28-30 May 2008, ISBN 9-788870-909999, pp. 67-82

[37] ISO IS 9001:2008, Quality management systems -- Requirements, International Organization for Standardization, December 2008

Appendix A: List of Acronyms

BEP	Break-Even Point
BPM	Business Process Model
BSC	Balanced Scorecard
CAR	Causal Analysis & Resolution (CMMI ML5 PA)
CFS	Critical Success Factor
CL	Capability Level
CMMI	Capability Maturity Model Integration (www.sei.cmu.edu/cmmi)
CMMI-DEV	CMMI for Development
CNPq	Conselho Nacional de Desenvolvimento Científico e Tecnológico (www.cnpq.br)
CONQ	Cost of Non Quality
COQ	Cost of Quality
DIKW	Data-Information-Knowledge-Wisdom
EFQM	European Foundation for Quality Management (www.efqm.org)
GP	Generic Practice
ICT	Information and Communication Technology
IEC	International Electrotechnical Commission (www.iec.ch)
ISO	International Organization for Standardization (www.iso.org)
IT	Information Technology
ITIL	IT Infrastructure Library
LEGO	(**L**iving **E**n**G**ineering pr**O**cess
LERO	The Irish Software Engineering Research Centre (www.lero.ie)
MA	Measurement & Analysis (CMMI ML2 PA)
MBQA	Malcolm Baldridge Quality Award (www.nist.gov/quality)
MCM	Maturity & Capability Model
ML	Maturity Level
MVV	Mission-Vision-Values
OPD	Organizational Process Deployment (CMMI ML3 PA)
PAM	Process Assessment Model
PDCA	Plan-Do-Check-Act cycle (Deming)
PI	Product Integration (CMMI ML3 PA)
PP	Project Planning (CMMI ML2 PA)
PRIME	Process Improvement in Multimodel Environments (http://goo.gl/p2GX3)
PRM	Process Reference Model
QMS	Quality Management System
RCA	Root-Cause Analysis
RD	Requirement Development (CMMI ML3 PA)
REQM	Requirement Management (CMMI ML2 PA)
SFI	Science Foundation Ireland (www.sfi.ie)
SG	Specific Goal
SP	Specific Practice
SPI	Software Process Improvement
SPICE	Software Process Improvement and Capability dEtermination (ISO/IEC 15504)
STO	Strategic-Tactical-Operational
TQM	Total Quality Management
VER	Verification (CMMI ML3 PA)

7 Author CVs

Luigi Buglione

Dr. **Luigi Buglione** is an Associate Professor at the École de Technologie Supérieure (ETS) – Université du Québec, Canada and is currently working as a Measurement & Process Improvement Specialist at Engineering.IT SpA (formerly Atos Origin Italy and SchlumbergerSema) in Rome, Italy. Previously, he worked as a Software Process Engineer at the European Software Institute (ESI) in Bilbao, Spain. Dr. Buglione is a regular speaker at international Conferences on Software/Service Measurement, Process Improvement and Quality, and is actively part of several International (ISO WG10-25, IFPUG, COSMIC, ISBSG, MAIN) and National (AutomotiveSPIN Italy GUFPI-ISMA, AICQ, itSMF Italy) technical associations on such issues. He received a Ph.D in Management Information Systems from LUISS Guido Carli University (Rome, Italy) and a degree in Economics from the University of Rome "La Sapienza", Italy. Further information @ SEMQ website (www.semq.eu). He can be reached at luigi.buglione@computer.org or luigi.buglione@eng.it.

Christiane Gresse von Wangenheim

Dr. **Christiane Gresse von Wangenheim** is a Professor at the Federal University of Santa Catarina (UFSC) and the coordinator of the Software Quality Group at the Brazilian Institute for Research and Technology on Digital Convergence (INCOD). Her research interests are software process improvement, project management and usability engineering. Previously, she worked at the Fraunhofer Institute for Experimental Software Engineering and the UNIVALI. She received a Ph.D. degree in Production Engineering at the Federal University of Santa Catarina (Brazil) and a Ph.D. degree in Computer Science at the University of Kaiserslautern (Germany). She's also a PMP – Project Management Professional and Assessor of the Brazilian Process Improvement Model MPS.BR. She can be reached at gresse@gmail.com

Fergal McCaffery

Dr **Fergal Mc Caffery** is a Lecturer with Dundalk Institute of Technology. He is the leader of the Regulated Software Research Group in Dundalk Institute of Technology and a member of Lero. He has been awarded Science Foundation Ireland funding through the Stokes Lectureship, Principal Investigator and CSET funding Programmes to research the area of software process improvement for the medical device domain. Additionally, he has received EU FP7 and Enterprise Ireland Commercialisation research funding to improve the effectiveness of embedded software development environments for the medical device industry. He can be reached at fergal.mccaffery@dkit.ie

Jean Carlo Rossa Hauck

Dr. **Jean Carlo Rossa Hauck** is a Lecturer at the Universidade do Sul de Santa Catarina and researcher at the Software Quality Group – Brazilian Institute for Research and Technology on Digital Convergence (INCOD). His research interests are in software process improvement and project management. He received his M.Sc. in Computer Science and his Ph.D in Knowledge Engineering from the Federal University of Santa Catarina. He can be reached at jeanhauck@incod.ufsc.br

MADMAPS - Simple and systematic assessment of modeling concepts for software product line engineering

Andrea Leitner, Reinhold Weiß, Christian Kreiner
Institute for Technical Informatics, Graz University of Technology, Austria

Abstract

Domain modeling is a key task in the development of a software product line. We identified two popular modeling paradigms to be predominantly used in practice: feature-oriented domain modeling and domain-specific modeling. The appropriate choice of the modeling paradigm is a crucial decision for the development of an efficient and easy to use domain model.

In order to take such a decision systematically, we propose MADMAPS, a simple method to assess the nature of the domain.

MADMAPS is based on multi-attribute utility theory. The core part is a set of four discriminating criteria describing the characteristics of a domain. The result is either a recommendation for one modeling paradigm, or to split the domain in homogeneous subdomains.

Four use cases have been used to extract assessment criteria, as well as to evaluate MADMAPS. The evaluation is based on the complexity of the resulting domain model. It can be shown that the model complexity with the proposed approach is always lower than the complexity of a model represented by the other approach.

Keywords

Domain modeling paradigm selection, Domain-specific modeling, Feature-oriented domain modeling

1 Introduction and motivation

Software product lines (SPLs) are a viable methodology to improve engineering of software intensive systems. Northrop et al. [1] highlight the importance of a well-structured and documented domain model, since this is the central part of an SPL. But what does this mean? What makes a domain model well structured? We argue that a well-structured representation of the domain model depends on the nature of the domain, and further, that the choice of an appropriate modeling paradigm is a first step towards a well-structured domain model.

The first step before starting to model a domain is to select a modeling paradigm. This decision is often taken implicitly. In some cases, even workarounds become necessary just because the chosen paradigm is not well aligned with the nature of the domain. Currently, the most common paradigms are Domain-Specific Modeling (DSM) and Feature-Oriented Domain Modeling (FODM).

Problems in the domain modeling paradigm selection process during one of our recent projects have been the motivation for this more systematic decision making process. We have extracted several characteristics of domains and derived criteria from them.

Besides having a concise and easy to understand domain model, we aim at an efficient way of product definition and derivation. This is worth striving for, as this step is done hopefully many times in an SPL. Both, domain modeling and product derivation, are heavily influenced by the modeling paradigm. Again, a "fit" paradigm and language to describe variability and variations in a concise manner helps to make this step more efficient.

The contribution of this paper is a simple method for systematic evaluation of a domain and decision support for a specific modeling paradigm. Fig. 1 shows the basic flow of the proposed decision making approach. The input is a domain, which has been bound in a previous scoping step, and the result is a recommendation for one modeling paradigm.

Summarized, we

- take advantage of past research efforts.

- get a simple decision making support, which can be used at an early stage of development.

- define simple criteria that still work fine for a general suggestion.

- use the original paradigm definitions without additional extensions to exploit their advantages (e.g. the advantage of FODA is its simplicity), and to be independent from specialized tool implementations.

Figure 1. Proposed flow for domain modeling paradigm selection with MADMAPS

Sec. 7 summarizes the most important related work. Sec. 2 gives some background information about the underlying concepts. Sec. 3 and Sec. 4 describe the MADMAPS decision making method and underlying theory. Finally, Sec. 5 presents four case studies that have been part of the development process and evaluation. Sec. 8 concludes our work.

2 Background

This section introduces some theoretical background that serves as a base for our approach.

2.1   Introduction to Multi-Attribute Utility Theory

Multi-attribute utility theory[1] (MAUT) can be used as a decision aiding technology, if one alternative from many should be chosen depending on multiple attributes. The most important steps are listed below [2]:

- **Identify alternatives** which should be evaluated (further noted as *alt*).

- **Establish assessment criteria** (attributes) that should be used in the evaluation process. It is advisable to focus only on the most important and relevant criteria.

- **Determine weighting factors** w_{crit}. Each attribute is weighted by its importance. The weighting factors are determined by a pairwise comparison of criteria.

- **Assessment of alternatives** with respect to the defined criteria $c_{crit}(alt)$. For the assessment a scale has to be defined.

- **Calculate utility value**. In this last step, the utility values *u(alt)* for all alternatives alt are calculated:

$$u(alt) = \sum_{crit=1}^{n} (w_{crit} * c_{crit}(alt))$$

Finally, a decision has to be taken. The resulting utility values serve as a quantitative base for this decision.

2.2   Domain modeling paradigms

As mentioned before, we support the decision making for a specific domain modeling paradigm. The alternatives are the two, in our opinion, main modeling paradigms: domain-specific modeling and feature-oriented domain modeling.

2.2.1 Domain-specific modeling

Domain-specific modeling (DSM) aims at the use of a higher level of abstraction and the direct usage of concepts and rules from a specific problem domain. Domain-specific languages (DSL) are used to model a system within that domain. A key characteristic of DSLs is their focused expressive power [3], enabling the generation of products directly from these high level specification [4]. We focus on graphical DSLs in this paper because of the experience and results available from our case studies.

2.2.2 Feature-oriented domain modeling

By feature-oriented domain modeling (FODM) we are mainly talking about feature-oriented domain analysis (FODA) proposed by Kang et al. [5]. This approach has become the basis of many other feature-oriented approaches (e.g. FORM [6]).

"Features are the attributes of a system that directly affect end-users." [5]

For the representation of features, a tree-structured feature model is defined using consists-of relations. The relations are either marked as mandatory, alternative, or optional. Further composition

---

[1] http://ddl.me.cmu.edu/ddwiki/index.php/Multiattribute_utility_theory

rules (e.g. "requires" or "mutually exclusive") are used to express relations between features that cannot be expressed in the tree structure itself [5].

Over the years, several extensions to the original FODA appeared (e.g. cardinality-based feature modeling [7]). These extensions are not investigated here.

3   Multi-attribute domain modeling approach for paradigm selection (MADMAPS)

This section describes a simple method aiding in systematic decision making on which domain modeling approach to choose. The input of our evaluation method is a well-scoped domain.

The setting in our evaluation is slightly different from the original MAUT approach. The investigated alternatives in MADMAPS are FODM and DSM. Defined criteria describe characteristics of domains. The resulting utility value indicates how well the approach fits to the nature of a given domain. So we still compare the alternatives – the domain modeling paradigms – according to a set of criteria defined below.

In contrast to the original MAUT we assess the criteria for each new domain to get a domain specific w_{crit} vector. This vector is then used to calculate the utility values. The $c_{crit}(alt)$ values remain constant as they describe the criteria score of the paradigms.

Identification of alternatives: We focus here on DSM and FODM as described in Sec. 2.2.

Establishing assessment criteria: The first step towards appropriate assessment criteria is the identification and comparison of the main characteristics of the two paradigms. We searched particularly for strong discriminating criteria that are observable at an early stage of SPL setup. A training set of three domains (described in Sec. 5 Case Studies 1-3) was used for criteria definition. Tab. 1 lists the resulting MADMAPS criteria C1-C4. A detailed description of the foundation of this criteria extraction is given in Sec. 4.

Criteria	**DSM**	**FODM**
C1 Fixed relation >= variable relation	4	17
C2 Several instances of elements	31	4
C3 Different binding times/views	8	15
C4 Domain model used by non-expert	8	15

Table 1: Criteria C1-C4 and assessments c_{crit} of domain modeling alternatives DSM and FODA.

Questionnaire for determining weight factors: Following the criteria C1 – C4 several questions have been formulated that we expect to be answerable in very early domain understanding phases. The questions are listed below:

- **Q1:** Are there more fixed relations than variable relations? (Formula 1 in Sec. 4.1) evaluates to true.)

- **Q2:** Should it be possible to use several instances of an element?

- **Q3:** Should there be more than one binding time or more than one view in the domain representation?

- **Q4:** Should the domain model be used by a customer who is not a domain expert? (e.g.: car configurator)

The questions are answered following the Likert scale[2].Tab.2 shows the possible answers and the corresponding weight. This is later used to calculate the utility values and, thus, assess the applicability of the domain modeling paradigm to the specific domain.

Assessment of alternatives Tab. 3 shows the assessment schema used in MADMAPS. These values are used to evaluate the alternatives (DSM and FODM) in respect to the criteria C1-C4. Tab. 1 finally lists the derived assessment values.

Strongly disagree	-2
Disagree	-1
Neither agree nor disagree	0
Agree	1
Strongly agree	2

Table 2: Likert scale for criteria weight-

ing factors w_{crit}.

Poorly or not at all	0-2
Fair	3-5
Good or complete	6-8

Table 3: Assessment schema of criteria fulfillment $c_{crit}(alt)$.

Calculate utility values and decision: Sec. 2.1 describes the calculation of utility values.

As mentioned before, the resulting utility value is only an indicator for a representation paradigm. The final decision has to be taken by a domain expert, but can be based on this systematic assessment of the domain.

4 MADMAPS - Assessment criteria rationale

This section links the chosen criteria to certain characteristics of DSM and FODM, respectively. In the course of our investigations we were looking for discriminating characteristics, which show a strong tendency towards one of the two paradigms.

4.1 Ratio between fixed and variable relations (C1)

A domain model in general consists of elements and relations. Relations can further be divided into fixed and variable relations. Fixed relations do not vary between different products. Variable relations are interesting in this context. Variable relations mean relations that are either defined between two elements where the kind of relation changes, or a relation where the target element changes. For a very rough estimate we define the following indicator for high complexity and variability:

$$|elements| \leq |relations_{variable}| \tag{1}$$

A somewhat easier to grasp indicator for the nature of the domain is the ratio of fixed and variable relations. If for a given domain the following is true, this might be an indicator for a given structure common to all products:

$$|relations_{fixed}| \geq |relations_{variable}| \tag{2}$$

4.2 Instantiation of elements (C2)

As stated in [8], domain objects are good candidates for abstraction. This is an important guideline for the design of a DSL in practice. Generally, there is not just one instance of such objects in the real world. Therefore, it should be possible to instantiate objects in the domain representation as well.

---

[2] http://www.socialresearchmethods.net/kb/scallik.php

DSLs are intentionally designed for instantiation, whereas FODM is not. In the original definition of feature orientation [5] there is no such concept. As stated in Sec. 2.2, an extension to represent instantiation in feature models has been proposed by [7]. This extension has several disadvantages. Moreover, the extension has only been implemented by one prototype tool.

4.3   Different binding times and views (C3)

The variability mechanisms are predefined in the FODM approach and derived and implicitly codified in the language in DSM.

One important differentiating factor seems to be the time when flexibility is useful and needed. This becomes especially important when one wants to represent different abstraction levels or binding times in one domain model. This can be accomplished easier with feature-oriented approaches.

Binding times are defined in the original FODA definition [5]. There are three types: compile-time, load-time, and runtime features. In case of the FODM approach it is easier to describe variability with different binding times, of course only if the generic platform supports those. A DSL need not have an existing code base. Instead, code is generated for each new product model. Therefore, it is harder to introduce different binding times in a DSM approach.

4.4   Target group (Domain model used by non-expert, C4)

Features are end-user visible characteristics of a domain. This means they are an important mean to support the communication between developers and customers. This is another reason why a concise and easy to understand representation is essential. A DSL mostly requires a deeper technical understanding that customers not always possess.

5   *Investigated use cases*

This section introduces and describes four different projects realized with an SPL approach. Each of these projects addresses a unique domain. Thus, the requirements of the projects are quite different.

5.1   Case 1: Configuration of an ERP system

The aim of this project was the systematic reuse of configuration knowledge for Enterprise Resource Planning (ERP) systems of a group of companies [9]. Each of these companies acts in the same industry, they share similar customers and they have similar business processes. The regarded ERP system has to be configured according to the business process of the specific company. Due to the complexity of this configuration process, systematic reuse of configuration knowledge helps to reduce efforts in terms of money and time, and to improve the quality of the ERP instances. Business processes form the SPL architecture.

5.2   Case 2: Fish farm automation

The aims of a fish farm automation system are to make the work of the fish farm owner easier and to save resources. The main functional requirements for a fish farm automation system are feeding and oxygen supervision including an alarm system. In addition, a water level supervision, pH-value measurement, and standard functions like switches and lights have to be realized.

An important characteristic of this domain is the existence of several instances of elements. For example, a fish farm consists of a certain number of ponds. The elements may be assembled in several ways. In fact, the focus is much more on the assembly of elements than on supported

functionality. Normally, no two fish farms look exactly the same. There may be several ponds where each pond may or may not have different supervision and feeding systems.

5.3   Case 3: PLC controlled inventory system

The domain in this case is a logistics system which is built of conveyors, rotary tables, cranes and high bay racking components. With this product line it should be possible to generate automation system software, documentation, etc. for various assemblies of these components.

The arrangement of elements is also more or less variable, which results in a high number of variable relations. The number of elements is very low, since there are only the aforementioned components available. However, these elements may be instantiated several times. The selection of different functionality is not required. Different binding times and abstraction levels are not important here as well. Fig. 2 shows a sample application model.
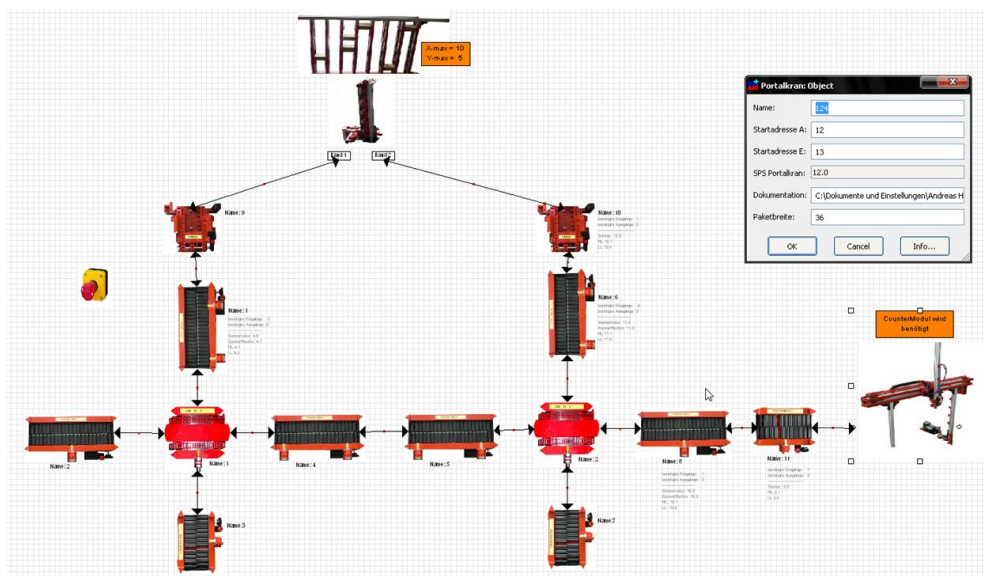
Figure 2: Sample application model for a PLC controlled inventory system (realized with MetaEdit+ DSL).

5.4   Case 4: Control unit for an HEV

The target in this project[3] is the development of a generic architecture for a hybrid electrical vehicle (HEV) [10] control unit. Since this is an embedded system, there are many connections to the environment (i.e. the overall system). This makes the domain very complex.

In the domain analysis phase we faced the problem that many criteria seem to be very important and necessary in some part of the domain, or another. There is certainly a focus on functionality, since the control unit provides functionalities. On the other hand, there is also a focus on assembly because the provided functionality is dependent on the layout of the drivetrain (e.g. full electric drive is only possible with a clutch between electric motor and combustion engine).

The result is not as clear as in the case studies above. Both utility values are positive and there is no real indication which approach is appropriate. Following the flow described in Fig. 1 we split domain into several subdomains, one for each viewpoint. The viewpoint identification resulted in a software

---

[3] http://www.iti.tugraz.at/hybcons

view, an ECU view, a mechanics view and a safety view. In the next iteration step MADMAPS was applied to each of the sub-views.

We applied MADMAPS to the software view, and to the mechanics view representing the drivetrain topology. Now, the situation looks quite different. For the software there is no need for several instances of elements or different assemblies of elements anymore and the drivetrain (mechanical view) can be described in a graphical representation. Fig. 3 shows an example for both representations. Due to the heterogeneity of subdomains a combined multi-modeling representation has been proposed in [11].
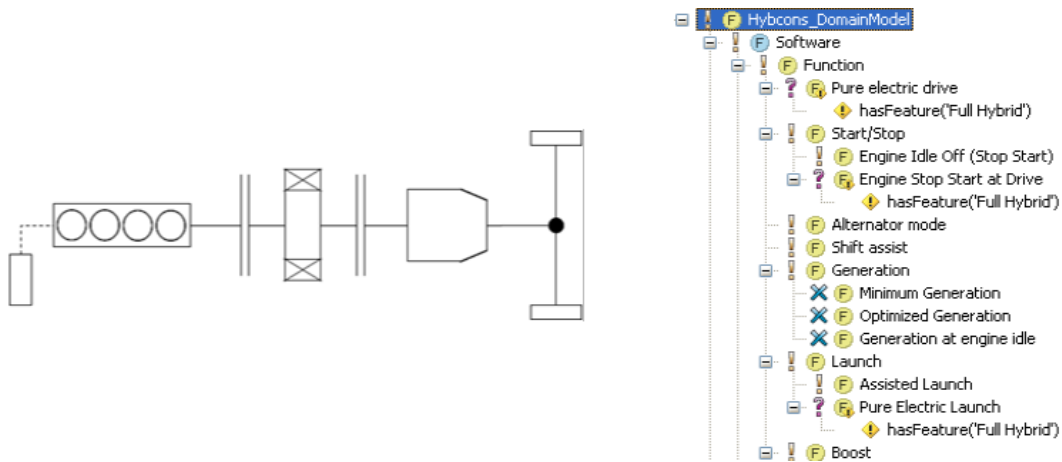
Figure 3: Sample application model for a hybrid electrical drivetrain (ECore-based DSL) and a sample software feature model (pure::variants).

6 Lessons learned

Complexity: Previously [12], we defined metrics to evaluate the resulting domain model. One useful quality metric is the complexity of the resulting model. Keeping complexity as low as possible is important in several aspects. First of all, it improves the usability and maintainability of the domain model. An investigation of the described domains shows that the modeling paradigm suggested by MADMAPS always results in lower complexity.

MADMAPS results: Tab. 4 shows an overview of the resulting utility values. Additionally, the resulting complexity values (CV) are given for both paradigms in order to verify the results. The utility values show that the approach gives a clear recommendation for all our test domains. For Case 4 we

Case	Domain	DSM	FODM	Recommended	Complexity (DSM/ FODM)
1	ERP system	-62	11	FODM	257/167
2	Logistics system	46	-41	DSM	65/ -
3	Fish farm automation	38	-56	DSM	70/ -
4	HEV CU (control unit)	55	49		78/80
	HEV CU-Software	-38	56	FODM	30/25
	HEVCU-Mechanics	11	-24	DSM	22/36

Table 4: MADMAPS utility values and complexity values for the case studies described.

interviewed experts working with this domain models and asked them how well they are satisfied with the representation and what they like best. An evaluation of this interview showed that the experts agree and welcome the combination of the two different representations. In particular the graphical representation for the mechanics part seems to be much more suited than a feature oriented approach.

General considerations: Why should one use our approach? First, there is no real overhead because all the information necessary to answer the questionnaire is gathered during domain analysis. Domain analysis is an integral part of the domain engineering process and has to be performed anyway. Due to the simple and abstract formulation of the questions, knowledge from domain analysis should be enough to answer them. One benefit is that the proposed criteria can be used during domain analysis to have a structured guideline how to investigate the domain. The big advantage however is that the decision is grounded on a systematic base.

7 Related Work

The importance of the appropriate representation in DSML specifications is mentioned in [13]: *"the correct representational paradigm depends on the audience, the data's structure, and how users will work with the data"*. We extend this statement to the next level and argue that not only the representation is important, but also the paradigm to create this representation in an effective way.

Some authors have investigated the differences between modeling approaches. In [14], feature-oriented domain analysis approach has been compared to Organization Domain Modeling (ODM). For the authors these seemed to be the most important approaches. ODM is also based on features. The feature definition is more general than in FODA. One major difference is that ODM postulates the need for a flexible architecture. It is stated that a generic architecture is not suitable for domains with a high degree of variability. This statement is similar to our observation that DSM is better suited for domains with many variable relations, because of a flexible architecture.

Czarnecki [15] investigates the relation between feature models and ontologies. The major conclusion from his work is that extensions of the original feature oriented approach are used to bring it closer to the expressiveness and formalism of ontologies. As stated before, these extensions are at the expense of the simplicity, which is a major advantage of feature models. Furthermore, the authors propose to combine FODM and DSM, which confirms our observations that it is sometimes not enough to model the entire domain with one representation. In contrast to our research, they do not split domains. Instead, two approaches are used to represent the same content in different views.

Haugen et al. [16] describe a separated language approach to specify variability in DSL models. They propose a Common Variability Language (CVL) and according variability resolution mechanisms embedded in the OMG metamodel stack. This allows to describe variability in potentially all MOF-based languages, including UML, as well as MOF-and UML-profile based DSLs. While being a general and clean approach to handle variability, it does not seem directly applicable to feature abstraction hierarchies and their complex constraints.

8 Conclusion

This work introduces a simple approach for the systematic selection of an appropriate domain modeling paradigm. Four criteria have been extracted from characteristics of the methods investigated -feature oriented domain modeling and domain-specific modeling. We use these criteria to assess the nature of a domain in respect to a systematic selection of a modeling approach.

The major advantage of the MADMAPS approach is a decision which is based on a systematic method. As a result we can show that the use of the proposed modeling paradigm always results in a domain model with lower complexity.

Acknowledgment

9   Literature

1. L. Northrop and P. Clements, Software Product Lines: Practices and Patterns. Addison Wesley, 2002.

2. J. W. Altschuld and B. R. Witkin, From Needs Assessment to Action: Transforming Needs into Solution Strategies, 1st ed. Sage Publications, 2000.

3. A. van Deursen, P. Klint, and J. Visser, "Domain-specific languages: an annotated bibliography," SIGPLAN Not., vol. 35, no. 6, pp. 26–36, 2000.

4. S. Kelly and J.-P. Tolvanen, Domain-Specific Modeling: Enabling Full Code Generation. Wiley-IEEE Computer Society Pr, March 2008.

5. K. C. Kang, S.G. Cohen, J.A. Hess, W. E. Novak, and A. S. Peterson, "Feature-Oriented Domain Analysis (FODA) Feasibility Study," Carnegie-Mellon University Software Engineering Institute, Tech. Rep., November 1990.

6. K. C. Kang, S. Kim, J. Lee, K. Kim, G. J. Kim, and E. Shin, "FORM: A feature-oriented reuse method with domain-specific reference architectures," Annals of Software Engineering, vol. 5, pp. 143–168, 1998.

7. K. Czarnecki and C. H. P. Kim, "Cardinality-based feature modeling and constraints: A progress report," in Proc. of the Int. Workshop on Software Factories, 2005, pp. 16–20.

8. S. Thibault, "Domain specific languages: Conception, Implementation and Application," Ph.D. dissertation, l'Universite de Rennes 1, 1998.

9. A. Leitner and C. Kreiner, "Managing ERP configuration variants: an experience report," in Proc. of the 2010 Workshop on Knowledge-Oriented Product Line Engineering, ser. KOPLE '10. New York, NY, USA: ACM, 2010, pp. 2:1–2:6.

10. M. Ehsani, Y. Gao, and A. Emadi, Modern Electric, Hybrid Electric, and Fuel Cell Vehicles: Fundamentals, Theory, and Design, 2nd ed. CRC Press, 2010.

11. A. Leitner, C. Kreiner, R. Mader, C. Steger, and R. Weiß, "Towards multi-modeling for domain description," in 2nd Int. Workshop on Knowledge-Oriented Product Line Engineering: KOPLE 2011, ser. SPLC'11, 2011, pp. 1–6.

12. A. Leitner, R. Weiß, and C. Kreiner, "Analyzing the complexity of domain models," in ECBS12, Proc., 2012.

13. S. Kelly and R. Pohjonen, "Worst Practices for Domain-Specific Modeling," Software, IEEE, vol. 26, no. 4, pp. 22 –29, 2009.

14. K. Czarnecki and U. Eisenecker, Generative Programming: Methods, Tools, and Applications. Boston, MA: Addi-son-Wesley, 2000.

15. K. Czarnecki, C. H. Peter Kim, and K. T. Kalleberg, "Feature Models as Views on Ontologies," in SPLC '06: Proc. of the 10th Int. on Software Product Line Conference. Washington, DC, USA: IEEE Computer Society, 2006, pp. 41–51.

16. O. Haugen, B. Moller-Pedersen, J. Oldevik, G. Olsen, and A. Svendsen, "Adding Standardized Variability to Domain Specific Languages," in Proc. of the 12th International Software Product Line Conference, 2008, pp. 139 –148.

10 Author CVs

Andrea Leitner

Andrea Leitner received her Dipl.Ing. (MSc) from Graz University of Technology in 2009. Since 2010 she is working on her Phd at Graz University of Technology. Her main research fields are Software Product Lines, variant management and knowledge oriented software engineering, with a strong focus on the needs in automotive industry.

Reinhold Weiß

O.Univ.-Prof. Dr.techn. Reinhold Weiss is a Professor of Electrical Engineering (Technical Informatics) and head of the Institute for Technical Informatics at Graz University of Technology, Austria. He received the Dipl.-Ing. degree, the Dr.-Ing. degree (both in Electrical Engineering) and the Dr.-Ing.habil. degree (in Realtime Systems) from the Technical University of Munich in 1968, 1972 and 1979, respectively. In 1981, he was as a Visiting Scientist with IBM Research Laboratories in San Jose, California. From 1982 to 1986 he was Professor of Computer Engineering at the University of Paderborn (Germany). He is author and co-author of about 170 scientific and technical publications in Computer Engineering. For e&i (Elektrotechnik & Informationstechnik, Springer-Verlag) he served several times as a guest editor for special issues on Technical Informatics and Mobile Computing, respectively. In 2001 and 2003, he organized two Workshops on Wearable Computing. His research interests focus on Embedded Distributed Real Time Architectures (parallel systems, distributed fault-tolerant systems, wearable and pervasive computing). He is a member of the International Editorial Board of the US- journal ''Computers and Applications" (ISCA). Further, he is a member of IEEE, ACM, GI (Gesellschaft für Informatik, Germany), OVE (Oesterreichischer Verein für Elektrotechnik, Austria).

Christian Kreiner

Dr. Christian Kreiner is key researcher in the field of model-based design and Software Product Line Engineering of embedded and industry automation systems. He received his Dipl.Ing. and Dr.techn. from Graz University of Technology in 1991 and 1999. From 1999-2007 he served as RD manager at Salomon Automation, a logistics software company. During that time, he led and coordinated a long-term research programme together with the Institute for Technical Informatics of Graz University of Technology.
His interests are in the fields of software product lines, model based design, runtime model execution, and agile processes in industry, with a strong focus on the needs in automotive industry.

Maturity assessment and process improvement for information security management in SMEs

Hervé Cholez and Frédéric Girard
Public Research Centre Henri Tudor
29 avenue John F. Kennedy, L-1855 Luxembourg
{herve.cholez, frederic.girard}@tudor.lu

Abstract

Information security is a central concern inside organisations, but it remains quite difficult for most small entities to implement and maintain information security. In this context, the Public Research Centre Henri Tudor and the Luxembourg's Ministry of Economy and Foreign Trade decided to enhance information security awareness and management in Luxembourg's SMEs. Therefore, our research work aims to propose a method adapted to SMEs to conduct a first assessment of the enterprises information security maturity and improve their process accordingly. This paper describes the framework developed and presents its validation in industry. The results of applying the method in industry are positive and show a lack in organisational maturity for the information security. The future challenge of this assessment method is to be integrated into an information security web platform and use the large amount of statistics to continuously improve and contextualise the proposed tool.

Keywords

Information security, maturity assessment, process improvement, SME

1 Introduction and research questions

Information Systems (IS) are everywhere and their roles are becoming larger due to the huge increase of the amount of information and ideas exchanged during the last decades. Information system security became a central concern inside organisations as these systems are now playing a vital role in the organisations' life. There is currently a strong need for a managed and reliable information security not limited to technical solutions. Moreover, information security must be adapted to each organisation context to match their business goals, but selecting security measures that suit organisation security needs is a challenging task. Establishing an Information Security Management System (ISMS) improves information security through the embedded business risk approach. Although ISMS are being adopted by large companies, it remains quite difficult for smaller entities to implement and maintain information security or even simply pay attention to this domain. This is particularly the case for the SMEs [1].

In Luxembourg, 95% of the country's enterprises are SMEs[1] and they represent 80% of the Luxembourg GDP (Gross Domestic Product) [2]. They are, therefore, economically significant. About information security, very few Luxembourgish SMEs establish an ISMS since it is still a long and costly process. SMEs need an initial affordable way of introducing information security and its management quickly and efficiently within their enterprise processes.

The Public Research Centre Henri Tudor in collaboration with the Luxembourg's Ministry of Economy and Foreign Trade decided to support the improvement of SMEs' information security including technology and management. From this perspective, our research work proposes a method specifically designed for the SMEs. The method aims to conduct a first assessment of an enterprise information security maturity and consequently improve its process accordingly. This paper answers the following research questions:
   1. How to evaluate the information security maturity of SMEs quickly and efficiently?
   2. How to introduce information security awareness in the SMEs?
   3. How to initiate and improve the information security management in the SMEs?

Section 2 of this paper presents the framework established to answer these questions. Section 3 is about the case studies performed and the lessons learned. Section 4 describes the planned case studies and summarises the future works. Finally, conclusions are drawn in the Section 5.

2 Framework presentation

This section presents the framework that was developed in the research project together with a short history. The framework is composed of concepts, a method, and a tool. The different concepts and objectives are presented before showing the method with a brief and concrete illustration of the tool.

2.1 History

The concept of a maturity assessment for the SMEs, also known as micro-assessment, was initiated by the Public Research Centre Henri Tudor and the University of Namur. At first, the aim was to evaluate the overall maturity level of software practices in SMEs [3]. First case studies were conducted in 1998 in Belgium [4]. After 32 evaluations and 7 re-evaluations between 1998 and 2004, approximately 25 organisations used this tool each year since 2004 in Quebec, Belgium, Spain, and France [5] [6]. Considering the success of this micro-assessment for software practice maturity, the Public Research Centre Henri Tudor decided to extend this concept to the maturity assessment for information security and its management.

---

[1] Small and Medium Enterprises: organisations that contain up to 250 employees.

2.2 Concepts

The main goal of the proposed framework is to assess the level of the information security maturity and to quickly provide an overview of the information security in enterprises. This assessment allows increasing enterprise awareness of information security and of the importance of management for the information security. As a result of the assessment, recommendations and practical measures are given to initiate or improve the information security management process in enterprises.

This framework focuses on SMEs as we assume they are a specific use case relevant for Luxembourg. In this article we will use the "SME" and "enterprise" terms as synonyms, but behind these terms we consider all types of organisations (e.g., associations, public institutions, etc.) up to 250 employees. Based on our previous works [7] [8], SMEs present strengths in the ability to reorganise their processes. SMEs are also flexible and reactive. However, they have one problem: they are limited in their resources in terms of money, time, and skills. In addition, SMEs often have a poor information security culture and believe to be not concerned by information security and the associated risks. The vocabulary of standards and large frameworks in information security is specialised and complicated in comparison to the reality of small enterprises practices [1]. These standards are thus often not accessible for the SMEs. These specific characteristics of the SMEs give multiple requirements and challenges for our framework design such as downsizing the standards and integrating these standards in a comprehensive and usable form to enterprises that are not familiar with them.

Applying this framework would be the starting point of consideration of information security and its management in the SMEs. The framework is thus kept light to be easily applied by the SMEs. In the assessment results it is essential to show the usefulness of efforts in this domain and to expose opportunities with potential next phases in regard to the SME business strategy.

To respect the SMEs characteristics, we deliberately designed a light method understandable to people who are not information security experts. As the financial and human resources in this kind of enterprises are limited, the method consumes a small amount of resources and represents a good value for money and time invested. In this context it is not relevant to propose advanced audits and on-site verifications as recommended by the standards. Our proposition is to base the method on the people's declaration since lying and cheating in the assessment is counterproductive for the enterprise. Nevertheless, it is important to focus on trust-building to guarantee more honest and reliable results during the interview.

SMEs have a range of heterogeneous profiles (e.g., context, sector, size, objectives). That is why this assessment is based on an adaptive method able to address any SME. A preliminary adaptation is done through the selection of interviewees. Other adaptations are performed with the questions' vocabulary and the depth of details requested during the interview. The framework is reusable to allow multiple assessments and successive comparisons over the time of the assessment results to observe the improvement, supported by the analysis and comparison of the final results' graphics depicting the enterprise profile.

Regarding the results, the final report has to be clear and concrete to be comprehensive, useful, and applicable to SMEs. Hence, it is important to generate and display graphics to provide a quick and clear visual overview of the SME information security state. Moreover, realistic short and mid-term recommendations are provided.

2.3 Method and tool presentation

As explained in Section 2.2, the method consists of interviews with the appropriate stakeholders that will provide the most accurate answers on the effective practices in the enterprise. To adapt the method to the enterprise needs and focus on most relevant aspects in the questionnaire, it is essential to understand the context of the SME. Therefore, an introduction phase and the selection of interviewees are crucial as the micro-assessment represents their point of view. To limit the impact on enterprise resources, one interviewee is generally sufficient to keep cost and effort at their lowest level. However, depending of the context, two stakeholders at the maximum are interviewed, especially if

one is specialised in technological aspects and the other one in organisational aspects. Obviously, a limited number of interviewees results in a restricted view of the enterprise practices [3], but in the SMEs context very few employees are competent in the information security domain.

The interview is based on a questionnaire composed of two sections. The first section focuses on the organisational side, while the second section concentrates on operational aspects. The whole questionnaire contains 27 open-questions. Each question is completed by sub-questions aimed at guiding the conversation and giving a better comprehension to the interviewer of the real enterprise practices. These sub-questions allow approaching the same question from different angles to adjust and refine information by rephrasing with different vocabularies. This set of questions and sub-questions is based on experience and on standards such as ISO/IEC 27001 [9] and ISO/IEC 27002 [10].

The ISO/IEC 27001 standard provides requirements to establish, monitor, maintain, and improve an ISMS. Relying upon quality management and ISO 9001 [11] principles, the standard following the PDCA[2] cycle, aims to continuously improve information security. This standard is the basis used to build the structure of the organisational section of the questionnaire following the PDCA cycle. However, to target the SMEs, the questions inside this structure are adapted to the SMEs' language and knowledge.

The ISO/IEC 27002 standard provides best practice recommendations on information security management. In the same manner as ISO/IEC 27001, our framework uses the structure of the ISO/IEC 27002 chapters to design the operational section of the questionnaire and adapt the sub-questions to the SMEs' context. An example from the assessment questionnaire is presented in the Figure 1. It is extracted from the operational section.
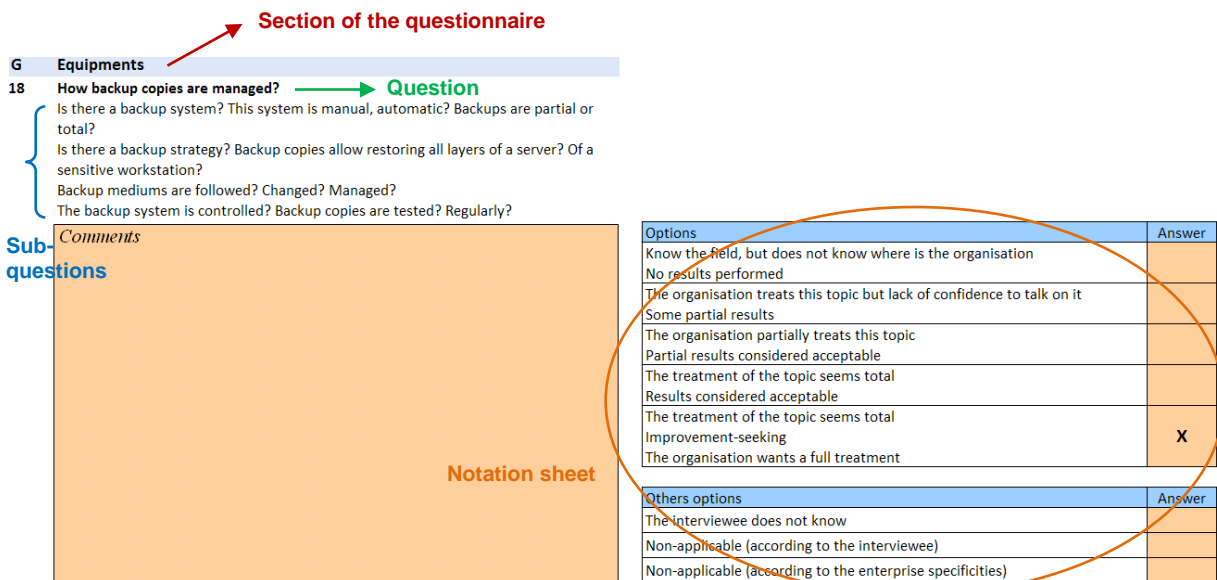
Figure 1: Example of a question from the assessment questionnaire

The interviewer follows the questionnaire and evaluates each question for the maturity level of the underlying activity on a 5-level scale ranging from 0 to 4 (as presented in the Figure 1). Increasing numbers denote increasing maturity. It is important to note that there are also three other possible answers to each question: "1. The interviewee does not know", "2. Non-applicable (according to the interviewee)", and "3. Non-applicable (according to the enterprise specificities)". These three additional answer possibilities lead to understanding why there is no security treatment in place in order to propose relevant improvements. The evaluation is performed by the interviewer and the interviewee is not aware of the ratings during the interview. This interview takes approximately a couple of hours. Finally, the answers of the whole questionnaire allow analysing the enterprise practices and assessing its maturity on different axes.

---

[2] PDCA: Plan, Do, Check, Act (also known as the Deming wheel) – An iterative management method used for continuous improvement of processes and products.

The maturity scale and the assessment method are inspired from the standard series on Process Assessment: ISO/IEC 15504 [12]. It provides requirements to conduct a process assessment and to design process models; guidelines for process improvement or capability determination; and exemplar process models. The ISO/IEC 15504 defines a 6-level capability scale from "incomplete process" to "optimising process". In regard to our SMEs context and their low maturity, this scale is adjusted. The level 4 (the higher) of the micro-assessment is comparable to the first capability level of the ISO/IEC 15504: results are performed with improvement-seeking. At the opposite side, the level 0 implies that absolutely nothing is performed, which is under the level 0 of the ISO/IEC 15504. In order to maintain consistency among assessment and ensure validity of collected data, the definition of each level is clear, unambiguous, and distinctive. Moreover, a level must be fully reached to be on the next level.

At the end of an interview, visual graphics of the enterprise maturity are generated (see Figure 2 and 3) to provide an immediate feedback. The graphical method employed is the radar chart [13] since it is visually striking and suitable to SMEs. These figures illustrate the organisational and operational profiles of a real case. The solid line represents the enterprise actual level in information security maturity and the dotted line is the goal to reach. By default, this dotted line is set at the level 3.5 on 4 everywhere. Figure 2 shows the organisational profile. A PDCA structured view with transversal practices is shown on the left while the graphic on the right exploits the detailed view inspired by the ISO/IEC 27001 chapters. In the same manner, Figure 3 depicts the operational profile with a detailed view on the right (based on a selection of ISO/IEC 27002 sub-sections) and it is summarised on the left graphic with concepts and vocabulary understandable by SMEs language and structure.

2.1: PDCA structured view   **2.2: Detailed view**
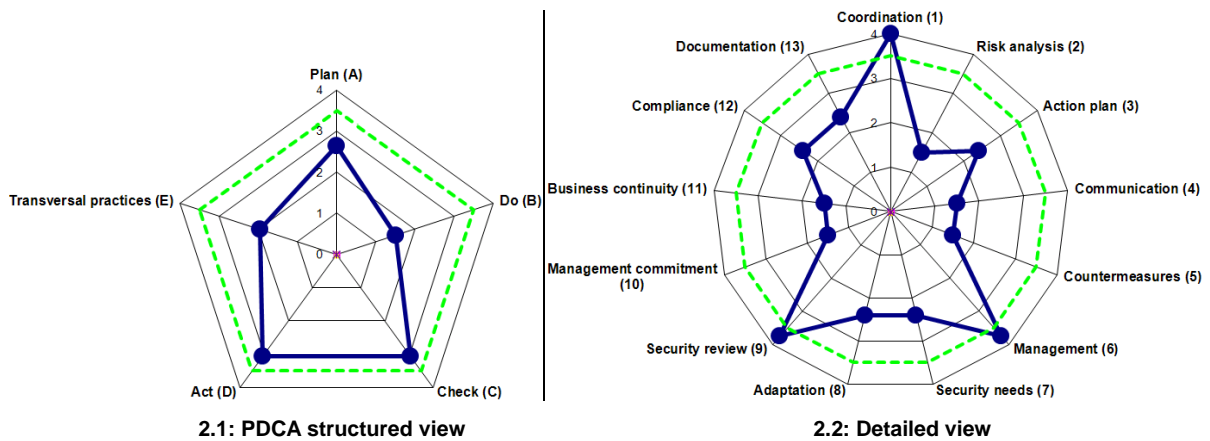
Figure 2: Example of an organisational profile produces after the assessment

3.1: Overview   **3.2: Detailed view**
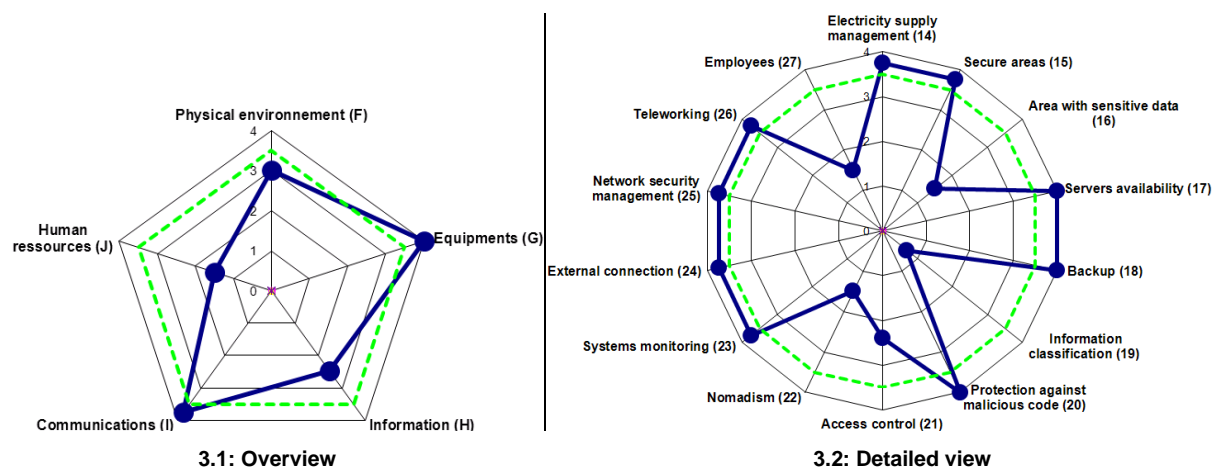
Figure 3: Example of an operational profile produces after the assessment

Finally, the summary of the results are drawn up in a report with a detailed analysis. However, this analysis must stay simple, concrete, and usable. A typical report first briefly presents the method. Next, the results of the questionnaire are presented and described in greater detail. For being clear

and concise, the identification of the enterprise SWOTs[3] in some bullets points is made to summarise and emphasise the observations. Last, the report provides a list of short and mid-term recommendations about organisational and operational practices aligned with the observations and interview conclusions. This list is limited to few contextualised reachable and realistic objectives to not discourage the SME from improving.

3 Case studies

The case studies to validate the framework developed in the research project are presented, and then the lessons learned are exposed.

3.1 First case studies

Six case studies were conducted in 2007 in Luxembourg SMEs to develop and adjust the maturity assessment framework in information security. After this period, a first refinement of the approach based on the analysis of the feedback and results were done to feed other methods, products, and projects related to the SMEs. In 2010, in order to answer to the market needs and requests, the micro-assessment was proposed by the CRP Henri Tudor as a commercial service. Since this period, seven other case studies were conducted and some of them by external IT services companies.

For a classical micro-assessment, approximately two man-days of work divided into four phases are necessary. The concepts of the method are briefly presented to the SME and the selection of the interviewees is carried out in the first step. Secondly, the interview is performed during a couple of hours. Afterwards, the interviewer writes the report and finally, the results are presented to the enterprise. Mobilising only one employee of the SME during three hours is usually enough to conduct the whole process (introduction, interview, and results presentation). In general, the interviewee is the information system manager or the person responsible for the information security. Sometimes, two people are preferred if organisational and operational responsibilities are shared by different employees. In such a case, it is necessary to extend the interview duration to three hours.

Lots of different kinds of organisations have been assessed in various sectors from 2007 to 2012. In very small organisations (up to 10 employees), the micro-assessment consumes only one man-day of work. We have also extended case studies to larger organisations (up to 3500 employees) that have a low maturity level in information security. Case studies results were positive and the method was suitable. In one particular case, the interview length was six hours instead of three due to the enterprise size and the level of details requested by the client. In these extreme cases, the total time required to achieve the micro-assessment is around four man-days of work.

3.2 Lessons learned

Each assessment gives opportunities to improve the method and to refine the questionnaire. Furthermore, in all cases, this micro-assessment brings benefits since the awareness on information security is a first main result on SMEs. In addition, according to positive feedback, the micro-assessment achieves its role of a first information security approach for the SMEs. The micro-assessment brings an overall structure, an estimation of the workload, and the stakes attached to this domain for the SMEs to deal with information security. SMEs will not obtain a high-level information security maturity in two hours: this method is only a starting point. Enterprises usually take into account urgent recommendations and initiate a long-term consideration of their information security. This is exactly the purpose of the micro-assessment. In addition, organisational and strategic reflections give inputs for enterprises beyond the information security domain.

