EuroSPI² 2014 Proceedings

Proceedings

The papers in this book comprise the industrial proceedings of the EuroSPI² 2014 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, FiSMA), iSQI as a large German quality association, the American Society for Quality SW Division, 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 21st 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 Centre de Recherche Public Henri Tudor, Luxembourg, is the host of the EuroSPI² 2014 conference. CRP Henri Tudor is currently leading research projects in the areas of Advanced Materials Technologies, Health care technologies, Information and Communication Technologies, and Business Organisation and Management, and provides leading experts in these areas as key notes to the EuroSPI 2014 event.

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	Béatrix Barafort, Centre de Recherche Public Henri Tudor, Luxembourg
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	Morten Korsaa, DELTA, Denmark
Industrial Programme Chair	Nils Brede Moe, SINTEF, Norway
Organizing Chair	Adrienne Clarke, ISCN, Ireland

Co-Organizing Chair	Eva Christof, ISCN Austria	
Local Organizer	Béatrix Barafort, CRP Henri Tudor, Luxembourg	
Local Co-Organizer	Marylène Martin, CRP Henri Tudor, Luxembourg	
Local Co-Organizer	Samuel Renault, CRP Henri Tudor, Luxembourg	

Industrial Programme Committee

Bæk Jørgensen Jens	Mjølner Informatics A/S	Denmark
Breske Eva	Robert Bosch GmbH	Germany
Christiansen Mads	Terma A/S	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
Hagenmeyer Philipp	ZF Friedrichshafen AG	Germany
Hallikas Jarmo	Falcon Leader Oy	Finland
Hällmayer Frank	Software Factory GmbH	Germany
Ivanyos Janos	Trusted Business Partners Kft	Hungary
Johansen Jørn	DELTA	Denmark
Johansson Mika	FiSMA	Finland
König Frank	ZF Friedrichshafen AG	Germany
Messnarz Richard	ISCN Ltd	Austria/Ireland
Morgenstern Jens	-	Germany
Nevalainen Risto	FiSMA	Finland
Poth Alexander	Volkswagen AG	Germany
Renault Samuel	Centre de Recherche Public Henri Tudor	Luxembourg
Baer 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 Friedrichshafen AG	Germany

EuroSPI² Board Members

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

ASQ SW Division http://www.asq.org

Editors of Proceedings

Richard Messnarz, ISCN, Austria/Ireland

Jørn Johansen, DELTA, Denmark

Morten Korsaa, DELTA, Denmark

Eva Christof, ISCN, Austria

Damjan Ekert, ISCN, Austria

Supporting Partners

CRP Henri Tudor, Luxembourg, http://www.tudor.lu/

European Certification and Qualification Association, http://www.ecqa.org

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, FiSMA), 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.

EuroSPI² 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 21st 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 I2E (Idea to Enterprise), AQUA (Knowledge Alliance for Training Quality and Excellence in Automotive), LSSH (Lean Six Sigma for Health Care), TransCert (Trans-European Voluntary Certification for a pool of professions related with SPI and Translators), and LEADSUS (Leadership in Sustainability). 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 Alexander Poth – New Co-editor of EuroSPI Books

Alexander Poth

Volkswagen AG,

Germany

In my opinion we can have more effective and efficient IT solutions with better methods for professional development and improvement of software systems and services.

I think a key to reaching this goal is to focus on methods that ensure the explicit and implicit demanded quality of IT systems and services leading to more customer and user satisfaction. In a user perspective added value by IT solutions is based on adequate quality. To deliver adequate quality we have to continuously realign the quality of the IT solution with the current quality demands of the users.

EuroSPI is a platform that brings together people from the industry and academic world to address this demand for more effective and efficient high quality IT solutions. This is the reason why I'm an active member in the EuroSPI community. My personal objective is to give ideas and feedbacks to the EuroSPI community to improve innovative concepts and methods for usage in the daily IT business to realize a higher added value with IT solutions.

Alexander Poth received the Dipl. Ing. (Master) degree in 2004 in computer engineering from the Technical University of Berlin. He is IT Quality Manager at Volkswagen AG. He was senior consultant for software quality management at Software Quality Systems AG from 2007 to 2012 after working as quality engineer for Mercedes Benz Technology GmbH. He is author of many conference and journal publications. His interests include software engineering, software quality management, software process improvement & innovation, risk- and complexity management and continuous delivery.

Contact: Alexander Poth, Volkswagen AG, Germany, e-mail: alexander.poth@volkswagen.de

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 20 years including maturity assessment according to BOOTSTRAP, SPICE and CMMI. DELTA has also been a partner in the EuroSPI² conference from the very beginning 20 years ago. We are now for the 6th time the publisher of the Industrial Proceedings from EuroSPI² making it part of the DELTA series about Process Improvement.

Jørn Johansen is Director Process Improvement and Senior Technology Specialist at DELTA. He has an M.Sc.E.E. from Ålborg University and more than 35 years of 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 19 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. The Talent@IT project developed the ImproveAbility™ model, which help organisations to improve more efficient. Latest Mr. Johansen was the Project Manager of SourceIT project, an 18 person-year project focusing on outsourcing and maturity.

He has also been a main driver in establishing and performing a large set of innovation checks in Danish companies. He has by him selves taken part in more than 50.

Mr. Johansen is also the co-ordinator of a Danish knowledge exchange groups: Improving the Software Development Process, Outsourcing and the newest group Software Measurement.

Mr. Johansen was lead editor on ISO/IEC 33014 Guide for process improvement, which was published November 2013

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

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

Béatrix Barafort

Centre de Recherche Public Henri Tudor, Luxembourg

Welcome to the 21st EuroSPI[2] Conference at Centre de Recherche Public Henri Tudor in Luxembourg-Kirchberg.

The Centre de Recherche Public Henri Tudor is situated in Luxembourg, at the heart of Europe. The CRP Henri Tudor is an autonomous institute of applied research. Essential link between research and society as a whole, its mission is to sustainably reinforce the economic competitiveness and the social fabric, at national, regional and European level. Its activities include applied and experimental research, doctoral research, development of tools, methods, labels, certifications and standards, technological assistance, consulting and watch services, knowledge and competences transfer. Its main technological domains are advanced materials technologies, environmental technologies, health care technologies, information and communication technologies as well as business organisation and management.

The Service Science & Innovation (SSI) department is an interdisciplinary centre dedicated to an approach that is grounded in the science of service innovation, based on the merging interdisciplinary field of science known as « Service Science, Management and Engineering » (SSME). SSI is made up of 126 researchers (as of 31/12/2013). For more than 18 years, (IT) process assessment and improvement science-based initatives have been developed.

Béatrix Barafort is the local chair for the EuroSPI 2014 Conference. She graduated as a Software Engineer in the "Conservatoire National des Arts et Métiers" (Lyon, France) and has worked in a software house in Lyon for development projects in banks and insurance companies prior to her current position. Since 1996 in CRP Henri Tudor, she has led R&D process assessment and improvement projects based on the ISO/IEC 15504 standard (Process Assessment), mostly in Software Engineering and IT Service Management.

Since 2003, she has been leading a Research Unit in the SSI department and is currently heading the "Business Process and Service Engineering" one encompassing the TIPA® initiative (Tudor's IT Process Assessment). She is actively involved in standardization activities in ISO/IEC JTC1 SC7 (Software and Systems Engineering) and in ISO/IEC JTC1 SC40 (IT Service Management and IT Governance). She is President of this latter for the National Mirror Committee for Luxembourg. She was editor of the published ISO/IEC 20000-4 standard for an IT Service Management Process Reference Model.

Contact Details:
Béatrix Barafort (E-Mail: beatrix.barafort@tudor.lu)

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 joins institutions and thousands of professionals from all over Europe as well as globally 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 offered in 24 countries across Europe by more than 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, China, Thailand, India, Singapore, Japan etc.).