The primary targets of our framework are SMEs. However, several tests of the method were con-

---

[3] SWOTs: Strengths, Weaknesses, Opportunities, and Threats

ducted with success in large enterprises and branch offices. In the same manner, some SMEs have a high maturity level in information security and therefore are not in our scope. Consequently, a new goal is to extend the framework to all organisations with a low maturity level in information security and its management.

Another result of these case studies is the fact that in absolutely all the assessed enterprises the organisational maturity level is substantially lower than the operational maturity. Actually, some activities around information security are often in place, but nothing is prepared, planned, or organised. It is generally considered a waste of time by SMEs. This provokes very low results in regard to the enterprise needs and sometimes it is counterproductive. The micro-assessment framework introduces the organisational view in an innovative way to show the power of such comprehensive reflections. Slight organisational improvements bring lots of concrete advantages since actions are adapted to the overall enterprise strategy. Therefore, reports generally encourage enterprises to make efforts in this domain.

4 Further case studies and upgrades

Case studies results show several opportunities to improve the micro-assessment method and the tool. As a first improvement, the framework will be adapted to be fully compliant with the ISO/IEC 27002 while staying at a comprehensive level for the SMEs. It is also essential to keep this method as a micro-assessment and as a first approach of information security. Consequently, the set of questions will evolve to focus on the identification of critical problems and associated recommendations. This micro-assessment could also become compatible with more systematic assessment approaches.

The Public Research Centre Henri Tudor in collaboration with the Luxembourg's Ministry of Economy and Foreign Trade initiated a research project aiming at offering a large panel of innovative services in the information security domain. These services will be integrated into a web platform built on the SaaS (Software as a Service) model and managed by the GIE Smile[4]. This platform will provide a better access and visibility to information security through centralised, simplified, and coherent services of high quality since service providers' competencies will be guaranteed with certification. Moreover, the platform takes part of the Luxembourg national policy to deploy and improve the information security in local organisations. This large diffusion will provide experience and feedback to continuously improve the offered services.

The micro-assessment will be integrated into a large set of coherent and complementary services linked together. Due to the high visibility, the platform will provide a large panel of case studies, and then a pool of data enabling statistics. This will bring a significant added value in information security and for the SMEs. The main future challenge for the micro-assessment is to set up an ontology that defines groups of organisations that share similar information security issues and objectives. Using this ontology and data from the web platform, the method will become highly adaptable to various contexts. For instance, the dotted line of the Figure 2 and Figure 3 that represents the goal to reach for the information security maturity of the SME will be continuously and automatically refined for each SME group in order to have relevant objectives according to their context.

5 Conclusion

This paper reports on the case studies conducted in the frame of a research project about maturity assessment and process improvement of information security management in SMEs. We have first introduced the research questions. We then have presented the framework by depicting the concepts, the method, and the tool. Next, the case studies were described with an analysis of the lessons learned. Finally, we have described the future work of this method.

The conclusions drawn from the case studies are first that our method is relevant for an SME context due to very small resource consumption. The micro-assessment is a very attractive tool for the SMEs

---

[4] GIE Smile: Security made in Lëtzebuerg - Economic interest group

to quickly and efficiently evaluate their maturity level in information security with an optimum return on investment. The method allows to introduce information security awareness in the SMEs and initiates the first information security management process in these organisations. The positive feedback from the SMEs has motivated us to pursue this work.

Further improvements with the on-going project and the SaaS platform will strengthen the scientific base of the method. Having an easy access to numerous Luxembourg SMEs will give a new dimension to the method with sectorial adaptations and continuous improvements. In addition, as shown in this research work, this framework could be applicable to enterprises of all sizes and extended to other sectors than information security.

6 Literature

1. Y. Barlette and V.V. Fomin. "Exploring the Suitability of IS Security Management Standards for SMEs". In Proceedings of the 41st Annual International Conference on System Sciences. Hawaii, USA. 2008.

2. S. Poggi. "Analyse de la structuration économique des PME et TPE au Luxembourg", CASES, 2005. Retrieved in: http://www.cases.public.lu/fr/publications/recherche/r2sic/WP51.pdf

3. N. Habra, A. Renault, S. Alexandre, and M. Lopez. "OWPL - Micro-Assessment". In Proceedings of the Workshop on Software Quality. International Conference Software Engineering, Orlando, Florida, USA, May 2002.

4. N. Habra, E. Niyitugabira, A.-C. Lamblin, and A. Renault. "Software process improvement in small structures: First results of a micro-assessment framework". In Proceedings of the European Conference on Software Process Improvement, Barcelona, Spain, 1999.

5. N. Habra, S. Alexandre, J.M. Desharnais, C.Y. Laporte, and A. Renault. "Initiating software process improvement in very small enterprises - Experience with a light assessment tool". Information and Software Technology, 50, 7-8, 763-771, 2008.

6. S. Alexandre, A. Renault, N. Habra, "OWPL: A gradual approach for software process improvement in SMEs". Proceeding of 32nd Euromicro Conference on Software Engineering and Advanced Applications (SEAA), 2006.

7. T. Valdevit, N. Mayer, and B. Barafort. "Tailoring ISO/IEC 27001 for SMEs: A Guide to Implement an Information Security Management System in Small Settings". In Springer, editor, Proceedings of the 16th European Systems & Software Process Improvement and Innovation Conference (EUROSPI'09), 2009.

8. N. Mayer. "A Cluster Approach to Security Improvement according to ISO/IEC 27001", 17th European Systems & Software Process Improvement and Innovation Conference (EUROSPI'10), Grenoble, France, September 2010.

9. ISO/IEC 27001:2005. Information technology - Security techniques - Information security management systems - Requirements. International Organization for Standardization, Geneva, 2005.

10. ISO/IEC 27002:2005. Information technology - Security techniques - Code of practice for information security management. International Organization for Standardization, Geneva, 2005.

11. ISO 9001:2000. Quality management systems - Requirements. International Organization for Standardization, Geneva, 2000.

12. ISO/IEC 15504:2003. Software engineering - Process assessment - Part 2: Performing an assessment. International Organization for Standardization, Geneva, 2003.

13. J. M. Chambers, W.S. Cleveland, B. Kleiner, and P.A. Tukey. "Graphical Methods for Data Analysis", Monterey, CA: Wadsworth. 1983.

7 Author CVs

Hervé Cholez

Hervé Cholez received his M.Sc. degree in computer science, specialised in the security of information and communication systems, from the University Paul Verlaine, Metz, France, in 2008. He is currently a research and development engineer at the Public Research Centre Henri Tudor (Luxembourg) in the Business Process and Service Engineering unit. His research interests are mainly in the fields of security risk management, information systems security, and information security standardisation. Recently, he contributed to research projects related to computer-based assessment information security and to cloud computing security risk management.

Frédéric Girard

For more than 10 years, Frédéric Girard has specialised in information security and its management. He started his career as security consultant on technical and architecture issues of various types of organisation, from multinational companies to SMEs.

The experiences in this field allowed Frédéric to evolve from the consultancy to the R&D world. Frédéric Girard is currently an R&D project leader in information security at the Public Research Centre Henri Tudor. He tries out new approaches for drawing business processes, designing new services, and tries to predict future behaviours of users. He is a coach for decision makers with the aim to increase the level of maturity of their organisations and their strategies linked with information security.

Frédéric covers training, information, and networks aspects by being co-Director of the Master ISMS Luxembourg (Professional Master in Information Security Management Systems), and General Secretary of CLUSIL (the network of information security professionals at Luxembourg).

Improving the Outsourcing Contracts of Software and Services through the Contract Engineering

Mejia Jezreel[1], Muñoz Mirna[1], Calvo-Manzano Jose A.[2], Cuevas Gonzalo[2], San Feliu T.[2]
[1] Centro de Investigación en Matemáticas
Avda. Universidad no. 222, 98068, Zacatecas, México
jmejia@cimat.mx, mamunoz@cimat.mx
[2] Facultad de Informática, Universidad Politécnica de Madrid
{joseantonio.calvomanzano, gonzalo.cuevas, tomas.sanfeliu}@upm.es

Abstract

Efficient contracts help to establish beneficial relationships between the acquirer and provider. Therefore, if an organization decides to acquire software and services products, the contract is a fundamental mechanism to ensure that expectations are realized. This paper shows the lacks in the outsourcing contracts in order to identify its improvements through Contract Engineering. Contract Engineering will try to avoid failures in the relationships between acquirers and providers caused by a bad definition of the contract. Contract Engineering provides: 1) a workflow that includes aspects or activities that should be considered before signing an acquisition contract, 2) A contract model proposal that describes the main contract's clauses, and 3) An evaluation method that allows verifying the contract coverage in order to select the best. Besides, a case study is presented.

Keywords

Contract Engineering, CMMI-ACQ, Outsourcing Contract model, Contract Clauses.

1 Introduction

Information Technology (IT) services outsourcing market has grown rapidly every year [4, 10, 14, 15, 20, 21, 28]. However, while the outsourcing is experiencing a considerable growth, the number of reported cases of failure is also increasing [14]. According to a study from the Software Engineering Institute (SEI) [25], 20 to 25 percent of IT acquisition projects fail within two years and 50 percent fail within five years. Important lacks in the contracts is a factor that contribute to project failure [25, 29, 30, 31,32]. The majority of project failures could be avoided if the acquirer learns how to prepare or evaluate properly the contracts [25].

According to an analysis of 40 organizations from different sectors [12, 13], problems encountered with outsourcing contracts prior to renegotiation often stem from misunderstandings between the acquirer and the service supplier. In addition, the organizations cited that a major impediment to a higher degree of success in IT outsourcing projects is the lack of knowledge or expertise in the development and structuring of the initial acquisition or outsourcing agreement or contract with a supplier.

Contracts are [22] "a framework which almost never accurately indicates real working relations, but which affords: 1) a rough indication around which such relations vary, 2) an occasional guide in cases of doubt, and 3) a norm of ultimate appeal when the relations cease in fact to work". Moreover, a contract is considered to be the only means to guarantee the expected achievement, and also the primary means to explain the acquirer-provider relationship [12, 16].

Efficient contracts structure the relationship between the acquirer and the provider in ways that are beneficial to both [16]. An important principle for IT outsourcing was formulated by Beulen and Ribers [3]: "If a company decides to outsource, the contract is the only mechanism to ensure that expectations are achieved".

The purpose of this paper is to present a analysis of results related to outsourcing contracts of Software and Services through the Contract Engineering that allows to prepare or evaluate contracts properly in order to identify its improvements.

This paper is organized as follows Section 2 shows a brief description of the Contract Engineering. Section 3 describes the analysis of Contracts. Section 4 addresses a case study. And finally, Section 5 covers the conclusions.

2 Contract Engineering

In order to identify improvements in the outsourcing contract the lacks in the contracts are identified. To achieve this, Contract Engineering has been established. The Contract Engineering seeks to 1) identify and define the aspects or activities before carrying out a contract signing, as well as, 2) provide a contract model that shows the basic clauses to take into account in the acquisition contract development, which is supported by 3) an evaluation method to carry out the evaluation of a contract in order to take the decision for accepting or rejecting it. In order to develop the Contract Engineering, the previous three aspects have been established.

2.1   Activities before signing the contract

The goal is to establish the activities to perform before signing a contract, in order to allow the acquirer to identify the activities that must be considered before establishing or formalizing an outsourcing contract. In order to establish these activities, CMMI-ACQ model was established as a starting model.

The CMMI for Acquisition model (CMMI-ACQ) provides a framework to facilitate the outsourcing strategies adoption, eliminating the existing barriers among the relevant stakeholders (service supplier, business departments, system areas, etc). The Acquisition concept in the model is broader

than the Outsourcing concept. While the Outsourcing concept focuses on the specific processes of the services supplier, the Acquisition concept or IT procurement covers the hardware and the Commercial off-the-shelf (COTS) solutions. Therefore, the CMMI-ACQ model offers a valid answer to the outsourcing processes.

CMMI-ACQ contains 22 process areas [25]: 16 Process areas are CMMI Model Foundation (CMF) and 6 Process areas focus on practices specific to acquisition addressing agreement management (AM), acquisition requirements development (ARD), acquisition technical management (ATM), acquisition validation (AVAL), acquisition verification (AVER), and solicitation and supplier agreement development (SSAD). Once the model is selected, the process area (AP) that provides the best practices before the signing a contract is chosen. The selected process area is: solicitation and supplier agreement development (SSAD).

The purpose of SSAD is to prepare a solicitation package, select one or more suppliers to deliver the product or service, and establish and maintain the supplier agreement [25].

The SSAD involves the following activities: Identify Potential Suppliers, Establish a Solicitation Package, Review the Solicitation Package, Distribute and Maintain the Solicitation Package, Evaluate Proposed Solutions, Establish Negotiation Plans, Select Suppliers, Establish an Understanding of the Agreement, Establish the Supplier Agreement.

To establish the main activities before carrying out a contract signing the dependences from SSAD process area to the maturity level 2 process areas were analyzed, in order to, establish the main activities, the required activities and the support activities [31, 32].

2.2 Contract Model

This model establishes the main clauses to be included in a contract grouped by 7 categories. Each category contains the clauses which are related to the same subject (see Figure 1). A clause is a set of components [30].

2.2.1 Contract structure

The proposed contract model is structured into seven categories: services, management, financial, duration, transition, security and communication (see Figure 1) and each of these categories has several clauses, which in turn contain components. The legal regulations were not considered in the establishment of these categories and its clauses and components, because they are different in each country and sector.
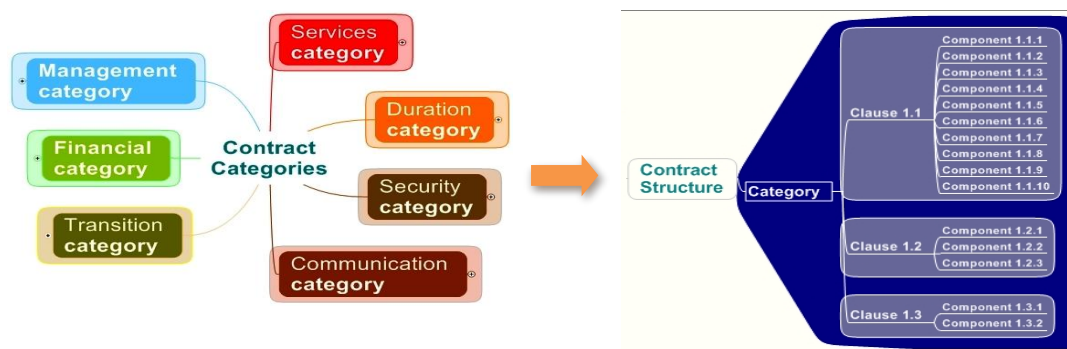
Figure 1. The contract structure

2.2.2 Contract categories
The categories and its main clauses of the proposed contract model are showed.

2.2.2.1 Service category

Clauses under services category should describe in precise terms types, scope, and nature of all the services required, times when these services should be available, and the level of performance (e.g. throughput rate, turnaround time, system availability, etc.) required [20]. Contractual clauses under service category include: *description of service (establishing key principles and agreements among the parties), functional requirements of the service system (specifying target service levels to be delivered, how often, to what extent, when and where) and terms and condition for installation and maintenance (specifies the processes that will be used to manage the implementation of new or modified services)* [5, 6, 12, 13, 20, 24].

2.2.2.2 Management category

Clauses under management category commonly set and assess the value that the relationship is generating for the various stakeholders to ensure that the relationship remains on course [12]. Contractual clauses under management category include: *customer support procedures and response time/performance (specifying tactical measures of service performance), audit rights (deep analysis of the provider by the acquirer), problem escalation procedures/risk management (specifying plans to deal with future risks), acceptance criteria/quality management (specifying the service level that must be fulfilled) and transfer of resources/ staff and assets (specifying which resources the acquirer will provide to the provider)* [1, 6, 20, 22].

2.2.2.3 Financial category

Clauses under financial category include: *pricing and payment terms (states all the financial agreements related to the service provided) and incentive/liability clauses (states appropriate incentives and penalties based on performance). When, how and to whom payments should be made, and the amounts and structure of payments involved are included here* [11, 20].

2.2.2.4 Duration category

Clauses under duration category *specify in the contract all the dates agreed in the negotiation of the relationship. When the contract will start, when it will be finished, the dates for delivering each service and the duration of them (specifying the dates for the duration of the relationship) are established.*

2.2.2.5 Transition category

Clauses under transition category refer *to elaborate procedures to be completed before the transfer or the conclude relationship can take place to assure the current terms related to the service or product. Also the renewal of the contract and the costs derived from penalties are established. Contractual clauses under transition category include termination conditions (which are the termination conditions and the penalties for both parties), transition clauses (which are the obligations from both parts duration and transition to another provider), reversibility clauses (compensation to be paid to either of the parties if the other terminates the contract) and business continuation clause/renewal option (the procedure for a renewal of the actual contract)* [1, 2, 3, 11, 20].

2.2.2.6 Security category

Clauses under security category safeguard *the interests of the outsourcing customer in case of a breakdown in the relationship. This also puts the outsourcing customer in a stronger position in any future contract re-negotiation [20]. Clauses under security category include intellectual property rights (identifying key process owners and their roles and responsibilities) and information security and confidentiality (conditions to maintain the security and confidentiality of the relationship)* [20].

2.2.2.7 Communication category

Clauses under communication category *deal with ground rules and procedures for future contingencies, which would lead to desired outcomes if followed (i.e., clauses for agreeing to agree)[12]. All the clauses that contain solutions for possible disputes, how to deal with them and when are included here. Contractual clauses under communication category include conflict arbitration mechanisms/dispute resolution (stating the parameters and conducting rules for involving a third party for resolving issues), renegotiation windows (articulating that the right processes, people and tools are*

in place to enable change to meet ongoing demands) and communication plan (documenting communication processes to facilitate consistent knowledge exchange) [3, 8, 11, 17, 20]

2.3 Evaluation Method

The goal of the evaluation method is to know the coverage level of contracts in order to select or reject the best proposal in a request for proposal. This method seeks to obtain the coverage level of the clauses and the categories of a contract. Two evaluation criteria have been defined to obtain the coverage level [30]:

• Evaluation Criterion 1 based on the contract model.

• Evaluation Criteria 2 based on the business goals.

2.3.1 Evaluation Criterion 1 based on the contract model

This evaluation criterion seeks to calculate the coverage level of each clause, each category and contract itself based on the sum of the percentages achieved by each component in the proposed contract model.

2.3.2 Evaluation Criteria 2 based on the business goals

This method seeks to identify the coverage level of the clause, category and contract based on the weight given to each contract model component, clause and category, according to the business goals. In other words, the acquirer will assign the percentage that reflects the importance degree of each component, clause and category.

As Figure 3 shows, the established percentage in each category will represent 100% to be distributed among the grouped clauses in it. As in the category, the established percentage in each clause will represent 100% to be distributed among the grouped components in each clause.

2.3.3 Coverage Criteria

To evaluate the obtained values by using methods 1 and 2, some criteria were created to define the type of coverage. If the value is equal to 100%, the coverage (clause, category or contract) is considered complete. If the value is bigger than 75% and less than 100%, the coverage (clause, category or contract) is considered large. If the value is bigger than 50% and less than 75% the coverage (clause, category or contract) is considered medium. If the value is bigger than 25% and less than 50% the coverage (clause, category or contract) is considered small. If the value is bigger than 1% and less than 25% the coverage (clause, category or contract) is considered incomplete and finally, if the value is equal to 0% the coverage (clause, category or contract) is considered null.

3 Analysis of Contracts

In order to analyse the contracts, a contract evaluation tool, which allows recording and getting automatically the contract coverage level, has been established [30]. This tool has been developed taking into account the previous contract model and evaluation method.

The contract is analyzed manually to check if it contains the components defined in the contract model. If a component is found, it is highlighted and a sticky note is allocated for writing the component name.

Once components are manually detected within the contract content, the following steps in the tool are performed [30]: information recording, percentage assigning and finally getting the results.

4 Case study

The outsourcing contracts analyzed are from Spanish companies related to the IT area. The contracts analyzed kept most of the available situations for an outsourcing relationship: a large company providing service to a large company, a large company providing service to a small company, a small company providing service to another small company and a small company providing service to a medium company. For security and confidentiality reasons of the companies involved in these contracts, all the companies' names will be kept on secret during the case study analysis.

The company related to contract 1 is a company with over 20 years of experience in the market for computers and telecommunications. Its commitment is to provide its acquirers a comprehensive and timely solution and support of a real added value. The company has implemented a quality assurance system based on ISO 9001 since 1998.

The company related to contract 3 is a small company that develops software projects (applications to web, mobiles and Windows systems) and provides assistance support (process and product audit, business analysis and requirements specification).

The company related to contracts 2 and 4 is a multinational consulting firm that offers its acquirers comprehensive business solutions covering all aspects of the value chain, from business strategy to systems implementation. The company is active in the sectors of Banking, Healthcare, Industry, Insurance, Media, Public Sector, Telecom and Utilities.

All contracts are analysed manually to check if they included the components for each clause then the found components are recorded in the contract evaluation tool.

4.1   Clauses results

The contracts did not cover well the clauses defined by the model and some of them lack information in specific categories. For example, all contracts should have paid more attention in:

- Termination Conditions clause. It is very important because the relationship between the acquirer and the provider is likely to be tense in termination cases.
- Transition clause, which is the part that deals with the agreements for software licenses, provision for the IT providers and identification of the transfers of responsibility
- Reversibility clause. It establishes the compensation to be paid by termination of the contract.
- Renegotiation windows clause which is in the category that provides a mechanism for the price and payment renegotiation.

Figure 2 shows the clauses coverage by contracts.
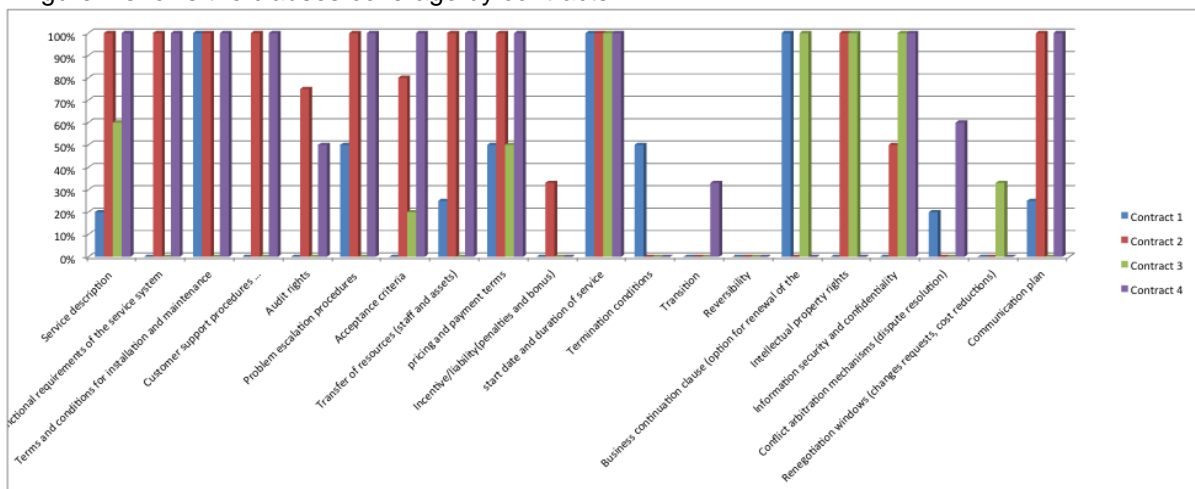
Figure 2. Clauses Coverage in the contracts

4.2 Categories coverage

The results obtained with the evaluation criteria 1 and 2 in the contracts 2 and 4 indicated a large coverage. However, the acquirer should make a contract review to include clauses in order to improve the overall contract, which was confirmed later in the case of the contract 2 having a renegotiation with the provider. Figure 3 shows the categories coverage by contracts.
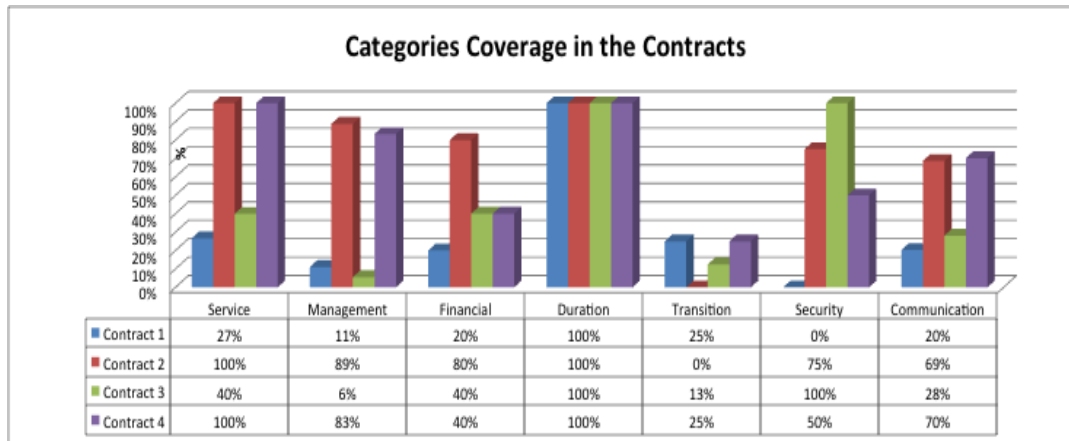
Figure 3. Categories Coverage in the Contracts

- Service category. Figure 3 shows that the contracts 2 and 4 have a 100% of coverage. During the evaluation these contracts contained all required information by the clauses in this category. However, contracts 1 and 3 have 27% and 40% of coverage respectively. Both contracts don't show any information about the functional requirements of the service system. The lack of information about the service category in these contracts is a problem, since the acquirer/provider does not know exactly which are the functions and services that they should receive/offer.

- Management category. The contracts 2 and 4 show a large coverage (89% and 83% respectively), so both can be considered well defined in this category. However, the contracts 1 and 2 have 11% and 6% of coverage respectively. According to the management category information, these contracts could have problems to manage the relationships between acquirer and provider.

- Financial category. The contract 2 was the best covered, with a complete coverage of 80%. The contracts 3 and 4 show a small coverage (both 40%). Both have covered the pricing and payment terms clause, however they did not show information related the incentive and liability clause. Contract 1 has a poor coverage (only 20%) in this category.

- Duration category. All contracts have achieved complete coverage (100%). The contracts show the information related with the start date and duration of the contract clause as defined in the model. Every contract must contain the initial and final dates for a successful outsourcing relationship and all contracts are in agreement with the model.

- Transition category. This category is not well specified by the contracts. The maximum coverage is 25% by the contracts 1 and 4 and the minimum coverage is 0% by the Contract 2. During the evaluation, all contracts don't have enough information related to this category.

- Security category. The contracts 2 and 3 are well defined according to this category, achieving a complete (100%) and large (75%) coverage respectively. The contract 4 has a medium coverage (50%). The security of the outsourcing relationship of the companies involved in contract 1 is seriously compromised, since it does not have any information about it.

- Communication category. The contracts 2 and 4 have a medium coverage (69% and 70& respectively). Moreover, this is another category not well specified by the contracts 1 and 3. These contracts don't cover all clauses grouped in this category. Contracts 1 and 3 will have serious problems during the dispute resolution since they do not provide any information about it.

5 Conclusions

After the case study of the contracts, it was possible to indicate that a clause named service level agreement (SLA) appear in three of them. This clause do not appear in the contract model with this name, but it can be defined as an aggregation of 8 clauses discussed before: description of service, functional requirement of the service system, start date and duration of service, terms and conditions for service installation and maintenance, customer support procedures and response time (performance), problem escalation procedures and acceptance criteria (quality measurement).

Moreover, the results obtained with the evaluation method in the case study confirmed: a) the poor clauses definition, b) the future problems between the acquirer and provider which was confirmed by a lawsuit between the acquirer and provider of the contract 1, c) the lack of information in each content of contracts analyzed, and d) the lack of a structure in the contracts.

The Contract Model enables the acquirer to determine and understand the main clauses to be included in a contract and a guide to develop or to establish the contract structure, the main clauses and its information. Besides this, an evaluation method based on this model, allowing verification of coverage in order to select the best contract for a future hiring. With the established criteria in the evaluation method, it is possible to determine the degree of contracts coverage in terms of the proposed contract model and a percentage weighting that reinforces the contract evaluation.

Finally it is concluded that the Contract Engineering provides a guide in order to allow the acquirer to identify the activities that must be considered before establishing or formalizing an acquisition or outsourcing contract. Besides the contract model and the evaluation method enable to prepare or evaluate contracts properly in order to identify improvements in the outsourcing contracts. Therefore the Contract Engineering allows minimizing the causes of failure in the relationships between acquirer and provider caused by a bad definition of the contract. In addition it ensures that customer expectations will be achieved.

6 Literature

1. Aubert, B., Houde, J. F. et al. Category of IT outsourcing contracts. System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference. (2003).

2. Barua, A., Susarla, A. et al. "Making the most out of an ASP relationship." IT Professional 3(6): 63-67. (2001).

3. Beulen, E. and Ribbers P. (2003). IT outsourcing contracts: practical implications of the incomplete contract theory. System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference. (2003).

4. Biolchini J., Mian P., Natali A., and Travassos G., "Systematic Review in Software Engineering," COPPE/UFRJ, Rio de Janeiro May. (2005).

5. Bryson, K. M. and Sullivan W. E. Designing effective incentive-oriented outsourcing contracts for ERP systems. System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference. (2002).

6. Buco, M., Chang, R. et al. A Generic SLA Semantic Model for the Execution Management of e-Business Outsourcing Contracts. E-Commerce and Web Technologies: 249-257. (2002)

7. Corbet and Associates, "Outsourcing's Next Wave," (2002).

8. Chin, W. and Goles T. "Information systems outsourcing relationship factors: detailed conceptualization and initial evidence." SIGMIS Database 36(4): 47-67. (2005).

9. Craze D., Mendonca M., Basili V., Shull F., and Jino M., "Extracting Information from Experimental Software Engineering Papers," in Chilean Society of Computer Science, 2007. SCCC '07. XXVI International Conference of the, pp. 105-114, (2007).

10. Gellings, C. "Outsourcing Relationships: The Contract as IT Governance Tool," in System Sciences. HICSS. 40th Annual Hawaii International Conference on, 2007, pp. 236c-236c. (2007).

11. Giunipero, L. C., Harris, A. et al. "Impact of organizational and contract flexibility on outsourcing contracts." Industrial Marketing Management 27(5): 373-384. (1998).

12. Goo, J. and Nam K. Contract as a Source of Trust--Commitment in Successful IT Outsourcing Relationship: An Empirical Study. System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference. (2007).

13. Grefen, P., Koetsier, M. et al. Contracts for Cross-Organizational Workflow Management. Electronic Commerce and Web Technologies: 110-121. (2000).

14. Huai, J. "An Incentive Model of IS Outsourcing Contract," in Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on, pp. 6588-6592. (2007).

15. Jens, D., Tim, G., Rudy, H., and Bandula, J. "Information systems outsourcing: a survey and analysis of the literature" SIGMIS Database, vol. 35, pp. 6-102, (2004).

16. Junchao, X. and Qing W. Contract-driven cooperation development process for software outsourcing. Computer and Information Technology, CIT. The Fifth International Conference. (2005).

17. Kern, T. The Gestalt of an information technology outsourcing relationship: an exploratory analysis. Proceedings of the eighteenth international conference on Information systems. Atlanta, Georgia, United States, Association for Information Systems. (1997).

18. Kern, T. and Willcocks L. "Exploring information technology outsourcing relationships: theory and practice." Journal of Strategic Information Systems 9(4): 321-350. (2000)

19. Kitchenham, B. A., Pfleeger S. L., et al, "Preliminary Guidelines for Emprical Research in Software Enginerring," National Research Council of Canada, (2001).

20. Lee, M. "IT outsourcing contracts: Practical issues for management." Industrial Management & Data Systems, vol 96, pp 15. (1996).

21. Liston, P., Byrne, J., Byrne, P., and Heavey, C. "Contract costing in outsourcing enterprises: Exploring the benefits of discrete-event simulation," International Journal of Production Economics, vol. 110, pp. 97-114, (2007).

22. Nellore, R. "Validating specifications: a contract-based approach." Engineering Management, IEEE Transactions on 48(4): 491-504. (2001).

23. Pino F., Garcia F., and Piattini M., "Software process improvement in small and medium software enterprises: a systematic review," Software Quality Control, vol. 16, pp. 237-261, (2008).

24. Schmidt, H. Service Contracts Based on Workflow Modeling. Services Management in Intelligent Networks: 132-145. (2000).

25. SEI CMMI Production Team. CMMI for Acquisition, **CMU/SEI-2010-TR-032** ESC-TR-2010-032 Version 1.3, November 2010.

26. Shein, E. "Railroad wields penalties to keep its service contract on track. (Southern Pacific Transportation Co)," PC Week, vol. v12, pp.14(1), (1995).

27. Stevens K. R., "Systematic reviews: the heart of evidence-based practice," AACN Clinical Issues: Advanced Practice in Acute & Critical Care, vol. 12, pp. 529-538, (2001).

28. Zhang P., Zeng Z.-x., and Huang C.-p., "Study on Critical Success Factors for IT Outsourcing Lifecycle," in Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007, pp. 4379-4382, (2007).

29. Calvo-Manzano, J. Cuevas, G. et al.. Outsourcing Contracts: A Systematic Review. In International Book Series "Information Science & Computing", Number 11. Bulgaria and International Journal "Information Technologies and Knowledge" Volume 3. (2009).

30. Calvo-Manzano, J. Cuevas, G. et al. CONEVTO: Contract Evaluation Tool for Software and Services Acquisition Organizations. 17th European Systems & Software Process Improvementand Innovation (EuroSPI 2010).

31. Mejia Jezreel, Muñoz Mirna, et al. A Multi-model Workflow before Establishing an Acquisition Contract Based on CMMI-ACQ Model. 18th European Systems & Software Process Improvement and Innovation (EuroSPI 2011).

32. Calvo-Manzano J., Cuevas G. et al. Processes Implementation Sequences to Achieve the Maturity Level 2 in CMMI-ACQ, European Systems & Software Process Improvement (EUROSPI2009).

7 Author CVs

Mejia Jezreel

He holds a PhD in Computer Science. He is actually researcher in Computer Science at the research center in mathematics (CIMAT), Zacatecas, México. His research field is Software Engineering. He is a member of MPSEI (Mejora de Procesos Software en el Espacio Iberoamericano) research group. His main research field is process improvement, focusing in the

topics of multi-model environment, project management, acquisition and outsourcing process, solicitation and supplier agreement development and agile methodologies. He has published several technical papers on acquisition process improvement, project management, TSPi, CMMI, muti-model environment. He has been a member of the team that has translated CMMI-DEV v1.2 and v1.3 to Spanish.

Muñoz Mirna

She holds a PhD in Computer Science. She is actually researcher in Computer Science at the research center in mathematics (CIMAT), Zacatecas, México. She is a member of MPSEI research group. Her research field is Software Engineering. Her current research interest consists to incorporate process improvement focusing on diminishing change resistance. She has published several technical papers on project management, multi-model environment and process improvement. She has been a member of the team that has translated CMMI-DEV v1.2 and v1.3 to Spanish.

Calvo-Manzano Jose A.

He holds a PhD in Computer Science. He is assistant professor in the Facultad de Informática at Universidad Politécnica de Madrid. He is teaching in the area of Software Engineering, specifically in the domain of software process management and improvement. He has participated in more than 20 research projects (European and Spanish Public Administration). He is author of more than 50 international papers. He is author of books related to software process improvement and software engineering topics also. He is a member of SOMEPRO research group, where his main research field is process improvement, focusing in the topics of requirements management, project management, process deployment, and solicitation and supplier agreement development. He is the head of the "Research Chair in Software Process Improvement for Spain and Latin American Region", where he has been a member of the team that has translated CMMI-DEV v1.2 and v1.3 to Spanish.

Cuevas Gonzalo

He received an Engineering degree in Telecommunications in 1965 and a PhD in Telecommunications in 1974. He also received an MS in Computer Science from the Polytechnic University of Madrid in 1972. He has been vice dean of the Computer Science faculty at the Polytechnic University of Madrid, where he is a full professor since 1970. He worked for Iberia Airlines as Software Development Director, supervising over 200 technicians, Data Processing Centre Director, Data Transmission Software Development Director, and the person in charge of strategic planning and software process improvement. He led European projects on software best practices from 1991 to 1995. His main research field is software engineering, including both technology (methods, techniques, and formalisms) and management. His current research interest is process models and methods, and transition packages. He is member of ACM, senior member of IEEE, member of the Telecommunication Engineering Association and member of the Computer Sciences Association. He has been a member of the team that has translated CMMI-DEV v1.2 and v1.3 to Spanish.

Tomás San Feliu

He holds a PhD in Computer Science. He has been working 15 years to Software Engineering field as programmer, associate and assistant professor and consultant. He has been involved in CMM appraisals in Spain and Latin America. As Software Engineering assistant professor in the Facultad de Informática at Universidad Politécnica de Madrid, his main research field is software process and software process improvement. His current research interest includes project management and risk management. He has published several books on process improvement and he has published technical papers in software engineering and software process improvement. He has been a member of the team that has translated CMMI-DEV v1.2 and v1.3 to Spanish.

The Service Catalogue as the basis for Financial Management in the Medium, Small and Micro enterprises: a practical approach using a process asset library*

***Authors are ordered alphabetically*

Magdalena Arcilla[1], Jose A. Calvo-Manzano[2], Jose Antonio Cerrada[1]
[1]Escuela Técnica Superior de Ingeniería Informática, C/ Juan del Rosal 16, 28040 Madrid, Spain
{marcilla, jcerrada}@issi.uned.es
[2]Facultad de Informática, Universidad Politécnica de Madrid, Campus de Montegancedo, Boadilla del Monte, 28660 Madrid, Spain
joseantonio.calvomanzano@upm.es

Abstract

Last decades have brought a transformation in the use of Information Technologies. However, IT spending is not under a proper control. This situation is even worse in medium, small and micro enterprises. A key aspect is to know what services the money is spent. For this, it is necessary to know what services the IT department provides. The objective of this paper is to show a model based on the concept of process asset library that allows a medium, small and micro enterprise to obtain its catalogue of services in a simple, flexible and inexpensive way.

Keywords

Service Catalogue; Financial Management; Medium, Small and Micro Enterprises; Service Management; ITIL

1 Introduction

The past few decades have supposed a drastic transformation in the use of Information Technology (IT). Since the early 80's until today, the data processing centres, large and expensive, have been replaced by personal computers that are affordable even for smaller companies. The current presence of computers in business supports their growth in terms of productivity, regardless of size or sector [1]. Also, it is not surprising that IT investments are an ever increasing cost of business. According to a business report conducted in the U.S. [2], the majority of CEOs (Chief Executive Officer) and CIOs (Chief Information Officer) consider IT as a strategic advantage.

Therefore it would be logical to assume that IT spending is under control, after all, it is essential for productivity growth and is considered an important asset, especially for Medium, Small and Micro Enterprises (MSMEs). However, this is not what happens: according to Schneider [3], 86% of U.S. financial executives reported that their IT expenditures are not under adequate control. This is alarming news for the corporate world, especially for MSMEs that are a total of 99% of companies worldwide [4] and, particularly, in Spain [5] (see Table 1).

Table 1: Spanish companies by layer of employees and their percentages (source: DIRCE 2010)

	Self-employed	Micro 1 - 9	Small 10 -49	Medium 50 -199	Large 200 - 499	Large + 500
# Employees	1,774,005	1,354,176	225,557	20,843	3,374	1,704
% Total	52.49%	40.07%	6.67%	0.62%	0.10%	0.05%
% Total Accumulated	52.49%	92.56%	99.23%	99.85%	99.95%	100%

It is also a common practice in companies that their Finance Department includes the costs of the IT infrastructure without making any distinction with other existing costs. This is worse, if company does not know what services associate to these costs (lack of service catalogue) [6].

In most MSMEs, the IT department is often seen as a cost centre [7]. A cost centre is a department whose cost is added to the overall cost of the company, but only contributes indirectly to the benefits (its costs must be paid whether the company does business as if it does not). Thus, according to Trastour and Christodoulou [8], IT managers are under increasing pressure to justify IT costs and show the other directors of the company that they can transform IT assets into business services, while they clearly link costs identified with the company profits.

On the one hand, there are many methodologies that quantify the return on IT investments, and on the other hand, those companies (mainly MSMEs) where the IT department is a cost centre, are left to chance the assessment and allocation of IT costs. Therefore, the MSMEs that consider IT as a cost centre [7] are lacking in the financial management of IT.

This policy of financial management (cost centre and traditional accounting system) together with the lack of a service catalogue does not help to understand the real costs associated with different IT services due to their complexity. It can also create a false sense of disconnection from the internal and/or external client, between price and quality. In fact, one of the main problems is not to determine the costs of IT services that users receive, it is that they are seen as a tool for their day to day, without being worried, mostly, to make and adequate use of them. Customers are not worried about the proper user of services and do not assess objectively whether the services received are balanced between price and expected quality because users are not being aware of the services they receive and their associated cost, and costs are not imputed. Any complaints, from customer or users, about such services will never have an objective basis, creating a gap between the business and the IT department.

When an organization detects that some services are not working well or the perception that the customer has about the services provided by IT is very poor, we begin to question where we are doing things wrong. So we need to take an internal look at the IT department and identify how to improve. To do this, indicators related to a poor or deficient service level are often used, and can assist in identifying where improvement is needed and take appropriate action. According to Esterkin [9],

among the typical indicators of a poor service level are the absence of a service catalogue and not controlling the costs (i.e., the process of IT financial management).

All these issues would be solved if a company gets to have a good IT service management. It should include the process related to service catalogue (available services) and the process related to financial management (budgets, costs and charging of the services). To do this, companies try to implement some of the models of best practices related to IT service management used as reference.

In section 2, the main service models in terms of financial management process and service catalogue process are presented. In section 3, the proposed model based on the concept of process asset library is presented. In section 4 a practical approach is presented. Finally, conclusions are presented in section 5.

2 IT service management models

The IT service management models provide guidance on how services can be managed effectively during its life cycle. The main service management models as they relate to financial management and IT service catalogue are: ITIL [10], Operational Management Capability Model from SUN [11], Operational Framework from Microsoft [12], ISO/IEC 20000 [13], CMMI-SVC [14]; and some additional standards such as COBIT, eSCM-SP, eTOM, ITSCMM, OPM3, PMBOK y PRINCE2.

Conceptually, the IT service management models describe "what" to do, but not explicitly set the "how". Table 2 compares the previous IT service management models, only those listed above as main, and shows the gaps to be addressed (criteria assessed). The criteria assessed are the typical activities related to the IT financial management process and IT service catalogue process applied to a MSME.

Table 2: IT Service Management Models

Criteria assessed	ITIL	OMCM (SUN)	OMF	ISO 20000	CMMI-SVC
Define the standard service catalogue related to a MSME with and IT department	○	○	○	○	○
Define the IT cost factors	◐	○	○	○	○
Provide guidelines to classifying IT spending	◐	◐	◐	○	○
Provide guidelines to assess IT spending	◐	◐	◐	○	○
Provide guidelines on IT charging	◐	◐	◐	○	○
Provide guidelines about how to be applied to a MSMEs	○	○	○	○	○
Indicate what to do	●	●	●	●	●
Indicate how to do	○	○	○	○	○
Legend: ● yes, ○ no, ◐ partially					

Next, both processes (IT financial management and IT service catalogue) are defined.

2.1   IT financial management

In order to implement IT financial management, it is necessary to:

- Predict and estimate the money to be required in order to deliver IT services for a period of time.

- Account for money spent on the provision of IT services and compare it with the budget. For this, it is necessary to identify the cost elements associated with services, including both their categories and the different types of cost.

- Influence the behavior of customers and users, making them familiar with all the services they are receiving from the IT department and sensitizing them on its cost in order to make good use of them.

In summary, IT financial management provides a way to measure, control and cover the costs of IT services. Therefore, it is important to define the services (service catalogue) that the IT department will provide the organization.

2.2 IT service catalogue

An IT service catalogue is like a restaurant menu: it presents the IT services that can be provided and supported to customers. This definitely influences the decisions that customers have about what IT can help. Typically a catalogue should include the systems already in operation, i.e. systems in production or operation. A common mistake when writing the IT service catalogue is to write it in computing or technical terms (words and phrases of the business must be used). If we say "OBIEE system", who knows what that is? Conversely, if we say "business intelligence system for the area of marketing", it sounds more in business terms, in terms of what the business understands.

In most MSMEs the IT service catalogue is not considered or it usefulness is not understood and where it exists, it has a very technical guidance or is never aligned with business objectives [9]. In addition, an added difficulty is that there are no operational guidelines to guide us in building the IT service catalogue oriented or not to MSMEs (what to do is in all models, but how to do is not established). So, defining an IT service catalogue becomes a very dificult task to be perfomed by a MSME.

Table 3 shows the data from the survey conducted by the Spanish National Statistical Institute (INE) on the use of Information Technology and Communication (ICT) and Electronic Commerce in Spanish companies during 2009/2010. Its purpose is to know the implementation of IT and electronic commerce in all Spanish companies.

For all the above and given the importance of an IT financial management, the purpose of this research is to create a standar catalogue of IT services for helping MSMEs to define its own IT service catalogue (that we have called defined catalogue of IT services). It is not required to develop a detailed catalogue, but it is important to get the list of services and their descriptions.

Table 3: Use of IT infrastructure by company size and number of employees

ICT Infrastructure	# Employees < 10	10 - 49	50 - 250	+ 250
Computers		98.5	99.8	99.8
Local area network		83.8	96.7	98.3
Wireless local area network		36.5	55.1	68.4
Internet	58.1	96.9	98.8	99.7
Mobile	66.3	90.8	96.1	97.3
e-mail	55.3	96.2	98.5	99.7
% enterprises with internet and web site	25	60.8	81.3	91.1
Automatic data interchange		43	56.2	68

3 Proposal model through a process asset library

The global proposal solution defines, implements, and evaluates the IT financial management,

including the IT service catalogue, through the implementation of a process asset library [16]. For this, the solution provided has been divided into three phases, as shown in Figure 1. In this paper, we are only going to be focused on the IT service catalogue.
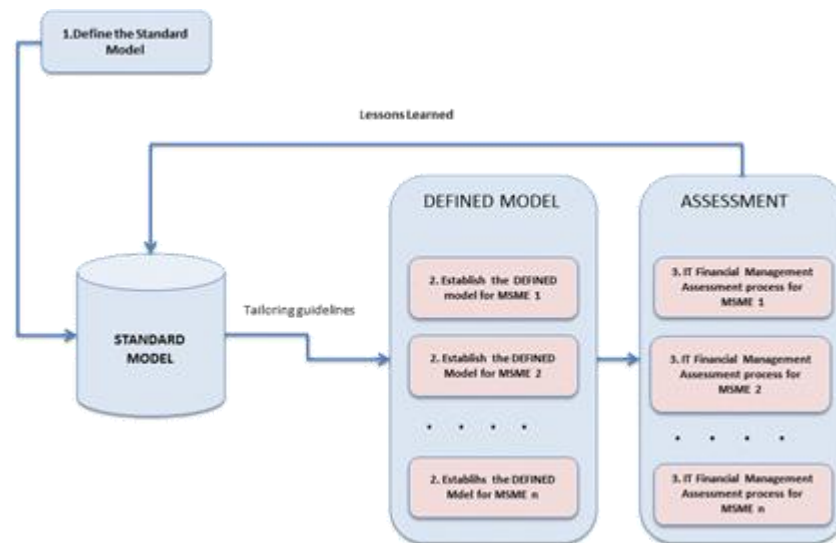
Figure 1: Model for implementing the IT financial management (including the IT service catalogue)

- The first phase (labelled standard) will be the foundation from which MSMEs will select their particular solution.

- The second phase (labelled defined) will be the solution for a specific MSME.

- The third phase shows the process related to the assessment of the IT financial management, and particularly the assessment of the IT service catalogue.

The standard and defined phases have similar activities and they are related to the IT service catalogue, the IT service budgeting, the IT service accounting, and the IT service charging.

In this article, the IT service catalogue related to the standard and defined phases is going to be addressed. Next, each activity for defining the standard catalogue and the defined catalogue is described.

3.1 Phase 1: Create the standard IT service catalogue

The service catalogue is the cornerstone for defining IT business needs. The services of the catalogue are grouped in a logical way according to the customer activity. So, a set of the services that the IT department provides to the business is established.

In this activity, the standard IT service catalogue is defined. This catalogue will be the foundation for defining (by selection) the catalogue for each MSME. The standard service catalogue is based on a survey related to the most used services in most MSMEs, as well as on information provided by the company where the model is going to be implemented. In particular, the entries (see Figure 2) for the standard IT service catalogue activity are:

- Statistical data from the Spanish Statistical National Institute (INE) for MSMEs that have been obtained taking into account a survey related to the use of IT and electronic commerce [5].

- Data required by the customers of the company where the model is going to be implemented (this is, the other departments of the company).
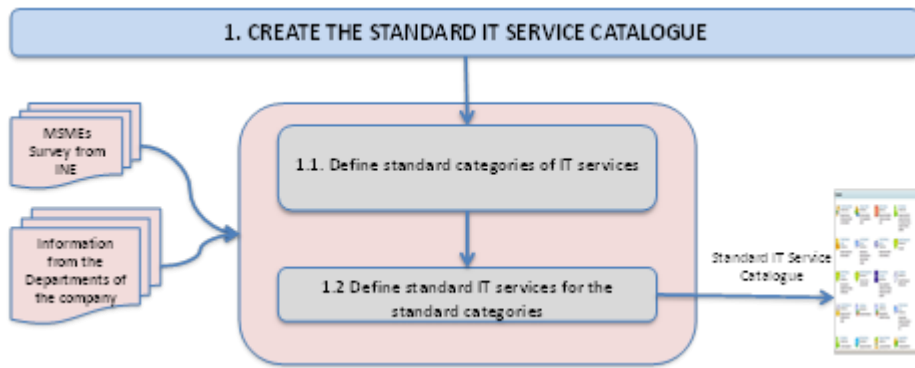
Figure 2: Phase 1: Create the standard IT service catalogue

Next, each activity for this phase is detailed.

- Define the standard categories related to the IT services.

The standard categories of services are shown in Figure 3. Taking into account that the initial number of services could be greater than 15, a set of general (standard) services categories is defined. Then, the applicable standard services have been added for each standard category defined (see Figure 4). Initially, there were more categories, for example maintaining and managing network, defining manuals and documentation, implementing and managing intranet services, but it has been decided to reduce the number of categories to the ones shown in Figure 3 because they are the most commonly repeated in enterprises.

Figure 3: Standard IT service categories

- Define the standard IT services for each standard category.