The main objective of the ECQA is to develop and maintain a set of quality criteria and common certification rules across the different 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 modularization is, according to European quality framework, split into units, elements and performance criteria. The certification exam then shows exactly which elements and units have been passed and which have not.

The ECQA has developed a set of quality criteria, which is 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.

Working today means cooperating with a many international partners. Thus, the understanding on both sides is essential. Certifications can help to better understand the different views of different professions. Because of this the ECQA Job roles aim at core competences in networking and understanding as well as concentrate on the needs of the industry.

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 coordinates and participates in various EU projects.

In the last 3 years, ECQA has developed towards an international certifier issuing certificates and establishing partnerships in all European countries as well as in India, South America, China, Japan and Arabia. This expansion on the one hand enriches ECQA and its job roles with new views and different cultural aspects but also shows that there will be the need of new approaches for the solution of international certification schema.

I wish you a good time at the EuroSPI² 2014, 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

Table of Contents

Wolfgang Raschke, Graz University of Technology, Austria; Massimiliano Zilli, Graz University of Technology, Austria; Johannes Loinig, NXP Semiconductors Austria GmbH, Gratkorn, Austria; Christian Steger and Christian Kreiner, Graz University of Technology, Austria

Onur Kaynak, Unal Aksu, Innova Information Technologies, Turkey

Su-jin Choi, Sogang University, Seoul, South Korea; Dae-Kyoo Kim, Oakland University, Rochester, USA; Sooyong Park, Sogang University, Seoul, South Korea

Jan Pries-Heje, Roskilde University, Roskilde, Denmark; Jørn Johansen, DELTA, Hørsholm, Denmark

Experience Session 4: Assessment

Jozef De Man Ghent University, Alcatel-Lucent, Belgium

Samuel Renault, PRC Henri Tudor, Luxembourg

Timo Varkoi, Spinet Oy, Finland; Risto Nevalainen, Finnish Software Measurement Association – FiSMA ry, Finland

Experience Session 5: Assets Management

Alejandro Ruiz-Robles, University of Piura, Peru; Maria-Isabel Sánchez-Segura; Fuensanta Medina-Dominguez, Carlos III University of Madrid, Spain

Experience Session 6: Safety

Masao Ito, Nil Software Corp., Tokyo, Japan

Richard MESSNARZ, ISCN LTD/GesmbH Ireland and Austria; Christian Kreiner, Graz University of Technology, Austria; Andreas Riel, Serge Tichkiewitch, EMIRAcle c/o Grenoble Institute of Technology GSCOP UMR5272, Grenoble, France; Damjan Ekert, ISCN LTD/GesmbH Ireland and Austria; Michael Langgner, Automotive Cluster Austria, Austria; Dick Theisens, Symbol BV, Netherlands

Experience Session 7: Innovation

Experience Session 8: Safety

Experience Session 9: Development Models

Manzano, Universidad Politécnica de Madrid, Spain; Tomás San Feliu, Universidad Politécnica de Madrid, Spain

ECQA Governance SPICE Assessor Skills
for Evaluating Integrated Risk Management Scenarios

János Ivanyos

Trusted Business Partners Ltd, Budapest, Hungary,
ivanyos@trusted.hu

Abstract

Professionals having been acquiring and evidencing their Governance SPICE Assessor skills are able to provide unique consulting and assurance services for supporting enterprises in achieving well established business goals and targets at an affordable level of risk treatment costs and effect of uncertainties by assessing and evaluating capability of enterprise governance processes.

1.   Risk Management Principles Supporting Enterprise Governance

Enterprise Governance principles are often referred as requirements and recommendations prepared by the supervision authorities or international professional organizations only for the publicly listed, the state-owned and the big multinational companies. However following these principles - far beyond the prescribed compliance requirements - is important for all market-driven economic entities due to establishment and maintenance of trusted business relations.