All the potential IT services to be considered in this research are defined for each standard category. Nevertheless, new standard service categories could be added, and also new standard services could be added for each category. In Figure 4, the standard IT services are shown for each standard category defined in the model.
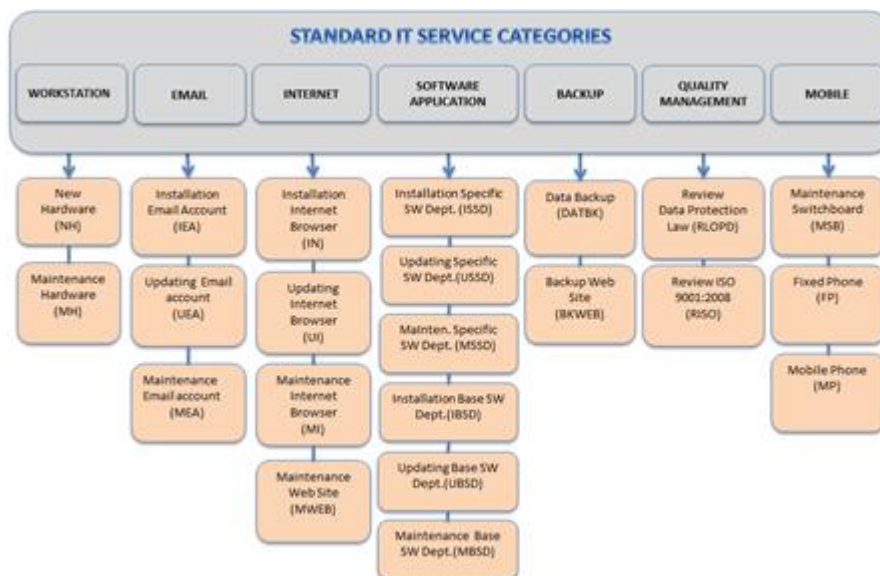
Figure 4: Standard IT services for each standard service category

The main difference between the updating services and the maintenance services are that the updating consist of installing new versions of the related service, while maintaining refers to solving an incident/problem related to the service.

3.2 Phase 2: Create the defined IT service catalogue

In this phase, the defined IT service catalogue is defined from the standard services established in the standard model. The IT department related to the MSME that is going to implement the model selects the services that is going to provide to the other departments of the MSME. This catalogue is called defined IT service catalogue and owned by each MSME.

Next, the activities related to this phase will be detailed through the practical approach.

4 Practical approach

The company that has experienced the model is called *THECOMPANY* for confidentiality reasons. It is a small company with 18 employees and 5 departments:

- Department of Accounting and Finance: is responsible for providing fiscal consultancy and the accounting of the company's clients, as well as its own accounting.

- Department of Labor: is responsible for carrying out payrolls, TCs, update the Social Security status of the company's clients, as well as all the activities related to its own human resources.

- Department of Mortgage Management: is responsible for everything related to the management of real state certificates and processing of inheritances related to the company's clients.

- Department of Legal: is responsible for carrying out legal advices, issuing criminal certificates, wills, and any paperwork related to Department motor vehicles (as, for example, renewal of a driving license).

- Department of IT: is responsible for providing support to the rest of departments. In addition it is also responsible for ensuring compliance to ISO Quality 9000 and LOPD (Organic Law of Personal Data Protection).

With respect to the infrastructure, *THE COMPANY* has a midrange server for hosting data and applications, 18 personal computers and 2 laser printers (all connected through the LAN). Each department only can have access (through licenses) to the applications needed for performing their daily work which are installed on the server. There are 4 matrix printers needed for printing official forms. The company also maintains a web site that provides all the services of the company, through the web www.thecompany.com. The web site is hosted on a third-party ISP server, which also provides for storage and security needs. Next, activities for establishing the defined catalogue of the company are shown.

- Define the defined IT service categories.

The categories the IT department is going to provide are selected from the standard IT service categories. All standard categories of the model have been selected.

- Define the defined IT services for the defined categories selected.

For each defined category of IT services, those services that are going to be provided by the IT department are selected from the standard IT services. In this case, all services have been selected except the ones with an "X" (see Figure 5).
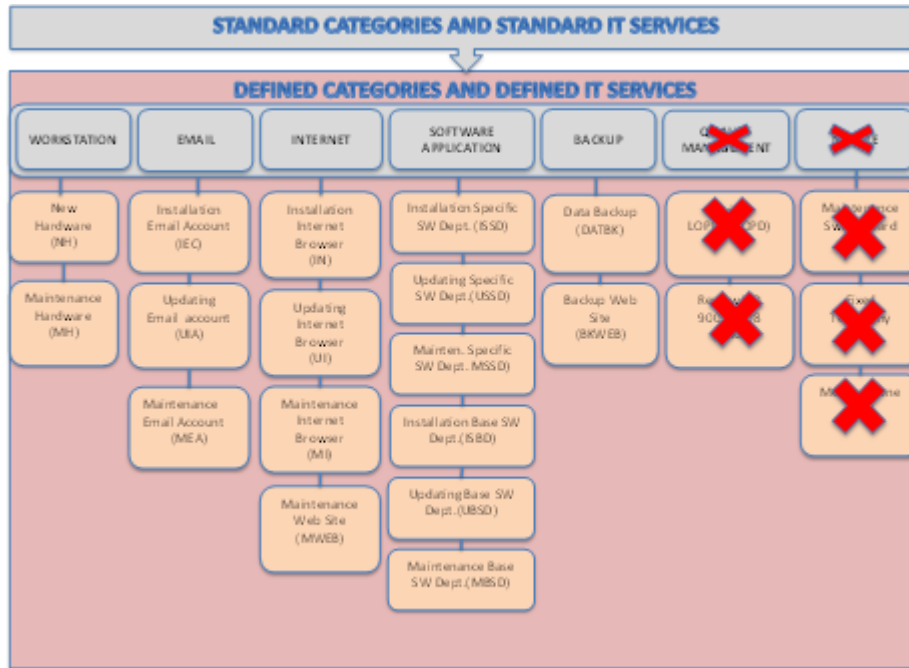
Figure 5: Defined IT services for each defined service category

5 Conclusions

IT is still considered by many managers as a representative part of the business expenses. Therefore, the service catalogue is a very important component to make visible the entire company the services that the IT department is providing. In this way, a picture of how IT is linked to the corporate strategy is shown.

A model for helping companies to implement their IT service catalogue is proposed. The model is based on the concept of process asset library, where the standard process is defined. The IT services provided have been obtained taking into account the needs of the company where the model has been implemented and a survey about the use of IT.

The proposed solution is very flexible, in the sense that standard services could be used by many enterprises and the defined services are the ones related to each enterprise. Each enterprise could add new services to the standard model and to the defined model.

The implementation of the service catalogue has enabled the company under the case study understanding the services provided by its IT department and the purposes for which they relate to the business. It is also the entry point for the implementation of the IT Financial Management process.

Acknowledgements. This work is sponsored by everis Foundation and Universidad Politécnica de Madrid through the Research Chair in Software Process Improvement for Spain and Latin American Region.

6 Literature

1. E. Brynjolfsson, L. Hitt, "Computing productivity: firm-level evidence", Cambridge: MA:MIT Sloan School of Management, 2000.
2. L. Cosgrove, W. Worthen, "Findings from CXO's IT spendingand lesson learned survey, CIO Magazine, 2001.

3. C. Schneider, "IT: Hold on to your wallet!", CFO Magazine, 2000.

4. S. Welch, "10-10-10: 10 minutes, 10 months, 10 years –a life-transforming idea", New York, Scribner: S1st Scribner hardcover ed, 2009.

5. Directorio Central de Empresas (DIRCE), Instituto Nacional de Estadística, http://www.ine.es/inebmenu/mnu_empresas.htm, 2011.

6. M. Sottini, "IT Financial Management: best practice", Zaltbommel:Van Haren Publishing, 2009.

7. H. Cha, D. Pingry, M. Thatcher, "What determines IT spending priorities?", Commun, ACM, Vol 52 Number 8, 2009, pp. 105-110.

8. D. Trastour, A. Christodoulou, ¡"Towards robust IT service portfolio management", Integrated Management of Systems, Services, Processes and People in IT, 2009, pp. 152-163.

9. J. Esterkin, "Buenas y malas prácticas en TI", http://www.deltaasesores.com/articulos/autores-invitados/iaap/, 2011.

10. OGC, "Service strategy", Office of Goverment Conmmerce, TSO, 2007.

11. Sun MicroSystem, OMCM Operations Management Capabilitites Model, http://www.sun.com/blueprints/0205/819-1693.pdf, 2009.

12. Microsoft Operations Framework, http://technet.microsoft.com/es-es/library/cc506049(en-us).aspx, 2010

13. UNE-ISO/IEC 20000-1, "Tecnologías de la Información – Gestión del Servicio- Parte 1: Especificación", AENOR, 2007.

14. CMMI Product Team, "CMMI for Services, version 1.3, CMU/SEI-2010-TR-034, Carnegie Mellon University, 2010.

7 Author CVs

Magdalena Arcilla

She holds a PhD in Computer Science. He is assistant professor in the Computer Science School at the Universidad Nacional de Educación a Distancia (Open University). She is teaching in the area of Software Engineering, specifically in the domain of software process management and improvement. She worked in a software development company (CustomWare) for eight years. She has participated as consultant implementing quality and service management systems (ISO 9000 and ISO 20000). She is author of papers related to process improvement, mainly in the service domain.

Jose A. Calvo-Manzano

He holds a PhD in Computer Science. He is assistant professor in the Computer Science School at the Universidad Politécnica de Madrid. He is teaching in the area of Software Engineering, specifically in the domain of software process management and improvement. He has participated in more than 20 research projects (European and Spanish Public Administration). He is author of more than 50 international papers. He is author of books related to software process improvement and software engineering topics also. He is the head of "Research Chair in Software Process Improvement for Spain and Latin American Region", where he has been a member of the team that has translated CMMI-DEV v1.2 and CMMI-DEV v1.3 to Spanish.

Jose Cerrada

He received the M.S. degree in industrial engineering and the Ph.D. degree from the Universidad Politécnica de Madrid, Spain, in 1979 and 1983, respectively. He is actually a Full Professor and Head of Department in the Systems and Software Engineering Department, Universidad Nacional de Educación a Distancia (UNED), Spain. His research interests are in RFID Technologies and Software Engineering. He is teaching in the area of Software Engineering, specifically in the domain of software process management and improvement. He has participated in more than 30 research projects (European and Spanish Public Administration). He is author of more than 60 international papers.

Knowledge Management in Software Process Improvement Initiatives in Small Organizations

Josefina Rodríguez-Jacobo, Ismael Edrein Espinosa-Curiel, Ulises Gutiérrez-Osorio,
Violeta Ocegueda-Miramontes, and José Alberto Fernández-Zepeda
Department of Computer Science, CICESE
Carretera Ensenada-Tijuana #3918 Ensenada, B.C. 22860, MEXICO
{jacobo, ecuriel, ugutier, vocegued, fernan}@cicese.mx

Abstract

In recent years, an increasing number of micro-sized and small-sized enterprises (MSEs) started a model-based Software Process Improvement (SPI) initiative. These initiatives are knowledge intensive activities that use and create knowledge related to multiple areas (SPI knowledge); however, very often MSEs do not manage their SPI knowledge causing its erosion and eventual lost. We performed a case study with five Mexican MSEs to analyze their SPI knowledge management process. We integrated the resulting information and created the SPI knowledge management system, Sharebox. Finally, to study the adoption of this system, we performed five interviews and two empirical studies. Preliminary results show that Sharebox is relevant in SPI initiatives, but several barriers limit its adoption.

Keywords

Knowledge management, software process improvement, SPI, small organizations.

1 Introduction

The micro-sized and small-sized enterprises (MSEs) play an important role in the development of software products and services. MSEs are organizations with fewer than 50 employees and, nowadays, they represent the 94% of the software development organizations [1]. The competition in the software market encourages many MSEs to start a model-based Software Process Improvement (SPI) initiative. The goal of these initiatives is to increase the productivity and quality of the organizations' software processes and products, to reduce their associated time and costs, and to increase the satisfaction of their customers [2]. MSEs have limitations, particularities, and special conditions that SPI managers need to consider to design effective SPI initiatives (*e.g.* MSEs have a reduced number of employees; they have severe restrictions in funds; they are very vulnerable to market conditions, etc.) [3], [4]. Because of these limitations and characteristics, MSEs use special SPI strategies and methodologies. For instance, MSEs usually implement tailored versions of some well-known Process Reference Models (PRMs) (*e.g.* CMMI, ISO/IEC 15504, or ISO/IEC 12207), or implement PRMs designed especially for their characteristics (*e.g.* MoProSoft [5], Competisoft [5], or MR-MPS.BR [6]). In spite of these innovations, there is a high failure rate of SPI initiatives, estimated as 70% [7].

The model-based SPI in MSEs is a complex, resource-demanding, and a long-term activity [8], [9]. During this activity, to meet the specifications of the PRM, these organizations make several major changes in their current processes (*e.g.* reallocation of activities flow), structure (*e.g.* change of roles and responsibilities), policies (*e.g.* changes in work performance expectations), culture (*e.g.* introduction of habits, beliefs, or values), employees (*e.g.* changes in status, benefits or influence), tools (*e.g.* introduction of new support tools), and software development methodologies (*e.g.* introduction of new methodologies) [10], [11]. Because of the magnitude and diversity of these changes, an SPI initiative

is a knowledge-intensive activity that uses and creates knowledge related to multiple areas (*e.g.* software engineering, project management, organizational change, human motivation, etc.). To make these changes, the employees involved (directly or indirectly) in the SPI initiative (we call them "SPI stakeholders") acquire new knowledge, work collaboratively, make decisions, learn from others, share their knowledge, and learn from the results and experiences obtained during the accomplishment of their SPI activities. However, very often, MSEs do not manage (identify, create, maintain, update, evaluate, access, transfer and apply) their SPI knowledge (we define this term as any knowledge necessary to make the organizational, social, personal and technical changes required by SPI initiatives). This situation may cause MSEs many difficulties (*e.g.* repeating the same mistakes, making ineffective decisions, depending on some very experienced people, increasing the time and cost of the initiative, and eroding the SPI knowledge or its eventual loss [12]), and sometimes, it can make the SPI initiatives fail [13]. We believe that the use of a Knowledge Management (KM) strategy tailored to the characteristics of MSEs can facilitate and increase the success of their SPI initiatives. The motivation for this research is to understand the KM process of the SPI knowledge in the MSEs. The core questions are:

- *How do MSEs perform their SPI knowledge management?*
- *Which are the factors that affect the SPI knowledge management in MSEs?*
- *How can we apply SPI knowledge management to foster a model-based SPI initiative in MSEs?*

To answer these questions, we conducted a case study involving five Mexican software development MSEs. During the case study, we analyze their SPI KM process, the problems that affect KM in MSEs, and how we can address them with a KM strategy. Later, we integrated this information and created the SPI KM system, Sharebox. Finally, we used a group of SPI experts and empirical studies in two organizations to evaluate the adoption of Sharebox. We organize this paper as follows. Section 2 introduces some theoretical background and related work on KM. Section 3 describes our study design and establishes our methodology. Section 4 discusses the findings of the study and presents our classification of SPI knowledge. Section 5 describes Sharebox. Section 6 explains the evaluation of adoption process of Sharebox. Finally, Section 7 provides some concluding remarks and some research directions for future work.

2   Theoretical background and related work

Nowadays, knowledge is a very important factor for organizations' competitiveness. Davenport and Prusak [14] consider knowledge as "a fluid mix of framed experience, contextual information, values and expert insight that provides a framework for evaluating and incorporating new experiences and information". Polanyi [15] classified knowledge into two dimensions: tacit and explicit. *Tacit knowledge* (TK) is more personal, difficult to formalize, and difficult to communicate and share with others. *Explicit knowledge* (EK) is codified and people can express it in words or numbers and shared in form of data, documents, manuals, etc. In addition, Nonaka [16] classifies knowledge as individual and collective. He also defines the knowledge creation as "knowledge spiral" in which there is a continuous interaction among individuals and continuous conversion from EK to TK and vice versa. This process has the following four basic conversion patterns: *socialization* (TK to TK), *externalization* (TK to EK), *combination* (EK to EK), and *internalization* (EK to TK).

2.1   Knowledge management

Organizations need to effectively capitalize their knowledge and use it as a source of grown and corporate profit; however, this action depends on the organizational efforts to explicitly manage their knowledge. Alavi and Leidner [17] define *Knowledge Management* (KM) as "an organizational systematic framework to capture, acquire, organize and communicate tacit and explicit knowledge to employees, so that employees can use it and thus be more effective and productive in their work, maximizing the knowledge of the organization". KM provides several benefits to organizations (*e.g.* better decision making, faster time to response to key issues, improvement of productivity, cost reduction, quality improvement, process improvement, learning, innovation, and increment of staff satisfaction) [18]. The KM process consists of the following six activities [18]:

1. *Identify knowledge.* The organization identifies the knowledge that is necessary to reach the objectives. The organization also identifies its present knowledge and knowledge gaps.
2. *Acquire/create knowledge.* This activity involves adding new knowledge to the knowledge base. Organization's employees create knowledge through learning, problem solving, innovating, or they obtain it from external sources.
3. *Capture/organize knowledge.* Organization's employees capture knowledge in explicit forms (*e.g.* written material or knowledge base systems). Additionally, organizations classify and group knowledge to allow employees to search and access specific knowledge.
4. *Access/transfer knowledge.* Organization's employees have access to knowledge search engines in the knowledge base. Additionally, the organizations distribute knowledge through education, training programs, knowledge-based systems, or expert networks.
5. *Apply knowledge.* The organization's ultimate goal is to apply knowledge (this is the most important part of the KM life cycle). In this situation, knowledge helps to create organizational capability (*e.g.* directives, organizational routines, and self-contained tasks).
6. *Maintain/update/evaluate knowledge.* This activity involves updating or evaluating existing knowledge. Organization's employees update or evaluate knowledge through learning, problem solving, or innovating.

To support the KM process, the researchers have developed several models, implementation guides, and technologies. Abecker *et al.* [19] argue that it is necessary to identify and structure the elements of KM and their interactions. They proposed a KM model that requires a hybrid solution that involves people and technology. Wong and Aspinwall [20] argue that the lack of support and guidelines can make organizations struggling with KM and unable to reach their full potential. They proposed a KM implementation framework that synthesizes the existing KM implementation frameworks and related KM literature. Lindvall *et al.* [21] argue that KM relies heavily on technology, but it is important to appreciate that technology alone will never be the solution to KM. They specify that a KM system implementation needs to address the socio-cultural and organizational components to assure its acceptance and success. Additionally, they present a literature review of KM technologies.

2.2   KM, SPI and MSEs

KM is an expanding and promising discipline in the software industry. Software organizations need to perform several knowledge activities related to documenting and improving their software processes in an effective and fast way to cope with pressing business issues (*e.g.* decrease time and cost, increase quality, make better decisions, and adapt to market changes) [22]. Meehan and Richardson [23] claim that KM is a core component of SPI initiatives where employees continuously create, capture and transfer SPI knowledge. Espinosa-Curiel *et al.* [24] argue that KM is a key factor that influences the success of SPI initiatives and impact the processes, the people, the organization, the improvement project, and the development process. Several researchers developed methodologies to support KM in SPI. Santos *et al.* [25] uses KM to support their proposed process-centered approach strategy, called SPI-KM. They based their strategy on well-known PRMs and standards. Montoni *et al.* [26] present a KM approach to support SPI initiatives. This approach captures knowledge related to critical success factors, issues, and best practices of SPI implementations. In addition, some researchers proposed KM tools. Montoni *et al.* [6] developed an SPI system that provides support to the KM of software development processes, preserves organizational knowledge, and fosters the initialization of a learning software organization. In another study, Montoni *et al.* [26] proposed a tool that integrates SPI and KM technologies. Wong and Aspinwall [20] argue that most of the literature on KM focuses on large organizations and it usually neglects KM in small settings; however, they specify that KM has the same importance in large and small organizations. Since MSEs have special characteristics and needs, their KM strategies have particularities in scope, scale, and how it will be carried out. Basri and O'Connor [27] conducted a case study in an MSE with an SPI initiative and identified seven characteristics of the MSE environment that influence the KM process. Although some researchers have conducted several studies on KM in software organizations, few of these studies focus on SPI in the context of MSEs. As far as we know, none of these studies explicitly investigates KM in SPI in a holistic way (considering the personal, social, and technical knowledge) and at the same time, focusing on the process, people and technology. Figure 1 shows the elements of an MSE and the elements of the model-based SPI initiative. Usually, MSEs do not consider the SPI KM process (black box on the right side of the figure), but it is important to perform an SPI initiative.

Figure 1. KM in a Model-based SPI initiative in MSEs
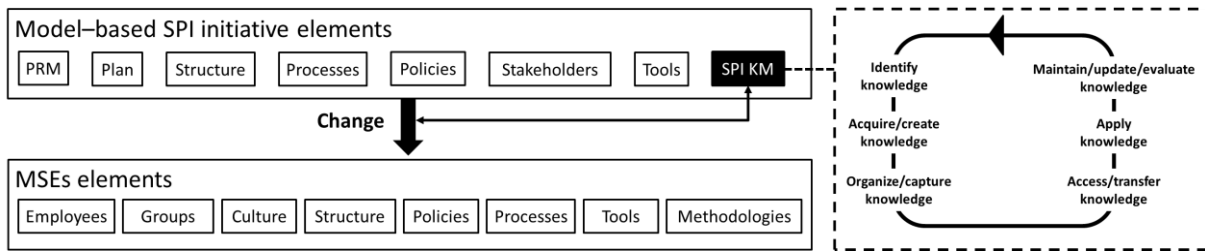
3   Methodology

To address our research questions, from 2010 to 2012, we conducted a case study involving five Mexican MSEs. All the organizations explicitly agreed to participate in the case study. The study includes two organizations that quit their SPI initiatives, one organization that was starting the SPI initiative, and the other two had a mature SPI initiative. We include organizations that quit their SPI initiatives to understand the influence of knowledge on the failure of SPI initiatives. All of the organizations were adopting the PRM MoProSoft [5]. This PRM, unlike others, includes the business management and resources management processes. For this reason, the adoption of MoProSoft involves top managers and middle managers not directly related to software development. There were 27 SPI stakeholders involved in the case study. The participants include: top managers, SPI leaders, SPI team members, experts, and users. Table 1 describes, for each organization, its number of employees, the duration of its SPI initiative in months (at the moment the study started or at the moment the organization quit the initiative), and the number of employees that participated in the study. We do not claim this is a statistically representative sample; therefore, we focus on the richness of the data. It is important to notice that these organizations faced real SPI issues on a daily basis; therefore, we have high confidence in the accuracy and validity of the data we gathered.

Table 1. Profile of the participating organizations

Name	Number of employees	Duration of the SPI initiative (months)	Number of participants	Continue in SPI
Organization 1	10	12	4	No
Organization 2	15	6	5	No
Organization 3	20	4	5	Yes
Organization 4	34	25	9	Yes
Organization 5	8	17	4	Yes

During the case study, we conducted semi-structured interviews. We interviewed employees of every SPI role. The interviews focused on the identification of the sources of knowledge, the activities to identify, create, maintain, update, evaluate, access, transfer, and re-use knowledge, the knowledge created and used by SPI stakeholders, the problems that affect KM in MSEs, and how we can address them with a KM strategy. The interview duration was on average half an hour; however, the researcher and the interviewee had the liberty to determine the pace of the interview. Before the interview, we made the arrangements regarding the time and place considering the interviewees' opinion. We recorded the interviews for later transcription. We transcribed the interviews and to analyze the data, we use the qualitative content analysis method [28]. In addition, to have a systematic data coding, we use a grounded theory (GT) [29] data coding process. We conducted two coding phases of GT: open and axial. To increase result precision of the research and to ensure the validity of the results, we used triangulation in the case study. Triangulation means taking different angles towards the studied object to provide a broader picture [30]. We used data source triangulation, because we had five data sources (organizations) and we collected the same type of data.

4   Findings and discussion

In this section, we describe and discuss the findings identified from the information analysis. We grouped these findings in the following four groups:

Characteristics of MSEs. MSEs mainly focus on core business processes. They are busy with their day to day operations; they usually have severe time and resource restrictions, and their employees are overloaded with work. The MSEs' managers (frequently, they are also the owners) usually have little formal training for an effective business management. The culture of MSEs is dynamic and fluid. Because of their strong influence, managers strongly impact and shape the MSEs' culture. Commonly, MSEs have a high rate of staff turnover. In MSEs, it is hard to assign dedicated staff to implement an SPI initiative. Since the stakeholders have low specialization in SPI, MSEs depend heavily on external SPI initiative consultants.

Characteristics of SPI initiative. Frequently, the SPI initiatives in MSEs have a simple five-level structure. Levels one to five correspond to the manager/sponsor, the SPI leader, the SPI team members, the process owners, and the users, respectively. Additionally, the experts or consultants collaborate with the people of the first three levels and usually they have a strong influence on the direction of the SPI initiative. The SPI activities are informal with low standardization and their division is unclear. Stakeholders normally know each other well, have face-to-face contact, and their collaboration is often good. The communication among them is short and direct and their interactions are informal and poorly documented.

SPI knowledge management. In general, MSEs lack of formal and integrated SPI KM activities. We identified the following SPI KM activities:

- *Identify knowledge.* MSEs focus their attention to define a general SPI plan, and frequently they do not make a plan of the knowledge needed to adopt a PRM. Usually, the consultants define the SPI knowledge needed by the SPI initiatives and they provide some support material and templates. Existing SPI knowledge is difficult to find and, when found, it is not reusable. Because of this situation, for MSEs, it is hard to determine the gap of knowledge that they have within their SPI initiative.

- *Acquire/create knowledge.* In general, to acquire knowledge, SPI stakeholders depend mainly on self-learning, the support of coworkers, and on the guidance of the SPI consultants and experts. The SPI stakeholders use external sources of information such as books, web pages, templates, documents, and formal training courses. However, these courses focus mainly on understanding the process of a specific PRM and lack of action knowledge. Moreover, the creation of knowledge occurs when SPI stakeholders perform their SPI activities and responsibilities.

- *Organize/capture knowledge.* SPI stakeholders focus mainly on the SPI activities rather than on the SPI documentation process. The documentation of the used and created knowledge during SPI activities is usually scarce, informal and individual. Stakeholders document most of the information in digital format. The SPI leaders and some SPI team members perform most of the documentation process, but they usually focus only on describing their activities. In general, the SPI stakeholders felt that a lot of the SPI knowledge was being wasted.

- *Access/transfer knowledge.* In general, MSEs do not have a tool to manage their SPI knowledge. Stakeholders transfer the SPI knowledge in informal ways (*e.g.* through open and direct discussions with other stakeholders and by providing informal guidance, printed or digital material, etc.). Sometimes the SPI stakeholders that attended the training courses replicate the information they learned within the company.

- *Apply knowledge.* Usually, the application of the SPI knowledge is individual. It does not become an organizational capability like self-contained tasks or routines.

- *Maintain/update/evaluate knowledge.* In general, MSEs lack of a systematic approach to maintain, update and evaluate their knowledge. Stakeholders update SPI knowledge in informal ways (*e.g.* through open and direct discussions with other stakeholders and by receiving informal guidance, printed or digital material, etc.).

Classification of SPI knowledge. We sorted the identified knowledge by using three hierarchical levels. We got this classification from the categories obtained by the qualitative content analysis, and the classification proposed by Espinosa-Curiel *et al.* [31]. The first level divides knowledge into three groups: personal, social, and technical. The *personal knowledge* is the knowledge that stakeholders require to know how to be and how to behave. The *social knowledge* is the knowledge that stakeholders require to know how to be with others. The *technical kno*wledge is the knowledge that stakeholders require to know what to do and how. The second level divides the personal, social and technical categories into subcategories. The third level makes a further division of the subcategories (see Table 2). In addition, we identified the following knowledge attributes:

- *Stage influence:* It refers to the stage(s) of the SPI initiative in which specific knowledge is important. Espinosa-Curiel *et al.* [24] specify that the stages of the SPI initiative are: pre-adoption,

general use, and continued use.

- *Roles influence:* It indicates which SPI roles have to know specific knowledge. Espinosa-Curiel *et al.* [31] specify that the SPI roles are: expert, sponsor, steering committee member, SPI leader, SPI team member, process owner, and user.
- *Focus:* It refers to the focus of the knowledge. Zack [32] classifies knowledge in the following six categories: declarative (know-about), procedural (know-how), causal (know-why), conditional (know-when), relational (know-with), and pragmatic (experiences and lessons learned).
- *Related knowledge.* It indicates which additional knowledge a stakeholder needs to understand specific knowledge.

Table 2. SPI knowledge classification

First classification	Second classification examples	Third classification examples
Personal	Self-awareness	Emotional awareness
	Self-management	Emotional self-control
Social	Social awareness	Organizational awareness
	Relation management	Team work and collaboration
Technical	Managerial	Organizational change management
	Process and SPI	Process modeling

The analysis shows that it is difficult for MSEs to understand the importance, the benefits, the requirements of SPI KM, and how to accomplish it. Managers strongly influence the direction of SPI initiatives and they assign the importance to the KM of the SPI. Stakeholders do not have time to think on KM strategies; however, they recognize the importance of KM, especially because of the high turnover rate. Stakeholders need knowledge from several areas that include technical, personal, and social aspects; however, they are poorly specialized in SPI and they usually do not dominate their assigned activities. This situation forces them to seek knowledge from diverse sources. The collaboration among the stakeholders is good; however, there are improper KM and knowledge losses over time because the interactions among stakeholders are informal and poorly documented. The informal ways in which stakeholders conduct their SPI activities may inhibit the implementation of a formal and comprehensive KM system. From the analysis of information, we identify that an SPI KM system should structure knowledge, facilitate KM at an organizational and personal level, should be "light" (requires little employee time), should be "easy" (requires little knowledge to use it), should offer the greatest extent of features (to facilitate the KM process) and should offer technological support. The KM system should be integrated, as much as possible, into the SPI activities carried out by the stakeholders.

5   SPI KM system Sharebox

To identify the main characteristics and functions of KM systems, we reviewed 15 papers[1]. Figure 2 shows the nine main categories of KM systems functions and a pair of examples for each category. We integrated the previous categories with the characteristics we identified in the case study. Finally, we developed an SPI KM system, called Sharebox[2]. This system can manage knowledge related with the changes of any element of the MSEs (see Figure 1). Sharebox has a web-based client-server architecture platform and its core is the Content Management System (CMS) Joomla. The user accesses the system through a web browser and the system is installed in a web server with a database server. The system offers the stakeholders the functionality to create and manage their profiles. Sharebox allows stakeholders to browse the knowledge categories, consult and evaluate specific knowledge, create knowledge through predefined templates, define access permissions, see statistics about the use of their created knowledge, consult their history of accessed knowledge, and subscribe to receive updates of important knowledge. When the stakeholders consult knowledge, they can add information or some attributes (*e.g.* related knowledge), add practical knowledge (*e.g.* lessons learned), attach files or export knowledge. Sharebox permits SPI leaders to add knowledge categories and subcategories. Sharebox also suggests, in some cases, relevant knowledge to reduce the searching time. Finally, Sharebox includes popularity lists of knowledge, some security features, and a simple persuasive evaluating system to motivate the use and creation of knowledge.

---

[1] To see the list of papers: http://gisep.com.mx/index.php?option=com_jdownloads&Itemid=231&view=viewdownload&catid=4&cid=17

[2] To see the screenshots http://www.gisep.com.mx/index.php?option=com_jdownloads&Itemid=231&view=viewdownload&catid=4&cid=12

Figure 2. Categories of KM systems functions

Storage	Classification	Visualization	Access	Security
• Knowledge templates • Multimedia content storage	• Knowledge classifications • Maps of experts	• Knowledge indexes • Knowledge reports	• Knowledge search • Knowledge navigation	• Authentication • Security levels

Communication	Collaboration	Evaluation	Integration
• Chats • Question boards	• Social networks • Collaborative knowledge creation	• Usability of information • Reputation evaluation	• Workflows • Legacy systems

6 Evaluation of the adoption of Sharebox

We evaluated the system in two stages. In the first stage, we interviewed SPI professionals. In the second stage, we conducted an empirical evaluation.

SPI Professionals' opinions. We qualitatively evaluated the proposed system by conducting five semi-structured interviews. The interviewees had previously participated in some SPI initiatives. The purpose of the interviews was to find out: 1) if the interviewees were aware of the SPI knowledge are-as; 2) if Sharebox was relevant to the SPI initiative; 3) if the interviewees were willing to try out Sharebox in practice. We checked that the interview questions were clear and understandable and we avoided questions that induce responses. We divided the interviews into two parts: in the first part, we collected their SPI experiences, identified the knowledge needed by stakeholders, and their opinion about the influence of Sharebox in the success of the SPI initiative. Prior or during the first part of the interview, we did not mention any details about Sharebox.

The majority of the interviewees agreed that there is a lack of specific action knowledge and experi-ence related to the conduction of an SPI initiative in MSEs. This lack of knowledge causes a strong dependence of the MSEs on SPI consultants. The interviewees agreed that usually, the organization only provides technical knowledge to the stakeholders (*e.g.* a hardcopy of the PRM and some tem-plates). MSEs often underestimate or do not provide knowledge to support the changes at personal, social and organizational levels. Because of these omissions, stakeholders do not know how to effec-tively perform some of their activities and how to fulfill their responsibilities, without increasing time and cost of the initiative. The interviewees also recognize that usually MSEs do not implement a KM strategy to capture and share the SPI knowledge and experiences. Usually this knowledge disappears over time or when employees leave the organization. Finally, the interviewees agreed that a KM strat-egy can ease the activities of stakeholders and contribute to the success of the SPI initiative.

In the second part of the interview, we presented Sharebox. Then, we collected their opinions about this system, the knowledge classification, and its application in the SPI initiative. The interviewees mentioned that it was useful that Sharebox supports the capture and share of knowledge. They men-tioned that it was useful that Sharebox classifies the knowledge with different criteria (*e.g.* according to categories and sub-categories, SPI stages, roles, focus). Additionally, they mentioned that Sharebox facilitates the management, searching and updating of knowledge. They argued that the current PRMs do not explicitly specify the knowledge considered by Sharebox. The interviewees are aware of some of the knowledge categories considered by Sharebox. They mainly identified the knowledge included in the technical categories and some of the personal and social categories. They emphasized that personal and social knowledge are fundamental for a successful SPI initiative because the employees are its driving force. The interviewees considered that Sharebox does not guarantee the success of the SPI initiative; however, they think that having a system that provides the right knowledge at the right time can help to improve their chances of success in the adoption of a PRM. The interviewees agreed that Sharebox can help to make better decisions, reduce the dependence on certain employ-ees, learn from past experiences, avoid repeating the same mistakes, and ease the training of new stakeholders. Sharebox seems to be easy to use and intuitively appealing to the interviewees, since most of them agreed to try out Sharebox in practice. This was an encouraging result from the inter-views; however, they were concerned about the seriousness and reliability of the information stored in the knowledge base, the anonymity of the information, and the time needed to capture knowledge.

Empirical evaluation. Two organizations are currently evaluating Sharebox in their SPI initiatives. The purpose of the evaluation is to determine whether Sharebox works as a support tool for the organizations and for the stakeholders, and to identify the factors that influence its adoption. The organizations started the adoption of Sharebox with the aim of reducing the time of learning, facilitating the capture, formalizing and increasing the knowledge for the stakeholders, and increasing the collaboration and communication among them. The organizations used Sharebox in the six stages of the KM (see Section 2.1). First, the organizations identified knowledge gaps and the knowledge that is important to their stakeholders. Later, they acquired and created their knowledge and captured it in Sharebox. Finally, in an asynchronous cyclic loop, they performed the following activities: they applied the knowledge, transferred the knowledge, and maintained, evaluated, and updated the knowledge. We interviewed the stakeholders of both organizations to know their opinions about the application of Sharebox. The interviewees agreed that the use of Sharebox facilitates the capture of knowledge because it provides templates. They also agreed that the search of knowledge is easy because it is similar to search information in the web, and they considered that the knowledge classification helped them to keep knowledge organized and to take better decisions. The stakeholders agreed that KM in their organizations is more effective, because Sharebox can provide the right information at the right time; however, they depend on the existing knowledge. The stakeholders agreed that Sharebox can help to support their professional development in a holistic way (personal, social, and technical) during the SPI initiative. Finally, the stakeholders agreed that Sharebox can support the training of new employees because they have staff turnover. We identified the following barriers in the adoption of Sharebox:

- *Integration.* The employees are overloaded with work and do not have enough time to store the acquired or created knowledge. Therefore, it is necessary to define a strategy to integrate the SPI KM activities within SPI activities and to define formal time periods to use Sharebox.
- *Support.* Managers should take an active role and exemplify the desired behavior for KM. They should also create an environment that encourages the preservation, use and transfer of knowledge. It is also necessary to provide training, support material (documents, manuals, videos, etc.) and templates.
- *Policies.* It is necessary to define a clear policy manual to regulate the use, capture, update and share of knowledge. The use of this manual may increase the quality and reliability of information. Also, it is necessary to provide mechanisms to manage access to knowledge and, when required, the anonymity of the information.
- *Awareness, commitment, and incentive.* It is very important that the whole organization has the commitment to include Sharebox in all the SPI activities, conduct activities to increase the awareness of the importance of KM, and implement incentives to recognize and to potentiate the use of Sharebox and knowledge creators.
- *Visibility.* It is necessary to identify the early adopters of Sharebox and include them in a strategy to strengthen the "visibility of the use" of Sharebox.

We performed the case study in Mexico. To generalize the result obtained from the case study and the evaluation of the adoption of Sharebox, it would be necessary further studies.

7 Concluding remarks and future work

During an SPI initiative, organizations use and create knowledge from several areas. Organizations need to manage this knowledge to improve decision making and to facilitate learning for SPI stakeholders. The stakeholders need knowledge on personal, social and technical levels. The personal knowledge helps stakeholders to know how to be and how to behave. The social knowledge helps stakeholders to know how to be with others. The technical knowledge helps stakeholders to know what to do and how. In this research, we propose a KM system called Sharebox. The objective of Sharebox is to support the management of the SPI knowledge. Organizations can use this system to identify, create, maintain, update, evaluate, access, transfer and apply knowledge. To successfully adopt Sharebox, we identified that MSEs need to implement awareness and motivation strategies, gain the commitment of the whole organization, provide support and training resources, develop policies of KM, monitor the use of the system, and recognize the employees that create knowledge. Finally, it is important to highlight that the impact of Sharebox depends on the amount and quality of knowledge managed by the organization. We are planning a more exhaustive evaluation of Sharebox in other organizations and during longer periods. Another short-term goal is to study the SPI

knowledge sharing among several software development MSEs. Finally, we are planning to integrate Sharebox with a social network system to facilitate the interaction and collaboration of the stakeholders and to study KM in this setting.

8 Literature

[1]   M. E. Fayad, M. Laitinen, and R. P. Ward, "Thinking objectively: software engineering in the small," *Commun. ACM*, vol. 43, no. 3, pp. 115–118, 2000.

[2]   M. C. Paulk, B. Curtis, M. B. Chrissis, and C. V. Weber, "Capability maturity model, version 1.1," *IEEE Software*, vol. 10, no. 4, pp. 18–27, 1993.

[3]   O. Demirors and E. Demirors, "Software process improvement in a small organization: difficulties and suggestions," in *Software Process Technology*, vol. 1487, V. Gruhn, Ed. Springer Berlin / Heidelberg, 1998, pp. 1–12.

[4]   R. V. Horvat, I. Rozman, and J. Gyorkos, "Managing the Complexity of SPI in Small Companies," *Softw. Process: Improve. Pract.*, vol. 5, no. 1, pp. 45–54, 2000.

[5]   H. Oktaba and M. Piattini, *Software Process Improvement for Small and Medium Enterprises: Techniques and Case Studies*. IGI Global, 2008.

[6]   M. Montoni, G. Santos, A. Rocha, S. Figueiredo, R. Cabral, R. Barcellos, A. Barreto, A. Soares, C. Cerdeiral, and P. Lupo, "Taba Workstation: Supporting Software Process Deployment Based on CMMI and MR-MPS.BR," in *Product-Focused Software Process Improvement*, vol. 4034, J. Münch and M. Vierimaa, Eds. Springer Berlin / Heidelberg, 2006, pp. 249–262.

[7]   M. Niazi, "Software Process Improvement: A Road to Success," in *Product-Focused Software Process Improvement*, vol. 4034, J. Münch and M. Vierimaa, Eds. Springer Berlin / Heidelberg, 2006, pp. 395–401.

[8]   D. Mishra and A. Mishra, "Software Process Improvement Methodologies for Small and Medium Enterprises," in *Product-Focused Software Process Improvement*, vol. 5089, A. Jedlitschka and O. Salo, Eds. Springer Berlin / Heidelberg, 2008, pp. 273–288.

[9]   M. Niazi, D. Wilson, and D. Zowghi, "Critical success factors for software Process Improvement Implementation: an Empirical Study," *Software Process: Improvement and Practice*, vol. 11, no. 2, pp. 193–211, 2006.

[10]   L. Mathiassen, O. K. Ngwenyama, and I. Aaen, "Managing change in software process improvement," *IEEE Software*, vol. 22, no. 6, pp. 84–91, 2005.

[11]   D. Moitra, "Managing organizational change for software process improvement," in *Software Process Modeling*, vol. 10, S. T. Acuña and N. Juristo, Eds. New York: Springer-Verlag, 2005, pp. 163–185.

[12]   A. Wickert and R. Herschel, "Knowledge-management issues for smaller businesses," *Journal of Knowledge Management*, vol. 5, no. 4, pp. 329–337, Jan. 2001.

[13]   S. Komi-Sirviö, A. Mäntyniemi, and V. Seppänen, "Toward a Practical Solution for Capturing Knowledge for Software Projects," *IEEE Software*, vol. 19, no. 3, pp. 60–62, 2002.

[14]   T. H. Davenport and L. Prusak, *Working knowledge: how organizations manage what they know*. Harvard Business Press, 1998.

[15]   M. Polanyi, *The Tacit Dimension*. NY, USA: Doubleday, 1966.

[16]   I. Nonaka, "A Dynamic Theory of Organizational Knowledge Creation," *Organization Science*, vol. 5, no. 1, pp. 14 –37, Feb. 1994.

[17]   M. Alavi and D. E. Leidner, "Review: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues," *MIS Quarterly*, vol. 25, no. 1, pp. 107–136, Mar. 2001.

[18]   M. Alavi and D. E. Leidner, "Knowledge management systems: issues, challenges, and benefits," *Commun. AIS*, vol. 1, no. 2, Feb. 1999.

[19]   A. Abecker, A. Bernardi, K. Hinkelmann, O. Kuhn, and M. Sintek, "Toward a technology for organizational memories," *IEEE Intelligent Systems and their Applications*, vol. 13, no. 3, pp. 40–48, Jun. 1998.

[20]   K. Y. Wong and E. Aspinwall, "Knowledge management implementation frameworks: a review," *Knowledge and Process Management*, vol. 11, no. 2, pp. 93–104, Apr. 2004.

[21]   M. Lindvall, I. Rus, and S. S. Sinha, "Technology Support for Knowledge Management," in *Advances in Learning Software Organizations*, vol. 2640, S. Henninger and F. Maurer, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 94–103.

[22]   I. Rus and M. Lindvall, "Knowledge management in software engineering," *IEEE Software*, vol. 19, no. 3, pp. 26–38, May 2002.

[23]   B. Meehan and I. Richardson, "Identification of Software Process Knowledge Management," *Software Process: Improvement and Practice*, vol. 7, no. 2, pp. 47–55, Jun. 2002.

[24]   I. E. Espinosa-Curiel, J. Rodríguez-Jacobo, and J. A. Fernández-Zepeda, "A framework for evaluation and control of the factors that influence the software process improvement in small organizations," *Journal of Software Maintenance and Evolution: Research and Practice. Accepted for publication.*

[25]   G. Santos, M. Montoni, S. Figueiredo, and A. Rocha, "SPI-KM - Lessons Learned from Applying a Software Process Improvement Strategy Supported by Knowledge Management," in *Product-Focused Software*

Process Improvement, vol. 4589, J. Münch and P. Abrahamsson, Eds. Springer Berlin / Heidelberg, 2007, pp. 81–95.

[26]   M. A. Montoni, C. Cerdeiral, D. Zanetti, and A. R. Cavalcanti da Rocha, "A Knowledge Management Approach to Support Software Process Improvement Implementation Initiatives," in *Software Process Improvement*, vol. 16, R. V. O'Connor, N. Baddoo, K. Smolander, and R. Messnarz, Eds. Springer Berlin Heidelberg, 2008, pp. 164–175.

[27]   S. Basri and R. V. O'Connor, "Knowledge Management in Software Process Improvement: A case study of very small entities," in *Knowledge Engineering for Software Development Life Cycles: Support Technologies and Applications*, Ramachandran., IGI Global, 2011, pp. 273–288.

[28]   H.-F. Hsieh and S. E. Shannon, "Three Approaches to Qualitative Content Analysis," *Qual Health Res*, vol. 15, no. 9, pp. 1277–1288, Nov. 2005.

[29]   A. Strauss and J. Corbin, *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. CA, USA: Sage, 1998.

[30]   P. Runeson and M. Höst, "Guidelines for Conducting and Reporting Case Study Research in Software Engineering," *Empirical Software Engineering*, vol. 14, no. 2, pp. 131–164, 2009.

[31]   I. E. Espinosa-Curiel, J. Rodríguez-Jacobo, and J. A. Fernández-Zepeda, "A competency framework for the stakeholders of a software process improvement initiative," in *Proceedings of the 2011 International Conference on Software and Systems Process*, New York, USA, 2011, pp. 139–148.

[32]   M. Zack, "What Knowledge-Problems Can Information Technology Help to Solve?," *AMCIS 1998 Proceedings*, Dec. 1998.

Author CVs

Josefina Rodríguez-Jacobo

She received a BS degree in Physics in 1986, a M.S in Computer Science in 1993 and Doctoral degree in Psychology in 2003. She is currently working in the Computer Science Department at CICESE. Her main research interest include software process improvement in small companies, software engineering, knowledge management, and the social and personal factors related to software engineering.

Ismael Edrein Espinosa-Curiel

He received a BS degree in Computer Science in 2005, and MS in Computer Science in 2008. He is currently a Ph.D. student in Computer Science at CICESE. His research interests focus on software process improvement in small companies, software engineering, knowledge management, and the impact of social and personal factors in software engineering.

Ulises Gutierrez-Osorio

He received a BS degree in Computer Science in 2009, and MS in Computer Science in 2011. His research interests focus on software process improvement in small companies, software engineering, and knowledge management.

Violeta Ocegueda-Miramontes

She received a BS degree in Computer Science in 2008, and MS in Computer Science in 2010. Her research interests focus on software process improvement in small companies, software engineering, and knowledge management.

José Alberto Fernández-Zepeda

He received a BS degree in Electronic Engineering in 1991, a M.S in Electronic Engineering in 1994 and a Ph.D. in Computer Engineering in 1999. He is currently an associate professor in the Computer Science Department at CICESE. Some of his research interests include software process improvement in small companies, the impact of social and personal factors in software engineering, design and analysis of algorithms.

Overview of Risk Management Frameworks and Challenges of Their Implementation in IT-Centric Micro and Small Companies

Jasmina Trajkovski[1], Ljupcho Antovski[2]

[1]Trajkovski & Partners Management Consulting
Sveti Kliment Ohridski 24/2/1, 1000 Skopje, Macedonia
jasminat@tpconsulting.com.mk

[2]Faculty of Computer Science and Engineering
University Ss. Cyril and Methodius
Rugjer Boshkovikj 16, 1000 Skopje, Macedonia
ljupcho.antovski@finki.ukim.mk

Abstract

This paper is an overview of the risk management concepts, specifically the risk management frameworks, and argues their usability when applied to IT-centric micro and small companies. The findings in terms of (i) specifics of the IT-centric companies and (ii) the challenges for the implementation of risk management frameworks, are elaborated. Further work is necessary to fine-tune a suitable framework that will be useful and could be sustainably implemented into IT-centric micro and small enterprises.

Keywords

Risk management framework, IT-centric micro and small companies, ISO31000, ISO27005, enterprise risk management, operational risk, IT risk, information security risk

1 Introduction

In this paper we review the risk management concepts, specifically the risk management frameworks, and discusses their usability when applied to IT-centric micro and small companies. The focus on these specific companies is due to the direct experience of the leading author in the last 5 years with over 20 micro and small companies that are heavily IT-centric in their operations. In each of these companies, there has been a process of application of risk management frameworks, and specifically conducting the risk assessment exercises. The findings in terms of (i) specifics of the IT-centric companies and (ii) the challenges for the implementation of risk management frameworks, are elaborated towards the end of the paper, and will be used as basis for further PhD research on integrated risk management frameworks and the proposal of a usable model for valuation of risks for the micro and small companies.

This paper is structured in several segments. In Chapter 2, an overview of risk and risk management is provided together with introduction of the various types of risks. The following chapter reviews the risk management frameworks and standards, and in Chapter 4, the related research work is presented to show in which direction the academic community is further developing the set standards and frameworks. The last two chapters provide the contribution of the authors in the area of identification of the specific of risk management in IT-centric micro and small companies, as well as the commonly faced challenges in practical implementation of integrated risk management in such companies.

2 Types of Risks and Risk Management

As this paper is focused on discussing the specifics of risk management in IT-centric micro and small companies and the related challenges of implementation of integrated risk management frameworks, I would like first to briefly describe the main concepts. The main concepts are divided into 2 groups: (i) definition of risk, types of risks and risk management, and (ii) risk management frameworks and standards.

Based on NIST publication on risk management, risk is defined as: "A measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of: (i) the adverse impacts that would arise if the circumstance or event occurs; and (ii) the likelihood of occurrence."[1]
As the paper discusses risks management frameworks for IT-centric micro and small companies, the main focus are organizational risks. There are various types of organizational risks such as program management risk, investment risk, budgetary risk, legal liability risk, safety risk, inventory risk, supply chain risk, and security risk. [1]

For the needs of the IT-centric micro and small companies management, all these risks could not be approached independently, and an integrated approach is necessary. This approach should be focused on the main drivers in the company i.e. continual operations thru IT operation and known business processes so that the employees can understand what they should do. The reliance on IT as well puts the information security risks among the top as well. For the purposes of the research questions, we make the assumption that the management of these IT-centric micro and small companies deals with the legal and financial risks intuitively, and that they are not necessary to be included in the integrated risk management framework and approach of the company.

Having said that, for the purposes of the paper, we will look into the IT risk, information security risk and operational risk, which are respectively defined as:

- IT risk—that is the business risk associated with the use, ownership, operation, involvement, influence and adoption of IT within an enterprise. [8]
- Information security risk—that is, the risk associated with the operation and use of information systems that support the missions and business functions of their organizations.[1]

- Operational risk - The most common definition, first published in The Next Frontier and also adopted in recent operational risk documents issued by the Basel Committee, is that "Operational risk is the direct or indirect loss resulting from inadequate or failed internal processes, people and systems, or from external events." [2]

The next concept to be introduced is the risk management. NIST special publication SP800-39 [1] defines the risk management as a comprehensive process that requires organizations to: (i) frame risk (i.e., establish the context for risk-based decisions); (ii) assess risk; (iii) respond to risk once determined; and (iv) monitor risk on an ongoing basis using effective organizational communications and a feedback loop for continuous improvement in the risk-related activities of organizations. Risk management is carried out as a holistic, organization-wide activity that addresses risk from the strategic level to the tactical level, ensuring that risk based decision making is integrated into every aspect of the organization.

With the development of risk management as an organizational discipline, a more defined concept evolved, named Enterprise Risk Management (ERM). There are many definitions of ERM, but a representative one is from the COSO framework: "Enterprise risk management is a process, effected by an entity's board of directors, management and other personnel, applied in strategy setting and across the enterprise, designed to identify potential events that may affect the entity, and manage risk to be within its risk appetite, to provide reasonable assurance regarding the achievement of entity objectives" [3].

3 Overview of Risk Management Frameworks and Standards

Nowadays, there are several types of risk management methodologies, some of them issued by national and international organizations such as ISO, NIST, AS/NZS, BSI, others issued by professional organizations such as ISACA or COSO, and the rest presented by research projects. Each of these methods has been developed to meet a particular need so they have a vast scope of application, structure and steps. The common goal of these methods is to enable organizations to conduct risk assessment exercises and then effectively manage the risks by minimizing them to an acceptable level [4].

3.1 General Risk Management Frameworks

Existing risk management standards offer several variations to the risk management process, but they can be generalized in four parts, as analyzed by Corpuz and Barnes in their 2010 paper on integration information security policy into corporate risk management [5]. The four parts are:

- Risk assessment process (derived from Risk Management Policy);
- Risk treatment process (facilitating the development and implementation of the Risk Mitigation Plans);
- Risk communication process (facilitating awareness of the Risk Communication Plan);
- Risk review and monitoring process (assuring accuracy of the overall risk assessment).

The most generic and internationally accepted standard is the one supported by the International Standards Organization – ISO, called ISO31000:2009 - Risk management - principles and guidelines [6]. This standard provides generic guidelines for the design, implementation and maintenance of risk management processes throughout an organization. Its approach to formalizing risk management practices facilitates broader adoption by companies who require an enterprise risk management standard that accommodates multiple 'silo-centric' management systems. It as well addresses the entire management system that supports the design, implementation, maintenance and improvement of risk management processes.

3.2   IT Risk Management Frameworks

As elaborated by Badenhorst in the comparative framework for risk analysis methods [7], the objective of IT risk management is to protect IT assets such as data, hardware, software, personnel and facilities from all external (e.g. natural disasters) and internal (e.g. technical failures, sabotage, unauthorized access) threats so that the costs of losses resulting from the realization of such threats are minimized. The specific domains of IT Risk Management covered in his approach to the domain of IT Risk Management include (i) Risk Approach Completeness, (ii) Information Technology Completeness and (iii) Information Security Completeness.

Looking at the professional associations, ISACA is the one of the more prominent in developing risk management frameworks. It's RiskIT framework [8] was published in 2009 and it provides an end-to-end, comprehensive view of all risks related to the use of IT and a similarly thorough treatment of risk management, from the tone and culture at the top, to operational issues. The framework is focused around 3 domains: Risk governance, Risk evaluation and Risk response.

3.3   Information Security Risk Management Frameworks

In the area of information security risk management frameworks there are two which are most recognized. They include the NIST SP800-39 and the ISO27005:2009, and each of them is explained shortly in the following paragraphs.

NIST SP800-39: Managing Information Security Risk [1]. The model is based on 3 tiers and 4-phased risk management process. The tiers are: Organization, Mission / business processes and Information Systems, while the phases are Frame, Assess, Respond and Monitor.

The phase Frame, addresses how organizations frame risk or establish a risk context. The purpose of the risk framing component is to produce a risk management strategy that addresses how organizations intend to assess risk, respond to risk, and monitor risk. The phase Asses addresses how organizations assess risk within the context of the organizational risk frame. Its purpose is to identify: (i) threats to organizations; (ii) vulnerabilities internal and external to organizations; (iii) the harm to organizations that may occur given the potential for threats exploiting vulnerabilities; and (iv) the likelihood that harm will occur. The phase Respond addresses how organizations respond to risk once that risk is determined based on the results of risk assessments. Its purpose is to provide a consistent, organization-wide, response to risk in accordance with the organizational risk frame by: (i) developing alternative courses of action for responding to risk; (ii) evaluating the alternative courses of action; (iii) determining appropriate courses of action consistent with organizational risk tolerance; and (iv) implementing risk responses based on selected courses of action. The final phase is Monitor and it addresses how organizations monitor risk over time. Its purpose is to: (i) verify that planned risk response measures are implemented and information security requirements derived from/traceable to organizational and regulatory requirements are satisfied; (ii) determine the ongoing effectiveness of risk response measures following implementation; and (iii) identify risk-impacting changes to organizational information systems and the environments in which the systems operate.

ISO27005 Information Security Risk management [9]. The standard extends on the requirements from the ISO27001:2005 for risk management and provide guidance on how to implement those requirements. This risk management process is characterized with iterativeness of the approach in the phases of risk assessment or/and risk treatment. This contributes to the depth and detail o the risk management exercise but still balance it with the time and effort spent.

3.4   Operational Risk Management Frameworks

The final concept defined in this section covers operational risks management frameworks. The leading two frameworks are the one published by COSO, and the one published by the Risk Management Association – RMA. They are briefly described in the following paragraphs.

COSO Enterprise risk management integrated framework [10]. In 2004, the Committee of Sponsoring Organizations of the Treadway Commission – COSO, released its Enterprise Risk Management Integrated Framework. This framework is based on principles and it's designed to apply to companies of all sizes, providing components to identify, assess, respond to and control risk. As such is intended to serve as a roadmap for management and company boards for identifying risk, avoiding pitfalls and seizing opportunities for growth.

RMA Operational risk management framework [11]. The main reason for developing management frameworks in any company is that, when well designed and implemented, they improve economic performance. There are four basic elements in the RMA Operational Risk Management Framework: 1. leadership, 2. management, 3. risk, and 4. tools. Leadership is about creating the right culture, overseeing management, and creating the right environment for the various processes of risk management. Management is a set of processes that channel and control the institution's risks. Risk involves understanding the pattern of operational risks the institution faces and takes on. Tools refers to the collection of guides, templates, libraries, services, training, and software that are available to help implement the other elements of the framework.

4 Related Work on Integrated Risk Management Frameworks

From the experience in direct operations with the IT-centric micro and small companies, it was seen that the standardized risk management frameworks are hard to integrate when the company needs to cover more than one type of risks. As as well, most of them are too cumbersome for implementation for one company with less than 50 employees. This has led to further research on integrated risk management framework, extending the existing concepts of the recognized and standardized risk management frameworks. Several published research papers on this topic are described briefly in the following paragraphs.

In 1999, a paper was published by Bandyopadhay and his associates in the Management Decision Journal on Integrated risk management framework for IT[12]. In their work they describe an IT risk management framework which focuses on four major risk management components: risk identification, risk analysis, risk-reducing measures and risk monitoring. The described framework does not emphasize any one particular component of the risk management process but concentrates on the sequential linkage of the four components to make up the entire system of IT risk management. In their opinion, their approach is an improvement over other approaches (that address one or more of the components in an isolated manner), because it should enable managers to smoothly move from one component to another by identifying and understanding the possible courses of action in the different steps.

In 2004, the Committee of Sponsoring Organizations of the Treadway Commission – COSO, released its Enterprise Risk Management Integrated Framework [10]. This framework is based on principles and it's designed to apply to companies of all sizes, providing components to identify, assess, respond to and control risk. As such is intended to serve as a roadmap for management and company boards for identifying risk, avoiding pitfalls and seizing opportunities for growth.

In 2007, in IBM Systems Journal, Abrams and associates published an optimized enterprise risk management framework [13]. This framework was designed to address risk and compliance management in a strategic, integrated, and comprehensive manner. In the paper they show how enterprises evolve along an enterprise risk-management maturity continuum from a state of mere penalty avoidance through a state of improvement until they finally reach a state of continuous, risk-based transformation. Their model depicts how people, processes, and technology interact in an enterprise. In the framework, they model an enterprise and its environment in five layers: jurisdiction, strategy, deployment, operation and events. On top of the model, they define an enterprise risk management approach which includes the following steps: gaining executive sponsorship, assessing the current state, developing an ERM vision, developing capabilities, planning implementation, monitoring and governance.

In 2011, in Applied Computer and Informatics, Saleh and Alfantookh published their comprehensive information security risk management (ISRM) framework for enterprises using IT [4]. They based their

structure of the framework on the STOPE (strategy, technology, organization, people, and environment) view. In their opinion, it is important to have such a structure for the framework, specifically when its focus is on information security issues, and at the same time, they have associated the process defined in the framework to the already recognized and proven six sigma phases - DMAIC (define, measure, analyze, improve, and control) cyclic phases.

In their research paper, Saleh and Alfantookh as well identified that in spite of the increasing number of standard and commercial risk management methods, the absorption of risk management in organization is limited due to lack of awareness, high cost, need for expertise, and long process, and that the trust in these methods is very low due to the poor results, bulky confused reports and the narrow technological scope.

5 Specifics of Risk Management in IT-Centric Micro and Small Companies

As part of the consulting work and support of implementation of risk management frameworks in over 20 IT-centric micro and small companies in the Balkans in the past 5 years, we have found that standardized frameworks and methodologies for risk management are not well received by the companies, as they are difficult and cumbersome for implementation.

These findings were the motivation to start researching which company and/or environmental characteristics had impact on the applicability of standardized risk management frameworks and methodologies. The research was based on the private sector companies where standardized risk management frameworks were being implemented, and it resulted in identification of 3 specifics which have to be taken into account for successful implementation of the risk management framework. These specifics are:

- Exposure to various types of risks
- Limited resources for risk management
- Low resilience of the organizations to operations and information security risks

The following paragraphs describe these specifics in more detail.

Micro and small companies, as any other company or organization, are exposed to various types of risks in their daily operations: operations risk, market risks, information security risks, strategic risks, etc. To deal with these risks the management of the company needs a risk management framework which can deal with these various types of risks in a consistent way in order to have a clear picture of the environment and threats that they are exposed to. What makes this a specific factor for micro and small companies is that fact most risk management frameworks are focused on a given group of risks, such as operational risks, financial risks, IT risks, information security risks, and even though most of the frameworks, as described in the previous chapters, have comparable phases and/or steps, they all have a different approach to risk assessment and valuation which complicates their implementation in a situation where various types of risks need to be taken into consideration.

Risk management frameworks as described in the previous chapters, imply existence of appropriate resources, such as human, technological, financial and time, in order to have a sustainable risk management process. Unfortunately, most of the micro and small companies are not capable to dedicate such resources on a full time or even part-time basis. Their access to automated tools for risk assessment is limited due to the costs of such tools. The competences of the staff as well are limited in terms of concepts, knowledge and skills for performing risk management exercises, and available training is as well costly. Finally, the overall needed investment in risk treatment is often more than the micro or small company can afford, and thus they find the exercise frustrating or unusual.

IT-centric micro and small companies as any other type of company are exposed to various types of risks but due to the nature of their operations and the reliance on information technology on one hand, and the intrinsic optimism and over-confidence that nothing bad will happen on the other, they in general have a very low resilience to information security risks. Due to the nature of their operations usually such companies are technically well equipped and cover various elements of the technical

controls for risk treatment, but the underlying organization, processes, and awareness of the staff are less developed which makes them specifically vulnerable to operations and information security risks.

6 Challenges of integrated Risk management in IT-Centric Micro and Small Companies

Additionally to the research regarding the specifics of IT-centric micro and small enterprises when it comes to risk management, other areas of interest were the challenges for integrated risk management to be implemented in such companies. The research was based on the feedback from the companies where standardized risk management frameworks were being implemented and on the elaborated risk management frameworks.

The main findings from this research are that several challenges are posed to those who are trying to define a comprehensive integrated risk management framework tailored to IT-centric micro and small companies. These challenges can be grouped to meet the following requirements:

─ Need for integrated approach to treat various types of risks
─ Need for comprehensive and usable methodology

Regarding the need for integrated approach to treat various types of risks, the integrated risk management framework for IT-centric micro and small enterprises should encompass the overall needs of the organization for risk management. In that line, it should be suitable for dealing with operations risk, IT risk and information security risk as they are most critical for resilience of operations and business continuity for IT-centric micro and small companies.

The integrated approach should as well include an adjusted valuation model as part of the risk assessment phase which can be used on various types of risks. This model should not be data intensive as experience shows that micro and small companies do not have access to historical data about risks, probabilities and impacts. It as well should not be based on complex calculations, require advanced skills, nor should it require too much time or people to conduct the risk assessment exercise.

Regarding the need for comprehensive and usable methodology, the integrated risk management framework for IT-centric micro and small enterprises should be comprehensive and should present an understandable set of steps and activities, with clear inputs and defined outputs. The elements of the organizational context should be clearly defined and should allow for small or even non-existent organizational hierarchies and procedures. The framework should allow for unclear segregation of duties, and for very high criticality of managers and/or owners, as well as lack of documented knowledge.

On the contrary, the framework and the defined phases, steps and activities should be usable in micro and small companies. From the experience in over 20 micro or small IT-centric companies in the Balkans, the average team for risk management included 3-5 people, usually departmental heads or other key experts in the company, with one of them in-charge for risk management on a part-time basis. The time allocated to the overall risk management effort on annually basis is approximately 5-10 days, inclusive of bi-annual review of the risk assessment and implementation of the risk treatment plan. Having these parameters in mind, the integrated risk management framework in order to be usable should be implementable with level of effort of 20-25man/days on annual basis.

7 Conclusion

This paper reviewed the risk management concepts, specifically the risk management frameworks, and discussed their usability when applied to IT-centric micro and small companies. The review covers both accepted standards and frameworks, and advanced new research in those areas. Towards the end of the paper, the contribution of the authors is provided. In the area of identification of the specific of risk management in IT-centric micro and small companies, the main findings are that these companies in terms of risk management are exposed to various types of risks and thus need a

framework which will comprehensively deal with all of these risks, they have limited resources for risk management and have low resilience to operational risk and information security risks. In the area of commonly faced challenges in practical implementation of integrated risk management in such companies, the findings are that the challenges are focused around the need for integrated approach to treat various types of risks and the need for comprehensive and usable methodology.

Based on the available risk management frameworks described in Chapter 3 and Chapter 4, as well as the defined specifics of IT-centric micro and small companies in Chapter 5 and the identified challenges in Chapter 6, one can conclude that further work is necessary to fine-tune a suitable framework which will be useful and can be sustainably implemented into IT-centric micro and small enterprises.

The development of such integrated risk management framework tailored to the IT-centric micro and small companies is will be the focus of our further research, and the outcome will be validated by attemts for its implementation in an adequate sample of such companies. The newly developed framework should take into consideration the competences and resources available in such companies, should be viable for dealing with various types of risks, and should not be dependent on huge amounts of historical data for the complex calculations of the value of risk.

The open questions which remain are concerned with the availability of adequate models for risk assessment i.e. valuation of risks so that the management of the micro and small companies can compare the various types of risks to which they are exposed, prioretize them and set appropriate risk mitigation controls.

8 Literature

1.   T.i. Joint Task Force, *Managing Information Security Risk, SP800-39*, 2011.

2.   M. Haubenstock, "The Evolving Operational Risk Management Framework," *The RMA Journal*, vol. 84, no. 4, 2001, pp. 5.

3.   , *Internal Control-Integrated Framework*, Committee of Sponsoring Organizations of the Tread way Commission (COSO), AICPA/COSO, 1992.

4.   M.S. Saleh and A. Alfantookh, "A new comprehensive framework for enterprise information security risk management," *Applied Computing and Informatics*, vol. 9, no. 2, 2011, pp. 107-118; DOI 10.1016/j.aci.2011.05.002.

5.   M. Corpuz and P.H. Barnes, "Integrating information security policy management with corporatre risk management for strategic allignment," *Proc. 14th World Multi-conference on Systemics, Cybernetics and Informatics (WMSCI 2010)*, 2010, pp. 7.

6.   ISO, "ISO 31000:2009 - Risk Management - Principles and guidelines," *Book ISO 31000:2009 - Risk Management - Principles and guidelines*, Series ISO 31000:2009 - Risk Management - Principles and guidelines, ed., Editor ed.^eds., ISO, 2009, pp.

7.   J.H.P.E.L.B.K.P. Badenhorst, "A comparative framework for risk analysis methods," *Computers & Security*, vol. 12, no. 6, 1993, pp. 6.

8.   ISACA, "The RISK IT framework," *Book The RISK IT framework*, Series The RISK IT framework, ed., Editor ed.^eds., ISACA, 2009, pp.

9.   ISO, "ISO/IEC 27005:2008 Information technology -- Security techniques -- Information security risk management," *Book ISO/IEC 27005:2008 Information technology -- Security techniques -- Information security risk management*, Series ISO/IEC 27005:2008 Information technology -- Security techniques -- Information security risk management, ed., Editor ed.^eds., ISO, 2008, pp.

10. M.K. Aguilar, "COSO releases a new risk management framework," *Accounting TOday*, vol. 18, no. 19, 2004, pp. 1.

11. C. Taylor, ;, "The RMA operational risk management framework," *The RMA Journal*, vol. 88, no. 5, 2006, pp. 3.

12. K. Bandyopadhyay, et al., "A framework for integrated risk management in information technology," *Management Decision*, vol. 37, no. 5, 1999, pp. 7.

13. C. Abrams, et al., "Optimized enterprise risk management," *IBM Systems Journal*, vol. 46, no. 2, 2007, pp. 15.

9 Author CVs

Jasmina Trajkovski,

MBA is an information security and IT governance consultant at Trajkovski & Partners, Skopje, Macedonia. She consults organizations from the public and private sector for implementation of IT Governance concepts, international management systems standards in the field of IT Service Management, Information Security, Business Continuity Management, and facilitates risk management exercises. She is the president of the IT Expert Council in MASIT – the national IT chamber of commerce and is a Board member of itSMF Macedonia. Jasmina is the Managing Director of Trajkovski & Partners Consulting, a boutique management consulting company with a regional scope of operations. Hers academic research is focused on development a usable integrated risk management framework tailored to micro and small companies and risk valuation model which can be used on various types of risks relevant for IT-centric SMEs. She has given numerous presentations and speeches at leading professional conferences in the region and has participated in several R&D projects.

Ljupcho Antovski,

PhD, is associate professor of software engineering at the Faculty of Computer Science and Engineering, St. Cyril and Methodius University, Skopje, Macedonia. He lectures courses in requirements engineering, software project management, e-business, semantically enabled computer architectures. He is the president of the IT technical committee at the Macedonian Institute for Standardization. Ljupcho is cofounder of iVote, leading integrator of elections management information systems in South-Eastern Europe. His academic research is focused on the requirements and usability engineering. Ljupcho has published widely in the fields of project management, e-government and e-voting. He has authored numerous papers in leading science journals and conference proceedings.

There is no Knowledge without Terminology: Key Factors for Organisational Learning

Blanca Nájera Villar[1] and Diana Brändle[2]

[1] *TermNet, Austria, events@termnet.org*
[2] *TermNet, Austria, projects@termnet.org*

Abstract

Any company or organisation in the world has to manage its knowledge. For some it may be sufficient to have the knowledge in their brains. But as products and services are subject to constant innovation, most of the companies and organisations acting in the global market or on an international level find themselves confronted with the task of constantly updating, referencing, tracking and managing knowledge. Marketing material, speeches on conferences and sophisticated tools make us believe that it is easy to keep track of information in a company or organisation, to use single source publishing, touch each content only once, have it centrally stored and reuse it for multiple purposes. How many of you have really achieved this goal or know organisations, where all content-related processes run smoothly and are fed by a central knowledge base? Is it actually realistic to follow the ideal of ONE central knowledge base in a company / organisation or just an utopian idea? We will have a closer look at challenges and possible solutions for innovative organisations and companies when managing units of knowledge represented in the form of technical words or „terms" in one or more languages.

Keywords

Terminology, multilingual knowledge base, single source publishing, organisational learning, learning organisation, terminological database, organisational knowledge, globalisation, multilingual, terms

Reference

Maturity Model for Quality in Brazilian Public Software

Angela M. Alves[1,2], Marcelo Pessoa[1], Clênio F. Salviano[2]

[1] USP: Universidade São Paulo
Av. Prof. Almeida Prado, Travessa 2, Nº 128 Cidade Universitária São Paulo – SP Brazil- 05508-070
angela.alves@usp.br, alvesam@gmail.com, marcelo.pessoa@usp.br

[2] CTI: Centro de Tecnologia da Informação Renato Archer
Rodovia D. Pedro I, km 143.6, CEP 13069-901, Campinas, SP, Brazil
angela.alves@cti.gov.br, clenio.salviano@cti.gov.br

Abstract

Brazilian Public Software (BPS) is an innovative experience in public administration. It combines features from the free software production model with the public goods concepts and is delivered by a portal that links different people and interests. This article summarizes a doctoral dissertation and describes how methodologies based on System Thinking, part of Complex Thinking Theory, were used to obtain empirical evidence that the BPS ecosystem evolves in learning cycles, with an emphasis on cooperation and connection. It proposes a systemic maturity model and provides concepts to a framework for understanding and improving BPS and others Public Software Ecosystems, as, for example, Certification in National Technology for Information and Communication Technology ecosystem that is under construction, and that uses the concepts defined in BPS, and that feeds back the concepts defined latter. The references for a systemic maturity model has been developed using an analogy with ISO/IEC 15504 references for capability maturity models, the Method Framework for Engineering Process Capability Models and Action Research. The finding from the research is consistent with the current evolution of ISO/IEC 33000 series towards others paths for maturity in addition to capability.

Keywords

Software Process Improvement, Software Ecosystem, Systemic Maturity Model, Complex System, Action Research

1 The Research Problem and Its Relevance

Brazilian Public Software (BPS)[1] [1] is an innovative experiment in public administration. It combines characteristics of the free software production model with the concept of public property and is delivered via a portal that brings different people and interests together. The BPS portal is a virtual space for the dissemination and improvement of software tools. In some communities, the development of new functionality or even new versions of tools is occurring. As such, software development processes also occur, although there is still no process in place for certifying quality. There are several types of studies on quality processes for free tools, however the goal of this research is to create an approach that takes into account the quality of an ecosystem [2] like that of BPS.

The most common approaches to certifying quality are based on formally established organizations[2] that have production processes that can be broken down linearly into smaller units. Thus, the maturity of an organization is directly proportional to the control of its processes and its capability to achieve its goals.

According to the Software Engineering Institute (SEI) at Carnegie Mellon University, a software process can be defined as a set of activities, methods, practices and transformations that people use to develop and maintain a piece of software and the deliverables associated with it (such as: project plans and documents, the code, test cases, and user manuals). As an organization matures, the software process becomes more defined, making it possible to implement the software process in a more consistent way throughout the organization. According to the SEI, the maturity of a software process is defined as the extent to which a specific process is explicitly defined, managed, measured, controlled, and carried out. This Maturity Based on Capability approach is guided by Capability Maturity Models (CMM) [3] [4] [5] [6].

Maturity represents the capability growth potential (scale of desired results that can be obtained by the application of the software process) and indicates the software process's risk to the organization and the consistency with which the software process is applied in all projects. In a mature organization, the software process is well understood – this understanding generally being achieved by means of documentation and training – and is continually monitored and improved by its users. The maturity of a software process implies that the resulting productivity and quality of the process can be continually improved through consistently increasing discipline through the use of the software process. The SEI's capability maturity model defines five levels of maturity to evaluate an organization's software processes. This definition of the maturity levels is based on the formalization of the software processes within the organization and the integration of these processes with the organization as a whole.

The BPS ecosystem has a fluid structure with software processes with varied dynamics that change over time. Standard company software processes and quality goals are part of the concepts of the CMM model which conflict with the dynamic of the ecosystem.

The goal for this research is to develop a conceptual structure for a maturity model for a complex system like BPS, a cloud-like virtual network with fluid, undefined limits and changing roles and settings which are based on voluntary collaboration and decentralized management via self-organization processes, intense information sharing, and feedback processes from the use of devices produced by the system.

---

[1] The BPS is a project of the Planning, Budget and Management Ministry (PBMM) of Brazil that introduces a new concept and operational structure to produce software, aimed at improving efficiency of the governmental system. The project began officially in 2006. At that time, the free software production model (FSPM) adopted by the Federal Government was strongly stimulated by national policies. Since their beginning in the end of 2006, the BPS has increased its number of associates to 120.000 users, and today it has 57 software solutions available, in many technological areas: geo-processing, health, public town management (sanitation, hospitals management, data management, etc)

[2] In this text organizational is [8] : an organizational unit typically part of a larger organization, although in a small organization, the organizational unit may be the whole organization. An organizational unit may be, for example: 1) − a specific project or set of (related) projects; 2) a unit within an organization focused on a specific lifecycle phase (or phases) such as acquisition, development, maintenance or support; 3) − a part of an organization responsible for all aspects of a particular product or product set.or organizational unit.

The process of forming a concept of quality for a virtual network needs to incorporate the complexity of this system. The maturity models based on existing capability were created based on the evaluation of strictly defined, quantified, and qualified structures with defined processes and interfaces with the external environment. For a system like BPS with undefined limits, non-linear relations and dynamics, a maturity model must be created in a different and innovative way.

The current version of ISO/IEC 15504 Standard [7,8] establishes process capability as a means towards process improvement. The maturity of a given organization is measured in terms of the capability of the defined processes for a given framework, such as CMMI (Capability Maturity Model Integration) and SPICE. Currently, the ISO/IEC 15504 Standard is under revision. A decision already taken is to consider capability as one example of process measurement framework. Therefore the revised version is establishing requirements to define process measurement framework.

Systems thinking [9] [10] [11] [12] [13] [14] [15] [16] suggests a means to maturity that is different from that which is based on capability. This finding is consistent with the current evolution of SPICE (ISO/IEC 33000 series [17] which concluded that rather than being the only means to maturity it is only one of many possible means. The group SPICE is currently developing the basic requirements that need to be considered when developing a new measurement framework to serve as a guide in creating these different forms of frameworks.

The research in question attempts to provide a framework to define the maturity of a public software ecosystem which takes into account research based on systems thinking to obtain empirical evidence of how the BPS ecosystem evolves in learning cycles that can bring forth a system maturity model and become a reference for understanding and improving BPS and other Public Software Ecosystems (PSE). The framework to be developed through this research should be considered by SPICE as a maturity alternative for digital ecosystems. In this case the maturity of the ecosystem is not based on the capacity of processes but on the robustness of its relationships to other entities of the ecosystem [20] with emphasis in cooperation and connection.

This article is organized as follows. This first section provides an introduction to the article. The second section summarizes the concepts used in research. The third section presents the research methodology. The fourth section establishes goals and process for the development of a theoretical maturity framework for BPS. Finally the fifth section presents some results and conclusions.

2   Conceptual Aspects

To provide theoretical grounding to the various fields of study involved in this research, it was necessary to take various theories into account. Given that the research objective is to study digital software production ecosystems, there arises the need to understand topics such as complex systems theory, systems thinking, digital ecosystems, and maturity models. The analysis of conceptual aspects aims to create a theoretical framework that establishes a base to guide research to be described in this paper. To this end, research was carried out on complexity theory, systems theory, digital ecosystems, and maturity models, as well as on their main concepts and ideas.

Given that the literature review would have to account for four conceptual spheres, systematic literature review (SR) was chosen as the preferred methodology. SR is used in medicine and its use began in the software engineering domain in 2004 with Kitchenham [18]. To carry out the SR, research questions were defined as well as an initial selection of the data to be studied. The data and catalogs were selected based on the specific data for the software engineering domain recommended by Kitchenham [18] and Brereton [19]. The objective of the research questions was to show the evolution and the tendency of the theoretical framework based on the data selected. The objectives of the literature review were met after the completion of the SR. The first objective, the verification of the originality of the proposed research, was demonstrated by the results obtained, and the results of the analyses carried out aided in the writing of the Conceptual Aspects chapter of the proposed research.

The process of forming a concept of quality for a virtual network needs to incorporate the complexity of this system. In the case of a virtual network, the conventional concept of a capability model is not adequate. The same thing occurs with BPS. To conceive of a capability model for BPS, it was necessary to make use of concepts from complex thinking theory, systems thinking, digital

ecosystems, and maturity models.

Complex thinking theory (CTT) is a new approach whose goal is to understand reality and its complex relationships. CTT came about based off of the finding that a new form of understanding the world and its processes was necessary since traditional approaches had shown themselves to be insufficient in dealing with the increasing complexity these processes represented. Moreover, CTT attempts to bring together aspects of order (linearity) and disorder (systems).

According to Ackoff [12], systems thinking works towards developing a new intellectual structure that attempts to describe 'organized complexity' as dynamic networks of interactions based on the concept of a system, and Capra [16] defines 'systems thinking' as "the understanding of a phenomenon in context," taking into consideration the entirety of interactions involved, in contrast to attempting to establish simple causal relationships between isolated parts.

Digital ecosystems are ecosystems where the environment is digital and is populated by digital species (software components, applications, online services). There are digital ecosystems oriented towards the production of content, business, and academic works, among other things [21]. Values have an important role in the self-regulation and self-organization of these ecosystems and are many times the primary element fueling the convergence of actors towards this virtual environment. Digital ecosystems represent an improved learning model as the flow of interactions and the creative energy for common goals held by the participants converge. The model that inspires these digital ecosystems is by nature a complex system; thus they naturally bring to mind mental models, values, and decision-making aligned with its characteristics (decentralization, autonomy, diversity, conflict identification and resolution).

Around the 1980´s, Watts Humphrey and others at the Software Engineering Institute (SEI) elicited and generalized good practices from few software intensive organizations that had been working well. Those practices were organized as sequential and cumulative maturity level as the Capability Maturity Model for Software (CMM or SW-CMM). As an evolution of CMM model two frameworks of models were established: ISO/IEC 15504 International Standard for Process Assessment [8] and the Capability Maturity Model Integration (CMMI) [22]. CMMI is aligned with ISO/IEC 15504 and the CMMI-DEV model [23] is the successor of CMM. ISO/IEC 15504 has been renamed by ISO/IEC as the ISO/IEC 33000 series [17]. The CMMI-DEV and ISO/IEC 15504-based models are the current most relevant models used in SPI projects.

PRO2PI-MFMOD is a method framework for engineering process capability models based on context and characteristics of a segment or domain. The current version is composed of four types of elements, each one, by a coincidence, with seven elements: sequential practices, customization rules, examples of utilization and examples of techniques. The method framework is described in Zoucas [24] e Salviano [25].

3   Methodology

The Action Research (AR) methodology was used to obtain a model appropriate for the BPS ecosystem. According to Thiollent [32], Action Research is a type of empirically-based social research that is conceived and carried out in close association with an action or with the resolution to a collective problem where researchers and participants representative of the situation or problem are involved in a cooperative or participatory way. Moreover, according to Thiollent, Action Research is a research strategy in production engineering that aims to produce knowledge and solve a practical problem. Action Research has four main phases: 1) the exploratory phase, 2) the in-depth research phase, 3) the action phase, and 4) the evaluation phase. The objective of the first phase of AR is to understand the context and diagnose the problems. In the second phase, data is collected from relevant sources using different methodologies. In the third phase, the action plan is defined and implemented. The fourth phase involves the codification of the practical and theoretical implementations from the various cycles of the previous phase.

The choice of Action Research as the research methodology was made for three reasons. The first reason refers to the perception that the phenomenon being studied does not have characteristics that allow it to be contextualized within the scope of traditional research methods since it is highly complex

and behaves like an ecosystem. The second reason refers to constraints imposed on the ecosystem by the nature of the object of study and by the client's expectations – that over the course of the project the actors in the ecosystem take part in the process, that knowledge exchange occur and that the understanding of the environment mature, that tacit understanding be explained, that a leveling of knowledge in the ecosystem occur, and that the model be created in a collaborative and shared way – which lead to the search for the best approach out of those available that would comply with the constraints and that would able to provide the desired results. The third reason is related to experimenting with a work methodology that could be used as the standard at the Public Policy Laboratory for ICT's from the Division of Software Process and Quality Improvement at the Renato Archer Information Technology Center – Poli.Tic/DMPQS/CTI. The methodology provides report generation which concerns the resolution of specific problems as well as generating and documenting new knowledge created over the course of the project which solves the problems specified by the client.

For this research project, an adapted version of the Perry and Zuber-Skerritt [33] model which has a framework to carry out graduate work was adopted and implemented using Action Research. This model makes use of the concept of an Action Research thesis to be developed by the researcher and that of an Action Research project nucleus which are used for organization as an action phase of the Action Research thesis.

Within the context of this research, the definition of a new perspective to identify maturity levels that should be used as a vector for the quality of digital ecosystems was the subject of the Action Research thesis and the issue stemming from the need for developing quality references for Brazilian Public Software was the Action Research project nucleus. Together they formed the Action Research approach to the research. Figure 1 shows this model. Currently the research is in the thesis/writing phase.
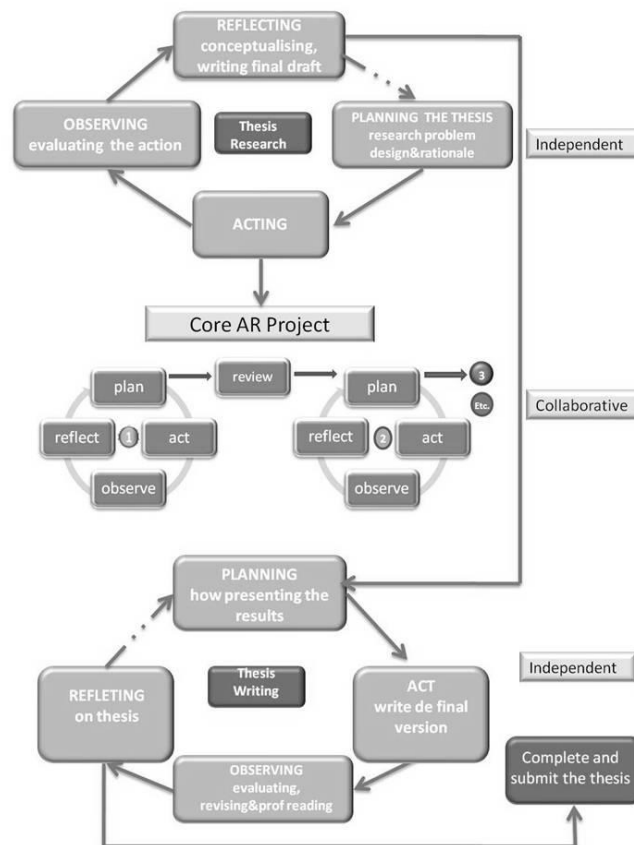
Figure 1. Relationship between Action Research and Thesis

Source: Perry e Zuber-Skerrit (1991: 76)

4 Goals and Process

The main problem that this research attempts to address can be understood via the following question:

How does one analyze and evaluate the maturity of public digital ecosystems that produce software collaboratively?

To answer the above question, it is necessary to also answer what we call intermediate questions. The research was conducted for complex system domain. In the functionalist sense of the word, complexity refers to a large set of variables whose relations cannot be mapped or monitored. For Demo [26], complexity is linked not only with the number of variables, but with a set of properties for interpreting a phenomenon as complex. The properties highlighted by the author are: the dynamics, the ambiguity, and the no-linearity. What is totally predictable and linear is not complex. These properties above help to characterize the complex a phenomenon as complex and were used, together the domain characteristics, to build the intermediate questions below.

Q1: Is it viable for the process of forming a concept of quality for virtual networks to incorporate the complexity of the system?

Q2: Are the capability models that traditionally aim to improve the quality of a linear, well-defined process adequate for cloud-like virtual networks with fluid, undefined limits and changing roles and mutant settings?

Q3: Is it possible to make progress on capability models for complex systems using concepts from complex thinking theory (CTT)?

Q4: Is it possible to identify standards that house and define the internal dynamics of this type of system using a systems approach?

Q5: What are the elements that make up this type of structure? What dynamic results from the interaction, sharing, and learning of these ecosystems?

Q6: How does one measure the capability of complex ecosystems?

Q7: How does one evaluate the maturity level of complex ecosystems?

Q8: What theoretical framework should be used to define the maturity model for complex ecosystems?

Q9: Is it possible to consider generalizing the concept of maturity to be developed by BPS given that BPS is only one example of Public Software Ecosystems (PSE)?

To respond to these intermediary questions it is necessary to include in the research objectives the construction of a theoretical structure that supports incorporating complexity concepts into the theory and practice of process improvement, especially that which relates to capability models of systems processes and systems maturity.

The objectives of this research are to: 1) contribute to the analysis and evaluation of quality in public software ecosystems within the context of maturity models using systems thinking methodology; 2) develop an experiment that will be used to aid in defining the requirements for a framework to evaluate processes for the new ISO/IEC 33000 standard, and 3) verify the possibility of generalizing the model for a specific case (BPS ecosystem) to generic cases of similar digital ecosystems.

The specific objective of this research is to develop a conceptual structure for a maturity model to analyze and evaluate the quality of a software production ecosystem according to the Action Research protocol.

This research makes a contribution that is empirical in nature, mapping and defining concepts to a maturity model that can be used for digital software production ecosystems in an attempt to understand and describe the phenomenon through the theoretical lenses of complexity theory [27] [28] [29] [30] [31], systems thinking, digital ecosystems and maturity models.

Through this study, the researchers were able to define the perspective that could be used to identify

maturity levels for systems of the type being studied and attach quantifiable measurements to them. The model and the method of evaluation were not developed because they were outside the scope of the project.

The objective for the researchers is to advance understanding regarding the improvement of software processes and expand upon the concept of capability and make a contribution in terms of studies related to maturity, working with other different concepts of process capability.

Over the course of this project, knowledge was generated and evaluated, undergoing assessment by the scientific community via the presentation of papers in conferences and later publication in the annals of conferences or in journals. Twenty one papers were presented. The papers' references are available at: http://goo.gl/sHPkR. The Action Research results related to the action part of the research project were published in two journals in the fields of information technology in addition to technical reports delivered to the client.

5 The proposed Concepts to a Systemic Maturity Model

After applications and the results obtained from the methodologies AR and PRO2PI MFMOD we concluded that the concepts for the maturity model of BPS incorporates normative vision and the levels that must be crossed to reach the desired vision. The maturity model consists of five levels and Figure 2 summarizes each of these levels. The maturity levels are defined from the values, qualitative and quantitative, undertaken by a number of critical variables that were set from the systemic map .At this first stage were defined 6 variables for level 1, 11 variables for level 2, 7 variables for level 3, 7 variables for the level 4 and 5 variables for level 5. Presented below are the variables that characterize each level of maturity, being, however, this only a first exercise that can be improved over time. The variables are: (a) Level 1: conceptual definition, legal framework, computational infrastructure, portal project, support human resources, project leader (champion); (b) Level 2: number of accesses, not attended expectation, necessity for infra-structure and human resources, quality of infrastructure and human re-sources, investment in infrastructure and human resources, quality of the co-ordination of community and interest groups, quality of governance of the bps network, quality of the interaction and the objects, user satisfaction and loyalty, number of developers, new solutions and interest groups; (c) Level 3:dissemination; number of opportunities; society demand; political support; conflict of interest; partnerships; software industry interest; (d) Level 4: addressing the needs of society by bps; number of adoptions; business; qualification;   number of providers in the virtual public market, re-sources optimization, understanding the BPS model; (e) Level 5: bps replication, component of public policies, budget autonomy, collegiate management, intra-governmental alignment.

Level 5 - Institutional maturity, autonomy, model replication in other areas, input to public policy.

Level 4 - Structured monitoring of results, increasing impacts, spread in society, structured network planning.

Level 3 - Partnerships established and formalized support, informal support identifiable, projection in media, expansion and dissemination of the model.

Level 2 - Legal and institutional framework set, minimal infrastructure functioning, rules and guidelines for quality of the artifacts.

Level 1 - Beginning - concept defined, minimal institutional support for project development, existence of project champion.

Figure 2- BPS Maturity Levels

For structured data and surveys conducted within the BPS, we can affirm that the ecosystem is between level 2 and 3. To that assertion more property is needed both to improve the methodology as well as its application in the field. The maturity levels were evaluated using as reference the critical variables that led to each level of maturity. The development process of the reference model was made by collaboratively and shared way. Different actors, from different positions in the ecosystem, participated in the process. A constant during development was the diversity of visions of the actors from the maturity of the same area of the ecosystem. This dichotomy was treated as an emergent process. The dichotomy has been turned into a new dimension that must be aggregated to the model. This dimension is composed of a bottom-up view and a top-down view. The same actor can have both views simultaneously. An actor with bottom-up vision is more sensitive and is more concerned with issues of shorter term or more operational. Actors with top-down view have a level of concern over the long term, more strategic and political.

6 Results

The research was able to address the main issue, i.e. construct a framework that is able to analyze and evaluate the maturity of digital public ecosystems that produce software collaboratively. The intermediary questions were also answered. For question Q1, incorporating all of the complexity that is intrinsic to virtual networks was found to be possible. For question Q2, it was found that capability as defined by current maturity models is not appropriate for BPS. It was then necessary to come up with mechanisms that allowed for a new perspective different from capability and which would be able to represent the maturity of this type of system. For the research in question, this new perspective is related to the system's learning cycles. For question Q3, it was found that the theoretical support from complex thinking theory provides the robustness necessary to address issues exhibited by this type of system. In search for an answer to question Q4, the researchers were able to identify the perspectives and learning cycles which showed the internal dynamic of this type of structure as their measurement evolved. Questions Q5, Q6, Q7, and Q8 were addressed by developing the conceptual structure for maturity models for systems similar to BPS, and Q9 was answered in the affirmative considering the broader theoretical concept of ecosystems and digital ecosystems and considering BPS to be a specific example of this type of system.

The objective of maturity models is traditionally to improve the quality of a linear, well-defined process. In the case of a virtual network without defined limits or roles and with mutant settings, the conventional concept of a maturity model is not appropriate. The same can be said for BPS. To conceive of a maturity model for BPS, it was necessary to use concepts from complex thinking theory. This approach allowed the researchers to form a general concept of BPS and see how variables interrelate and interact. The analysis of the system map and the identification of critical variables in learning cycles allowed the researchers to establish the internal dynamics of the ecosystem. The maturity levels were defined based on the learning cycles. The reference model of the BPS ecosystem is a model created to guide the evolution of the ecosystem as well as permit a new user to construct an integrated view of the various elements that make it up. This conceptual structure of the maturity model is innovative since it switches the maturity model from organizations with well-defined forms, objects, and functions to another type of productive arrangement: digital public ecosystems. The research also generated a second innovation: incorporating multiple concepts of maturity levels from a single maturity model which depends on the position held by the actor within the ecosystem. The model allows one to evaluate the maturity of ecosystems with an emphasis on cooperation and sharing, unlike conventional maturity models which emphasize command and control.

The BPS project was inspired by the broader theoretical concept of ecosystems and digital ecosystems and one can say that BPS is made up of elements that are related via the infrastructure that supports them, by the relationships among actors, and relationships between actors and the infrastructure. The resulting dynamic is conveyed through interaction, sharing, and learning, which are the characteristics of an ecosystem. Thus, one can say that BPS is a specific example of a public software ecosystem. In view of the above, the model created for the ecosystem can be generalized for all public software ecosystems and includes an innovation: multiple concepts of maturity level.

The initial and partial affirmations made up to this point provide a sufficient level of confidence that the conceptual structure for a systems maturity model is headed in a promising direction. For the SPICE community, the process carried out and the current results can provide reference for process measurement framework requirements of the revised version of ISO/IEC 15504. At the same time, the systems maturity model should use these requirements when they are published as a reference for future strengthening of the model.

The practical results from the concepts developed in BPS project and thesis refers to using these concepts in the Certification in National Technology for Information and Communication Technology project ecosystem.

Acknowledge

Literature

1   Software Publico Brasileiro: URL: http://goo.gl/hQtwM

2   CHANG, E., West, M., 2006. Digital Ecosystems a next generation of the collaborative environment. Proceedings of iiWAS 2006.

3   HUMPHREY, Watts S., Characterizing the Software Process: A Maturity Framework, in IEEE Software, Vol. 5, No. 2, pp. 73-79, March 1988. [reprinted in Software State-of-the-art: Selected papers, edited by Tom DeMarco and Timothy Lister, Dorset House Publishing, pp. 62-75, 1990].

4   HUMPHREY, Watts S., Managing the Software Process, Addison Wesley, Reading, Mass. USA, 1989. (ISBN 0201180952).

5   PAULK, Mark C. et al.The Capability Maturity Model - Guidelines for Improving the Software Process, CMU-SEI, Addison-Wesley, 441 pages, 1994.(ISBN: 0201546647).

6   SEI, Software Engineering Institute, TR-24 e TR-25, 1993

7   SALVIANO, C. F.: Uma proposta orientada a perfis de capacidade de processo para evolução da melhoria de processo de software (Portuguese) (A proposal oriented by process capability profiles for the evolution of Software Process Improvement), PhD thesis, FEEC Unicamp, 2006.

8   ISO/IEC 15504, Composed of seven parts (15504-1 to 15504-7) parts, under the general title Information technology — Process assessment, 2004-2008.

9   FORRESTER, Jay W. Industrial Dynamics. New York, London. The MIT Press and John Willey & Sons, 1961.

10 SENGE, Peter M. A Quinta Disciplina - Arte, Teoria e Prática da Organização de Aprendizagem. São Paulo, Best Seller, 1990. (ISBN:9788576843719)

11 BUCKLEY, W..Sociologia e a Moderna Teoria de Sistemas. Editora da Universidade de São Paulo, SP, 1971.

12 ACKOFF, Russell L. Creating the Corporate Future. John Wiley & Sons, 1981

13 CHECKLAND, P. Systems Thinking, Systems Practice. New York, John Willey & Sons, 1981.

14 FLOOD Robert L. & CARLSON, Ewart R. Dealing With Complexity: Na Introduction to the Theory and applications of Systems Science. Plenum Press, New York, 1988.

15 LANE, David C. & OLIVA, Rogelio. The Greater Whole: Towards a Synthesis of SD and SSM. Proc. of the 1994 Conference of the Intl. System Dynamics Society, 1994

16 CAPRA, Fritjof.   A Teia da Vida. São Paulo, Cultrix, 1996.

17 DORLING, A..Next Generation 15504 - the 33001 series of Standards – UPDATE, in http://www.spiceusergroup.org, August 24, 2009.

18 KITCHENHAM, B.A., 2004, Procedures for Performing Systematic Review.Tech.Report TR/SE-0401, Keele University, 2004. URL: http://goo.gl/UcVoo

19 BRERETON, Pearl; KITCHENHAM, Barbara A.; BUDGEN, David; TURNER, Mark and KHALIL, Mohamed. Lessons from applying the systematic literature review process within the software engineering domain. J. Syst. Softw. 80, 4 (April 2007), 571-583. DOI=10.1016/j.jss.2006.07.009 http://goo.gl/mFNAI

20 SALVIANO et alli, Practices and Techniques for Engineering Process Capability Models, in CLEI Electronic Journal, Volume 13, Number 1, Paper 3, April 2010, 13 pages.URL: http://goo.gl/hQtwM

21 BRISCOE, G.; SADEDIN, S.; PAPERIN, G.. Biology of Applied Digital Ecosystems, Digital EcoSystems and Technologies Conference, 2007. DEST '07. Inaugural IEEE-IES, vol., no., pp.458-463, 21-23 Feb. 2007 DOI: 10.1109/DEST. 2007.372015

22 The International Organization for Standardization and the International Electrotechnical Commission, ISO/IEC 15504 - Information Technology - Process Assessment, published as seven parts in different years: 15504-1:2004, 15504-2:2003, 15504-3:2003, 15504-4:2004, 15504-5:2006, 15504-6:2008 and 15504-7:2008

23 CHRISSIS, M. B., Konrad, M., Shrum, S., CMMI: Guidelines for Process Integration and Product Improvement, 2nd Edition, Addison-Wesley, 2007.(ISBN: 978-0-321-71150-2)

24   ZOUCAS, A. C., Thiry, M., and Salviano, C. F., Técnicas para Engenharia de Modelos de Capacidade de Processo de Software, Proceedings II Workshop on Advanced Software Engineering (IWASE), Santiago Chile, 13 nov. 2009 p.11-18. [SSM]

25 SALVIANO et al, A Method Framework for Engineering Process Capability Models, in EuroSPI 2009, The 16th European Systems and Software Process Improvement and Innovation, Industry Proceedings, University of Alcala, Madrid Region, Spain, 2-4 September 2009, p. 6.25-6.36 (2009.eurospi.net)

26 DEMO, P. Política Social do Conhecimento, Petrópolis, Vozes, 2000.

27 PRIGOGINE, I., Stengers, I. (1984) Order out of Chaos. - man's new dialogue with nature. New York: Bantam Books, 1984. (ISBN: 9780553340822)

28 BAR-YAM, Y. Dynamics of complex systems. Readings, Massachusetts: Addison-Wesley, 1997

29 MORIM, Edgar. O método I: A Natureza da Natureza. Publicações Europa-América Ltda, Portugal, 1977.

30 MORIN, Edgard. The Concept of System and the Paradigm of Complexity. In MARUYAMA, Mogoroh Context and Complexity: Cultivating Contextual Understanding. Springer-Verlag, New York, 1992.

31 MORIN, Edgard. Ciência com Consciência. BCD União de Editores S.A, Rio de Janeiro, RJ, 1996.

32 THIOLLENT, M. Metodologia da pesquisa-ação, 14ª ed. São Paulo: Cortez, 2005, Pósfacio, 1985, 132 p.

33 Perry, C. and Zuber-Skerritt, O. (1991), Action research in graduate management programs. First World Congress on Action Research and Process Management - Conference Proceedings, Brisbane: 1, 67-83.

Author CVs

Angela Maria Alves

Doctoral student in Information Technology at USP/POLI in Production, Master's in Quality from the State University of Campinas/Mechanical Engineering College, Electrical/Electronic Engineering from the Educational Foundation of Barretos, Researcher at the Renato Archer Center of Information Technology in the field of software process improvement. More information available at: http://lattes.cnpq.br/2583134243307649.