"Trusted Business" is highly substantial for all stakeholders, such as the owners and investors, the employees, the customers and suppliers, the creditors, and the authorities and associations of public interest for social, economic and ecologic sustainability. As the aware business risk taking is an essential element of the economic growth and innovation, it is definitely stressful how the involved parties "grease the skids" for successful management of uncertainties effecting business goals, like operational, environmental, legal, societal, human, health, etc. risks - in either micro or macro environment. The lower is the level of business trust measuring acceptance and undertaken of unavoidable uncertainties in business relationships, the higher is the cost of risk-taking due to mistrust (like in the form of higher interest rates, insurance and enforcement costs, etc.), which leads to lower efficiency and competitiveness by the unsubstantiated

increase of operational costs. For these reasons, even the SMEs can benefit from presenting their governance capability conforming to international standards without exaggerated implementation and assurance costs by adapting the **Governance Model for Trusted Businesses** [1] aligned with their own business goals and environment.

For effective Risk Management integrated with Enterprise Governance, according to **ISO 31000:2009 Risk Management** standard [2], the company should at all operational and organizational levels comply with the following principles:

a) **Risk Management creates and protects value**.
Risk management contributes to the demonstrable achievement of enterprise objectives and improvement of business performance.

b) **Risk Management is an integral part of all organizational processes**.
Risk management is not separated from the main activities and business processes of the organization. Risk management is part of the responsibilities of management and an integral part of all business and governance processes at all operational and organizational levels.

c) **Risk Management is part of decision making**.
Risk management assists management make informed choices, prioritize actions and distinguish among alternative courses of action.

d) **Risk Management explicitly addresses uncertainty**.
Risk management explicitly takes account of uncertainty, the nature of that uncertainty, and how it can be addressed.

e) **Risk Management is systematic, structured and timely**.
A systematic, timely and structured approach to risk management contributes to efficiency and to consistent, comparable and reliable business results.

f) **Risk Management is based on the best available information**.
The inputs to the risk management process are based on information sources such as historical data, experience, stakeholder feedback, observation, forecasts and expert judgement. However, decision makers should take into account, any limitations of the data or modelling used or the possibility of divergence among experts.

g) **Risk Management is tailored**.
Risk management is aligned with the organization's external and internal business context and risk profile.

h) **Risk Management takes human and cultural factors into account**.
Risk management recognizes the capabilities, perceptions and intentions of external and internal people that can facilitate or hinder achievement of the organization's business objectives.

i) **Risk Management is transparent and inclusive**.
Appropriate and timely involvement of stakeholders and, in particular, decision makers at all operational and organizational levels, ensures that risk management remains relevant and up-to-date. Involvement also allows stakeholders to be properly represented and to have their views taken into account in determining risk criteria.

j) **Risk Management is dynamic, iterative and responsive to change**.
Risk management continually senses and responds to change. As external and internal events occur, business context and knowledge change, monitoring and review of risks take place, new risks emerge, some change, and others disappear.

k) **Risk Management facilitates continual improvement of the organization**.
Management should develop and implement strategies to improve the risk management capability aligned with all other (e.g. process) improvement aspects of the enterprise.

The above listed principles adopted from ISO 31000:2009 Risk Management standard comprise the base for interpreting the Integrated Risk Management scenarios aiming at specific enterprise objectives within different timescales for the operational and organizational levels of business operation. Therefor risk management is integrated into the Enterprise Governance by supporting effective decision making and improvement of business performance.

ISO 31000 Risk Management principles do not directly address the term of *"risk appetite"*, however the **COSO** [3] definition and interpretation is not really far from applying these principles in context of Enterprise Governance:

COSO's Enterprise Risk Management - Integrated Framework defines risk appetite as follows:

> *"The amount of risk, on a broad level, an entity is willing to accept in pursuit of value. It reflects the entity's risk management philosophy, and in turn influences the entity's culture and operating style. … Risk appetite guides resource allocation. … Risk appetite [assists the organization] in aligning the organization, people, and processes in [designing the] infrastructure necessary to effectively respond to and monitor risks."*