Marcelo Pessoa

Professor. Dr. Marcelo Schneck de Paula Pessoa is an electronics engineer, M.Sc., Ph.D., Associate Professor, Polytechnic School of USP - EPUSP. Professor, Department of Production Engineering and professor of EPUSP Federal University of Lavras. He has over 30 years experience in the fields of computer, automation and telecommunications and has worked for companies such as Cosipa, Siderbras and Research Center Telebrás. Effective member of the Study Management Life Cycle Software ABNT, participates in the development of national and international standards in the area of software and represented Brazil in international meetings. Coordinator eLabSoft where he conducts research on: factory software and software process. Also participates in research on IT strategy. Certified by the author on 20/10/11

Clênio F. Salviano

Clenio F. Salviano works at CTI (Centro de Tecnologia da Informação Renato Archer) (www.cti.gov.br) since 1984 in Brazil and teaches Software Process Improvement (SPI) courses at industry and academia. At CTI, he coordinates a SPI Research, Development and Innovation technical group. He is member of ISO/IEC 15504 (SPICE) group and a SCAMPI CMMI Lead Appraiser. He helped more than 80 organizations to perform SPI programs. He published and presented more than 60 articles and tutorials in relevant conferences worldwide. He received his PhD at Unicamp in 2006 proposing PRO2PI Methodology as an evolution of current SPI (http://pro2pi.wikidot.com/, clenio.salviano@gmail.com).

A Process Reference Model for managing Living Labs for ICT innovation: A proposal based on ISO/IEC 15504

Javier García, Alvaro Fernández del Carpio, Antonio de Amescua, Manuel Velasco
Computer Science Department
Carlos III University of Madrid
Avda. de la Universidad, 30, 28911 Leganés (Madrid)
jgarciag@inf.uc3m.es, arfernan@inf.uc3m.es, amescua@inf.uc3m.es, velasco@ia.uc3m.es

Abstract

Living Labs are innovation infrastructures where software companies and research organizations collaborate with lead users and early adopters to promote participative strategies to define, design, develop and validate new products and services that maximize the socio-economic conditions of the partnership.

Interest in Living Labs in ICT innovation is increasing and software related companies are considering these ecosystems as a relevant approach to improve their products and services through the active involvement of different stakeholders during the product definition lifecycle. Nevertheless, software companies need to adopt effective practices to participate effectively and take advantage of the participation in this kind of innovation clusters and there is not any reference model or standard related to the processes or practices to manage a Living Lab.

This article presents a reference model of effective practices to manage effectively the synergies of software companies with the other stakeholders (user communities, research organization and public administrations) participating in a Living Lab.

The article also describes the process used to create the reference model trough the identification and the analysis of a multiple case study considering six Living Labs and discusses the lessons learned during the creation of the process reference model.

Keywords

Living Lab, Open Innovation, User-Driven Innovation, Software Companies, Reference Model, ISO/IEC 15504, Experience Paper

1 Introduction

Nowadays, most of successful software products were developed by "lead users" collaborating with technology and software providers in the development of innovative products. The active involvement of end-users in the innovation lifecycle is an important asset for competitiveness in software companies. Living Lab is a pragmatic approach to implement this kind of strategy in the software industry.

The Living Lab concept originates from Prof. William Mitchell, MediaLab and School of Architecture and City Planning of MIT [8]. A Originally, Prof. William Mitchell used Living Labs to analyze how

people use IT technology when they stay in buildings. The information of these experiments was used to design buildings more comfortable and useful. Later, this Living Lab concept was used to define real life environments for involving end-users in the validation and co-design of new technological products promoted by IT companies and research organizations [1], [5].

Current Living Labs for ICT innovation are implemented as a cluster of organizations that are key actors in the technological innovation processes, such as companies of the software industry, communities of end-users, computer science research organizations and public administrations supporting innovation policies. The Living Lab infrastructure uses this stable network of stakeholders to provide different kind of services to enrich and complement the traditional execution of ICT innovation projects with services to promote the active participation of end-user communities to obtain their feedback during all the phases of the innovation initiatives lifecycle. Examples of these services are: the incubation of ideas of new ICT products and services to improve the socio-economic conditions of end-user communities, validation of technological solutions by different target of end-users, support to the wide-scale roll-out of innovative ICT products, etc. A Living Lab also has a resource perspective that includes processes and services for providing the specific infrastructures required for implement open and user-driven innovation. These resources include physical facilities for co-design, test-beds, collaboration tools supporting the interaction among the stakeholders of an innovation initiative, knowledge management platforms and human resources supporting the Living Lab services.

Due to the clear benefits of the Living Lab approach, the European Commission promoted a European Network of Living Labs that was launched in Helsinki at the end of 2006 under the Finnish Presidency. Since then, the network has enrolled new "waves" of Living Labs every year [2]. Thus, the European Commission promoted the implementation of the Living Lab approach in ICT research (FP7) and collaborative innovation programs CIP during the period 2008-2014 [3], [4].

Interest in Living Labs in ICT innovation is increasing and software related companies are considering these ecosystems as a relevant approach to improve their products and services through the active involvement of different type of stakeholders during the product definition lifecycle. Nevertheless, it is important to state that the Living Lab approach applied to ICT innovation is being introduced in the activities done by companies of software industry to create new products and services [1]. Consequently, there is not any reference model or standard related to the processes or practices to manage a Living Lab.

First, the paper describes the strategy and processes to develop the reference model. This strategy was based on a multiple case study approach that was carried out under the C@R research project. The main objective of C@R was the technological research and innovation in collaborative working environments in rural areas, adopting the Living Lab approach [9]. Six Living Labs were considered in this project: Cudillero (Spain), Åboland (Finland), Frascati (Italy), Homokhátság (Hungary), Wirelessinfo (Czech Republic) and Sekhukhune (South Africa). In these living labs, the focus was on innovating current business processes or public service provision, based on functionalities in collaborative work and business environments adapted to the local context and needs.

Second, the paper provides a summarized description of the process reference model for Living Labs management that is structured in process categories, specific and generic practices. To create this Process Reference Model, the authors decided to adapt the philosophy and structure proposed by ISO/IEC 15504 [13, 14, 15] because: 1) The structured approach to define and catalogue effective practices helps the Living Lab stakeholders to adopt effective practices following a process improvement approach; 2) ISO/IEC 15504 provides a framework for benchmarking among different Living Labs; 3) It is feasible to integrate open and user driven innovation practices in software process improvement programs already started in software development and service provision organizations.

Finally, the paper discusses the lessons learned during the definition of the process model proposed in this paper. These lessons learned include difficulties on the implementation of the effective practices presented in the model and their appropriateness to guide the Living Lab stakeholders in the implementation of this approach.

Section 2 provides a summary of the conceptual approaches reported in the literature to manage Living Labs. Section 3 describes the strategy to develop the Process Reference Model presented in this article. Section 4 presents a Living Lab process model including effective practices to manage a Living Lab for ICT innovation. Section discusses the lessons learned during the creation of the Process Reference Model. Finally, section 6 provides the main conclusions obtained from this work.

2 Current approaches to implement Living Labs

Living Labs, in which users' experiences lead the future directions of the products, are increasingly being used and successful service innovation development depends on understanding both existing and user needs [10]. In analyzing the literature available on Living Labs, two categories can be identified:

1) Living Labs as open innovation platforms. The Living Lab is created as a network of stakeholders for creating new software products and services through the active involvement of end-users through all the phases of the innovation lifecycle.

2) Living Labs exposing testbed applications to the users. The Living Lab is considered as an open testbed of ICT products and services where applications were exposed to end-users for validation. Nowadays, there are several examples of this kind of Living Labs: mobile computing, ubiquitous computing, collaborative working environments and cognitive systems engineering.

As a Living Lab is an open innovation infrastructure shared by a cluster of different software industry stakeholders, several elements are essential to implement an effective Living Lab:

a) A set of physical, software tools and human resources required to implement the services that a Living Lab provides to their stakeholders for enriching their innovation projects through the implementation of end-user involvement.

b) A set of processes necessary to prepare, organize and run each service provided by the Living Lab to support the development of each innovation initiative in the Living Lab scope. Even more, these processes should also include the activities required for governing the network of stakeholders composing of the Living Lab and for defining and managing the business model to assure the economical sustainability of the shared innovation infrastructure.

Based on the dual view of a Living Lab as a shared ICT innovation infrastructure and catalyzer of ICT innovation projects through the implementation of participative and open innovation principles, the Living Lab processes can be categorized into two perspectives [11]: a) The organizational dimension focuses on the creation and management of a self-sustainable infrastructure for networked innovation; and b) The operational dimension includes all the activities to carry out each innovation initiative considered in the scope of the Living Lab innovation organization.

As the Living Lab concept is strongly related to the user driven innovation principles, it is essential to address user involvement during the whole innovation lifecycle in the projects managed in a Living Lab. User community building, participative design, and using the tools in new work and business practice are key activities of user involvement [5]. Practical constraints hinder continuous and active engagement of end-users in an innovation process [11]. A reasonable successful strategy has been to distinguish between two levels of users, the first is the strategic level of stakeholders (users of the living lab as innovation facility, e.g. business associations, policy makers, local governments), and the second, the end-user who is directly involved in living lab innovations (users of the living lab innovations) [6].

The most relevant techniques, which contribute to implementing user driven innovation principles in the creation of Living Labs, were referenced in the literature (Figure 1).
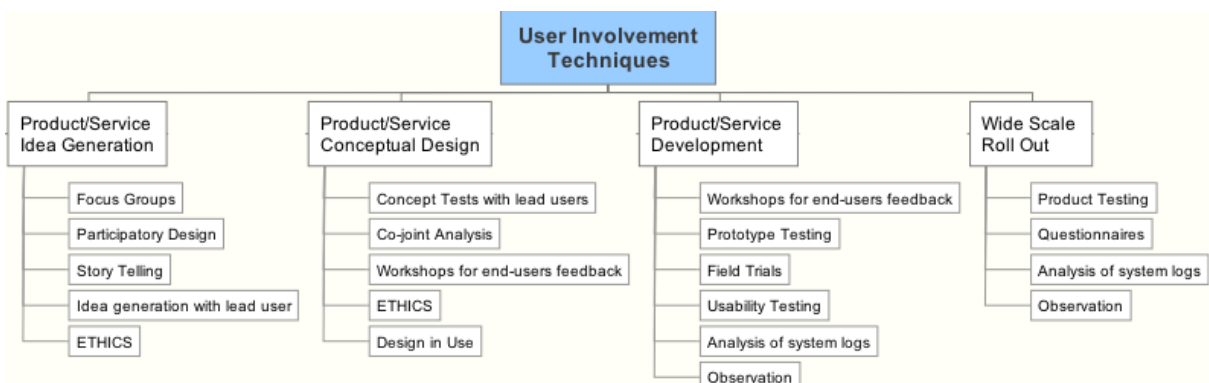
Figure 1: Living Lab Techniques

As conclusion of the state of the art analysis, there is an absence of references models or standards on the effective practices to manage a Living Lab that can be used to implement this kind of innovation organizations in the ICT and software industry. The provision of systematic approaches based on empirical results to guide the creation and continuous improvement of a Living Lab is not discussed in detail in the state of the art [12]. ISO/IEC 15504 metamodel constitutes an effective framework to gather and organize the effective practices for continuous improvement of the capability of an organization to develop and provide ICT based products and services.

In order to contribute to the effective creation and continuous improvement of Living Labs, the question guiding this work is: "What is the feasibility to apply an ISO/IEC 15504 based approach to create an reference model of effective practices to create and manage Living Labs as open organizations for ICT Innovation?". This kind of adaptable and upgradeable framework of effective practices will be very useful for Living Lab practitioners to guide their efforts on the creation and management of this kind of organizations for open ICT innovation [7].

3 Approach to define the Living Lab Process Reference Model

In order to apply an ISO/IEC 15504 based approach to create a Process Reference Model of effective practices to create and manage Living Labs, a multiple case study approach was implemented.

This multiple case study was carried out under the C@R research project (2006-2010) that was dealing with the problems of introducing innovations based on ICT Collaborative Services in Rural Areas to improve current business process, productivity that aims to increase economic growth, and the quality of life. Rural living is characterized by widely distributed work and life activities. One of the main goals of C@R project was the creation of six Living Labs. These living labs are located in Spain, Italy, Hungary, the Czech Republic, Finland and South Africa.

The multiple case study considered for creating the Process Reference Model was executed during three years. Figure 2 presents the planning of the case study execution.
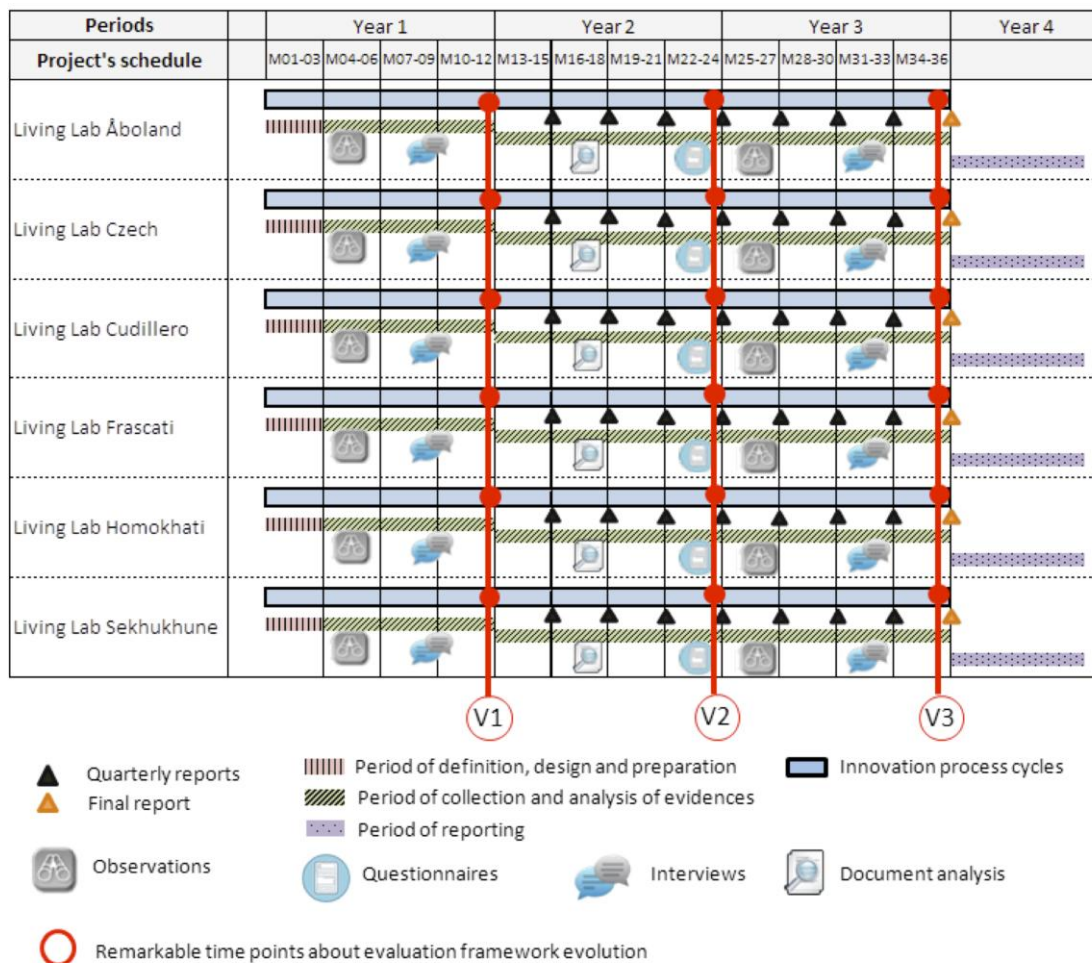
Figure 2: Living Labs considered in the multiple case study

During each year, a version of the reference model was used to determine the capability and effectiveness of each Living Lab considered in the C@R project. The model was defined following an incremental and iterative approach in three-monthly cycles. At the end of each cycle, each Living Lab provided a quarterly evaluation report. The Living Lab managers prepared these quarterly reports. Each of these reports was reviewed by a team composed of the C@R staff in charge of giving methodological support to the Living Labs considered in the case study. The reviews were done to look for incomplete or inconsistent information according to the actual situation of each Living Lab. Finally; periodic and individual meetings with the Living Lab leaders were carried out to clarify the information in each report requiring further details. Also, objective evidences that demonstrate the processes to operate and manage a Living Labs beyond the information included in the periodic reports were e-mails, minutes of the meetings and logs of the supporting tools used to manage the Living Labs.

Participant observation method was applied to assure the validity of the data included in the periodic report. This method was used to observe how Living Labs were created and worked, the problems they faced and the actions taken to overcome them. Interviews and questionnaires methods were used to collect information and the opinions of Living Lab members on their personal experiences, degree of satisfaction, problems and other issues related to working in a Living Lab. Interviews were also used to corroborate if the impressions obtained by the observers were right.

Once the case study was finished, the information on effective practices to implement a Living Lab were analysed, catalogued and documented. To facilitate the final elaboration of the reference model, its appropriateness to characterize the efficient practices and capability levels of a Living Lab was analysed. This analysis determined if the practices specified in the Process Reference Model really represent those carried out in C@R project's Living Labs. To this end, each practice by Living Lab was monitored and rated to express how much a practice was executed. Then, all values by practice were totalized and a representative value was obtained, which meant that the practice was evidenced

barely, moderately or completely in most of or all Living Labs analysed.

4 Living Lab Process Reference Model

The application of ISO/IEC 15504 philosophy and structure [13, 14, 15] was considered appropriate to fulfil the following requirements defined by the practitioners of the Living Labs studied: 1) The provision of a framework for cataloguing and disseminating the effective practices for enabling the management and participation of Living Labs; 2) The provision of a framework to support the implementation of continuous improvement initiatives in a Living Lab; and 3) The provision of mechanisms to let Living Lab stakeholders and sponsors to evaluate the performance and effectiveness of a Living Lab.
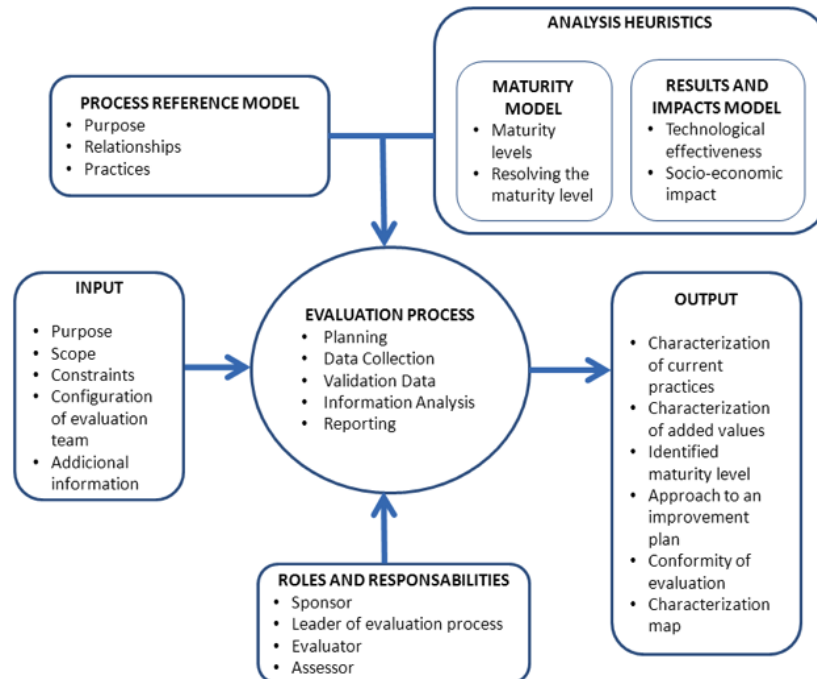
Figure 3. Improvement and Evaluation Framework for Living Labs adapted from ISO/IEC 15504

The structure of the Living Labs improvement framework is presented in figure 3 and it was adapted from [13, 14].

The process reference model describes the processes and practices to implement this type of ICT innovation organization. This information is essential to implement evaluation and continuous improvement initiatives. It formalizes the effective practices to create and manage a Living Lab grouped in five categories of processes (see figure 4). A more detailed description can be found at [16].

a) Innovation initiatives Management (IIM): this category is the core process area in which all other processes are developed. It defines the essential practices to execute the innovation process, which complement all other process areas of the Reference Model. It does constitute the essential and core process of a Living Lab.

b) Organizational Management: this category contains the activities of defining, planning and implementing a Living Lab as a self-supporting organization. The process areas in this category are: Strategy Management (SM), Stakeholders Management and Governance (SMG), Technology Infrastructure Management (TIM) and Knowledge Management (KM).

c) Technical Development: this category contains all the activities related to the definition, specification, design and development of technological solutions. The process areas included into this category are: Project Management (PM), Technological Requirements Management (TRM), and Technological Solutions Development (TSD).

d) Monitoring and Evaluation (M&E): this category contains a process area which provides all the

necessary practices to analyse the innovation practices carried out in a Living Lab, and the mechanisms to determine the success of the results obtained.

e) Deployment and operation (D&O): this category contains the activities for the deployment and operation of the technological solutions developed in a Living Lab. The process area contained in this category is Deployment and Operation of Technological Solutions.
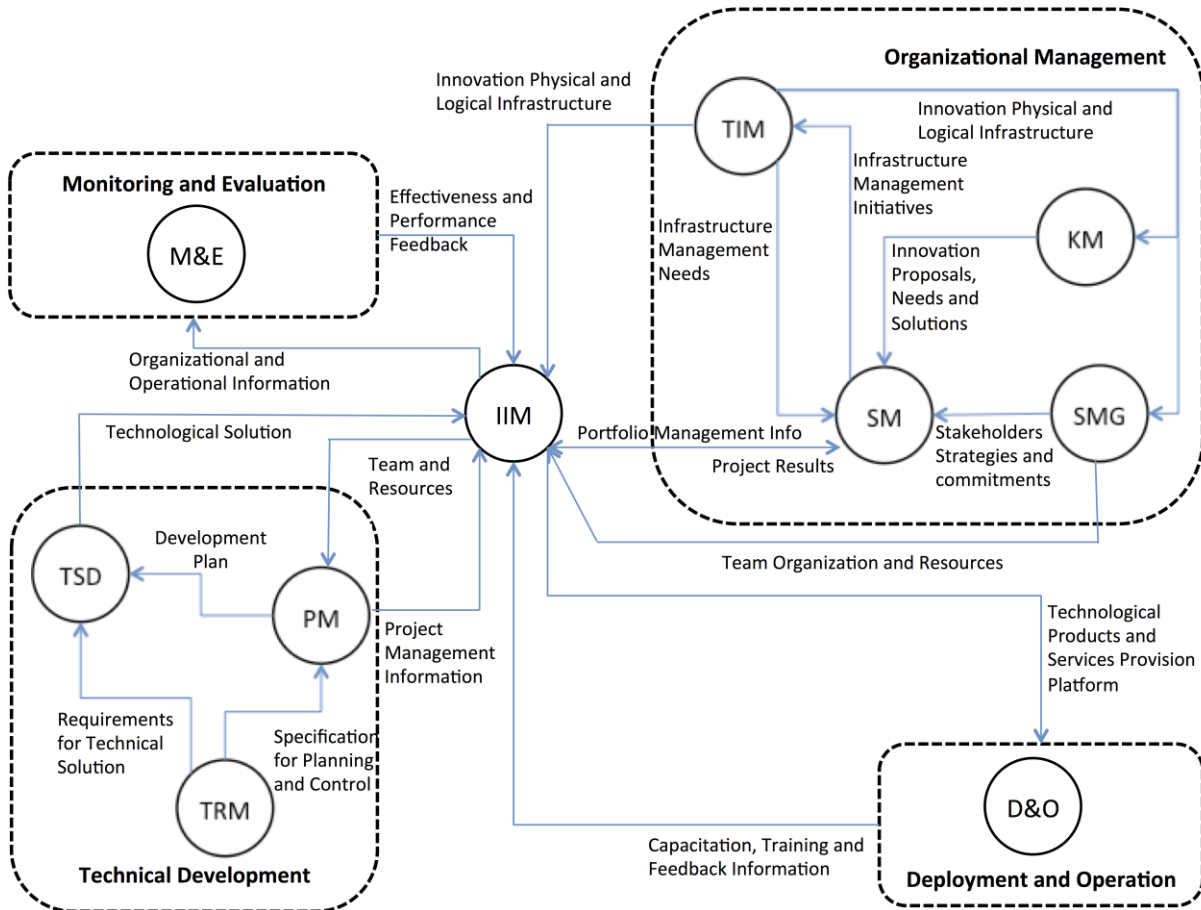
Figure 4: Overview of the Living Labs Process Reference Model

5 Lessons Learned

This section presents the lessons learned during the process of creating the Living Lab. These lessons learned were obtained from the quarterly reports obtained during the multiple case study complemented with participants observation and interviews. These lessons learned are grouped in two categories: A) Usefulness of the formalization of the Living Lab effective practices following an ISO 15504 approach; and B) Usefulness of the model developed to guide the continuous improvement initiatives of a Living Lab for ICT innovation.

A) Usefulness of the approach to formalize the Living Lab effective practices

The formalization of the effective practices to manage and participate in a Living Lab following an ISO 15504 like approach was considered positive in several cases:

* Software products and services vendors. For this kind of stakeholders, the implementation or participation in a Living Lab infrastructure is considered as part of an improvement initiative to define new products and service lines or improve the already existing by involving end-users for product definition and refinement. In this sense, the provision of this kind of formalization was considered useful because it can be integrated with other already existing continuous improvement initiatives.

- Living Lab investors and public administrations. For this kind of stakeholders, the implementation of the proposed approach could help to determine the internal effectiveness of a Living Lab organization and it is considered as an starting point to implement mechanisms to evaluate the effectiveness of a Living Lab.

Nevertheless, it is necessary to indicate that several Living Lab stakeholders and practitioners were reluctant to implement this approach because they considered that Living Labs for open innovation should be focused on promoting social and ad-hoc innovation initiatives where the enhancement of creativity is more important than the implementation of specific processes. In this sense, the adoption of the Process Reference Model proposed is more feasible in Living Labs where the collaboration among industry, research organizations, user communities and public administrations is required. In other Living Labs more related to social innovation and hacking culture, the application of the proposed approach would be more difficult and problematic.

B) Usefulness of the model developed to guide the continuous improvement initiatives of a Living Lab for ICT innovation

According to the evidences obtained during the multiple case study, the process areas defined in the Process Reference Model contain the most common effective practices carried out by each considered Living Lab. In several specific Living Labs, those practices that were not applied completely or not were initiated prove the variability in their execution due to the different nature of each Living Lab, considering the internal and external situations such as availability of resources, availability of social communities participating in the innovation process, specific conditions in the software development organizations and strategic views of each Living Lab.

It is important highlight that the practices included in the Process Reference Model were implemented in different stages. These stages were:

a) Creation of initial Living Lab infrastructure. In this stage, the Living Lab stakeholders' community is set defining an initial governance model; the innovation goals are agreed among user communities and software companies and the services to be provided by a Living Lab are identified establishing their priorities. The requirements for the Living Lab infrastructures are stated and a small-scale infrastructure is implemented. A first set of innovation initiatives is implemented using this small-scale infrastructure to determine the Living Lab feasibility. At this stage, the Living Lab infrastructure is provided using the funds of specific innovation projects of the Living Lab members Specific process areas considered in this stage were: Innovation Initiatives Management, Government and Strategic Planning, Project Management, and Technological Solutions Development.

b) Enlargement of Living Lab infrastructure. In this stage, the objective is to create a self-sustainable infrastructure that provides added-value services to the innovation ICT projects carried out in the scope of the Living Lab. It requires the enrichment of the small-scale innovation infrastructure providing additional resources in terms of human resources fulltime dedicated to the Living Labs management, physical and technological infrastructure for providing user involvement services to the innovation ICT projects along the innovation lifecycle. In this stage, the processes to define, operate and maintain sustainable Living Lab services are defined and piloted. Specific process areas in this stage were: Community Management, Planning and Technological Infrastructure Configuration, Knowledge Management and Technological Solution Requirements Management.

c) Self-sustainability of Living Lab infrastructure. This stage is based on the adoption of organizational processes to implement a Living Lab business model to assure the economical sustainability of the Living Lab infrastructure, the provision of user involvement initiatives considering large user communities and providing a measurement and evaluation framework to support the analysis of the added value infrastructure for the stakeholders and the implementation of continuous improvement initiatives for the services provided by the Living Lab. Nevertheless, it is necessary to consider that there are few Living Labs that have achieved this stage of development.

6 Conclusions

This paper has presented a Process Reference Model of effective practices to create and manage Living Labs as open organizations for ICT Innovation. Through the application of principles and foundations provided by ISO/IEC 15504, it was possible to elaborate a reference model that catalogues and formalizes the processes to manage a Living Lab in the software industry. This reference model was created by means of a multiple case study, which comprised three analysis parts.

The Process Reference Model describes properly the practices carried out in each Living Lab of the C@R project, showing it to be efficient and real to manage this innovation environment at strategic and operational levels.

The reference model developed covers the absence of a formalized approach to carry out a holistic view to facilitate the effective participation of software companies in open and people-led innovation environments such as Living Labs. The model proposed demonstrated its adaptability to different Living Labs considering the specific characteristics of each Living Lab and the availability of resources to implement user-driven ICT innovation approach. The formalization of the framework contributes to facilitating benchmarking activities, comparing the current processes of a Living Lab against the efficient practices recognized by the Living Labs community.

The next step for the development of this reference model is the constitution of a users group that could be proposed in the scope of SPICE standard user groups for the software industry and the ENoLL [2] to facilitate the analysis of more Living Labs. At the Spanish Level, this enlargement initiative is being carried out under the activities of ESdILAB (Spanish network of Living Labs and Social Spaces for Research and Innovation – http://www.espaciossociales.es).

In the scope of the creation of this user group, the refinement of the Process Reference Model would be done through the analysis of other case studies, questionnaires and interviews to Living Lab managers and participants.

References

[1] Almirall, E., Wareham, J. (2008) Living Labs and Open Innovation: Roles and Applicability. The Electronic Journal of Virtual Organizations and Networks (eJov),10, 3, 21-46.

[2] ENoLL (2011) European Network of Living Labs. November, 20. (available at: http://www.openlivinglabs.eu).

[3] European Commission (2008a) Information and Communication Technologies, Research Work Programme 2009-2010. April, 15. (available at: ftp://ftp.cordis.europa.eu/pub/fp7/ict/docs/ict-wp-2009-10_en.pdf).

[4] European Commission (2008b) Competitiveness and Innovation Framework Programme (CIP). ICT Policy Support Work Programme 2009. June, 18. (available at: http://ec.europa.eu/information_society/activities/ict_psp/documents/ICT%20PSP%20WP2009%20-%20v21nov08.pdf).

[5] García, J., Schaffers, H., Bilicki, V., Merz, C., Valenzuela, M. (2008) Living Labs Fostering Open Innovation and Rural Development: Methodology and Results. Proceedings of the 14th International Conference on Concurrent Enterprising, Lisbon, Portugal, June 23-25.

[6] García, J., Schaffers, H., Fernández, A. (2009) Assessment of Results and Impacts of the C@R Rural Living Labs. eJOV Executive – The Electronic Journal for Virtual Organizations and Networks, 11.

[7] García, J., Navarro, M., Schaffers, H., Kulkki, S. (2007) Methodological Framework for Human and User-Centric Rural Living Labs. Proceedings of eChallenges e-2007 Conference , The Hague, Netherlands, October, 24 – 26.

[8] Mitchell,W.J. (2011) Smart Cities Lab at MIT. March, 7. (available at: http://cities.media.mit.edu/).

[9] Navarro, M., Lopez, M., Ralli, C., Peña, C., Schaffers, H., Merz, C. (2010) Innovation Strategy for Rural Development Based on Living Labs for Human Empowerment. In: TRAGSA and FAO (eds.), Living Labs for Rural Development. Madrid, pp.1-9.

[10] Nystrom, A.-G.; Leminen, S. (2011) Living lab — A new form of business network. 17th International Conference on Concurrent Enterprising (ICE), June 20-22.

[11] Schaffers, H., Budweg, S., Kristensen, K., Ruland, R. (2009a) A Living Lab Approach for Enhancing Collaboration in Professional Communities. Proceedings of the 15th International Conference on Concurrent Enterprising, Leiden, The Netherlands, Jun 22-24.

[12] Schaffers, H., García, J., Merz, C. (2008) An Action Research Approach to Rural Living Labs Innovation. Proceeding of the eChallenges e-2008, Stockholm, Sweeden, October 22-24.

[13] ISO/IEC 15504-2:2003 Information technology -- Process assessment -- Part 2: Performing an assessment, ISO 2003

[14] ISO/IEC 15504-5:2006 Information technology -- Process assessment ISO 2006

[15] ISO/IEC 12207:2008 Systems and software engineering -- Software life cycle processes, ISO 2008.

[16] Fernandez del Carpio, Alvaro (2012). Aproximación Formal para la Gestión y Evaluación. PhD Thesis. Carlos III University of Madrid, 2012. In Press

Author CVs

Javier García Guzmán

PhD. holds a BSc in Engineering and a PhD in Computer Science (Carlos III University of Madrid). He has 14 years' experience as a researcher and consultant in public and private companies. He has participated in numerous research projects related to open and user driven innovation approaches for creating new solutions for software services, collaborative working environment and e-learning technologies, financed by public (European and national) and private funds. He has published several books and international scientific papers related to Living Labs, software engineering and collaborative working environments.

Alvaro Fernández del Carpio

He is a PhD candidate at the SEL-UC3M (Software Engineering Lab) group at Carlos III University of Madrid and holds a MSc in Computer Science and Technology (Carlos II University of Madrid, 2007). His current interests are: Software Process Improvement, Collaborative Work Environments, Assessments Models and Living Labs. Contact him at arfernan@inf.uc3m.es.

Antonio de Amescua

Antonio de Amescua is a full professor in the Computer Science Department at Carlos III University of Madrid. He has been working as software engineer in a public company (Iberia Airlines) and in a private company (Novotec Consultores) as a software engineering consultant. He founded Progresion, a spin-off company, in order to offer advanced Software Process Improvement services. He received his PhD in computer science from the Universidad Politécnica of Madrid. His research interests include Software Process Improvement, Software Reuse, Software Project Management.

Manuel Velasco de Diego

He received an Engineering Degree (1993) and PhD (1998) in Computer Science at the Polytechnical University of Madrid. He is a lecturer in Software Testing at Carlos III University of Madrid since 2001. His main research areas are software testing and reusability. He has published several papers in international journals and congresses on these subjects. He has participated in numerous research projects, financed by the European Union and the Spanish Government. Currently, his main research focuses on relating software testing and reusability topics to the design of spatio-temporal databases.

SPI through Process Tailoring Framework with Objective-Based Scoring Metric

Tisanai Krisanathamakul, Chayakorn Piyabunditkul,

Apinporn Methawachananon, Pawarat Nontasil
Software Engineering Laboratory (SWE)
National Electronics and Computer Technology Center (NECTEC)
Pathumthani, Thailand
chayakorn.piyabunditkul@nectec.or.th

Abstract

This research proposes systematic process tailoring framework that helps the practitioner decides for the most appropriate tailoring level matching to the project environments. Based on the organization's set of standard process (OSSP), the software processes and artifacts are given the weighting value that relevant to the organization quality criteria called "objectives". In project level, software processes and artifacts are tailored according to quality criteria that are set according to the project specific environments. By adopting the concept of value-based software engineering (VBSE), the objective-based scoring metric is calculated in terms of quality/effort. The ordering value of quality/effort helps practitioner decide to do the process tailoring in systematic way according to the quality criteria needed in the project.

Keywords

1   Introduction

In a current competitive software business, organizations have to find ways to keep up their product and services quality. Conducting Software Process Improvement (SPI) program with international development standard assures business in competing with their rivals. To achieve and maintain such a software development standard likes CMMI (Capability Maturity Model Integration) is a major challenge for every software organizations especially SMEs (Small and Medium Enterprises) businesses that have so many limitations due to their lack of resources nature. Besides that, even the software processes that conform to the software development standard can be set up in organization, but to tailor those software processes to software development project is not trivia in which each project has their own specific environment. In SMEs environment, to tailor the organization software process depends on experiences of practitioner and usually in the informal approaches. That means the previous process tailoring approach is hard to be effective repeatable framework for less experience practitioner. To implement CMMI, the IDEAL model, which composes of Initiating, Diagnosing, Establishing, Acting and Learning phase are generally used by practitioners. The IDEAL model shows an iterative process for process improvement as continuously evaluated process. The proposed process tailoring framework with objective-based scoring metric also abides to this IDEAL model in that the setup and evaluation of quality criteria called "objective" is done periodically to achieve a better fit criteria for the organization environment.

Target of process tailoring is a set of software processes and artifacts generated from process that suits to a certain environment. Tailoring activities may include organizing, adding, deleting, merging or altering those processes and artifacts. However, the process tailoring generally involves with activities that try to reduce the effort to accomplish the goals of standard process. Specific environment determines the selection of key practices to accomplish specific goals [2].

CMMI provides tailoring guideline [9], that help practitioners develop and apply to their process tailoring. This is done via Organization Process Definition (OPD) and Integrated Project Management (IPM), which describes the relationship among elements needed in process tailoring as shown in figure 1. Organization's Measurement Repository (OMR) holds the measurement data and related information needed in estimation for project related parameters these data also help organization setup the tailoring guidelines in which the historical data informs what can be expected through the process tailoring. The Organization's Process Asset Library (OPAL) is a utility tool for managing all the Organization Process Asset (OPA); these assets provide all the resources needed in the project implementation.
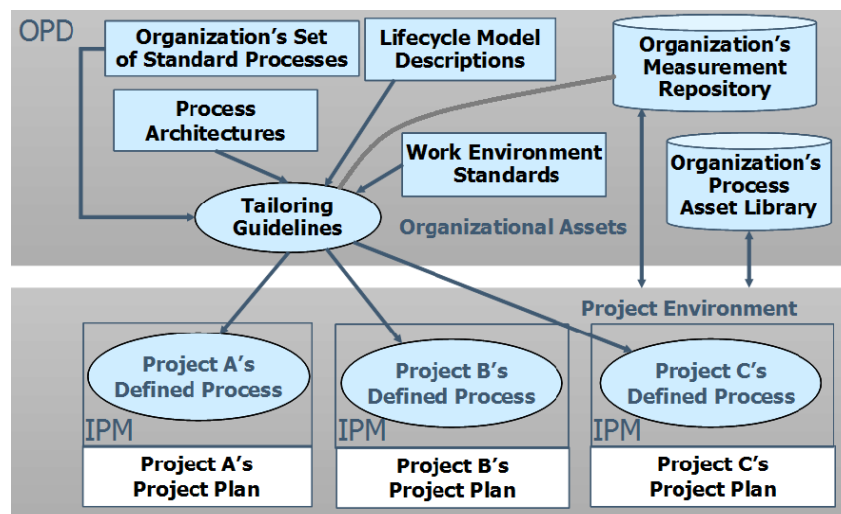
Fig. 1. The relationship among elements in OPD and IPM process areas

The important point to address in process tailoring analysis is the degree of differences from the organization standard process, which can be measured in forms of formality, frequency or granularity to perform certain process, and those differences might not reduce the quality of software product below the acceptable criteria. With the maintaining of acceptable quality criteria, we try to reduce the implementation effort to the level that best match current organization capability through software process tailoring [2][5].

CMMI has been proven to be a successful software developing framework for well-developed software organizations. However, for SMEs businesses, normally they have far less resources from well-developed counterpart. To be successful in applying CMMI in the SMEs environment, it needs a lot of process reduction from the standard one. Several research works stated the important activity of process tailoring through the practical framework developed for SMEs [3][4][11][12][13][14][16].

This research, proposes the tailoring framework for the certain environment by using quality criteria, called "objectives", that highly related to those specific environment. To optimize the process tailoring, the objective-based score metric are proposed in helping the practitioner to decide the most appropriate tailor template for their work.

By using quality criteria as objectives for specifying the most relevant processes and artifacts that contribute to required criteria, this will reduce the effort to perform processes or create artifacts to the suitable level and helps small organizations like SMEs to better manage their resources. To tailor the processes with the goal to achieve quality requirements, this gives the flexibility to organizations in setting the tailoring framework that well match to their environment. The quality criteria depend on the characteristics of the project needed to deal with. In this research paper, we set the quality criteria to support the software development process in terms of time schedule, resources, quality control,

technology risk, and knowledge management. Apart from the quality criteria, regarding to organization standard compliance [8], the software development process and the tailoring activities must ensure that the resulting processes used in each project still conform to the organization process definition which is certified by CMMI [9][15].

2 Research Problem

To apply organization software process to the software development project is a major challenge. Especially in SME organization, that has limitation in various aspects related to resource availability, process tailoring plays a crucial role in achieving the organization standard criteria with minimum effort. The idea of tailoring the development process according to the context of the project was studied and be provided with framework [6]. However, our proposed framework helps practitioner do the process tailoring in systematic way by providing the process tailoring templates appropriate to the identified project characteristics. With the objective scoring metric, process tailoring are made according to the quality criteria required to the project. By concern only quality criteria that required by project, this process tailoring aims the lower the resource expenses in economical way, which is one of the aspect ideally want to achieve through SPI implementation such as CMMI [7]. Figure 2 outlines the proposed tailoring framework. OSSP indicates the standard process with its properties of quality characteristics. Those characteristics are mapped with its counterparts in project level then the degree of tailoring is determined based on the relevant scores through scoring metric. The objective-based scoring metric is the result from the calculation of the quality contribution score of processes and artifacts to the required quality criterion in the project.
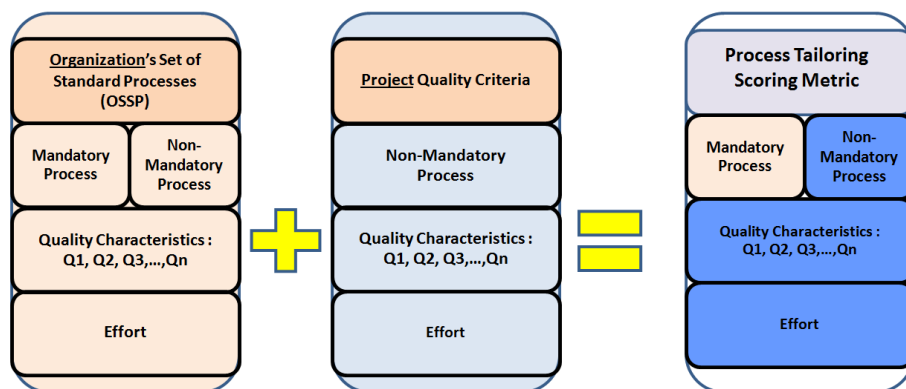
Fig. 2. The process tailoring score based on relevant between quality criteria of project level and organization level

3 Proposed Process Tailoring Framework

Figure 3 shows the proposed objective-based tailoring framework. OSSP provides the process tailoring template available for selection based on project characteristics. Also it determines the properties of the process and artifact within the template. The properties of process and artifact include the mandatory/non-mandatory status that indicates the availability for tailoring process, the relation to the quality criteria in the form of weighting value, and the effort to perform process or create artifact in term of workload.
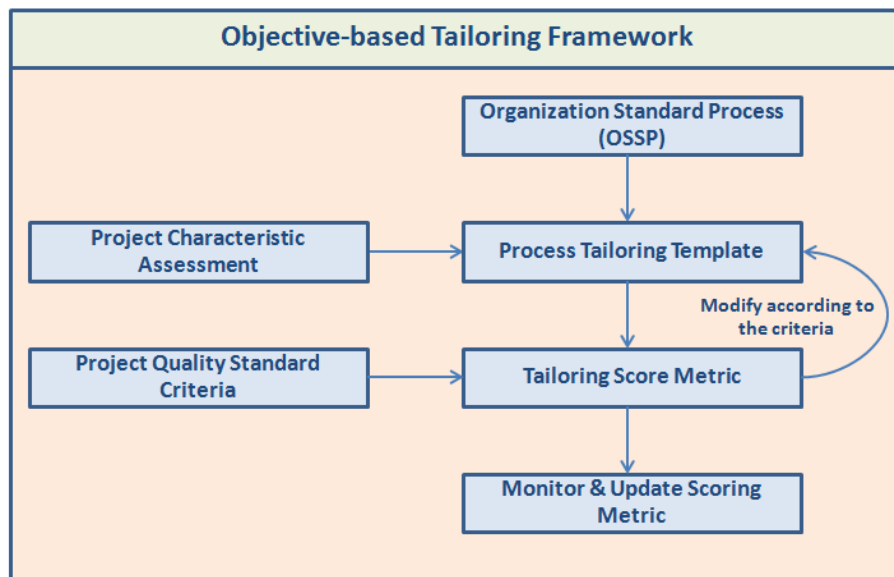
Fig. 3. lll Objective-based tailoring framework

The quality criteria (QC1 to QCn) are organization-oriented perspective that depended on organization outlook to the quality that should to be maintained for the success of the project. We summarize the possible quality criteria that might needed in software development project as in the Table 1. Table 2 displays the example of weighting distribution of process and artifact to the quality criteria. By the mandatory/non-mandatory status of each process and artifact, only the non-mandatory ones are listed here.

The assigned value shows the relationship degree of process or artifact to the quality criteria we want to maintain in project. The weight values are distributed to each process and artifact as weight summary to one (WPQC1+WPQC2+WPQC3+…+WPQCn = 1) for that it can indicates the comparison of relationship degree of process and artifact to the certain quality criteria. OSSP set up this weight distribution values based on measurement data collected through past experiences. These weight distributions will periodically be updated according to the result of the quality of the process tailoring, in which come from the review and evaluation process that must be done correspondingly.

Table 1. Example of Quality Criteria

	Interpretation of Quality Criteria
QC1	**Time Schedule:** Project that has to be submitted in a very short time period.
QC2	**Quality Control:** Project that needed to be closely monitored through the development stage.
QC3	**Resources**: Project that has very limited resources.
QC4	**Technology Risk:** Project that is very sensitive to the risk of technology failure.
QC5	**Defect:** Project that must be controlled to have minimum error.
QC6	**Knowledge:** Project that will be used as the role model for any reference or best practice for knowledge management.

The process tailoring starts from selection of tailoring template from OSSP weight distribution. From the process flow in figure 3, the quality criteria required to the project will be treated as project characteristics and these quality criteria are used to select a set of relevant processes and artifacts as initial process tailoring template.

Table 2. Example of weighting distribution of process and artifact to quality criteria

Process / Artifact	Organization Quality Criteria Weighting Distribution			
	Quality Criteria 1 (QC₁)	Quality Criteria 2 (QC₂)	Quality Criteria 3 (QC₃)	Quality Criteria n (QCₙ)
Process 1	W_{P1QC1}	W_{P1QC2}	W_{P1QC3}	W_{P1QCn}
Process 2	W_{P2QC1}	W_{P2QC2}	W_{P2QC3}	W_{P2QCn}
.	.	.	.	.
Process n	W_{PnQC1}	W_{PnQC2}	W_{PnQC3}	W_{PnQCn}
Artifact₁	W_{A1QC1}	W_{A1QC2}	W_{A1QC3}	W_{A1QCn}
Artifact₂	W_{A2QC1}	W_{A2QC2}	W_{A2QC3}	W_{A2QCn}
.	.	.	.	.
Artifactₙ	W_{AnQC1}	W_{AnQC2}	W_{AnQC3}	W_{AnQCn}

Then we modify the process and artifact within the tailoring template based on quality criteria distribution in the project. The quality distribution is determined project owner. Table 3 is the example of the quality criteria distribution in the project.

The weight values also distributed in the form of weight summary to one ($P_{QC1}+P_{QC2}+P_{QC3}+…+P_{QCn}= 1$). The quality criteria in the project are always deduced from the set of organizational quality criteria as listed in table 1 but may not need to be all included into the project (some P_{QC} may have value as zero).

Table 3. Example of quality weighting distribution in the project

	Project Quality Criteria Weighting Distribution			
	Quality Criteria 1 (QC₁)	Quality Criteria 2 (QC₂)	Quality Criteria 3 (QC₃)	Quality Criteria n (QCₙ)
Project	P_{QC1}	P_{QC2}	P_{QC3}	P_{QCn}

The process quality contribution score (*CS*) is calculated from the weight distribution of processes to the quality criteria set in OSSP and quality criteria distribution in the project judged by project owner. From the organization experience, OSSP can indicate the effort level (*E*) to perform each process or create each work product and set it as the property of process and artifact. This effort level can be measured in different ways, for simplicity, we count the effort level in term of estimated workload to finish the job.

The modification of process and artifact through the process tailoring, either by deleting or altering those process and artifact, will yield the changing of effort level. This effort level help project owner better decide the final process tailoring degree which balances the quality criteria needed and the effort expenses. Table 4 shows the example of the quality contribution from process to the project and Table 5 is an objective-based scoring metric calculated from this tailoring framework.

Table 4. Example of process quality contribution score to the project

Process	Process Quality Contribution Score (CS) to project			
	Quality Criteria 1 (QC₁)	Quality Criteria 2 (QC₂)	Quality Criteria 3 (QC₃)	Quality Criteria n (QCₙ)
Process 1 of	CS_{P1QC1}	CS_{P1QC2}	CS_{P1QC3}	CS_{P1QCn}

Process	Process Quality Contribution Score (CS) to project			
	Quality Criteria 1 (QC_1)	Quality Criteria 2 (QC_2)	Quality Criteria 3 (QC_3)	Quality Criteria n (QC_n)
Project				
Process 2 of Project	CS_{P2QC1}	CS_{P2QC2}	CS_{P2QC3}	CS_{P2QCn}
.	.	.	.	.
Process n of Project	CS_{PnQC1}	CS_{PnQC2}	CS_{PnQC3}	CS_{PnQCn}

Table 5 shows the example of process tailoring result from the quality/effort scores, which represent the quality gain from the workload expense. In this example, there are four quality criteria QC_1, QC_2, QC_3, and QC_4 while we are deciding to tailor process P_1, process P_2, process P_3 and process P_4 of the project. Each process has score distribution that shows the relationship between process and quality criteria in term of quality/workload. This information helps project owner decide the appropriate tailoring degree for the project. The explanation of how to calculate the scoring metric is in the research approach section.

Table 5. Example of Objective-based scoring metric

	Quality/Effort score			
	Process 1	Process 2	Process 3	Process 4
Weighting score to the Quality Criteria	0.3164	0.3362	0.1661	0.1813

By considering the scoring metric, the practitioner can decide can decide the process that should be performed. This may be judged according to the organization rule, for instance if the rule indicate the decision level as 0.3, practitioner may decide to omit the process P_3 and process P_4 from the project. In current research state, we provide the tailoring scores for practitioner to decide the degree of tailoring level.

In the future, we will develop the determination mechanism to help justify the level of tailoring degree. As such mechanism, it will provide the tailoring score that can be used to help practitioner justify the acceptable tailoring degree. The acceptable tailoring degree may not only depend on the effort reduction but also on certain standard, such as CMMI maturity level.

4 Research Approach

The tailoring framework proposed here used objective-based scoring metric in helping project owner decides the tailoring degree that is suitable for their project environment. The weighting quality criteria in this framework obviously depend on the justification of the practitioner and unique to their conditions. So the weighting judgment needs results from the field experiment for appropriateness. The manipulation of weighting distribution between the project and organization (W_{PQC}, P_{XQC}) will be the key to make a justifiable scoring metric. Several approaches can be adapted to fill in this weighting manipulation. The idea such as value-based software engineering (VBSE) framework can be used to develop the manipulation mechanism [10].

Concept of VBSE was adopted in many research areas that has underlying argument about the spent effort should yield the maximum value that satisfies all the stakeholders. By this regard (maximize value with the reasonable effort), we applied the VBSE concept into our research approach as the quality criteria represent the value that stakeholders (organization/ practitioner) will be satisfied with. The effort is straightforward interpreted as workload (E) needed to perform the process or create artifact. Firstly, the weight distribution of the process and artifact relate to the quality criteria must be identified in the OSSP level (as listed in table 2). These values are from the mutual consent of the working group assigned by organization. Next, the project owner decides the weighting criteria suitable for the project environment (Table 3). Then we can calculate the quality contribution score (CS) from each process and list it in the table 4 by multiplication between $P * W$, for instance the quality contribution score of process P_1 to quality QC_1 (CS_{P1QC1}), as $P_{QC1} * W_{PQC1}$. Lastly, we calculate the quality gain with the workload expense (quality/workload) as the result of value (CS/E) where E is the workload or effort to perform process or create artifact (table 9). The scoring metric (table 5) can be calculated by normalizing the (CS/E) value as a weight distribution to one. To illustrate how to achieve the result in table 5, the example of weighting distributions are filled into the structure of table 2 and table 3 as table 6 and table 7 respectively but are shown only the process part as the artifact part can be calculated in the same manner.

Table 6. Weighting distribution of process in OSSP

Process / Artifact	Organization Quality Criteria Weighting Distribution			
	Quality Criteria 1 (QC_1)	Quality Criteria 2 (QC_2)	Quality Criteria 3 (QC_3)	Quality Criteria n (QC_n)
Process 1	0.4	0.3	0.1	0.2
Process 2	0.5	0.2	0.2	0.1
Process 3	0.3	0.3	0.1	0.3
Process 4	0.2	0.2	0.5	0.1

Table 7. Weighting distribution of quality criteria in project

	Project Quality Criteria Weighting Distribution			
	Quality Criteria 1 (QC_1)	Quality Criteria 2 (QC_2)	Quality Criteria 3 (QC_3)	Quality Criteria n (QC_n)
Project x	0.5	0.3	0.1	0.1

The quality contribution score (CS) of each process to project are calculated as a result in table 8. Assuming the workload expense (E) to perform each process is displayed as in table 9. We can normalize the quality gain from workload expense as the example result shown in the Table 5.

Table 8. Process quality contribution to project

Process	Process Quality Contribution Score (CS) to project			
	Quality Criteria 1 (QC_1)	Quality Criteria 2 (QC_2)	Quality Criteria 3 (QC_3)	Quality Criteria n (QC_n)
Process 1	0.20	0.09	0.001	0.002
Process 2	0.25	0.06	0.02	0.01
Process 3	0.15	0.09	0.01	0.03
Process 4	0.1	0.06	0.05	0.01

Table 9. Workload expense of each process

	Workload Expense (E)			
	Process1 (E1)	*Proces2 (E2)*	*Press 3 (E3)*	*Process4 (E4)*
Workload	15	15	25	18

The decision level may be set up for helping the tailor level judgment. As the value in scoring matrix shows the quality gain from the workload expense, organization can judge the reasonable quality/effort level that appropriates to their environment.

5 Conclusions and Future Works

This research proposed process tailoring framework with objective-based scoring metric. By adopt the concept of VBSE; this systematic tailoring approach calculates the quality/effort distribution score that show the value gain from spent effort. This helps practitioner make a better decision to what the appropriate degree of tailoring suitable for their project environment.

In the future, research work aims to quantify the relationship between quality loss level from the process tailoring and the standard reference level such as CMMI maturity standard. This will help practitioners justify the appropriate level of tailoring by not only concern the effort reducing but also commit to the acceptable reference standard. From our point of view, the question that must be answered is how to tailor process and still does not fail the required standard criteria. The acceptable degree of the quality loss from the process tailoring can be the solution to that question. So to quantify the relationship between CMMI maturity standard level and the quality criteria from our tailoring framework will be a challenging one.

To enhance the usefulness of proposed framework, we plan to implement it with the methodology that refines the appropriate weighting values. As the matter of fact, how well the scoring metric represents the quality/effort distribution depends very much on the correctness of the weight distribution that OSSP must provides in the organization level. This paper does not describe in details how to achieve those values but it is critical to have the methodology to determine the appropriate value and how to refine the value according to the results of implementation.

6 References

1. Bob McFeeley, "IDEAL: A User's Guide for Software Process Improvement", CMU/SEI-96-HB-001, Carnegie Mellon University, Pittsburgh, February 1996, pp. 1–6. (references)
2. Mark P. Ginsberg and Lauren H. Quinn, "Process Tailoring and the software maturity model", CMU/SEI-94-TR-024, Carnegie Mellon University, Pittsburgh, November 1955, pp. 1–53. (references)
3. M. Sivashankar, A.M.Kalpana, Dr. A. Ebenezer Jeyakuma, "A Framework Approach Using CMMI for SPI to Indian SME's", 2010 International Conference on Innovative Computing Technologies (ICICT), 2010.
4. M.P.Thapliyal, Pratibha Dwivedi, "Software Process Improvement in Small and Medium, Software Organizations of India", International Journal of Computer Applications (0975 – 8887), Vol. 7– No.12, October 2010, pp. 37–39.
5. Maria Habib, Sana Ahmed, Amna Rehmat, Malik Jahan Khan and Shafay Shamail, "Blending Six Sigma and CMMI - An Approach to Accelerate Process Improvement in SMEs", Proceedings of the 12th IEEE International Multitopic Conference, December 2008, pp. 386–391.
6. N. du Preeza, D. Luttersb and H. Nieberdinga, "Tailoring the development process according to the context of the project", CIRP Journal of Manufacturing Science and Technology, Vol.1 Issue 3, 2009, Pages 191-198.
7. National Defense Industrial Association (NDIA), "The Economics of CMMI", October 2009, pp. 1–23. (references)
8. Il-Chul Yoon, Sang-Yoon Min and Doo-Hwan Bae, "Tailoring and Verifying Software Process", APSEC 2001, Eighth Asia-Pacific Software Engineering Conference, December 2001 Macao, China.
9. Mary Beth Chrissis, Mike Konrad and Sandy Shrum "CMMI: Guidelines for Process Integration and Product Improvement", 2nd Aedition, Addison-Wesley, Pearson Education, Inc.
10. S. Biffl, A. Aurum, B. Boehm, H. Erdogmus, P. Grunbacher (eds.), "Value-Based Software Engineering", Springer, 2006.
11. Gleison Santos et al., "Implementing Software Process Improvement Initiatives in Small and Medium-Size Enterprises in Brazil", Sixth International Conference on the Quality of Information and Communications Technology.
12. Stefan Biffl, Dietmar Winkler, Reinhard Hohn, and Herbert Wetzel, "Software Process Improvement in Europe: Potential of the New V-Modell XT and Research Issues", Softw. Process Improve. Pract. 2006; 11: 229–238.
13. Deepti Mishra and Alok Mishra, "Software Process Improvement in SMEs: A Comparative View", ComSIS Vol. 6, No. 1, June 2009.
14. T. Gorschek and C. Wohlin, "Identification of Improvement Issues Using a Lightweight Triangulation Approach", Proceedings European Software Process Improvement Conference (EuroSPI), pp. VI.1-VI.15, Graz, Austria, 2003.
15. F. G. Wilkie, D. McFall, McCaffery, "An evaluation of CMMI process areas for small- to medium-sized software development organisations", Software Process: Improvement and Practice, Special Issue: Special Issue on Software Product Lines, Volume 10, Issue 2, pages 189–201, April/June 2005.
16. Pettersson F., Ivarsson M., and Gorschek T., "A Practitioner's Guide to Light Weight Software Process Assessment and Improvement Planning", Journal of Systems and Software, vol. 81(6), 2008, pp. 972-995.

Analysis of SPI strategies and their modeling
- Exploring the art of SPI promotion -

Noriko Wada (Sony Corporation),
So Norimatsu (Norimatsu Process Engineering Laboratory),
Toshiyuki Kishi (Toshiba Corporation)

Abstract

Software process improvement (SPI) strategies were collected from organizations participating in the Japan SPI Consortium (JASPIC), and their key concepts and elements were extracted. These elements were further categorized into strategic promotional factors and contextual factors, along with outcome indicators, and an SPI Strategy Model was devised as a framework for analyzing them. One objective of this model is to understand the art of SPI promotion by exploring the interrelationships among these factors. Another objective is to utilize the findings for the evaluation of SPI activities with a view to making them more successful.

1 Introduction

Many SPI experiences are reported in the SPI community. Successes reported in these cases are of different kinds and degrees, including ISO certification, maturity level achievement, quality improvement, and productivity increase. Factors that contributed to these successes are also mentioned, including strong sponsorship and engineering process group (EPG)'s communication skills and leadership.

However, there are some cases in which a similar level of success could not be achieved despite the existence of these factors. Other cases seem to indicate the existence of other factors accounting for success. We considered SPI strategy to be one of these critical factors accounting for success and, in order to understand what specific factors would contribute to success, analyzed the kinds of strategies used in various SPI cases.

2 SPI Strategy

2.1 Definition of SPI strategy

There are various definitions of "strategy." Here we define "strategy" as "a technique to coordinate resources and activities from long-term and global perspectives" or "the plan that reflects such coordination." According to this definition, SPI strategy can be defined as "a technique to coordinate resources and activities from the perspective of advancing capabilities to promote SPI activities, considering the influences of short-term tasks and their results, for the purpose of achieving improvement objectives (i.e. improvement targets) associated with the higher-level objectives (i.e. business objectives)" or "the plan that reflects such coordination."

For example, an organization can adopt the approach of "improving quality through increasing organizational maturity level using an improvement model." This approach could involve activities such as gap identification, definition of new processes, standardization, and deployment with a view to achieving a higher maturity level. There could be different ways of coordinating these activities.

2.2 Case studies of SPI strategy

The strategies observed in the six sampled SPI activities are described in this section and the SPI cases are introduced. These activities were performed in some of the organizations currently participating in the JASPIC consortium.

2.2.1 Departmental SPI

a) Top-down type
With strong direction from upper management, an organizational structure for SPI was established at the outset. A full-time software engineering process group (SEPG) was assigned, standard processes were defined, and then these processes were deployed in development projects.

Organizational scale	Dozens of people who develop IT systems
Target	Increasing capability maturity model integration (CMMI) level
Motivation	Direction from upper management
Diagnosis	Mini assessments are conducted.
SPI structure	Organization management and full-time SEPG are at the heart of the SPI promotion structure.
Role of SEPG	Define and revise organizational standard processes Provide process training

	Audit processes and work products
Training/support for SEPG	Attendance at training courses provided by Corporate SEPG Information sharing with other organizations in the company
Results	Achievement of CMMI maturity level Establishment of an organization's standard processes and rules Implementation of milestone management tool

b) Bottom-up type
Cases of this type were prompted by development team members' awareness of the problems. Minimal resources were used for the activities.

Organizational scale	20 people who develop embedded software
Target	The definition of a project process based on XP, implementation, and passing on of experiences to the next project
Motivation	To eliminate chaos in the later phases of a project and to reduce overtime work or working on weekends
SPI structure	Three core members of the project team lead SPI activities, discussing with all project team members
Role of SEPG	Increasing awareness and involvement of the project team members, and promotion
Training/support for SEPG	External benchmarking Providing cases of other organizations in the company
Results	Decrease of rework Continuation of SPI at project level

2.2.2 Divisional SPI

a) Top-down type at the start of new organization
An SPI group was established as part of a new organizational structure, and SPI activities were started as an organizational activity.

Organizational scale	500 people who develop embedded software (Organization created by amalgamation of organizations)
Target	Establishment of organization processes Establishment of a milestone management tool
Motivation	Strong direction by top management to avoid process anarchy when building a new organization.
Diagnosis	Mini assessments are conducted.
SPI structure	Two full-time EPGs (Directly reporting to sponsor)
Role of SEPG	Define and deploy organizational processes Promote management tool
Training/support for SEPG	Attendance at training courses provided by Corporate SEPG Sharing information with other organizations in the company
Results	Establishment of organization processes Implementation of a milestone management tool

b) Bottom-up type by a small group
SPI activities were conducted by project team members who were aware of the problem and corporate organizations provided external support.

Organizational scale	500 people who develop embedded software
Target	Continuation of SPI activities in the absence of sponsorship
Motivation	Support from a corporate group and individual motivation for SPI
SPI structure	One project team member serving as division EPG
Role of SEPG	Support for development projects (Produced development guides and provided training; assisted projects in terms of process management, including issue management).
Training/support for	Attendance at training courses provided by Corporate SEPG

SEPG	Sharing information with other organizations in the company
Results	Continuation of SPI activities through the project (although no apparent increase in maturity level of the organization)

2.2.3 Corporate SPI

a) 3-layer SPI promotion type

This project was prompted by a corporate initiative to solve business issues. SPI activities were promoted by 3-layer structure covering the entire company.

Organizational scale	3000 people responsible for various application domains
Target	Continuous SPI activities
Motivation	Awareness of mid- to long-term business issues
Diagnosis	Mini assessments were conducted.
SPI structure	3-layer structure was established.
Role of SEPG	(1) Corporate EPG supported divisional and group EPG as an SPI expert. - EPG development (provided training, operated forums to share know-how and experiences among EPGs) - Support for promotional activities (Mini assessments, consulting) (2) Corporate EPGs monitored the status in the company and reported to top managements
Training/support for SEPG	Cooperation with external research facilities External benchmarking
Results	Establishment of 3-layer EPG structure and continuous conduct of SPI activities in each organization

b) Building SPI culture type

Corporate EPG promoted SPI by forming a network of engineers whose awareness was high

Organizational scale	3000 people
Target	Developing culture of SPI through community-based approach (transcending organizational barriers) (1) Acceleration of SPI in targeted business fields (2) Increasing corporate-wide efficiency of SPI activities through sharing of SPI assets.
Motivation	Awareness of mid-term business issues
SPI structure	Corporate SPI office with promotional function
Role of SEPG	(1) SPI support for targeted organizations (support for working group operation and support for management) (2) Identification of project/divisional SPI efforts and promotion of results sharing through "Community" and "Forum" (3) Introduction of external SPI cases and new technologies, and support for pilot activities
Training/support for SEPG	External benchmarking.
Results	(1) Improved capabilities of targeted organizations (2) Senior management's increased awareness of SPI activities

3   SPI Strategy Model

To select the most effective SPI strategy for the organization, it is necessary to identify candidate strategies, and analyze their differences in terms of outcomes. This analysis tends to be more complicated than a simple comparison because the SPI strategy should address the unique situation of the individual organization. To facilitate this effort, we devised the SPI Strategy Model as a framework for describing, comparing, analyzing, and evaluating SPI strategies.

3.1 **Fundamental Concept of SPI Strategy Model**

Elements that describe the components of SPI strategy and the outcome of SPI can be grouped into the following three categories.

    (a) C: Contextual factor: elements that describe the environment and situations in which the SPI activities take place

    (b) P: Promotional factor: elements that describe the major decisions to promote SPI activities

    (c) O: Outcome indicator: elements that indicate the degree of success as a result of SPI.

    Promotional factors are the basic elements that the organization can determine in order to support the SPI activities. Contextual factors may not be arbitrarily selected by the organization in the short run, but they affect decision-making on promotional factors and the conduct of SPI strategy, thus influencing the outcome.

For a single set of SPI activities in an environment described by Contextual factors (Cn) and with a selected strategy composed of Promotional factors (Pn) resulting in the outcome (On), the scenario can be expressed as the following relationship.

  Cn -> {Pn -> On}

   (*) "->" denotes a causal relationship.

   (*) Subscript "n" denotes that there could be multiple factors.

If these elements can be expressed quantitatively, this relationship can be described in a functional form, such as $O_i = F_i (C_n, P_n)$.

Although these elements include the common factors found in multiple SPI cases as described in section 2, different organizations can adopt their own unique factors when choosing outcome indicators. Also, the degree of importance of each factor (i.e. weight) with respect to the contribution to the result may vary depending on the context.

Furthermore, the categorization of these elements may change, and there may be an intricate interaction among these three categories. The organization can "select" some of the Contextual factors (e.g. the organization can set the scope of SPI activities, so this can be considered as part of the Promotional factors). Some aspects of the environment can "change" as a result of the activities, or their changes could be set as the target (e.g. since "change of the organizational culture" can be one of the Outcome indicators, the Contextual factor can also be treated as Outcome).

The objective of the SPI Strategy Model is to classify the elements of these factors and understand their interrelationships.

3.2 **Elements of SPI Strategy Model**

Closer analysis of the SPI strategies that are introduced in Section 2 can extract additional elements, as shown in Appendix A. The types of these elements are discussed in this section.

Promotional factor:

Promotional factors can usually be identified by analyzing what stakeholders considered most carefully in conducting SPI activities, or through the design of the SPI plan. Some factors can be set implicitly without documenting them.

Table 1: Promotional factors

Factor	Description
Motivation	How to motivate stakeholders
Norm	Criteria to determine the validity of SPI (e.g. process model, quality attributes)
Diagnosis	Method to evaluate the validity of SPI based on criteria
Structure	Organizational structure for SPI promotion (e.g. hierarchical, horizontal, vertical)
Education	Mechanism to communicate norms and new processes
Support	Support from external entities
Assetization	Mechanism to make the outcome of SPI (i.e. process assets) available for reuse

Deployment	Mechanism to deploy the outcome of SPI and expand the scope of activities
Involvement	Mechanism to involve stakeholders

Contextual factor:
Contextual factors can usually be identified by analyzing why the specific promotional factor was chosen. When two different SPI cases with the same promotional factor lead to different outcomes, a causal analysis may identify a hidden contextual factor.

Table 2: Contextual factors

Factor	Description
Scope	Scope, structure and scale of the target organization or activities
Resources	Resources provided for the SPI activities
Stakeholders	Important stakeholders and their expectations and needs
Business Strategy	Needs, expectations, constraints imposed by the business strategy
Culture	Organizational paradigm (e.g. behavior in response to changes, communication mechanisms, team vs. individual behavior)
SPI Experience	Historical SPI experiences and accumulated process assets

Outcome indicator:
SPI activities are ultimately evaluated in terms of their direct and indirect influence on the attainment of business objectives. Short-term evaluation may be complicated and multifaceted because multiple stakeholders may have different expectations and criteria.

Table 3: Outcome indicators

Indicator	Description
Business Performance	Degree of contribution to the business performance
Improvement Performance	Degree of achievement measured against the norm, including secondary effects if any
Satisfaction	Degree of satisfaction of the stakeholders
Issues	Problems, issues, and risks arising from SPI activities

3.3 **Application of SPI Strategy Model and further development**

Application of SPI Strategy Model in the organization can include the following:
(1) Supporting development of strategy and plan at the time of startup. By referring to other SPI cases, it becomes easier to develop the strategy most suitable for the organization's environment and its objectives.
(2) Evaluating - a current SPI strategy. By diagnosing/evaluating the factors during and at the end of SPI activities, strategic positions can be identified, and the need to change the strategy or the plan can be determined.

Steps to improve the framework and SPI Case database can be as follows:
(1) Collect cases:
SPI cases can be collected and shared across various organizations through JASPIC's activities, public conferences and workshops.
(2) Analyze cases:
Additional elements can be extracted and their interrelationships (e.g. identification of causal relationships and evaluation of the degree of contribution) can be analyzed. This analysis would probably require discussion with the SPI promoter or the stakeholders.
(3) Analyze strategic patterns:
Patterns of SPI strategy can be identified through comparison. A specific pattern is presumed to have a set of factors with intended and coherent causal relationships. An organization can also shift to another pattern when the context changes.
(4) Analyze relationships:
Correlations can be identified and the degree of such relationships among various factors determined. By identifying critical factors from the comparison of successful and unsuccessful

instances, empirical laws that commonly manifest themselves as tips or common sense (i.e. certain situations will lead to certain commonly recognized results) can be explained in terms of the model, which can facilitate development of comprehensive guidelines for handling SPI strategy.

This paper introduced the first two of these steps, and we will continue other analyses.

4 Conclusion

We considered the SPI strategy to be a mechanism for maximizing the SPI outcome by selecting appropriate SPI promotional factors in a recognized organizational context, and extracted elements that highlight these characteristics. Through the exchange of SPI cases and discussion of their strategies, we strengthened our recognition of the importance of SPI strategies, and confirmed the effectiveness of a forum such as a consortium. The SPI Strategy framework is effective for highlighting the characteristics of SPI cases, and exploring their patterns. By formalizing what has been shared as anecdotal information, we can deepen our understanding of the success factors in SPI activities. We intend to refine the model through further collection of SPI cases, analysis of factors and identification of their patterns and interrelationships.

5 Acknowledgements

6 Literature

(1) Hideto Ogasawara, Takumi Kusanagi, Minoru Aizawa: Proposal and practice of SPI framework - Toshiba's SPI History since 2000-, EuroSPI2012, Vienna, Austria, (2012)
(2) N.Wada: Exploring the relationships between organizational culture and SPI activities according to IDEAL model, SEPG Conference CMU/SEI, Boston, USA. (2003)
(3) Takahiro Nakayama; OpenSPI - Growth and deployment of bottom-up SPI, SPI-Japan07, Niigata, Japan (2007)
(4) Koichi Nagachi, Hirotake Goto: A case of SPI community approach - A realization of cross-organizational innovative change -, SPI-Japan08, Kobe, Japan, (2008)

7 Author CVs

Noriko Wada
 Sony Corporation
 Ms. Wada has 10 years of experience in SPI promotion at Sony Corporation, where she currently manages SPI activities in the semiconductor business group as a senior software process manager.

So Norimatsu
 Norimatsu Process Engineering Laboratory, Inc.

Mr. Norimatsu has been involved in various SPI activities as an EPG member, trainer, and consultant. He is a certified CMMI high maturity lead appraiser.

Toshiyuki Kishi

Toshiba Corporation

Since 2002, Mr. Kishi has participated in SPI activities at Toshiba Semiconductor and Storage Company to implement CMM-based software process improvement.

Appendix A: Strategic elements of cases

		Promotional Factor									Contextual Factor						Outcome Indicator			
		Motivation	Norm	Diagnosis	Structure	Education	Support	Assetization	deployment	involvement	Scope	Resource	Stakeholders	Business Strategy	Culture	SPI Experience	Business Performance	Improvement Performance	Satisfaction	Issues
1	Top down type	Top indication	CMMI	Consultation	MSG, EPG	Case introduction	Stndard process	Organization	Simultaneous deplo	Top indication	Division (gene ralizati	General overhead	Top	customer and emplo	Top-down	Process audit	Confidential	Level achievement	Increase in persons-in-charge	Full-time person abatement
2	Bottom up Type	Problem-solving	agile	The number of entrie	Devel opper, suppo rter	Case introduction	Top assistance	Process for each theme	Needs base	Needs base	Whole company	A tool, training	Engineer, sponsor	employee satisfaction	active engineer	On-site consultant	Confidential	The number of asset and	improvement in operating	With totality evaluation
3	Top down type at the start of new organization	Top indication	SW-CMM	Mini assessment	MGR, EPG	original training	Head office organization	Organization process	As long as EPG	Top indication	Division (gene ralizati	Sponsor direct	Sponsor	customer and	A theoretical spot	Process construction	Confidential	Asset construction	The development purpose	declines by top change
4	Bottom up by a small group	Problem-solving	CMMI	number of division QA	One EPG person	original training	Stndard process	Organization process	Web page disclosure	Early adapter	Division (gene ralizati	Department al burde	Engineer, EPG	maintenance of the status	Rush work cultur	On-site involvement	Confidential	The base to activities enlarge	Acceleration of division	With no change of maturit
5	3 layers SPI promotion type	Give&take	CMMI, IDEAL	Assessment	Three hierarchy EPG	Whole company educa	External cooperation	Activities data, a	Word-of-mouth comm	Give&take	Whole company	Whole company activiti	Three hierarchy EPG	employee satisfaction	Research work peopl	On-site consultant	Confidential	Imploving the maturity	Increase in a improved organiz	Promoter fatigue
6	Building SPI culture type	information sharing	Community growth	The number of forum	Head office promotion	information sharing	Benchmark	Activity report	Exchange of idea	Incite a sense of	Whole company	A tool, training	Engineer, sponsor	employee satisfaction	active engineer	Facilitation	Confidential	Rate of employee satisfa	revitalization of the exchange of	With no totality evaluation

Implementation of a Lightweight Assessment Method for Medical Device Software

Valentine Casey, Fergal Mc Caffery,
Regulated Software Research Group & Lero
Dundalk Institute of Technology
Dundalk
Ireland
val.casey@dkit.ie , fergal.mccaffery@dkit.ie

Abstract

This paper outlines the development and implementation of Medi SPICE-Adept. Medi SPICE-Adept is a lightweight assessment method that has been designed for usage with the Medi SPICE software process assessment and improvement model which is currently being developed for the medical device industry. While the Medi SPICE model is detailed and comprehensive in its approach there is industry demand for a lightweight medical device process assessment and improvement method. To address this requirement Medi SPICE-Adept has been developed. Details on how this has taken place and the procedures for implementing a Medi SPICE-Adept assessment are presented. Information is also provided regarding how a Medi SPICE-Adept assessment was undertaken in an Irish based medical device company. A summary of the issues identified from this assessment and the actions taken to facilitate process improvement is also presented. Finally, plans for future work are discussed.

Keywords

Medical Device Software, Software Process Improvement, SPI, Lightweight Process Assessment Method, Medical Device Software Process Assessment and Improvement.

1 Introduction

Due to the potential threat that medical devices can pose to patients, clinicians and third parties their development is highly regulated. In recent years there has been a significant increase in the role and importance that software plays in the healthcare industry [1]. The outcome of this has been the functionality, complexity and size of software components in medical devices has substantially increased [2]. This development has been recognized by the European Union (EU) in their latest amendment to the Medical Device Directive (MDD) (2007/47/EC) [3]. As a result standalone software may now be classified as an active medical device in its own right in the EU. Given the importance and relevance of this measure the European Commission released a guidance document for the qualification and classification of standalone medical device software MEDDEV 2.1/ [4] in January 2012. In the United States (US) the Food and Drug Administration (FDA) is responsible for the regulation and approval of medical devices and has published software specific guidance documents for medical device software developers. These are the *General Principals of Software Validation* [5], *Off the Shelf Software Use in Medical Devices* [6] and *Guidance on the Content of Premarket*

Submissions for Software Contained in Medical Devices [7]. To address the increasingly important role that software now plays the FDA recently published the *Medical Device Data Systems Final Rule* [8] and *Draft Guidance in Relation to Mobile Applications* [9].

Given the mission critical nature of medical device software compliance with the relevant regulations, and international standards of the location where a medical device is to be marketed is obligatory [10]. In the EU the receipt of the CE mark is essential and in the US FDA approval is required. There are approval bodies performing similar roles in other countries including China, Canada, India, Japan, and Australia. A key international standard for achieving regulatory compliance is IEC 62304:2006 [11] and its aligned standards ISO13485:2003 [12], ISO 14971:2007 [13], EN 60601-4:2000 [14], IEC 62366:2007 [15], and IEC 60812:2006 [16]. Information is also provided in the technical report IEC/TR 80002-1:2009 [17] and IEC 61508:2010 [18]. Despite the provision of these international standards, regulations and guidance documents the information they offer is high-level and no specific methods for performing the essential activities required have been provided [19].

It is therefore not surprising given the importance that achieving regulatory approval plays that organisations developing medical device software have focused on achieving compliance rather than implementing efficient processes and undertaking process improvement [20]. Previously this was not a critical issue due to the limited proportion of software in medical devices and it was acceptable to take a compliance centric approach. This is no longer the case and there is now a particular requirement for highly effective and efficient software development processes to be in place. These processes need to be defined in a regulatory compliant manner and then adopted to produce the required deliverables in order to achieve approval [19]. To address this requirement Medi SPICE [21], a medical device software process assessment and improvement model is currently under development and is discussed in section 2. While there is a specific requirement for Medi SPICE which is a comprehensive and detailed model there is also industry demand for lightweight medical device software process assessment methods [22]. Medi SPICE-Adept has been developed to help address this requirement and this is discussed in section 3 along with the procedure for its implementation. Section 4 outlines how a Medi SPICE-Adept assessment was undertaken and provides a summary of the process improvement plan which was collaboratively developed based on the findings report. Section 5 provides a summary and context for future work based on this research.

2 Medi SPICE

Existing generic Software Process Improvement (SPI) models are available which include the Capability Maturity Model Integration (CMMI[®]) [23] and ISO 15504-5:2006 [24] (SPICE), but these were not developed to provide sufficient coverage of all of the necessary areas required to achieve medical device regulatory compliance [25]. To address the requirement for a medical device software process assessment and improvement model the Regulated Software Research Group (RSRG) at Dundalk Institute of Technology (DkIT) undertook extensive research in the area [19]. This resulted in work commencing on the development of Medi SPICE a medical device specific software process assessment and improvement model which is being devleeoped in collaboration with the SPICE User Group. This approach is in line with that taken for the development of Automotive SPICE [26] which is a software process assessment and improvement model which is domain specific to the automotive industry.

Medi SPICE is based upon the latest version of ISO/IEC 15504-5 (currently under ballot) and ISO/IEC 12207:2008 [27]. It also provides coverage of the relevant medical device regulations, standards, technical reports and guidance documents. These include IEC 62304:2006 and its aligned standards, the FDA regulations [28] and guidance documents and the European MDD and guidelines. The objective of undertaking a Medi SPICE assessment is to determine the state of a medical device organisation's software processes and practices in relation to the regulatory requirements of the industry and to identify areas for process improvement [29]. It can also be used as part of the supplier selection process when an organisation wishes to outsource or offshore part or all of their medical device software development to a third party or remote division [30].

Medi SPICE contains a Process Reference Model (PRM) which consists of forty two processes and fifteen subprocesses which are fundamental to the development of regulatory compliant medical

device software. Each process has a clearly defined purpose and outcomes that must be accomplished to achieve that purpose. Medi SPICE also contains a Process Assessment Model (PAM) which is related to the PRM and forms the basis for collecting evidence and the rating of process capability. This is achieved by the provision of a two-dimensional view of process capability. In one dimension, it describes a set of process specific practices that allow the achievement of the process outcomes and purpose defined in the PRM; this is termed the process dimension. In the other dimension, the PAM describes capabilities that relate to the process capability levels and process attributes, this is termed the capability dimension.

In line with ISO/IEC 15504-2:2003 [31] Medi SPICE process capability is defined over 6 levels:
- Level 0 Incomplete;
- Level 1 Performed;
- Level 2 Managed;
- Level 3 Established;
- Level 4 Predictable;
- Level 5 Optimizing.

The Medi SPICE PRM and PAM are being released in stages and each stage is extensively reviewed by interested parties from the SPICE User Group, representatives from international standards bodies and industry experts. This collaborative approach is seen as a key element in the development of Medi SPICE to ensure coverage of both the SPI and medical device software regulatory requirements [29]. Medi SPICE is a comprehensive and detailed model and its overall objective is to provide a conformity assessment scheme to support first, second or third party assessments. It is envisaged that results from these assessments may be recognized by the relevant regulatory bodies.

3 Medi SPICE-Adept

As outlined in section two there is a specific requirement for a detailed and comprehensive process assessment and improvement model which is specific to the medical device domain which Medi SPICE is being developed to address. As with other SPI assessments models i.e. CMMI® and IEC 15504-5:2006 a full Medi SPICE assessment will require considerable planning and resources to successfully undertake. While Medi SPICE is being developed with the objective of being as efficient as possible the necessity for rigour dictates the level of planning, resources and analysis required for its successful implementation. While the need for and importance of Medi SPICE is understood [21], it was also appreciated by the RSRG that there is a specific requirement for lightweight assessment methods in the medical device software industry [32]. In particular there was industry led demand for a lightweight assessment method based on Medi SPICE. This was communicated directly to the RSRG by numerous medical device organisations. To address this specific requirement Medi SPICE-Adept was developed. There were two additional objectives in undertaking this task. The first was the opportunity to leverage the extensive research [19] and level of detail which developing Medi SPICE provided. The second was the opportunity to identify and facilitate the use of agile and lean methods for medical device software development. The use of agile and lean methods in this context is an area that the RSRG are also currently researching to assist organisations increase the efficiency of their software development practices [33].

To be effective Medi SPICE-Adept required the employment of a lightweight approach for undertaking software process assessment and improvement. This included the use of a limited number of personnel to carryout and participate in the assessment while also maximising the benefit of the time and effort of those involved. It was envisaged that Medi SPICE-Adept would eventually encompass all the Medi SPICE processes. It was therefore recognized that an assessment could take place over a day or a number of days depending on how many processes were being assessed. It was also important that organisations could select the specific processes which were of most benefit for achieving their business goals. The focus of the method had to be on the evaluation of the essential practices, key work products and the achievement of the outcomes which were necessary for the attainment of the specific process purpose being assessed. Medi SPICE-Adept therefore needed to be process dimension centric in its focus. Finally the objective of undertaking a Medi SPICE-Adept assessment was not to receive formal certification or a rating, but rather to identify an organization's

strengths and weaknesses and to facilitate process improvement. Having defined the criteria which had to be met the next step was to undertake the development of Medi SPICE-Adept.

3.1   Developing Medi SPICE-Adept

The RSRG having previously successfully developed and implemented three lightweight software process assessment methods Adept [34], Med-Adept [35] and Med-Trace [36] the objective was to leverage that experience and utilise it for the development of Medi SPICE-Adept. It was in this context that work commenced on the development of Medi SPICE-Adept. It was recognized that this assessment method needed to cover more processes and provide more detailed analysis than those methods which had been previously developed. While this was the case Medi SPICE-Adept was still required to be lightweight to fulfil its purpose. The first task was to identify the initial Medi SPICE processes that would be utilised. The goal was to select a limited number of processes that would be most beneficial and relevant to industry. To achieve this, industry experts were consulted and ten processes were selected:

- Requirements Elicitation;
- System Architectural design;
- Systems Requirements Analysis;
- Software Requirements Analysis,
- Software Construction;
- Software Integration;
- Software Testing;
- Configuration Management;
- Change Request Management;
- Verification.

While these were the initial processes selected Medi SPICE-Adept will provide coverage of all the Medi SPICE processes and subprocesses.

The Medi SPICE PAM had been developed for each of the initial processes which were based on best practice as outlined by the latest version of ISO/IEC 15504-5 and the specific requirements of the medical device regulations, standards, technical reports and guidance documents. As a result each process had a defined purpose and outcomes, specific practices and work products were also included for the achievement of these outcomes and purpose. In addition each outcome and specific practice was cross referenced to the regulations, standards etc. on which it was based. To facilitate the assessment each of the initial processes were evaluated and specific questions identified based on the Medi SPICE PAM. In addition questions on the current or potential use of agile and lean methods were also identified and included. This work was undertaken by five members of the RSRG team with extensive experience of SPI and knowledge of medical device software development and included two experts in the area of lean and agile methods. Having defined the assessment instrument the next step was to develop the specific procedure for undertaking a Medi SPICE-Adept assessment.

3.2   The Procedure for Undertaking a Medi SPICE-Adept Assessment

Based on the RSRG's previous experience of developing and undertaking lightweight software process assessments [32] the seven stage procedure for undertaking a Medi SPICE-Adept Assessment was defined. It was decided the assessment team should normally consist of two assessors who share responsibility for conducting the assessment. The seven stages of the procedure are as follows: As a precursor to undertaking an assessment a preliminary meeting between the lead assessor and the company takes place. This is the first stage in the procedure and during this meeting the lead assessor discusses the main drivers for the company wishing to undertake an assessment. In this context the expectations regarding what can be realistically achieved are discussed and the procedure for undertaking the assessment is outlined. If there is agreement a schedule is drawn up. At the second stage the lead assessor has a meeting with the staff and management from the company who will be participating in the assessment where an overview of the Medi SPICE-Adept

assessment method is presented and details of what their participation will involve is outlined. On the agreed date the onsite assessment commences which is the third stage in the procedure. For each process the lead assessor conducts interviews based on the scripted Medi SPICE-Adept questions with the relevant personnel and evaluates the responses. The second assessor who also participates in the interviews prepares interview notes and may ask additional questions when clarification is required.   In addition work products may also be requested and briefly reviewed as part of this stage. A maximum of five processes are assessed in a single day with the interviews for each process taking approximately one hour. At the fourth stage the findings report is prepared off site based on the data gathered at stage three. Each process is reviewed in turn and where relevant particular strengths and issues (weaknesses) are identified based on the evaluation and interview notes. Suggested actions to address these issues and to facilitate process improvement are outlined and discussed.   The possibility for the use of appropriate agile and lean practices is also considered.   These are then documented and included in the findings report. This is a joint effort between the assessors and may include other SPI and/or lean and agile experts if required. The findings report is then presented to the management and staff who took part in the assessment which is the fifth stage in the procedure. Having provided adequate time for the findings report to be read and considered by the organization at the sixth stage the contents of the report is discussed in detail with the relevant management and staff.   At this point specific objectives for process improvement are collaboratively defined based on the findings report which results in the development of a process improvement plan.   Given the lightweight nature of Medi SPICE-Adept improvements that offer the greatest benefits in terms of compliance, quality and the achievement of business goals are selected for inclusion in this plan. At the seventh stage in the procedure the organisation having implemented the process improvement plan have the opportunity of having the processes reassessed. Based on this, a final detailed report is prepared which highlights what has been achieved and an updated improvement plan is also provided.

4   Implementation of a Medi SPICE-Adept Assessment

Having developed the Medi SPICE-Adept Assessment method and the procedure for its implementation the first assessment took place in an Irish based medical device company Western Medical (a pseudonym). The company have been developing and selling medical devices for over thirty years. Each of their products contains both hardware and software and the role that software plays has considerably increase over the last number of years. They market their medical devices in the EU and the US so their products must conform to the MDD to receive the CE mark and the FDA regulations.   Having agreed that an assessment would take place it was decided that the ten processes would be assessed over a two day period. This was undertaken by two assessors from the RSRG. Based on the results of the assessment a findings report was prepared and presented. The focus of the report was that for each process the company's strengths and issues were highlighted, in addition suggested actions to facilitate process improvement were also provided.   Based on the findings report the process improvement objectives and process improvement plan were collaboratively defined and developed with the company. A summary of the issues identified for each process and the actions taken to address these issues and facilitate improvement were outlined in the process improvement plan as follows:

Requirements Elicitation - A serious problem which emerged from the assessment was that the requirements specification produced was too high level. It did not capture the level of detail required to facilitate product development and both the hardware and software engineers had to try to guess what some of the specific requirements were. Marketing took the place of the company's customers and would not commit on their exact requirements. To address this, the use of prototyping and user scenarios to define key product features to facilitate the development of a detailed and comprehensive requirements specification was agreed. Senior Management support was also sought to reinforce the importance of ensuring marketing's full participation in defining and signing off on requirements.

Systems Requirements Analysis - The need for a systems requirements specification document was identified and this was an important omission as it was also necessary for traceability.   The high level requirements specification was used in this process and it was not adequate as it lacked the necessary level of detail. In addition system requirements were not prioritised. It was agreed that a

systems requirement specification would be developed based on the detailed requirements specification and that system requirements would be prioritised.

Configuration Management – While a configuration management tool was in place some key work products were not under configuration management and for those that were some important information was missing. In addition key features of the tool i.e. automatic merging was not being used. It was agreed that a configuration management strategy would be drawn up and implemented that ensured all the relevant work products were correctly managed and that the required level of detail was maintained. It was also agreed that all the relevant features of the tool would be updated and utilised.

Change Request Management - An ad-hoc change request management system was in place as a result requests were not prioritised, the level of detail provided was limited and the status of accepted change requests could not be determined until they were complete. To address these issues it was agreed that a formal change request management strategy would be developed and implemented. The selection of a tool to assist with this process would also be investigated. .

System Architectural Design - This process was dealt with in a very informal manner and a system architectural design specification was not produced. The IEC 62304:2006 requirement for a safety classification of each product was not addressed. This omission was important as this standard is now harmonized with the MDD and approved by the FDA. It was recommended and agreed that a formal system architectural design process would be put in place to facilitate the production of a system architectural design specification and to ensure that each product receives the relevant safety classification.

Software Construction - There was no defined strategy in place for performing unit testing. As a result it was carried out in an ad-hoc manner as the content and level of testing was left to the discretion of the individual tester. Documentation was not maintained and therefore test cases and results were not recorded. To address these issues it was agreed that a unit test strategy and procedures would be developed and implemented to ensure the required level of unit test coverage would be consistently provided and all unit test cases and results would be recorded.

Software Integration - There was no defined strategy for performing software integration or a documented integration plan. This process was therefore performed in an informal manner and the content and level of testing was left to the discretion of the individual tester. It was proposed and agreed that a software integration strategy should be developed and a software integration plan defined. The integration test procedures should be evaluated for correctness and completeness and they should be consistently implemented. Integration test cases and test results should also be documented and recorded.

Software Testing - Specific software testing did not take place after software integration. The next tests undertaken were on the complete system which incorporated both hardware and software. The need for a software testing process was discussed and agreed in the light of the requirements of the medical device standards and in line with best practice. It was therefore proposed that a software testing strategy and procedures should be developed and implemented and test cases and results documented and recorded.

Verification - In general there were good verification procedures in place. One issue highlighted by the assessment was that code reviews were only performed very late in the process. To address this, it was agreed that code reviews would take place prior to unit testing.

The findings report was positively received by Western Medical as was the whole assessment procedure. The collaborative nature of the development of the process improvement plan provides motivation for its successful implementation. The plan is currently being implemented and when this is complete the opportunity to have the processes reassessed is available. While in this paper we have focused on the negative issues we identified the company had very good risk management and traceability procedures in place. It is important to state that Medi SPICE-Adept highlights the strengths as well as the weaknesses in an organization.

5 Conclusion

It is important to stress what the differences will be between a Medi SPICE assessment and Medi SPICE-Adept. Medi SPICE is a comprehensive and detailed domain specific process assessment and improvement model for medical device software development. When the model is complete a Medi SPICE assessment method will be developed which will facilitate in-depth analysis and assessment of each process and this will include the determination of its capability level. As a result the findings from a Medi SPICE assessment will be extensive, comprehensive and detailed. On the other hand Medi SPICE-Adept which is also based on the Medi SPICE model has a different purpose it is a lightweight assessment method. Its focus is more high level and its role is to provide a snap shot of key aspects of medical device software development processes and to assist with regulatory compliance and process improvement in this context.

Medi SPICE-Adept is the largest and most detailed lightweight assessment method developed by the RSRG. It is the result of industry demand and was developed to meet the requirement from a more extensive, but lightweight medical device software assessment method. A Medi SPICE-Adept assessment has recently been successfully implemented in Australia by our colleagues in Griffith University. Feedback from the Australian assessment was very positive and a specific request was that we include additional project management processes. This is in line with our strategy for Medi SPICE-Adept and we plan to incorporate the remaining thirty two Medi SPICE processes and twelve subprocesses in the coming year. Given the level of demand it is also our objective to carry out additional Medi SPICE-Adept assessments both in Ireland and in collaboration with our international colleagues.

6 Literature

1.   Abraham, C., E. Nishiharas, and M. Akiyama, *Transforming healthcare with information technology in Japan: A review of policy, people, and progress.* International Journal of Medical Informatics, 2011. **80**(3): p. 157-170.

2.   Lee, I., G. Pappas, R. Cleaveland, J. Hatcliff, B. Krogh, P. Lee, H. Rubin, and L. Sha, *High-Confidence Medical Device Software And Systems.* Computer, 2006. **39**(4): p. 33 - 38.

3.   European Council, *Council Directive 2007/47/EC (Amendment).* 2007, Official Journal of The European Union: Luxembourg.

4.   European Commission, *MEDICAL DEVICES: Guidance document- Qualification and Classification of stand alone software (MEDDEV 2.1/6).* 2012, Brussels, Belgium.

5.   US FDA Center for Devices and Radiological Health, *General Principles of Software Validation; Final Guidance for Industry and FDA Staff.* 2002, CDRH: Rockville.

6.   US FDA Center for Devices and Radiological Health, *Off-The-Shelf Software Use in Medical Devices; Guidance for Industry, medical device Reviewers and Compliance.* 1999, CDRH: Rockville.

7.   US FDA Center for Devices and Radiological Health, *Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices.* 2005, CDRH: Rockville.

8.   US FDA, *21 CFR Part 880 Medical Devices; Medical Device Data Systems Final Rule.* Federal Register, 2011. **76** (31): p. 8637 - 8649.

9.   US FDA, *Draft Guidance for Industry and Food and Drug Administration Staff Mobile Medical Applications, July 21.* 2011, http://www.fda.gov/downloads/MedicalDevices/DeviceRegulationandGuidance/GuidanceDocuments/UCM263366.pdf, Accessed 20th February 2012.

10.   Vogel, D.A., *Medical Device Software Verification, Validation and Compliance.* 2010, Norwood MA: Artech House. 432.

11.   IEC 62304:2006, *Medical device software—Software life cycle processes.* 2006, IEC: Geneva, Switzerland.

12. ISO 13485:2003, *Medical devices — Quality management systems — Requirements for regulatory purposes.* 2003, ISO: Geneva, Switzerland.

13. ISO 14971:2007, *Medical Devices — Application of risk management to medical devices.* 2007, ISO: Geneva, Switzerland.

14. BS EN 60601-1-4:2000, *Medical Electrical Equipment, Part 1 - General requirements for safety.* 2000, BSI: London.

15. IEC 62366:2007, *Medical devices - Application of usability engineering to medical devices.* 2007, IEC: Geneva, Switzerland.

16. IEC 60812:2006, *Analysis technique for system reliability - Procedure for failure modes and effects analysis (FMEA).* 2006, IEC: Geneva, Switzerland.

17. IEC/TR 80002-1:2009, *Medical device software Part 1: Guidance on the application of ISO 14971 to medical device software.* 2009, BSI: London.

18. IEC 61508:2010, *Functional safety of electrical/electronic/ programmable electronic safety related systems.* 2010, IEC: Geneva, Switzerland.

19. Mc Caffery, F., J. Burton, V. Casey, and A. Dorling, *Software Process Improvement in the Medical Device Industry*, in *Encyclopedia of Software Engineering*, P. Laplante, Editor. 2010, CRC Press Francis Taylor Group: New York. p. 528 - 540.

20. Denger, C., R. Feldmann, M. Host, C. Lindholm, and F. Shull. *A Snapshot of the State of Practice in Software Development for Medical Devices*. in *First International Symposium on Empirical Software Engineering and Measurement*. 2007. Madrid, Spain.

21. Mc Caffery, F. and A. Dorling. *Medi SPICE: An Overview*. in *International Conference on Software Process Improvement and Capability Determinations (SPICE)*. 2009. Turku, Finland.

22. Casey, V. and F. Mc Caffery. *Med-Trace: Traceability Assessment Method for Medical Device Software Development*. in *European Systems and Software Process Improvement and Innovation Conference*. 2011. Roskilde University, Denmark.

23. CMMI Product Team, *Capability Maturity Model® Integration for Development Version 1.2*. 2006, Software Engineering Institute, Pittsburch PA.

24. ISO/IEC 15504-5:2006, *Information technology - Process Assessment - Part 5: An Exemplar Process Assessment Model.* 2006, ISO: Geneva, Switzerland.

25. Mc Caffery, F. and A. Dorling, *Medi SPICE Development.* Software Process Maintenance and Evolution: Improvement and Practice Journal, 2010. **22** (4): p. 255 – 268.

26. Automotive SIG, *Automotive SPICE Process Assessment V 2.2*. 21 August 2005.

27. ISO/IEC 12207:2008, *Systems and software engineering - Software life cycle processes.* 2008, ISO: Geneva, Switzerland.

28. US FDA, *21 CFR Part 820 Quality System Regulations.* Code of Federal Regulations, Title 21, Revised April 1, 2011. **8**.

29. Mc Caffery, F., A. Dorling, and V. Casey. *Medi SPICE: An Update*. in *International Conference on Software Process Improvement and Capability Determinations (SPICE)*. 2010. Pisa, Italy: Edizioni ETS.

30. Casey, V., *Virtual Software Team Project Management.* Journal of the Brazilian Computer Society, 2010. **16**(2): p. 83 – 96.

31. ISO/IEC 15504-2:2003, *Software engineering - Process assessment - Part 2: Performing an assessment.* 2003, ISO: Geneva, Switzerland.

32. Mc Caffery, F., V. Casey, and M. Mc Hugh. *How Can Software SMEs become Medical Device Software SMEs*. in *European Systems & Software Process Improvement and Innovation Conference, (EuroSPI)*. 2011. Copenhagen: CCIS.

33. Cawley, O., I. Richardson, and X. Wang. *Medical Device Software Development - A Perspective from a Lean Manufacturing Plant*. in *11th International SPICE Conference on Process Improvement and Capability dEtermination 2011*. 2011. Dublin City University, Ireland.

34. Mc Caffery, F., I. Richardson, and G. Coleman. *Adept – A Software Process Appraisal Method for Small*

to Medium-sized Irish Software Development Organisations. in *European Systems & Software Process Improvement and Innovation (EuroSPI 2006)*. 2006. Joensuu, Finland.

35. Mc Caffery, F. and V. Casey. *Med-Adept: A Lightweight Assessment Method for the Irish Medical Device Software Industry*. in *European Systems & Software Process Improvement and Innovation Conference, (EuroSPI)*. 2010. Grenoble, France.

36. Casey, V. and F. Mc Caffery, *A lightweight traceability assessment method for medical device software.* Journal of Software Maintenance and Evolution Research and Practice, 2011.

Acknowledgment

7 Author CVs

Dr Valentine Casey

Dr. Val Casey is a Senior Researcher with the Regulated Software Research Group in Dundalk Institute of Technology. His previous roles include Senior Lecturer and Research Area Leader at Bournemouth University, Researcher with Lero - the Irish Software Engineering Research Centre at the University of Limerick where he also lectured. He has over 20 years experience in the software industry. This included a number of senior management positions and the provision of consultancy services focusing on software process improvement, software testing, agile methods and global software development.

Dr Fergal Mc Caffery

Dr Fergal Mc Caffery is the leader of the Regulated Software Research Group in Dundalk Institute of Technology and a member of Lero. He has been awarded Science Foundation Ireland funding through the Stokes Lectureship, Principal Investigator and CSET funding Programmes to research the area of software process improvement for the medical device domain. Additionally, he has received EU FP7 and Enterprise Ireland Commercialisation research funding to improve the effectiveness of embedded software development environments for the medical device industry.