> *"This definition raises some important points. Risk appetite:*

> - *is strategic and is related to the pursuit of organizational objectives;*
> - *forms an integral part of corporate governance;*
> - *guides the allocation of resources;*
> - *guides an organization's infrastructure, supporting its activities related to recognizing, assessing, responding to, and monitoring risks in pursuit of organizational objectives;*
> - *influences the organization's attitudes towards risk;*
> - *is multi-dimensional, including when applied to the pursuit of value in the short term and the longer term of the strategic planning cycle; and*
> - *requires effective monitoring of the risk itself and of the organization's continuing risk appetite."*

> *Source: Enterprise Risk Management - Understanding and Communicating Risk Appetite (COSO 2012)*

In this interpretation risk appetite is definitely not just a set of acceptable (practically hardly measureable and comparable) risk levels, but much more a strategic thinking about how the uncertainties around the business objectives and their effects on these objectives should be managed at all operational and organizational levels. Implementation of enterprise governance and control processes should follow the related management decisions aligned with specific business conditions, formalized as Risk Appetite Statements (at strategic level) or Management Assertions (at operational and organizational levels), instead of complying with generic models normally used as simple check-lists by assurance providers (e.g. auditors).

The **ISO/IEC 15504 standard** [4] based process improvement and capability determination methodology provides a conceptual framework for determining organizational risk appetite by setting target capability levels for key business and governance processes. Evaluation of the gaps between target and actually assessed capability profiles provides input for the next risk treatment planning and implementation cycle at the concerning operational or organizational level.

2.   Applicable Governance Objectives and Enabling Processes

The well established and recognized control frameworks and process reference models could be used for implementing and evaluating effective and efficient enterprise governance, if only the management established its own governance related objectives. Unfortunately, structures of generic control frameworks and reference models promoted by assurance professional bodies are not easily interpretable by enterprise management for setting their business' specific governance objectives.

The **Governance Model for Trusted Businesses** keeps both enterprise management and assurance (e.g. audit) logics in mind by presenting governance processes in line with the specific objectives relevant for the company, together with an exact mapping to processes of control frameworks (reference models) accepted and used by assurance providers (e.g. auditors) for compliance attestation.

The Governance Model aims 11 governance objectives:

Supporting Business Sustainability:	Supporting Organization's Internal Control System:		
✓ **Competitiveness**	✓ **Accountability**	✓ **Commitment**	✓ **Risk Awareness** ✓ **Control Efficiency**
✓ **Exploitability**	✓ **Process Integrity**	✓ **Competency**	
✓ **Satisfaction**	✓ **Accuracy**	✓ **Data Protection**	

Figure 1: Applicable Governance Objectives for Trusted Business

While business sustainability objectives are significant for keeping business operation economically effective and successful, the organization's internal control objectives have the major focus on how effectively internal control system is enabling achievement of strategic, operational effectiveness, reliability and business process performance related Enterprise Goals. Each governance objective should be determined in context of the specific organizational or operational level by considering the adequate time-horizon. The governance objectives should be supported by *"Usefulness" and "Efficiency"* measures and other risk criteria for improving business processes and management activities within the enterprise governance framework.

The Governance Model provides ISO/IEC 15504 conformant process descriptions and application practices for providing assurance over trusted and sustainable business operation.

The following three governance processes are defined related to business sustainability objectives:

- *Competitive Operation* – Ensuring market recognition of the business operation.
- *Exploitable Operation* – Organization realizes optimal value from business operation.
- *Satisfactory Operation* – Ensuring user/customer satisfaction based on agreed levels of business operation.