Process Assessment Model Optimization towards introduction of Light Assessment in Space Development

Yuko Miyamoto[1], Tatsuya Kaneko[1], Masafumi Katahira[1]

*[1] Japanese Aerospace Exploration Agency, 2-1-1 Sengen, Tsukuba, Ibaraki, 305-8505 Japan
1{Miyamoto.Yuko, kaneko.tatsuya, Katahira.masafumi}@jaxa.jp*

Abstract

JAXA is now introducing Software Process Assessment as an extensive product assurance activity to ensure the success of our space missions. The assessment is a tool to find out process issues and to encourage improvement activities both JAXA and suppliers. To maximize advantages of a ISO/IEC 15504[1] compliant assessment, JAXA have been developed own Process Assessment Model (JAXA-PAM). To drive improvement activities by performing assessments, it is needed to realize light and efficient assessment methodologies that include its PAM and procedure of assessment. JAXA-PAM is not just translation of ISO/IEC 15504 part5 but add interpretations based on characteristics of space systems and Japanese manufacturing culture. Furthermore, we add some modification on the capability indicators to realize light assessment. In this paper, JAXA's original approach to implementing process assessments, its adaptability for Japanese space development, and its effectiveness are described.

Keywords

Process Assessment Model (PAM), JAXA-PAM, Process Improvement, Software Process Assessment, Process Assurance, Space Development

1 Introduction

The space domain places great emphasis on the quality of both hardware and software in order to satisfy rigorous mission success, safety, and reliability requirements. To ensure mission success, by ensuring safety and reliability of space systems, is the one of the most important responsibility of JAXA. For this purpose, JAXA demands suppliers to apply numerous standards and performs various assurance activities. Since hardware is a main part and has a long history of space development, JAXA has concrete procedure to assure hardware products. For software, its functionality is becoming larger and more complex but its development period becomes shorter. To maintain quality of software in such situation, JAXA is performing Independent Verification and Validation (IV&V) as product assurance and now introducing a software process assessment as improvement oriented process assurance. The improvement-oriented assessment is focusing on understanding deployed processes, identifying process strengths and weaknesses, and making suggestions for future improvements. JAXA has performed ISO/IEC 15504[1]-based improvement-oriented process assessments as a trial in cooperation with suppliers, the assessed organization since 2008. In the trial assessment, software engineers and quality assurance engineers of suppliers are joined as assessors so that they could deeply understand their deployed processes and underlying issues in the processes to be able to

drive future improvement activities.

Through the trial assessments, JAXA is establishing plan how to get the assessment institutionalized. One of the biggest issues is the amount of efforts needed in an assessment. To make an assessment light and useful tool of improvement, JAXA starts re-organizing JAXA-PAM.

2 How to introduce improvement-oriented assessment in JAXA?

2.1 Overview of Process Assessment Framework in JAXA

2.1.1 JAXA Process Reference Model and JAXA Process Assessment Model

To introduce ISO/IEC 15504 (IS) [1] based assessment, we need to determine the Process Assessment Model (PAM). Since ISO/IEC 15504 allows freely to choose or develop original PAM, obeying normative described in ISO/IEC 15504 part 2, JAXA has developed own PAM to reflect characteristics of our products by modifying the ISO/IEC 15504 part 5. The first version of JAXA-PAM includes information regarding the characteristics of space development such as tips for identifying domain-specific, process-related risk, but it was more like ISO/IEC 15504 part 5 with translation. In consequence, it takes much time to analyze the First JAXA-PAM and performing assessments. Therefore, JAXA has started revising and re-organizing JAXA-PAM.

For the Process Reference Model (PRM), we chose the JAXA Software Development standard as a PRM in the first JAXA-PAM. The JAXA software Development standard that complies with ISO/IEC 12207 [2] covers not only the on-board software but also all JAXA-developed space system products, including rockets, satellites, and their ground systems. However, it was not appropriate for the standard to be treated as a PRM because JAXA's documentation rules for standards don't match the requirements for PRM that written in ISO/IEC 15504 part 2. Therefore we decide to develop new PRM that is simple copy of PAM as shown in Figure 2.1, then clarify the relationship between the PRM and the JAXA development standard.
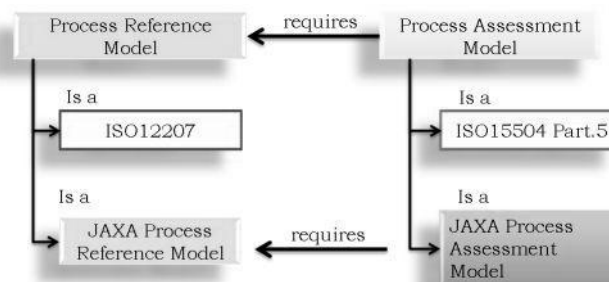
Figure 2.1: Assessment Framework Structure in JAXA

Because each space product has very different features in terms of the development process, JAXA defined various development standards to ensure the development of space systems. For the software development, JAXA has developed a hierarchical software standards structure to deal with such differences. In the lower layers of the standards structure, JAXA defines domain-specific software development standards for each domain considering characteristics of its software development processes. Figure 2.2 shows the relationship between the assessment framework, the standards and process instance that would be assessed.
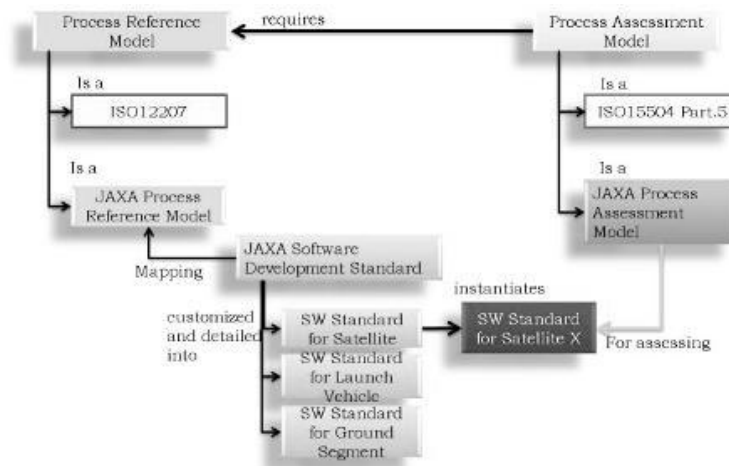
Figure 2.2: Relationship of PRM, PAM and standards

2.2   Trial Assessments of JAXA

JAXA has performed assessments for satellite on-board software development as trial assessments. In this section, the overview of the assessment strategy in JAXA and assessment procedure is explained.

2.2.1   Organizational strategy of the assessment

From the JAXA's point of view, we need to ensure the products have enough quality and dependability to launch. For the software, especially for the embedded software, it is hard to verify their quality and dependability by testing final products, i.e., the satellites and rockets. JAXA's strategy of assurance is to set quality gates in accordance with development phases and to accumulate the evidence of process and product assurance in each gate.

Furthermore, JAXA is aiming to assessments as not only the checking function, but also the improvement trigger to motivate implementing improvement cycles and performing continuous improvement. This improvement cycle will be the basis of quality implementation of products. Since we are in starting phase, JAXA organizes trial assessment as an process improvement tool. After trial phase, JAXA will implement assessments as an institution of quality assurance, instead of software audit. To enhance the use of assessment, we are trying to make assessment light, easy and effective to apply.

2.2.2   Assessment procedure

- Assessment team development

It is important for the assessment team to gather personnel and to develop knowledge of assessment team. In trial assessments, we tried several types of teams. One is the trilateral team that consists of assessors from JAXA (sponsor organization), a supplier (assessed organization) and an assessment provider (third organization). The team became big and rather inefficient in term of time management. However, it was effective in terms of knowledge development because the team would have wider viewpoints from three different backgrounds. Another type of team consists of two or three expert from a supplier (assessed organization). This type of team requires each assessor to have high skill of

assessment to achieve high efficiency. Sponsors and a lead assessor need to agree which type of the team is appropriate according to the purpose of the assessment and develop skills of assessors. To support assessor's basic knowledge, JAXA-PAM provides guides of key practice and related requirements described in JAXA development standard.

- Preparation:

In the first step of the assessment, assessors lean about the project or the organization that is assessed. Before the analysis of PAM, assessors need to know about the characteristics of products, organization structure, and situation of the target project. After that, Base Practices (BPs) and Generic Practices (GPs) in JAXA-PAM and related requirements in JAXA software development standard are analyzed. Then assessors identify the "Check Points" which are the critical points from perspective of the assessment objectives and project situation.

- Data collection (Review of Documents and Interviews):

The assessment team performs the document review and interview to collect evidence data. In the document review, the process instances are investigated based on a development plan document, work products, any enabling artifacts of processes such as review recodes and supplier's standards. Data are collected from point of view of JAXA-PAM and the "Check Points" identified in preparation task.

Based on the results of the document review, the strategy of the interview is determined. Time allocation is also part of the strategy. In some cases, the assessment team does not cover every BPs and GPs, but focus on the "Check Points" and possible weakness points that are identified in document review. In addition, interviewers try to find issues that the interviewee is dealing with in his daily development jobs, even if the issues are not directly related to JAXA-PAM or requirements of JAXA's standard.

- Consolidation:

Collected data are analyzed and consolidated as the assessment results. Assessment team verifies collected data and evaluates them in terms of BPs and GPs. In the JAXA's Assessment, rating of Process Attributes (PAs) is not mandatory. The result of PA rating is treated as supplemental information of evaluation of process strength and weakness. And PA rating is also needed to be certified as ISO/IEC 15504 compatible assessment.

- Reporting:

The assessment team provides the assessment report. The report includes evaluation results and identified opportunities of improvement. The assessment recodes such as all observed data, assessment log and analysis results are provided to assessed organization.

3 Re-organization of Measurement Framework of JAXA-PAM

3.1 Concept of Re-organization of Measurement Framework

From the experience in trial assessments, JAXA-PAM related issues have been found. JAXA is now revising JAXA-PAM. Figure 3.1 shows overview of JAXA-PAM. At first, we have worked on revising the process performance attribute. The main improvement points of process performance attributes were 1) make clear structure of Base Practice (BP) in a process, 2) reduce duplication of practices and 3) balance the weight of BP[3]. Next, JAXA is now working on reviewing capability dimension. In this chapter, the motivation and the concepts of re-organization of measurement framework are described.
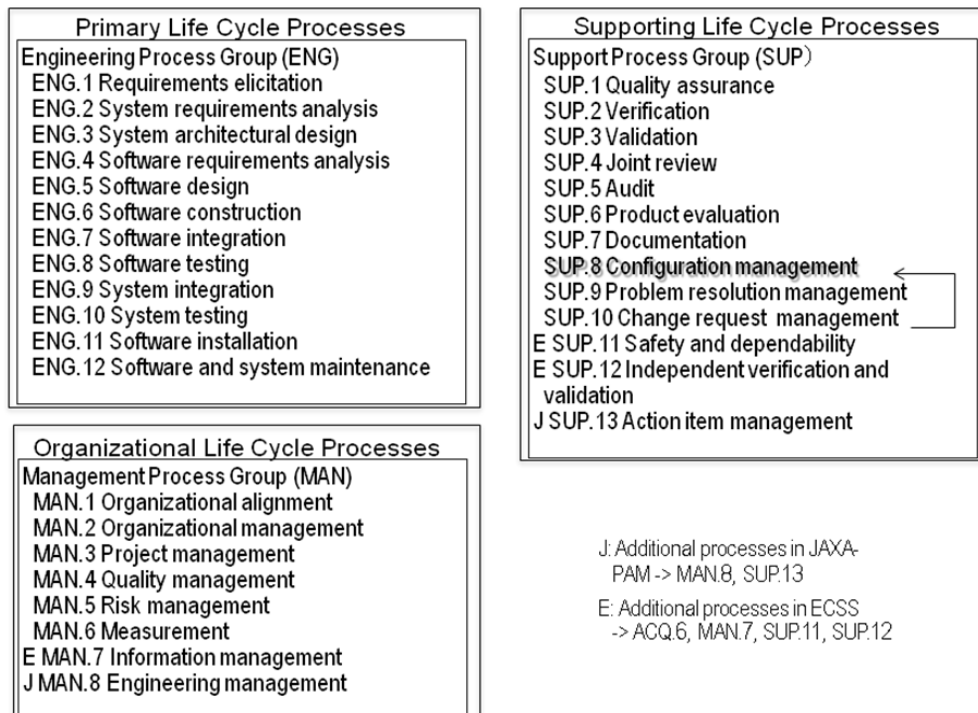
Figure 3.1 JAXA-PAM Overview

3.1.1 Issues in Assessment

One of the most important issues that make people hesitate introducing assessment is the effort needed in assessment. To introduce the assessment as an institution of JAXA, it is needed to establish light assessment procedure that makes possible to perform assessment light without losing our objective of introducing assessment. From the experience of trial assessments, we found that followings make an assessment difficult and high load.

- Miss match between improvement needs and definition of capability level

JAXA has requested suppliers to have organizational standard and develop systems following the organizational standard in procurement agreement. Because of such history, suppliers already establish own standard process and deploy the process as following the standard. Then in the trial assessment, JAXA requests to evaluate the capability of standard process definition and process deployment. In the assessment, the first version of JAXA-PAM was used, i.e., the measurement framework was the same as ISO/IEC 15505 part5, therefore, we evaluate the level up to 3. Because of the continuity requirement, assessment team starst evaluation from level 1 to 3 and we found that GPs of Level-2 and Level-3 are deeply related and it was hard to explain distinction and relation in interview. Moreover, it was not so important to distinguish the difference of Level-2 and Level-3 for someone who actually performs processes and drives improvement activities. In addition, it forces to spend lots of time to analyze, interview and evaluate capabilities on assessors.

In Level-4 capability of ISO15504 part5, i.e., to control, manage and predict process performance based on statistical metrics data, is not fit to Japanese space development. The number of products or development is insufficient to generate quantitative criteria from statistical data.

- Complexity of the capability indicators

The concept of capability definition in ISO/IEC 15504 part5 is clear, however, in the viewpoint of indicators, the Generic Practices (GPs) of Leve-2 and Level-3 of are closely related. Therefore assessors usually combine related level-2 and level-3 questions when they interview, otherwise, interview topics

would be duplicate and interview duration would also doubled. This arrangement of interview topics also forces to spend time in every assessment.

To adjust such inconveniences describe above without losing the concept of ISO/IEC 15504 part5, we start to re-organize measurement framework.

3.2 Re-organization of the Measurement Framework of JAXA-PAM

3.2.1 Customizing the Capability Dimension

As previously described, there are several inconveniences in the measurement framework of the first version of JAXA-PAM. To adjust JAXA-PAM to our space development, we started with customization of capability dimension. The concepts of the customization are 1) Easy to understand from the point of view of assessed engineers, 2) to allow non-statistical approach for quantitative process management and 3) to realize light assessment As shown in figure 3.2, we define 3 capability level, called J-Level. In the figure, capability level of ISO/IEC 15504 part5 and the New JAXA-PAM is compared.

Level	ISO/IEC15504	J-Level	JAXA-PAM
5	Optimizing	**3**	**Established for Improvement Cycle**
4	Predictable		
3	Established	**2**	**Established for Improvement Base**
2	Managed		
1	Performed	1	**Performed at Minimum Level**

Figure 3.2 Capability Level Comparisons

The PAs are also arranged in accordance with J-Level. Figure 3.3 shows schematic PA mapping between JAXA-PAM's PA (J-PA) of J-Level 2 and related ISO/IES 15504 part5 PAs. As shown in the figure, J-Level 2 consists of 2 process attributes, Process definition attribute and Process performance management attribute. In this customization, PAs that have close relationship are merged and simply structured.

ISO/IEC15504-5 PA (L2, L3)

JAXA-PAM J-PA (J-L2)

PA2.1 Performance. mng. attr.
· Objectives
· Plans, monitoring, adjustment
· Performance

PA2.2 Wk. product mng. attr.
· Work product requirements
· Work product control
· Work product review

PA3.1 Process definition attr.
· Std. process definition
· Methods to monitor effectiveness

PA3.2 Proc. deployment attr.
· Tailoring
· Resource allocation
· Monitoring effectiveness

(21 practices)

J-PA2.1 Process definition attribute
· Work product requirements
· Standard process definition
· Methods to monitor effectiveness

J-PA2.2 Defined proc. perf. mng. attribute
· Tailoring
· Resource allocation
· Monitoring effectiveness
· Plans, performance, monitoring, adjustments
· Education + training plans, performance

(11 practices)

Figure 3.3 PA Comparisons between ISO15504-5 Levels 2+3 and JAXA-PAM J-Level 2

As mentioned before, the statistical approach doesn't suit for space development. On the other hands, we already introduced process performance measurement to monitor the process and use data for self-improvement. To adjust our situation and objective of assessment, we merge Level-4 and Level-5 of ISO/IEC 15504 part5 as J-Level-3 and define process attributes to assess process improvement and optimization capability without statistical approach of prediction.

Figure 3.4 shows schematic comparison of Process Attributes (PAs) of ISO/IEC 15504 part5 and JAXA-PAM in J-Level 3.

ISO 15504-5 PA

JAXA-PAM J-PA

PA4.1 Proc. meas. attr.
· Proc. measurement objectives
· Identifying measures
· Collecting measurement results

PA4.2 Proc. control attr.
· Analyzing meas. results of processes and products
· Identify + perf. corrective acts.

PA5.1 Proc. innov. attr.
· Def''ing proc. impr. objectives
· Identifying proc. impr. opps.
· Est. impl. strategies

PA5.2 Proc. optim'n attr.
· Mng'ing impl. of agreed chgs.
· Eval. effectvns. of proc. chgs.

(21 practices)

J-PA3.1 Process improvement attribute
· Process improvement objectives
· Identifying process impr. opportunities
· Planning improvement strategies
· Quantv. goals + mechanisms for meas.

J-PA3.3 Process optimization attribute
· Applying changed processes
· Evaluating process change effectiveness

(6 practices)

Figure 3.4 PA Comparisons between ISO15504-5 Levels 4+5 and JAXA-PAM J-Level 3

3.2.2 Refining capability indicators

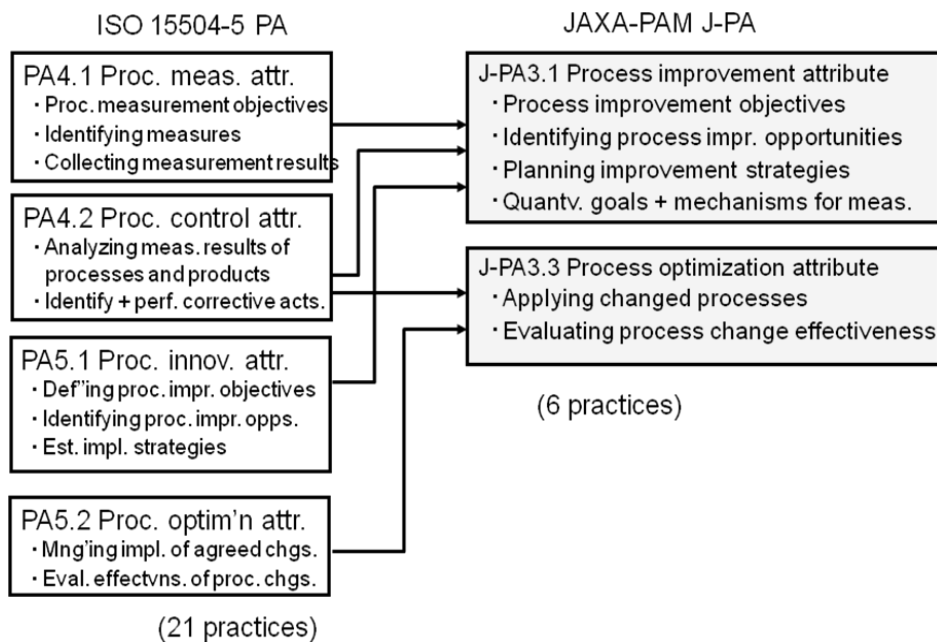In the JAXA-PAM, Base Practice (BP) and Work Product (WP) are defined as the performance indicator and GP is defined as capability indicator. As described in chapter 3.1, JAXA have already revised BPs and WP. For the GPs, re-structure is done in accordance with J-Level and J-PA.

4 Summary and Future work

JAXA is now implementing a new improvement-oriented assessment framework as an institution. Some issues have been identified through trial assessments. JAXA analyzed assessment results and has begun customization measurement framework of JAXA-PAM.

The main adjustment of capability dimension are 1) adjust capability level definition and related process attributes and 2) refine capability indicator, as described in chapter 3

Through the series of customization, JAXA-PAM becomes easy for assessors to understand and use the PAM. By using the new JAXA-PAM, the assessment becomes more efficient and useful enough to be institutionalized. The easiness of understanding and performing assessment will encourage peoples who desire to improve their development.

The effectiveness of the customization of BPs are already verified in trial assessments. For the GP part, the effectiveness will be examined in future assessments. Based on this JAXA-PAM, we are now considering to be certified internationally as the first Japanese PAM.

5 Literature

1. ISO/IEC 15504:2003. Information Technology - Process assessment. Available at: http://www.iso.ch/iso/en/. Last checked 2007-06-02

2. ISO/IEC 12207:1995. Information Technology - Software Life Cycle Processes International Organization for Standardization. Available at: http://www.iso.org/iso/en/. Last checked 2007-06-02.

3. "Process Assessment Model Development towards Improvement Oriented Assessment in Space Domain", Yuko Miyamoto, Yumi Koishi, Motoko Takeuchi, Masafumi Katahira, EuroSPI2, 2010.

6 Author CVs

Yuko MIYAMOTO

Ms. Miyamoto is an associate senior engineer in the Digital Innovation Center (JEDI) at the Japan Aerospace Exploration Agency (JAXA). She is in charge of software process improvement (SPI), including assessment and software Independent Verification and Validation (IV&V). She is an ISO/15504 certified provisional assessor and has experience with process assessments in the space domain.

Tatsuya Kaneko

Mr. Kaneko is engineer in the Digital Innovation Center (JEDI) at the Japan Aerospace Exploration Agency (JAXA). He is in charge of software process improvement and software process assessment. He is an ISO/IEC15504 certified provisional assessor. He has experience with process assessments in the space domain and enterprise system.

Masafumi KATAHIRA

Mr. Katahira is a senior engineer and Manager for Software Engineering for JEDI at JAXA. He worked on satellite operations and developing the JEM and H-IIA launch vehicle at NASDA, and complex systems at the Massachusetts Institute of Technology. He is engaged in software IV&V, software process improvement, and real-time operating systems for spacecraft. He is a member of the IEEE.

SPICE in Turkey
Reasons, Actions, Expectations of a SPICE Conference in Turkey

Kerem KEMANECİ
SPICE Assessor (Provisional)
TURKISH STANDARDS INSTITUTION
kkemaneci@tse.org.tr

Mariye Umay AKKAYA (MSc.)
SPICE Assessor (Provisional)
TURKISH STANDARDS INSTITUTION
ukuruoglu@tse.org.tr

Abstract

Having an idea of a young country, tends to develop and improve rapidly in IT sector. With rapid growth rate in the sector putting quality in minds is something very important to reach country wide strategic goals in IT sector. Before it's too late making a common sense of quality in IT and software sector in particular by arranging a SPICE Conference in Turkey is a great need.

Keywords

Software Quality in Turkey, SPICE Conference in Turkey, ISO 15504, SPICE, Subventions and Regulations for IT Sector in Turkey, Turkish People, Interest of IT, Medium-Term And Long Term Goals Of Turkey, Improvement of IT Sector in Turkey.

1 Introduction

In recent years, in rapidly growing Turkish economy, IT Sector is becoming increasingly important. Communication, construction, automotive, transportation, heavy industry, and textile industries are such large sectors and have been the locomotive of Turkey and contributing for the rapid growth of Turkish economy recent years. In those widespread sectors, quality in every meaning and customer satisfaction is understood very well during last 20 years. Because of young educated population, steady and reliable circumstances via strong politics, rising purchase power, foreign and domestic investment etc. Turkey is a great market alone.

Turkish People

As of December 31, 2011 Turkey's population is 74,724,269 people. Turkey is a young populated country with median age is about 28.5 now and around %74 of total population age is below 45. In Turkey percentage of using computer is %46,4 as of July,2011. Below 45 years old people percentage is more than %60 and percentage is trending upward rapidly as you see the table below.

Son üç ay içinde bireylerin yaş grubuna göre bilgisayar ve İnternet kullanım oranları (%)
Individuals using the computer and Internet in the last 3 months by age groups (%)

| | Yıl Year | Yaş grubu - Age group | | | | | | | | | | | | | | | | | |
| | | 16 - 24 | | | 25 - 34 | | | 35 - 44 | | | 45 - 54 | | | 55 - 64 | | | 65 - 74 | | |
		Toplam Total	Erkek Male	Kadın Female	Toplam Total	Erkek Male	Kadın Female	Toplam Total	Erkek Male	Kadın Female	Toplam Total	Erkek Male	Kadın Female	Toplam Total	Erkek Male	Kadın Female	Toplam Total	Erkek Male	Kadın Female
Bilgisayar / Computer	2004	-	44,4	21,1	-	26,4	13,1	-	19,2	7,1	-	12,9	2,8	-	4,0	0,7	-	0,8	0,1
	2005	-	43,8	25,0	-	27,6	13,9	-	19,2	7,1	-	14,2	3,3	-	5,0	1,2	-	2,2	0,2
	2007	54,6	67,3	40,7	35,1	44,7	25,5	26,8	36,6	17,1	17,1	26,8	7,5	6,0	9,6	2,0	1,5	2,2	0,6
	2008	57,9	69,6	47,0	43,3	54,8	31,8	31,6	41,6	21,5	20,4	28,3	12,5	7,4	12,6	2,5	1,8	3,0	0,9
	2009	62,2	76,4	49,1	46,6	58,6	34,5	31,8	42,1	21,3	20,2	28,9	11,6	6,7	10,6	3,1	2,2	3,2	1,4
	2010	65,2	78,5	52,7	52,0	62,4	41,6	36,9	46,9	26,9	23,2	33,6	12,7	8,3	13,5	3,4	2,7	4,1	1,6
	2011	67,7	77,9	58,3	57,1	67,5	46,7	41,7	52,6	30,6	24,1	34,3	13,9	11,2	17,2	5,4	3,0	5,0	1,4
İnternet / Internet	2004	-	38,3	15,9	-	21,5	9,9	-	13,9	4,9	-	9,3	1,7	-	2,7	0,6	-	0,9	0,1
	2005	-	37,4	18,8	-	22,5	10,6	-	14,3	5,0	-	10,1	2,4	-	3,8	0,9	-	1,8	0,1
	2007	50,4	63,5	36,2	32,3	41,5	23,1	23,8	33,2	14,4	14,8	23,2	6,5	4,8	7,5	1,8	1,4	2,1	0,6
	2008	54,8	67,1	43,4	41,4	52,3	30,4	29,3	37,7	20,8	19,4	27,1	11,6	6,9	11,9	2,2	1,6	2,5	0,9
	2009	59,4	74,1	46,0	45,1	57,2	32,9	30,2	40,3	19,9	18,6	26,7	10,5	6,2	9,5	3,1	2,0	3,1	1,2
	2010	62,9	76,6	49,9	50,6	60,9	40,2	34,7	43,5	25,7	22,4	31,9	12,9	7,8	12,6	3,2	2,7	4,2	1,6
	2011	65,8	76,5	55,9	55,1	65,4	44,9	39,7	50,4	28,9	22,7	32,1	13,2	10,4	16,0	5,0	2,7	4,5	1,2

Hanehalkı Bilişim Teknolojileri Kullanım Araştırması Sonuçları, 2004-2011 - Results of the ICT Usage in Households and by individuals, 2004-2011
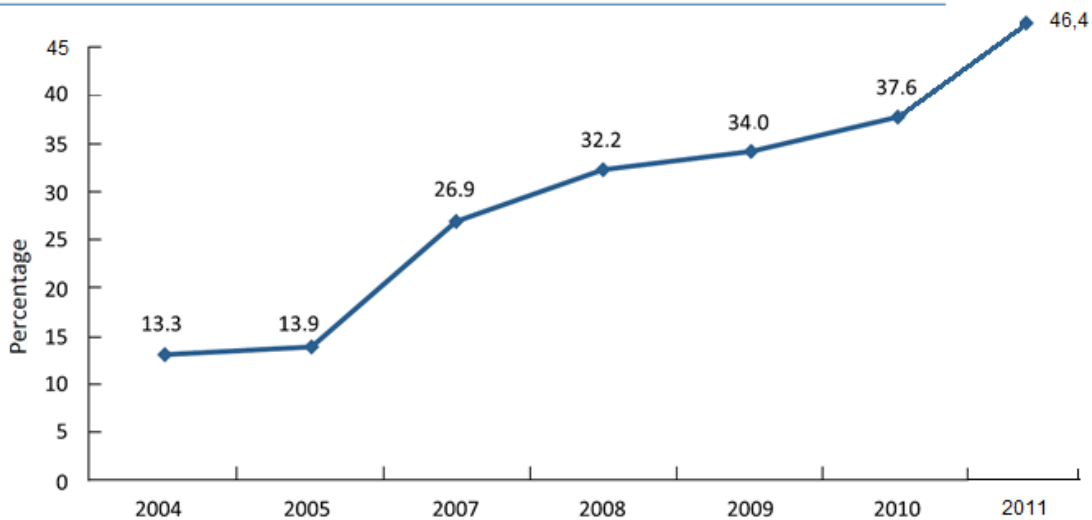

Figure 1: Computer Usage within the last 3 months

IT Sector is Growing

Because of being a great market new investment projects have great possibility to survive by using quality tools. By the web technologies and decent competition, customers have options and have their own purchasing criteria.

New circumstances makes it obligation to reach customers rapidly and to be reachable easily and safely, to product fast and consistent service and products more automation is used in production like factories, services like hospitals. In government area large government software projects are being developed. Now many government services are being emerged under www.turkiye.gov.tr . Goal of this project is to make possible for all citizens to reach government services on one web site safely and easily. Last few years government institutions are collaborating by using some common services or sharing their databases for integrity and consistency.

In those conditions developing many different kind of software projects for every sector and every case is becoming so vital. Recent days many software projects in government are in use and many projects are ongoing and still under development and many projects are planned for future. Some of old fashioned projects are needed to be replaced with new open and flexible projects.

## Government Investments in IT Technologies

As you can see from the graphic, total amount of Government IT investments is raising every year. This year total amount for IT project investments will reach 2,484 million TRY (around 1080 million EUR, Apr 2012).
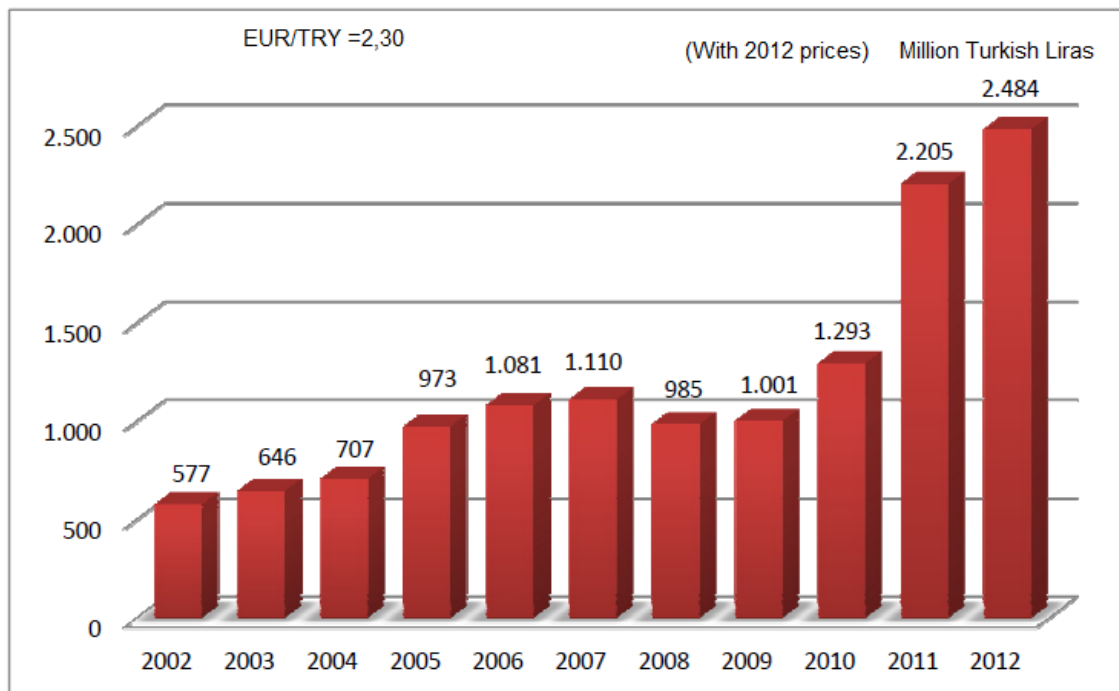
Amount Of Investments per year 2002-2012

At second graphic you can see top ten projects according to amount of investments this year.

Million Turkish Liras
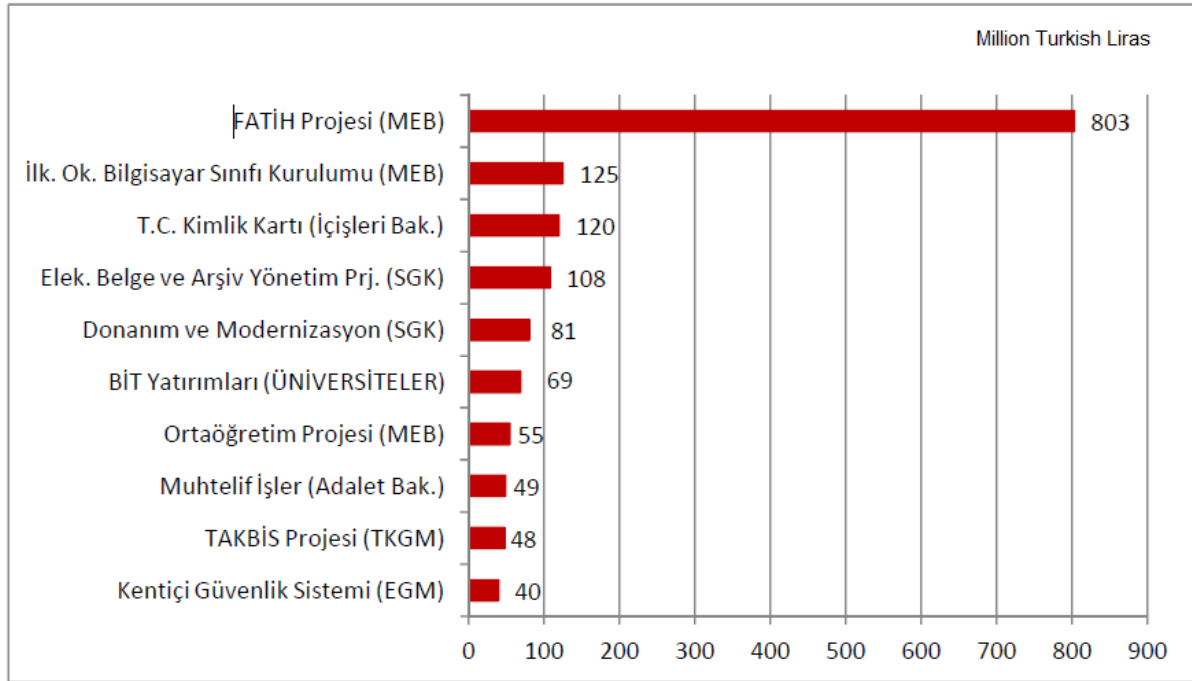
**Amount of per Government IT Project in 2012, Top 10 Projects**

Technology Parks in Development

Technology parks are being established generally in University campus areas or independent areas. Idea of technopark concept began in 90s. In time, number of technoparks raised and size of technoparks enlarged. With leadership of universities, Ministry of Science, Industry and Technology made some regulations. In this scope, Law of Technology Development Zones (Turkish Law No: 4691) was published in 2001. So technoparks found a legal base. After 2001, 39 technoparks were established in Turkey.

By the Law 4691, the companies active in technoparks are excluded from income taxes and corporation taxes for the software and research and development activities. %50 of employees insurance payment is excluded. For the software developed by technopark companies V.A.T. is excluded.

Technopark statistics in Turkey

- Number Of Companies **1451**
- Number of Employees **12743**
- Total Export **540 million USD**

TOTAL TECHNOLOGY DEVELOPMENT ZONES

The number of technoparks in Turkey, established between 2001-2010
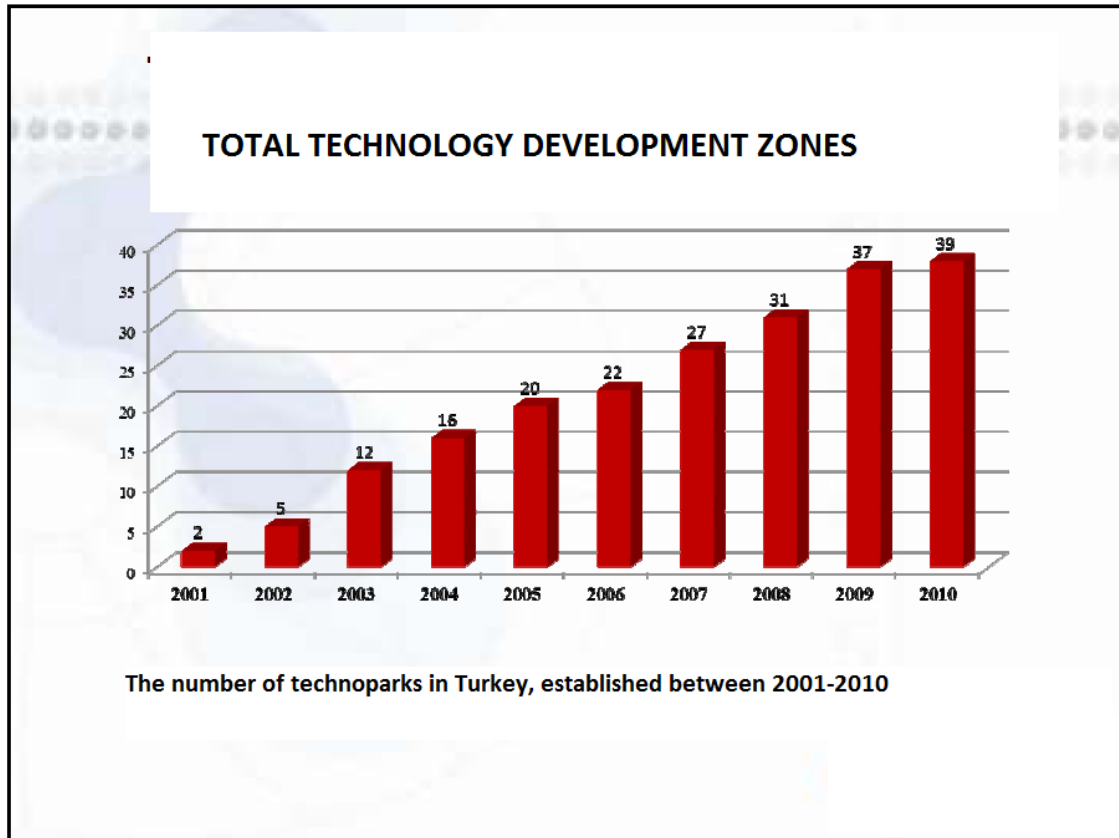
Quality is Important

In the long run some of IT companies in Turkey went a few steps more than the rest those already realized importance of quality, productivity, flexibility, transparency, customer satisfaction, innovation, proactive protection against errors which could cause to decrease customer faith. For those companies rapid growth was inevitable.

Unlike widespread sectors such as communication, construction, automotive, transportation, heavy industry, and textile industries etc. in IT sector, meaning of quality's positive effects are being understood in wide was something not very common, by manufacturer side and government side who determines regulations.

Recently in Turkey two companies have SPICE certificate those are Level-1 and Level-2 and nearly 10 companies CMMI certificate. Those companies who qualified for SPICE certification telling that they are really satisfied and even saw better reactions than expected. INNOVA is one of two companies that have SPICE Level-2. You can read their story of SPICE certification, which they prepared for this conference.

Regulations Become Opportunities

Government of Turkey also realized the growth of software sector in government and private sectors. So government made obligatory to comply with number of standards for bidder companies.

For government side, in "Ministry of Development, Strategy of Information Society - Action Plan" job sharing for IT projects and gave responsibilities for special branches in IT sector country-wide development. Those branches are in main titles:

- Social Transformation
- Effects of IT on Business World
- Modernization in Public Administration
- Citizen Oriented Service Transformation
- Worldwide-Competitive IT Sector
- Competitive, Widespread and Cheap Communication Infrastructure and Services
- Research and Development and Innovation

Under those 7 branches there are 111 sub branches and few actions for each sub branches those shared by different government institutions.

In the "Preparing Government Information and Communication Technologies Projects Guideline, 2010 (DEVLET PLANLAMA TEŞKİLATI MÜSTEŞARLIĞI Bilgi Toplumu Dairesi Başkanlığı, Kamu Bilgi ve İletişim Teknolojileri Projeleri Hazırlama Kılavuzu, 2010)" which belongs to "Ministry of Development, Information society Department" says:

"In this context, for 2011-2013 period, it's obligatory to comply with CMM-I Level-2 certification of quality (SPICE level-2, AQAP 160) for new IT projects which will be included in the investment program or which has not been yet tender that includes application software development component, estimated project amount of ₺5,000,000 and above or equivalent. For those, project amount below ₺5,000,000 it's required to apply TS ISO 12207 Software Life Cycle Standard."

Subventions

Beside those regulations there are also subventions for new investments, research and development projects. KOSGEB (Small and Medium Enterprises Development Organization of Turkey) has some options for new investors and investors who have new ideas, R&D projects. Total amount of subventions could be up to ₺750 000.

There are also Development Agencies in Turkey, and those agencies are working for different parts of Turkey. The goals of those agencies are supporting regional development of Turkey by supporting investments, new projects, consulting companies, making the region attractive for investments, supporting to create manpower for investments, improving the coordination between government, civil society organizations and private sector.
Percentage of subventions for industrial companies up to %40 of total cost, for government and civil society organization subvention is up to %75 of of total cost of the projects. (See some samples here: www.cka.org.tr/flipbook/CKA_MALIDESTEK/index.html)

TÜBİTAK (The Scientific and Technological Research Concil of Turkey) is another government Institution and a great chance for companies those want to produce R&D projects. TÜBİTAK is the greatest Technopark of Turkey and leading Institution for other R&D Institutions and Companies, also biggest collaborative. TÜBİTAK also has a department responsible of subventions for R&D investors. Subventions sizes ₺30,000– ₺2,500,000 TL if it's a government project then the upper limit is ₺15,000,000

Amounts of R&D Subventions given by TÜBİTAK to Universities

Years	Active Projects	Subvention amount of active projects *	Average amount of projects **	Annual Cost	Annual cost for per project
2000	843	15,1	17,9	6,2	7,3
2001	1.001	20,1	20,1	8,5	8,5
2002	1.242	30,9	24,9	12,2	9,8
2003	1.227	36,2	29,5	11,5	9,3
2004	1.353	44,2	32,7	15,7	11,6
2005	2.353	197,1	83,8	86,3	36,7
2006	3.091	356,3	115,3	170,1	55,0
2007	3.363	528,2	157,1	159,5	47,4
2008	3.165	506,7	160,1	164,1	51,8
2009	2.708	438,9	162,1	143,6	53,0
2010	2.533	417,9	165,0	147,0	58,0
2011	2.604	436,4	167,6	168,0	64,5

*Million TL   **1000 TL

Turkish Standard Institution (TSE)

TSE was established in 1954 within the Turkish Union of Chambers of Industry and Commerce and of Commodity Exchanges. TSE formally became an independent agency in 1960 under the Law 132.

TSE is a full member of International Organization for Standardization (ISO), International Electrotechnical Commission (IEC), and affiliate member of European Committee for Standardization (CEN) and European Committee for Electrotechnical Standardization (CENELEC). TSE is also a member of World Packaging Organization (WPO) and is a signatory to the Code of Good Practice contained in the World Trade Organization (WTO) Agreement on Technical Barriers to Trade and acts as the National Enquiry Point in connection with standards under the said Agreement. TSE also actively follows the quality efforts in Europe. As a member of the European Organization for Quality (EOQ) since 1976, TSE now holds the position of EOQ Vice-President.

Certification of IT Products within TSE

We have total 12 provisional assessors working for TSE in Product Certification Department, Personnel and System Certification and IT Department. Under Product Certification Department TSE has IT Certification Unit, certificating according to standards below:

- SPICE (TS ISO/IEC 15504)

- Systems and software engineering -- Software life cycle processes (ISO/IEC 12207)

- Common Criteria: Information technology -- Security techniques -- Evaluation criteria for IT security (TS ISO/IEC 15408)

- Document Management Standard (TS 13298)

- Software engineering -- Software product Quality Requirements and Evaluation (SQuaRE) -- Requirements for quality of Commercial Off-The-Shelf (COTS) software product and instructions for testing (TS ISO IEC 25051)

- Ergonomics of human-system interaction -- Part 151: Guidance on World Wide Web user interfaces (ISO 9241-151)

- Information technology -- Security techniques -- Security requirements for cryptographic modules (ISO/IEC 19790)

- Information technology - Security techniques - Test requirements for cryptographic modules (TS ISO/IEC 24759)

- Systems engineering - System life cycle processes (TS ISO/IEC 15288)

SPICE Conference in Turkey

For now we have only 2 companies already had SPICE certification, but it's just a start. Those companies INNOVA and NEL are good samples for candidate companies. INNOVA's SPICE Experience Report will be a good guide for companies which have tendency to get SPICE certificate.

Our motivation is to make SPICE well known in Turkey so we have great opportunities for this goal. Regulations and subventions are spectacular weapons to use. Our goal is getting along very well with medium term and long term country strategy. Country strategy is to make information technology usage very common for government, private sector and public in every level of society. Also being a country manufacturing high-tech products with own software.

There is great need for quality standards in software sector in Turkey for short, medium and long term. SPICE is a great answer for this need. From this point of view, almost everything seems positive for future to reach the main goal.

A SPICE Conference in Turkey will cause a great reaction from Turkish Government and Turkish software companies and SPICE standards will be understood very well. More subvention and regulation for SPICE standards can be granted particularly after this conference and companies can be attracted to have SPICE certification. After this conference number of SPICE certificated companies will raise and by certificated companies a software quality improvement mentality will place clearly in country wide. Finally, this mentality will help Turkey to reach one of very important goal defined in country strategy in medium-term and long-term.

2 References

1. www.tse.org.tr
2. www.tuik.gov.tr
3. http://www.dpt.gov.tr/Kalkinma.portal
4. http://www.cka.org.tr/flipbook/CKA_MALIDESTEK/index.html
5. www.kosgeb.gov.tr
6. http://www.tubitak.gov.tr/tubitak_content_files//ARDEB/destek_prog/genel_bilgi/ARDEB_sunum.pdf

3 Author CVs

MARİYE UMAY AKKAYA, Msc

TSE-Turkish Standards Institution
Responsible of Information Technology Certification System
Has graduated from Bilkent University with Computer Engineer and Information Science degree with MSc. Started to work at Turkish Standards Institution (TSE) From 2003. Still acting as the Technical and Quality Responsible of Turkish IT Certification System and also responsible of many IT Standards` Certifications such as TS ISO/IEC 27001(ISMS), SPICE-TS ISO/IEC 15504, TS ISO/IEC 12207, TS ISO/IEC 15288, TS ISO/IEC 20000, TS ISO/IEC 25051, TS ISO/IEC 9241-151, TS ISO/IEC 19790, TS ISO/IEC 24759-Crypto Modul Certifications etc, as Inspection-Certification Expert and Provisional Assessor of Automotive SPICE. Is still member of CCRA, CCMB, CCDB, CCES and CCMC on behalf of TSE and Turkey.

KEREM KEMANECİ, Computer Engineer

TSE-Turkish Standards Institution
Assessor at Information Technology Certification System
Graduated from Computer Engineering Program at Selçuk University in 2008. Started working at IT Department of Culture and Tourism Ministry of Turkey in Jan 2009. Then transferred to IT Department of State Supply Office. Now working for TSE since October 2010 and still going on Production Management Master Program at Gazi University in Ankara.

Conceptual Models for Innovation Process and Process Innovation

Koichi Kishida

Software Research Associates

k2@sra.co.jp

1 Looking back for future

How can we talk about the issues of innovation? There are some questions in front of us:

- How can we define and understand innovation in software development?
- How does process address the innovation of software products?
- How can software process itself be innovated?

About a century ago, philosopher Otto Neurath made his famous statement in his short essay "Anti-Spengler"(1921):

- We are like sailors who on the open sea must reconstruct their ship but are never able to start afresh from the bottom.

This statement was to describe the status of philosophical knowledge in early 20the century. But the image of this "Neurath's Boat" almost completely fits to current status of software engineering. We are now sailing legacy software boat on stormy ocean of the networked application environment, and are never able start a fresh from the bottom, namely go back to the port for the construction of a new boat.

But, do not panic!

Human beings have created countless variety of conceptual worlds using symbols (words, images, etc). These worlds are never created from the vacuum, They are made as new versions of already existing worlds by adding some new things and/or deleting unnecessary things, Software engineers have been creating their own versions of worlds. According to Michael Jackson, software is an imaginary machine, which represents conceptual model of the targeting world. We are now living in a multiple co-existing worlds as Nelson Goodman described in his controversial book "Ways of Worldmaking".

Innovation belongs to the future. Time flows. How fast is the speed? It is one hour per hour. Which direction? It is against us, from future to present to past/ We can not see future because it is still invisible for us. We need to look back in to the past to find some new innovative idea for our future,

A young Japanese philosopher Nakamoto Tominaga in early 18th century found an important principle of human thoughts as a result of his exhaustive study of large volumes of philosophical documents in Confucianism and Buddhism. He named it as "Add-on Principle". He noticed that any innovative conceptual idea refers some ancient origin as a support of innovative strength comparing to others [4].

Nakamoto's work was inspired by his fore-runner Sorai Ogiu, a philosophical giant of the age. Sorai was the leading person of a kind of Linguistic-Turn movement in Japanese philosophy.[5] He made a famous statement about the nature of abstract conceptual terms. "Tao" (the Way) is the central abstract concept in Confucianism. Sorai stated his thoughts on Tao as follows:

Every Confucianism philosopher talks about his own version of Tao. Those represent only some aspects of Tao. True Tao in its full meaning can only be found in the words of sacred kings in ancient China. They knew the difficulty of teaching the meaning of Tao to their people. So, they invented "Ceremony and Music" as educational tools for management. Confucius himself once told that starting point of political reform of a country is to rectify the terms. If terms are not rectified, then the language will not be flourish. If language is not flourish, then projects will not be successful. ….

2 Process Models

Almost all software process models proposed so far were created based upon the concept of "Repetition". The typical example is CMM. The second level of maturity in CMM is called as "Repeatable": It says that the process is at least documented sufficiently such that repeating the same steps may be attempted.

It is true that software development process looks like to take repetitive execution of similar steps again and again. It seems better to control these repetitive steps using well-made documentation. It seems to be a naïve inductive attitude to think future will be same as past. Then, where the innovation will come into the scene?

Here is a famous paradox about the color of emerald proposed by Nelson Goodman:

The word *grue* is defined relative to time t as follows: "An emerald is grue", if it is green and was examined before time t, or blue and was not examined before t.

The word *bleen* has a complementary definition: "An emerald is bleen", if it is blue and was examined before time t, or green and was not examined before t.

All emeralds examined thus far are green. This leads us to conclude (by induction) that also in the future emeralds will be green, and every next green emerald discovered strengthens this belief.

Goodman observed that (assuming t has yet to pass) it is equally true that every emerald that has been observed is [grue](). Why, then, do we not conclude that emeralds first observed after t will also be grue, and why is the next grue emerald that comes along not considered further evidence in support of that conclusion?

The problem is to explain why induction can be used to confirm that things are "green" but not to confirm that things are "grue".

How about our "Process Emerald? Is it green, or grue?

French philosopher Gilles Deleuze claimed as follows in his famous book "Difference and Repetition";

Identity not be first, it exists as a principle but as a second principle, which revolves around the Different. Such kind of thought should be the nature of a Copernican revolution, which opens up the possibility of Difference having its own concept, rather than being maintained under the domination of a concept in General already understand as Identical.

Deleuze pointed out that repetitive occurrence of identical things will never occur only the different things will be repeated. The concept of difference has not been accepted on its own for a long time. It was only accepted after being understood with reference to self-identical matters, which make difference as difference between. When we think about innovation, it is necessary to reverse this situation to understand "difference in itself", because it is the mother of innovation.

3 Case of Structured Programming

I myself have experienced this "repetition-difference" problem during my work in structured programming practice in late 1960s. I have used a simple common repetitive model of program execution process, proposed by Egdar Dijkstra, such as:

Initialize:

Repeat while data exists:

Input

Manipulation

Output

Finalize

This simple repetitive model was successfully applied to almost all programs for typical data processing application systems by providing appropriate different internal structures for Input, Manipulation, Output modules and communication mechanism between them. This experience revealed that it is important to recognize common repetitive structure of program execution process model among different

application contents and also to discover different design structure for common modules depending particular characteristics of application.

4 Two types of Innovation

Let's think about process steps described in traditional repetitive models like CMM. Same key process steps will be repeated again and again. The names of process steps are same, but actual contents of activities performed in those steps are different in each repetition.

If some change happened in the developing environment, practitioners should start looking for some new innovative idea to modify their activities to adapt the situation. We should be careful not to stick around same identical process execution style. Always we should think about the difference, which is the essential nature of social environment around us.

Innovation for making software process adaptable to environmental change is considered as a weak kind of small-scale innovation. We only need to incorporate some "watch-dog" function into our process model to monitor changes which occur during actual execution of each process steps carefully. Some steps are likely to change, and others are not, depending to the characteristics of each process steps and environment.

Sometime, more strong kind of innovation will happen. It is the real innovation, which changes structure of process model completely. There are various reasons for those happenings. Maybe some unexpected event, that lead us to big structural modification of process model, or an emergence of new tool, which introduce revolutionary change in developing process as a whole.

I remember the opening session of the 2nd International Process Workshop held in LA in 1986, Program committee people of the workshop showed following hierarchical diagram at first:

<div align="center">

Tools

Environment

Methodology

Process Model

Real World Process

</div>

They wanted to categorize workshop discussions into the levels of this hierarchy. At once, Bob Balzer (USC/ISI), assigned as program co-chair with me at the next year's ICSE, made a strong objection to PC. He said: "I'm a tool developer, considering to develop a new tool, which will radically change the world, Existing process models, development methodology, and also support environment will be completely renewed after my new tool. So, this hierarchy is just a nonsense. You people can not forbid innovation in future"

5 Models for conceptual overlay operation

November 1921, Paul Klee gave a short opening speech for his lecture series at Bauhaus. He told to students as follows:

> We are artists. Our approach in analysis is different from those of natural science. Scientist breaks down the target object into components and analyze their characteristics. Our motive of analysis is different. Looking into the masterpiece of other painter, we will analyze the process of creation for the purpose of start walking by our own foot

To prepare for the radical innovation in software development process, we need to have new conceptual meta-model. Our daily practice of process execution usually follows rather scientific component-breakdown paradigms: Process into process steps, and key practices, It seems necessary to have a holistic conceptual framework to lay over such a daily practice. Using these holistic overlay model, we will be able to get the real image of process as a whole and can find out implicate (enfolded) order among process components as suggested by David Bohm.

Following are the three candidates for such conceptual models for overlay:

(1) Flower Model: This model was proposed at a domestic workshop held in Japan. It has an imaginary central archive holding all knowledge and data related to the project on the hand. ,All process activities are arranged around this central archive. Execution sequence of each activity is completely random. They pick up necessary information from the archive, do some processing, and put back result to the archive. This model may represent a context-free meta-model of software process.

(2) Tourist Model: This model was proposed at an ISPW meeting. At first, it has an original touring plan. But It always looking for some interesting things or happenings along with the tour route. Actual process execution is likely to go away from original route. It does not assure the goal of original tour plan. But maybe it will bring more fruitful result back.

(3) Rhizome Model: This model is based upon the metaphor of "Rhizome", which was proposed by Felix Guatari and Gilles Deleuze in the book "Thousand Plateau" It is a model of human thoughts replacing traditional metaphor of "Tree". Rhizome has following principles:

- Principles of connection and heterogeneity: any point of a rhizome can be connected to anything other, and must be

- Principle of multiplicity: only when the multiple is effectively treated as a substantive, "multiplicity" that it ceases to have any relation to the One

- Principle of a signifying rupture: a rhizome may be broken, but it will start up again on one of its old lines, or on new lines

- Principle of cartography and decalcomania: a rhizome is not amenable to any structural or generative model; it is a "map and not a tracing"

These principles are very holistic nature and fit into our needs for conceptual framework.

6 Final Remark

This is my position statement for the discussion of software process innovation issues. I believe that we need to have sound philosophical foundation to establish right direction to innovate our process practice. Otherwise, our sailing boat will be lost in the stormy ocean of the Internet age.

7 Reference

1. Otto Neurath, "Anpi-Spengler", 1921.

2. Michael Jackson, "Software Requirements &Specifications", 1995.

3. Nelson Goodman, "Ways of Worldmaking", 1978.

4. Testu Najita, "Kaitokudo", 1998.

5. Tetsuo Najita (ed), "Tokugawa Political Writings", 1998.

6. Nelson Goodman, "Facts, Fiction, and Forecast", 1983.

7. Gilles De;euze, "Difference and Repetition, 1994.

8. Edgar Dijkstra,, "Structured Programmning", 1972.

9. Paul Klee, "Das Bildnerishe Denken", 1956.

10. David Bohm, ".Wholeness and Implicate Order, 1980.

11. Felix Guatari & Gilles Deleuza, "Thousand Plateau">1987

Ten Good Reasons for Terminology Management to improve your Business Processes

Gabriele Sauberer and Diana Brändle

International Network for Terminology, Vienna, Austria
gsauberer@termnet.org

Abstract

The paper presents "10 good reasons for terminology management" in the fields of operational, supporting and management processes, focussing on the role and value of terminology management for business process improvement and innovation.

Many companies and organisations still think that managing terminology is the sole responsibility of language service providers and that it is only import for localisation or translation. This is not the case at all, as terminology management plays a vital role in managing knowledge and improving business processes. It applies to the product development cycle and innovation management, no matter, how many languages are involved. We will look at the benefits of maintaining corporate terminologies and making them accessible to anybody involved in internal or external communication. Effective terminology management, in addition to being a quality assurance procedure, is primarily a preventative measure that ought to be taken at the authoring stage of a product's development cycle. The true benefit of terminology management is rooted in its ability to anticipate and check the proliferation of consistency issues at the source level. Beyond that, terminology management can also serve as an equally viable means of correcting terminological disagreement at the translation stage of a product lifecycle.

Proactively managing terms at the source has a profound effect on the overall quality of products and services. It is a source-level precaution that will save a significant amount of time and money – not only when it comes to localisation, but long before [5].

Keywords

Terminology, Terminology Management, Knowledge and Communication Management, International Network for Terminology, business processes, improvement

1   Introduction to Terminology and Terminology Management

Each branch and business has it's specific vocabulary and technical terms. Some of them may be standardised, many of them may be clearly defined in technical dictionaries or knowledge bases of individual business branches. But a lot of abbreviations, acronyms, product or service names and other terms representing corporate knowledge, functions, tools, services, processes, etc., are so specific and individual that they can only be defined and managed by the company or organisation concerned. This so-called enterprise terminology represents corporate or organisational knowledge in the form of concepts and their terms.
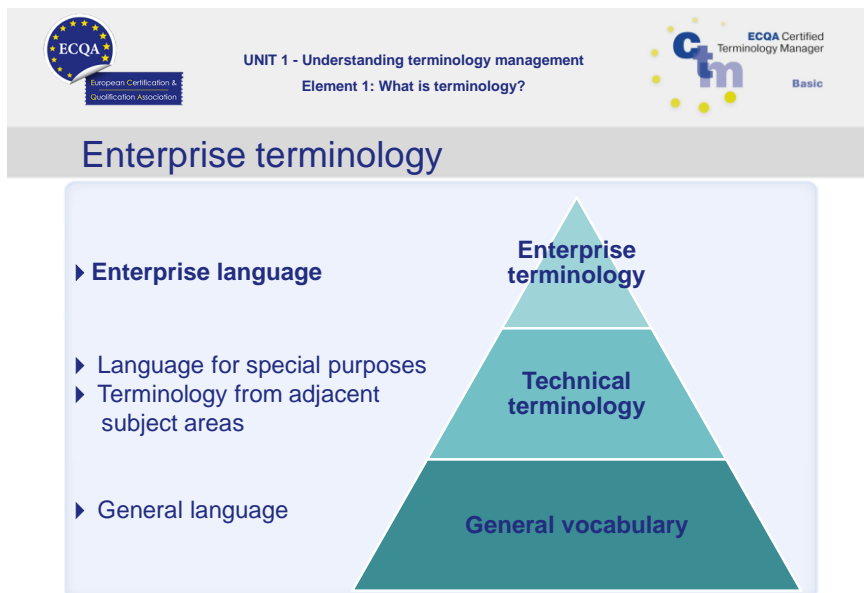

Fig. 1: Enterprise Terminology [4]

1.2   Definitions

Terminology

In a standard published by the International Organization for Standardization (ISO), terminology is defined as "A set of designations belonging to one special language." (1st meaning) and as Science studying the structure, formation, development, usage and management of terminologies in various subject fields (2nd meaning), in: ISO 1087-1:2000 Terminology work – Vocabulary – Part 1: Theory and application.

In knowledge generation, communication and management, we, the experts, use and create our own terminology, i.e. our own specialised language. It is a language for the special purpose of scientific or business communication. Thus, we all do terminology work, but mostly not paying much attention to that work, i.e. of the "structure, formation, development, usage and management" of the terms we use or create in our particular subject field. Probably we are not even aware of the existence of such a science called terminology.

Business Process Improvement

Business Process Improvement is defined as "improving quality, productivity, and response time of a business process, by removing non-value adding activities and costs through incremental enhancements." [6]

2 Terminology Management in operational Processes

When a new product or service is being developed or manufactured it has to be named and described, its components have to be listed and classified.

In most cases the first name (term) is only a project name that later has to be changed according to the requirements of customs, the law department, marketing & sales, etc. If the new product, function or service is part of a classification system or Enterprise Resource Planning system, it may also be coded, so that a term may go through different stages, which all should be recorded.

Example from automotive industry: 7 synonyms used either chronologically or in parallel for one concept, in one language and within one company.

1. Development creates new product called and uses the project name   hill hold
2. Manufacturing calls it   Hill Hold Control (HHC)
3. Purchasing asks suppliers for offers for   starting-off aid (Hill Hold Control)
4. In the Enterprise Resource Planning system this function is stored as   PR3245M43
5. In the spare parts warehouse there is an item called   xyz hold assist
6. After-sales uses in its Customer Relationship database   hill hold assist function
7. Marketing & Sales decides to call it   hill-start assist

It is then the task of terminology management:

- to assign all synonyms with information on their usage (i.e. the code name in the ERP system may still be "PR3245M43", but it must be clear that this code is used for the concept called "hill-start assist")

and/or

- to decide that only one term, "hill-start assist", may be used and all synonyms are deprecated and must not be used.

3 Terminology management for Supporting processes

Imagine you have bought a new tool like this:
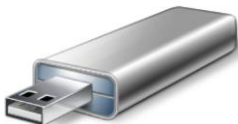
Unfortunately, something went wrong and you need to download the driver software for it.

In the printed documentation that was shipped with this product there is some information on downloading driver software for your new "memory key".

As you do not really understand the instructions, you search for information on "memory keys" in the Internet.

You are forwarded to a website offering "flash drives", which does not offer downloading the required driver software for your tool.

You search in the online help for "flash drive", but can only find

"USB flash drive"

with a lot of information, but no information on download options for driver software.

You call the support and ask them for help concerning your

"USB flash drive",

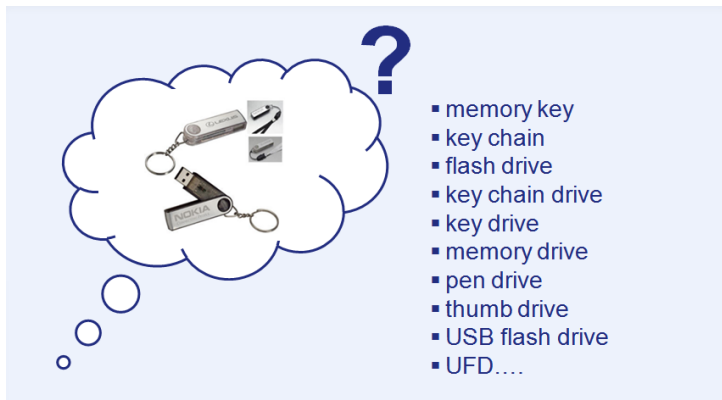and you receive the answer to simply look for information on the "pen drive".

Fig. 2: Synonymous terms for one concept

Or imagine, you have to deal with different languages: Having a German software version installed, what are you going to do if a German error message appears, but the online support database contains only information in English?

This example shows once more, what role terminology management plays in improving business processes.
How many misunderstandings, customer complaints, time and money can be saved through terminology management!

Possible solutions to solve this problem might be:

Company A produces this tool:

Fig. 3: Memory key, http://www.portable-tools.de/tl_files/usb/tmp/usbpendrive_mount.png

The only allowed term throughout company A in English is   "USB flash drive"

and has to be consistently used

- in all types of documentation (printed instruction manual, online documentation, labels, online-help, etc.)
- by all departments (custom, marketing, purchasing, after-sales, customer service, hotline, etc.)

or possibly necessary synonyms must be recorded and assigned with information on their usage and made accessible or at least referenced so that someone looking for , e.g. "thumb drive" is forwarded to "USB flash drive".

Company B may call this tool   "key drive" and
company C   "pen drive".

4   *Terminology Management for Management Processes*

In order to improve quality, productivity, and response time of a business process, non-value adding activities and cost have to be removed. This goal can be achieved through incremental enhancements. [10]

In terms of terminology management, this means:

- to prevent inconsistencies and possible misunderstandings and costs resulting from them
- to implement terminology workflows and define terminology processes
- to set up a central terminological database with controlled access rights
- to define necessary meta data
- to involve all employees, departments, external service providers and subsidiaries that have a stake in terminology processes
- to leverage synergies and bundle forces (using interfaces with ERP-systems, knowledge bases, glossaries, etc.)
- to take existing standards (terminology standards like e.g. Termbase Exchange Format (TBX) or standardized terminology) into account in order to foster technical and semantic interoperability between different systems used in different organisations, projects and environments

The International Organization for Standardization (ISO) has created a number of standards that outline terminology management as best practice in localisation, e.g.:

ISO 704:2009
Terminology Work – Principles and Methods
This 38-page document is a good introduction to terminology management, including detailed guidelines for writing definitions.

ISO 1087-1:2000
Terminology Work – Vocabulary
This overview describes the major concepts of and behind terminology management.

ISO 12616:2002
Translation – Oriented Terminography
This document provides a wealth of information on managing terminology, with a focus on integration with translation environments.

ISO 12620:2009
Computer Applications in Terminology – Data Categories

This standard specifies the data categories that should be employed to ensure easy data exchange between systems that store and process terminology.

In addition to these standards for terminology management, ISO publishes literally hundreds of standards that contain specifications for monolingual and multilingual glossaries.

5 Terminology Management and Business Process Improvement

There is a direct link between terminology management and business process improvement: only if information on existing products and services is accessible and well maintained, you can use it in order:
- to develop new products and services based on existing experience and knowledge;
- to prevent deprecated technology, services, tools and terms from being re-used;
- to create a basis for continuous improvement and sustainable innovation.

Terminology management is supporting the business process improvement at all stages and answers basic and core questions, e.g.:

- Who is responsible for and involved in managing the concepts and terms of a process?
- How and by whom terms and concepts are created?
- How and where the terms and concepts are stored?
- Which metadata are they assigned to (e.g. subject field, source, who entered term and when, etc.)?
- Which language/s and type/s of texts are involved?

- How a new process should be defined – which enterprise terminology should be used?
- How are new terms to be created?
- Who decides on new terms?
- Who enters terminology, where and how?
- Who decides on usage status of terms (i.e. deprecated, preferred, etc.)
- Who is responsible for which language, which type of text, which format, etc.?
- Who is involved (employees, external service providers, suppliers, customers, subsidiaries, etc.)?
- How many types of documents, publications, communication channels are concerned (e.g. GUI, operating manuals, online help, hotline, etc.)?
- Who needs training, tools, etc.?

6 Nine good reasons and one Advice for Terminology Management in Operational, Supporting or Management Processes [7]

1: IMPROVE YOUR TECHNICAL AND CORPORATE COMMUNICATION AND DOCUMENTATION
Increasing specialization in all fields of knowledge as well as high innovation rates in many technical fields are leading to a need for ever more differentiated technical vocabularies (terminologies). Communication is becoming more and more difficult, not only between specialists and laypeople, but even between experts in one and the same discipline. This is particularly true when communicating across and beyond language and cultural borders. Today, technical communication comprises around 80% of all information exchanged across the new communication paths of a borderless and multilingual knowledge society. In the area of technical communication, the need for technical information and documentation is growing. The quantity and difficulty of specialist texts have increased, along with the demands placed on the technical documentation (laws, norms, customer

and corporate language). This is why experts in technical documentation must become familiar with the terminology of their field. Frequently, parts and components have different names in one and the same company. From everyday language we know that many common words have several meanings (homonyms). But technical words (terms) are by no means always clear. For example, for someone not familiar with fibre optics, it is by no means clear that fiberoptics, optical fibre and glass fibre all refer to the same concept. For the good of specialist communication, it is very important that the meaning of technical words be defined as early as possible, the results be documented and made available to potential communication partners. Terminology management serves here to a certain extent as a central hub for the multitude of corporate communication processes. It eliminates and prevents communication distortions that result from terminological ambiguities. It creates the precondition for clear communication between

- research and development,
- production and marketing
- the company, its cooperating partners, and suppliers
- the company, its markets, and its customers.

2: ORGANIZE FLEXIBILITY AND CHANGES WELL One small modification, such as, for example, changing one part of a technical component, will affect all models in which this part can be found. This means in plain English that all language versions of all model descriptions must be revised. At this time, these modifications are implemented more or less manually by companies or fed into a multitude of parallel-existing, but non-uniform, IT systems. This approach is very expensive and also conceals the risk of errors and confusion among all stakeholders. This problem can be considerably reduced through a uniform terminology in which the language of the company, the corporate wording is defined and documented – and then linked to other in-house information systems. Whether external or stand-alone, there are already numerous software solutions available for this purpose, such as terminology databases and terminology management systems on small, medium, and large IT systems. The terminologies stored in terminology databases help us saving and retrieving the information together with links and further notes. Terminology management software has been available on the market for a long time. It is structured in such a way that it is simple to use and produces good results, making possible a synchronization of terminology data between various IT systems within a company (e.g. CRM/ERP/inventory-management-systems etc.). Systems such as MultiTerm by SDL/Trados and Term star by their competitors Transit offer the possibility to save images and charts in addition to blocks of text.

3: ASSURE THE QUALITY AND LIABILITY OF YOUR PRODUCTS AND SERVICES At International, European and national levels, lawmakers place special requirements on the development of terminology, especially in the area of technical documentation. Well know in the US and Canada since long, product liability and its impacts has become a hot topic also in Europe and the rest of the world. In Europe, EU standards, product liability, and CE certification require companies to deliver, as an integral part of their products, documentation that meets safety requirements. Defective documentation is deemed a product defect that leads to complaints or even claims for damages. As described in the 1998 EU Product Liability Act, a product's manufacturer must compensate for damages that arise through defects in construction, fabrication, and even instructions. Such instruction defects are, among other things, derived from incorrect or unclear terminology. For this reason, documentation should be based on a predefined terminology in which the meaning of the terms is clearly established. The demands placed on the terminological quality of information and documentation have grown and have made terminology into a production and marketing factor and also a business factor in terms of quality, safety, liability and profitability.

4: BE PART OF THE INTERNATIONAL COMMUNITY: BENEFIT FROM STANDARDS In your day-to-day life as knowledge worker, you might need practical guidelines and common standards on how to implement existing terminological methods and tools, which are relevant to a broader and international community. The core standard on terminology work, ISO 704 Terminology work — Principles and methods, is being used by a growing number of diverse user groups. For beginners, there is a British Standard "Terminology work — Fundamentals made simple" (BS 8430:2005). For those who are interested in mechanisms for creating, selecting and maintaining data categories, as well as an interchange format for representing them, ISO 12620:2009 is relevant (Terminology and other language and content resources – Specification of data categories and management of a Data Category Registry for language resources). Many international, national, and domain-specific

standardization bodies make monolingual and multilingual terminologies available. ISO publishes hundreds of terminologies for a wide range of fields.

5: SHOW, SHARE AND SELL YOUR KNOWLEDGE AND PRODUCTS IN COMMON CLASSIFICATIONS. A close connection exists between classification systems and terminology systems. Well-conceived parts information systems are based on a classification system that classifies the individual parts in a hierarchic system according to their characteristics and definitions – exactly like the term systems common in terminology management. The technical words that denote the individual parts usually exist in several languages in multinational and strongly export-oriented companies. Lack of business data transparency restricts a company's profitability and performance capability. Classification systems created through reliable terminology work, together with the correct multilingual terminology, offer a company the advantage that the same and similar parts can be found quickly, that multiple designs can be avoided when developing new systems, that inventory management and procurement can be optimized, and that the documentation process – which also includes the translation – runs more efficiently and delivers high-quality results. A uniform classification is also required for doing business online (B2B and B2C). Electronic trade is successful when buyers are able to easily and quickly find products and services they are looking for, place their orders, and receive the deliveries in their native language. At the same time, salespeople should be able to offer their products and services around the world in standardized online catalogues. Harmonized terminological methods and the terminologies themselves are indispensable conditions for achieving this goal.

6: THINK GLOBAL, ACT LOCAL: USE TRANSLATION AND LOCALIZATION The importance of translating technical literature and documentation, the international interlacing of companies and associations, as well as the growing need on the part of customers to receive access to information in their native tongue place high demands on translation and localization. Terminology is the key success factor in these areas. Software handling must be user friendly and the handbooks must be precise, consistent, and understandable. A precondition for this is that the meaning of the technical terms has been explained to the users and that technical words are used uniformly throughout the interface, the help function, and in the documentation. In addition, it goes without saying that the software interface and manuals must be available in the user's native tongue. Nearly all large software companies around the world invest immense amounts in localizing software and have their own specialized terminology departments. New software products or versions must appear almost simultaneously on many different markets, and the interface and documentation must be translated into each local language. This can only be achieved when the multilingual terminology is already precisely established during software development and used uniformly during localization. This type of terminology can´t be found in corresponding technical dictionaries because it is innovative, company-, and often even product-specific. Different country versions can be quickly and almost simultaneously brought to market, when terminology methods are used during software localization. The localization costs shrink to such an extent that it even pays off to develop versions for "small" markets such as Slovakia, the Czech Republic, or Estonia. The product's quality (and documentation) is thus considerably increased, which also affects product liability, user satisfaction, and support costs (hotline).

7: BENEFIT FROM TERMINOLOGY APPLICATIONS IN THE PUBLIC SECTOR. The need for an exact terminology in the area of politics and public affairs is obvious indeed. Laws and regulations must be based on clearly defined concepts and the correct nomenclature for these concepts must be used in their wording. To avoid complications and misunderstandings, messages and statements must be clear in terms of the terminology used. Many ministries and public offices recognized terminology as an essential working field and deployed terminologists and terminology representatives in their language services to manage the terminology specific to the technical fields by using terminology databases. UNESCO, the United Nations Educational, Scientific and Cultural Organization, with its mission and its own variety of member states, languages and cultures, was one of the first international organizations to confront the topic of terminology. Differences in terminology often arise especially in international communication and can lead to very sensitive political situations.

8: THERE IS NO KNOWLEDGE WITHOUT TERMINOLOGY: CARE FOR YOUR KNOWLEDGE. Without appropriate terminologies, your staff cannot be properly trained nor can your experts and scientists work with precision. There especially is a need for terminology management and clarification

in many innovative areas of our knowledge societies. For instance in medicine, a variety of synonyms develop for the same phenomena because often the same research is conducted at different locations. But because research, clinical medicine, the pharmaceutical industry, legislators, and insurance providers must work closely together, clear communication based on precisely defined terminology is especially required in the medical environment. Flawed terminology can cause misunderstandings that can lead to deaths, damage to health, and enormous costs (see above, product liability, reason number 3).

9: MAKE BETTER USE OF WHAT IS THERE ALREADY. Usually, the chances are good that some units or individuals within your organization already have established terminology collections for their own domain and purpose. Scientists, writers and translators, engineers and software developers, marketing and legal experts – they all could keep lists of their products, services, functions or classifications. This existing material represents a sound basis for a consolidated, updated and expanded common terminology database, which in the future serves as a knowledge base for the entire organization or community. Uwe Muegge, Corporate Terminologist at Medtronic, emphasizes the importance of awareness of and share with all stakeholders: "Once a terminology management program is in place, it is essential that managers and communicators alike are made aware of the existence of such a program, and provided with easy access to the organization's terminology resources. (…) However, just having a website on the intranet that either provides an interface for searching the corporate terminology database, or simply lists all terminological entries in alphabetical order, can significantly improve the adherence of communicators to the established corporate terminology standard" [3]

10: YOU ARE NOT ALONE: JOIN THE TERMINOLOGY COMMUNITY TO GET HELP AND ADVICE. To get a first overview on "Who is who in terminology" have a look at the international terminology community at the website of TermNet, the International Network for Terminology, at the TermNet members list at: http://www.termnet.org/english/about_us/members_list.php. TermNet members are companies, universities, institutions and associations from all over the world who engage in the further development of the global terminology market. The products and services of this market are considered and promoted by TermNet and its members as integral and quality assuring parts of any product and service in the areas of a) information & communication, b) classification & categorization and c) translation & localization. The members of TermNet and their experts are connected with the key players and the respective terminology associations at national and regional level. TermNet currently is establishing an International Center of Excellence for Terminology: Research, Technologies and Services (www.termnet.org).

7   References

[1] http://www.termportal.de/eLearning/02_definition.html
[2] ISO 1087-1:2000-10 „Terminology work - Vocabulary - Part 1: Theory and application"
[3] U. Muegge, Why manage terminology? Ten quick answers, in: Globalization Insider 7 (2007). Available at: http://works.bepress.com/uwe_muegge/3
[4] ECQA Certified Terminology Manager-Basic Course, Skill Unit 1: Understanding terminology management
[5] CSOFT, "10 Good Reasons to Manage Your Terminology"
[6] http://www.businessdictionary.com/definition/business-process-improvement-BPI.html#ixzz1qxoIKjx0
[7] G. Sauberer, Ten good reasons for terminology, in: proceedings of the Spring 4th International Conference on Knowledge Generation, Communication and Management, on April 8th, 2010, in Orlando, Florida, USA: http://www.iiis.org/CDs2010/CD2010IMC/KGCM_2010/PapersPdf/GB787QV.pdf and G. Sauberer, There Is No Knowledge Without Terminology. How Terminological Methods and Tools Can Help to Manage Monolingual and Multilingual Knowledge and Communication, in: Journal of Systemics, Cybernetics and Informatics 2011: http://www.iiisci.org/journal/sci/Abstract.asp?var=&id=OL787QV

Integrating different assessment approaches to evaluate safety-critical software development in nuclear domain

Risto Nevalainen and Timo Varkoi
Finnish Software Measurement Association – FiSMA
{risto.nevalainen, timo.varkoi}@fisma.fi

Abstract

We can evaluate safety-critical software development by several approaches, e.g. software product quality evaluation, safety assessment, or development process assessment. In this paper, we discuss how process evidence could be treated when harmonizing different evaluation approaches. The context of this work is safety-critical software development for nuclear power industry. In addition to the rigorous process, safety and nuclear domain specific standards need to be followed. The requirements are partially overlapping and they shall be demonstrated during an assessment. Integrating product and process views helps to integrate evidence collection for various evaluation approaches. Process assessment can be adjusted to highlight the software development practices that are relevant in achieving safety or reliability of the software product. Process assessment, with appropriate rigour, can support safety assessment and provide usable sets of evidence also for product evaluation.

Keywords

software process assessment, functional safety, SPICE, software product quality

1   Introduction

Safety evaluation and assessment must be as much fact-based as possible. Still, it is impossible to collect enough evidences to have complete detection and proof of all potential safety risks and problems. Real life is a combination of judgement-based expertise and fact-based, explicit and well documented knowledge in standards, methods and tools. This can be illustrated as a quadrant, see figure 1.

The aim is to minimise "unknown" and maximize "known" by using different approaches, see figure 1. "Known" means that

- either the evaluator or auditor knows and can verify safety requirements, identify potential risks and problems / potential faults; or
- safety requirements, potential risks and problems / potential faults are known by others and written in standards, models and tools.

This classification leads to three subcategories of "known":

- Basic audit: both evaluator and models/methods know what is required and looked for. If evaluation is done in a professional way, this is an easy way to analyse. Also certificate should be easy because all necessary evidences and proofs exist.
- Expert judgement: No support from standards, models and tools is available. Safety evaluation is based on personal and team competences. Critical factor is profound knowledge of evaluators and their adequate training.
- Model based audit: We can trust on explicit, cumulated knowledge in standards, methods and tools. By their proper use we can cover at least some areas of safety evaluation.
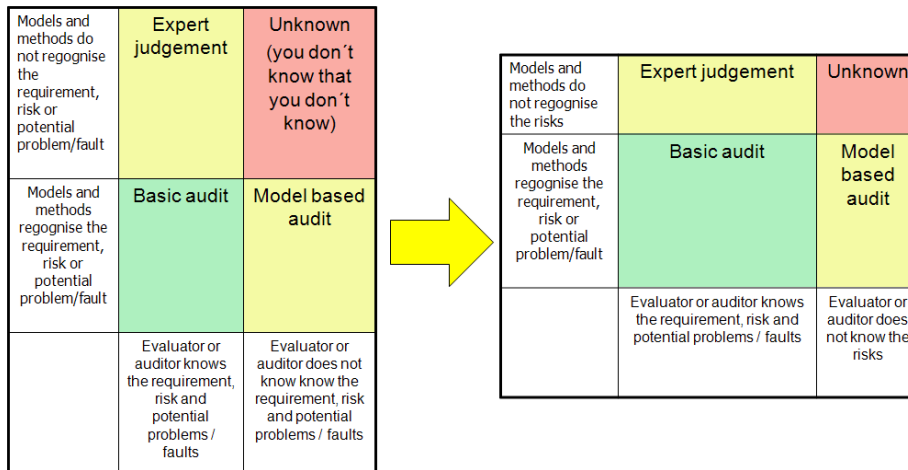

Fig. 1. The worlds of known and unknown in safety evaluation and assessment

Minimising "unknown" can be done by extending either judgement-based or fact-based coverage of safety evaluation and assessment, or both. This is visualized in figure 1.

2 Process and Product View of Evidences

We can evaluate safety-critical software development from several viewpoints. The key output, the software product, can be evaluated against a predefined set of e.g. quality requirements. Safety assessment can study both the product and the processes used in development and use of the software based often on domain specific standards. Process assessment focuses typically on the product development phase. All these approaches (table 1) produce valuable information in building trust on the safety of the product. So far, harmonization of these approaches is missing for safety-critical software.

Table 1. Selected approaches and their key characteristics

Topic	Product evaluation approach	Safety assessment approach	Process assessment approach
Main purpose of the approach	To analyse and show compliance of product (artefact) by using selected criteria	To demonstrate compliance with a selected reference (standard)	To demonstrate capability to develop, deliver and improve
Main focus in safety-critical domain	Product quality, especially reliability metric and data, for example MTBF	Compliance with generic or domain specific safety standard, certification	Process evidences to demonstrate achievement of safety management and engineering
Specifics of each	Internal, external,	Inspections, re-	Professional

approach	in use metric	views, V&V evidences, technical practices and methods	practices, work products, capability levels
Commonalities with other approaches	V&V metric, measurement and analysis practices	Engineering methods and competences	Process results, like mandatory work products
Typical standard(s) and models	ISO/IEC 25000 family (SQUARE)	IEC 61508, IEC 61513, IEC 60880, ISO 26262	ISO/IEC 15504 (SPICE), Automotive SPICE, Nuclear SPICE

Product evidences can also serve as safety assessment and process assessment evidences and vice versa. So, it is meaningful to harmonise those approaches to be more supportive for each other. An example could be traceability, which is direct requirement in safety assessment standards and in process assessment models. Another good example is testing coverage, which can be classified both product and process evidence for verification and validation (V&V) activities.

Product quality model ISO/IEC 25010 [12] includes eight characteristics in internal and external metric and five characteristics in in-use metric. Reliability is one characteristic, including Maturity, Availability, Fault tolerance and Recoverability as sub-characteristics. Safety is less obvious sub-characteristic in in-use model, belonging to Freedom from risk characteristic. It is called there "Health and safety risk mitigation".

Safety is then most relevant in existing and operational systems. This view has unnecessary limitation, because safety can be built in the system and software by a rigor development process. Alternatively, safety can be seen as a combination of process quality and product quality. They have also dependencies and relationships, as illustrated in figure 2.
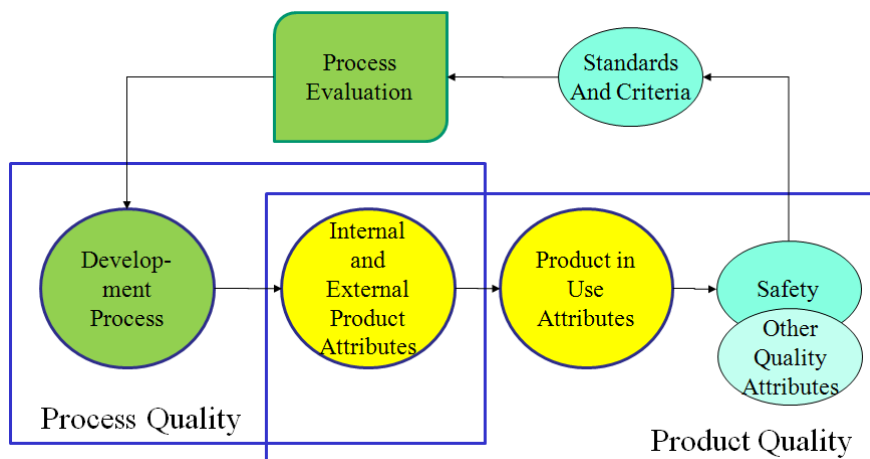

Fig. 2. Process and product quality, integrated view

SPICE assessment evaluates process capability by process attributes [8]; this can be interpreted as an indication of process quality. Process attributes are similar to product quality attributes defined in SQUARE. Integrating product and process views helps in integration of evidence collection for various evaluation approaches.

3 Integrated Approach in Evidence Collection and Coverage

Integrated approach in evidence collection means, that we collect evidences only once and use is either together or separately to evaluation needs. Process evaluation needs typically more management evidences, depending on evaluation scope. SPICE standard defines clearly, what is the minimum evidence to rate process attributes. Product evaluation is based typically in specific evidences, like reliability calculations, V&V data and product technical documentation.

Minimum option is to keep SPICE and domain specific safety standards separate and leave space for ad hoc integration in each assessment and qualification case. This option is more or less the current situation. Assessment team should have a portfolio of assessment methods and use them as needed.

The other extreme is tight integration. It would mean that each element and requirement in SPICE and domain standards is linked with each other. Also overlaps are removed and combined when possible, to minimize unnecessary recording of evidences.

Integration may require elements and requirements also from other sources than standards, mainly based on regulatory and qualification needs. An example is information security, which is highly relevant nowadays. Because current SPICE does not have any process for information security, it will be taken either from ISO/IEC 27001 [13] or ISO/IEC 15504 Part 8 [10].

Integration can be illustrated as a three-level model, see figure 3. First we can integrate a generic approach in SPICE (supplemented by safety processes of ISO/IEC Part 10 [11]) with functional safety requirements in IEC 61508 [4]. This can be further integrated with nuclear domain requirements and classifications. All levels have their own internal structure.
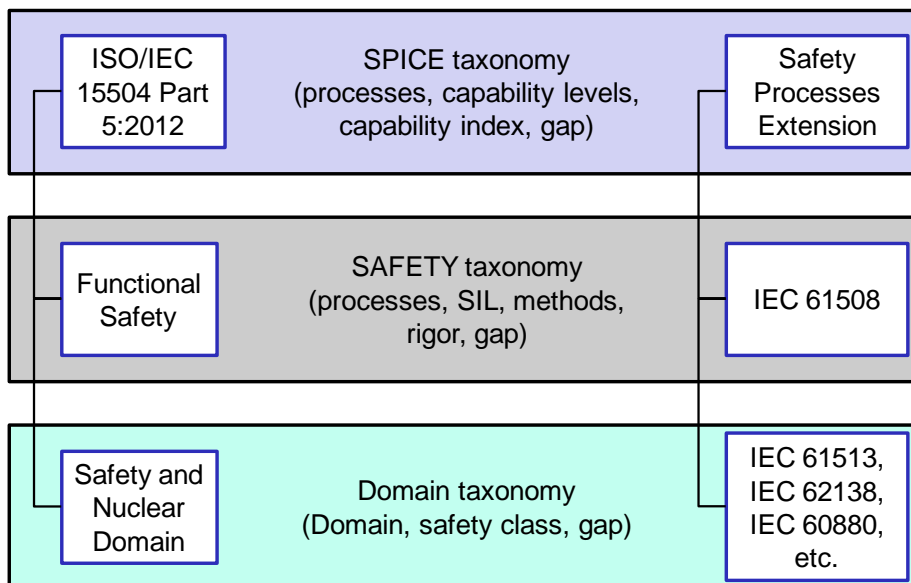

Fig. 3. Main levels of integration and classifications by safety standard types [14]

Additionally, we need to consider also other aspects than abstraction level in the integration. One viewpoint is the current state-of-the-art in tools and technologies. For example, FPGA processors can be used in some safety-related applications in nuclear power plants. Current safety standards [3, 5, 6] do not say anything about implementation technologies. Their suitability and adequacy must be considered otherwise, mainly by interpreting their capabilities separately in each case.

4 Using Different Classifications

Integration of multiple approaches produces an issue: each approach contains plenty of information, definitions and at least slightly divergent terminology. Efficient use of integrated approaches requires methods and tools for support. Next, we propose a possible solution to manage the various taxonomies.

Each existing approach and abstraction level has its own classifications. Some examples are:

- SPICE: Processes, process attributes, capability levels [8, 9]
- IEC 61508: Processes, SIL levels, methods, rigor [4]
- IEC 60880: Processes, safety class, method requirements, nuclear domain specific requirements [3]
- ISO 26262: System vs. software, processes, ASIL, methods [7]

Integration of approaches means also better harmonization between classifications. For example, a process in SPICE can achieve capability level 3. It should have some relevance also in satisfying process specific requirements in IEC 61508 and in ISO 26262. Also, if some requirement in ISO 26262 is satisfied, it should be an evidence also for functional safety and SPICE. Otherwise, there is no integration.

Ideally, we can develop evidence collection smarter to avoid unnecessary recording work and to fulfil product specific qualification needs. Product evaluation is typically done using a safety case or an assurance case approach. SPICE PRM can provide major part of evidences for safety case calculations, but it can only be achieved gradually.

All different classifications can be expressed as a "taxonomy machine" (figure 4). It could be a smart tool, being able to combine different evidences and (re)using them to satisfy different classifications. In this phase of Nuclear SPICE development [1], taxonomy machine is just a metaphor and needs further elaboration.
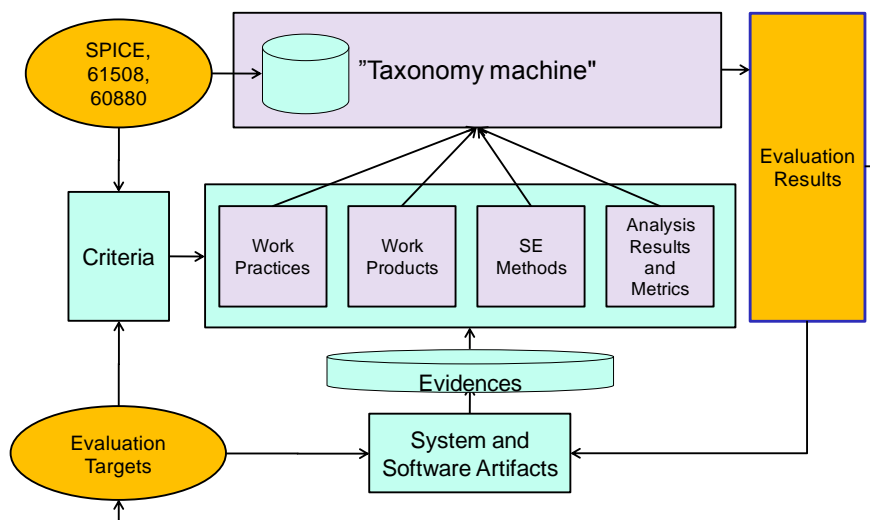

Fig. 4. Taxonomy machine concept to combine different types of classifications and evidences [14]

An essential part of process assessment is to record and classify evidences. They can be categorized by type and source. Figure 4 proposes a four-type classification system for evidences:

- Work practices: they can be base and generic practices in SPICE or process requirements in IEC 61508 and in IEC 60880.
- Work products: they can be same as in SPICE. Some work products could be mandatory, others optional.
- Software engineering methods. Main source are tables in IEC 61508 Part 3 Annexes A, B and C. Also rigor and relevance for SIL is here. This evidence type allows also new methods to be included in Nuclear SPICE, if and when some research projects can validate new methods.
- Analysis results and metrics: The key is relevance for validation, using various analyses of measurement results. Examples are PHA, FMECA, reliability calculations and V&V metrics.

5   Requirements for an Assessment Process

The Assessment Process gives guidance to the assessment and specifies the minimum requirements for the performance of an assessment. The main goal is to ensure credibility and repeatability of the assessment result.

Planning is needed to identify the scope of the assessment (which processes to assess), when and where the assessment takes place, who will participate, material required and so on. Data collection may consist of interviews, revision of related documents and measurements. Data validation ensures that consistent and correct data has been collected. Process attribute rating means that the elements of an implemented process are analysed and their contribution to the achievement of the goals of the process are evaluated. Reporting is needed to declare and record the results of the assessment.

For example, the Nuclear SPICE assessments are performed to evaluate the capability of systems and software development process applied in systems and software engineering in nuclear industry domain. The domain is safety-critical and presents strict requirements for the capability of the processes. [2]

The level of rigour in the assessments may vary and the level is independent of the assessment scope. E.g. the Nuclear SPICE assessment method [1] defines three classes for the assessment rigour:

- Class 1: Compliant with ISO15504
- Class 2: Team and expert driven
- Class 3: Expert judgement

The assessment requirements may include stakeholder, functional, non-functional, and generic standard-based requirements. Functional requirements deal with assessment scope selection, assessment rigour, and assessment input and output. Non-functional requirements can define assessment usability including performance, confidentiality, reliability, risks and legal aspects.

Table 2. Requirements for safety-critical assessments

Requirement class	Requirement	Rationale
Stakeholder	Assessments are applicable in the domain.	The assessment is applicable in safety-critical nuclear industry domain for systems and software development processes. The assessment must produce a capability rating for an assessed process.
	Assessment satisfies stakeholder needs.	The assessment provides adequate information to all defined stakeholders of an assessment.
	Assessments are ISO/IEC 15504-2 conformant.	When needed, the assessment must meet the requirements of the standard. Assessments may be performed with less rigour.
Scope	Assessments are adaptable.	The assessments can be applied to various classes of process assessments in safety related software context. The level of rigour may vary; the level of rigour is independent of the assessment scope.
	Assessment scope is variable.	The assessment scope may consist of 1-n processes. Nuclear SPICE has three pre-defined process sets: Core, Management, Full conformance. Also, the target capability level may vary for each

		assessed process.
	Assessments are flexible.	Assessments support modular approach: different sets of assessment indicators can be used. Indicators may interface to several domain related standards.
Rigour	Assessment is directed.	Assessment process must be documented and the rating guidance must enable consistent interpretation of the assessment result.
	Assessment has rigour.	Assessment must provide appropriate rigour to be credible for all stakeholders.

Process assessment can be adjusted to highlight the software development practices that are relevant in achieving safety or reliability of the software product. Process assessment, with appropriate rigour, can support safety assessment and provide usable sets of evidence also for product evaluation.

6   Summary and Future Work

In this paper, we present some approaches used to evaluate safety-critical software development. We aim to solve how process evidence could be treated when harmonizing different evaluation approaches. Our experience and presented views are based on assessments of safety-critical software development for nuclear power industry. In this context a rigorous process is required, but not adequate: many safety and nuclear domain specific standards must be taken into account, too. The requirements are partially overlapping and they shall be demonstrated during an assessment.

Integrating product and process views helps to integrate evidence collection for various evaluation approaches. Process assessment can be adjusted to highlight the software development practices that are relevant in achieving safety or reliability of the software product. Process assessment, with appropriate rigour and well defined assessment process, can support safety assessment and provide usable sets of evidence also for product evaluation.

We propose an integration of evaluation approaches and a method to manage various taxonomies in safety-critical assessments. Our future work includes specification and development of tools to support the "taxonomy machine". We also intend to promote the integration of process and product quality thinking in the international standardization.

7   References

1. FiSMA report 2011-1: Risto Nevalainen, Timo Varkoi: S4N method description – Nuclear SPICE PRM and PAM. Espoo 2012.

2. FiSMA report 2011-2: Timo Varkoi, Risto Nevalainen: S4N Assessment Process – requirements for Nuclear SPICE assessment. Espoo 2012.

3. IEC 60880, Nuclear power plants – Instrumentation and control systems important to safety – Software aspects for computer-based systems performing category A functions. 2006.

4. IEC 61508-3, Functional safety of electrical/electronic/programmable electronic safety-related systems Part 3: Software requirements. 2010.

5. IEC 61513 FDIS, Nuclear power plants - Instrumentation and control for systems important to safety - General requirements for system. 2011.

6. IEC 62138, Nuclear Power Plants – I&C Systems Important to Safety – Software Aspects for Computer Based Systems Performing Category B and C Functions. 2004.

7. ISO 26262, Road vehicles — Functional safety [several parts]. 2009.

8. ISO/IEC 15504-2, Information technology – Process assessment - Part 2: Performing an Assessment. 2003.

9. ISO/IEC 15504-5 FDIS, Information technology – Process assessment - Part 5: An exemplar Process Assessment Model. 2012.

10. ISO/IEC 15504-8 PDTR, Information technology — Process assessment — Part 8: An exemplar process assessment model for IT service management. 2011.

11. ISO/IEC 15504-10, Information technology — Process assessment — Part 10: Safety extension. 2012

12. ISO/IEC 25010 FDIS, Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models. 2010.

13. ISO/IEC 27001, Information technology -- Security techniques -- Information security management systems – Requirements. 2005.

14. Risto Nevalainen, Timo Mäkinen, and Timo Varkoi: Towards SPICE for Nuclear (S4N) – Integrating IEC 61508, IEC 60880 and SPICE, 18th EuroSPI Conference Industrial Proceedings, Selected Experience Workshop Papers, June 27-29, 2011, Roskilde University, Denmark, pp. 11.17-28.

8   Author CVs

Risto Nevalainen

Mr. Risto Nevalainen (Lic. Tech.) has long experience in software measurement and quality topics. He has been managing director of Spinet Oy for last 7 years. He is also managing director of Finnish Software Measurement Association FiSMA. His working experience includes position as managing director of Finnish Information Technology Development Center during 1989-1995. Before that he had different research and management positions for example in Prime Minister´s Office, Technical Research Centre (VTT) and Technical University of Helsinki (HUT). Mr. Nevalainen has participated in ISO/IEC 15504 (SPICE) standard development since the beginning. He is Competent SPICE Assessor and ISO9001 Lead Assessor.

Nevalainen has participated in development of SPICE based assessment model for nuclear domain for several years. Current phase of development is to define an assessment model and method, Nuclear SPICE, for highest class safety-critical software. Work is part of the Finnish nuclear safety SAFIR 2014 research program.

Timo Varkoi

Mr. Timo Varkoi (Lic. Tech.) has studied Computer Science at Tampere University of Technology and graduated Master of Science in 1989 and Licentiate of Technology in 2000. His working experience includes both software development and management responsibilities in industrial software organizations. Recently he has been working as a project manager in research and expertise service projects to improve software processes with Finnish software companies. He is a competent SPICE assessor, an active member of the Finnish Software Measurement Association (FiSMA), and a founding member of the SPICE Academy. His current interests include software process assessment and improvement related research, expertise and training combined with development of software intensive organizations.

Mr. Varkoi is also the editor of ISO/IEC 29110-3 VSE Assessment Guide and the soon to be published update of ISO/IEC 15504-5 Exemplar Software Process Assessment Model, and a member of the Standards Management Group (SWG5) of ISO/IEC JTC1 SC7. He participates in the development of the SPICE based *Nuclear SPICE* assessment model.