Eight processes are defined related to the internal control objectives:

- *Control Risks* – The organization and its staff adequately address risks to the governance objectives relevant for financial reporting and trusted business operation and consider those risks in management of business operation.
- *Control Management* – The management of the organization is able to control business processes in a way which is adequate to the objectives of internal control over financial reporting and trusted business operation.
- *Control Competence* – Sufficient skills and knowledge relevant for the objectives of internal control over financial reporting and trusted business operation are available and used.
- *Information Reliability* – Data architecture and disclosure elements relevant for financial reporting objectives and trusted business operation, and for supporting data processing integrity are accurate and consistent.
- *Process Control* – Design and operation of process-level controls relevant to the objectives of financial reporting and trusted business operation, and processing integrity principle are effective.
- *Data Protection* – The organization and its staff are committed to security, confidentiality and privacy principles to avoid unauthorized access to and misuse of confidential data effected by business operation.

- *Integrity Assurance* – The organization and its staff are committed to comply with ethical and business integrity requirements relevant to the objectives of financial reporting and trusted business operation, and availability principle.
- *Control Efficiency* – Efficient usage of control resources relevant to the objectives of financial reporting and trusted business operation.

ISO/IEC 15504 Capability Levels and related Process Attributes can be applied as qualitative and quantitative measures for setting affordable enterprise specific requirements (risk appetite) relevant for achieving the business goals within a tolerable deviation (risk tolerance).

3.   Implementing Enterprise Goals driven Integrated Risk Management scenarios

The term of *"scenario"* is used for systematically considering integrated risk management aspects of governance practices implemented at given operational or organizational level. These considerations are focusing on the customized design of governance objectives, processes and practices enabling the achievement of the concerning enterprise goals, and the presentation of related risk criteria as management assertions (or risk appetite statements). Based on the evaluation of how these risk criteria are fulfilled, the next improvement or correcting actions (risk treatments) are planned and performed. The ISO 31000 standard describes applicable requirements for these risk management activities at organizational and process levels.
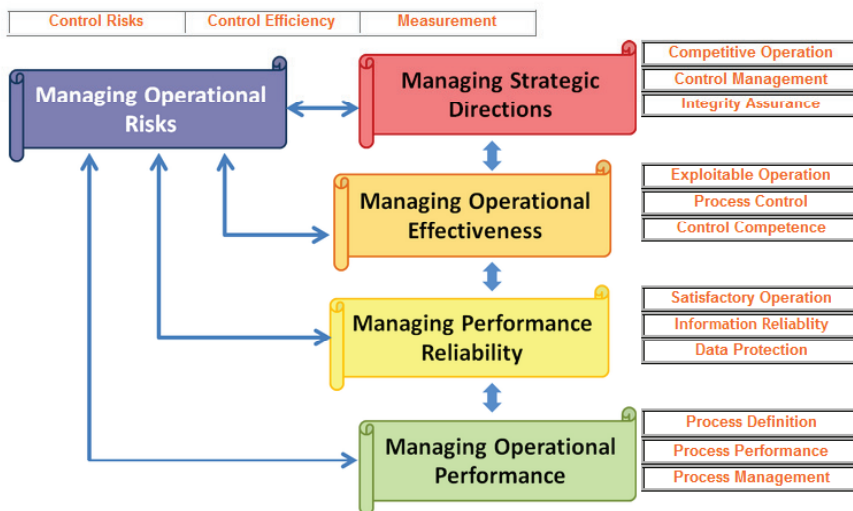
Figure 2: Enterprise Goals driven Integrated Risk Management Scenarios

The suggested solution describes how **Integrated Risk Management scenarios** are established by mapping already implemented or newly developed management practices to governance objectives - through company specific enterprise goals. By this way also the compliance and assurance works will be aligned with the enterprise specific business objectives and might keep the less meaningful elements of general governance or control frameworks out of scope. However by comparing existing practices to those offered by these frameworks, the management and - if requested due to company size or corporate laws - the supervision bodies might benefit from getting wider professional knowledge and best practice suggestions for improving enterprise governance.

The proposed Integrated Risk Management scenarios are distinguishing different operational and organizational levels having specific targets and time-horizons. At each level "Usefulness" and "Efficiency"

goals and measures should be defined allowing management to see recognized professional framework practices as enablers instead of just compliance requirements.

Operational Performance is related to the core and supporting business processes of a business unit. The processes might be described by using different methodology and tools; however the *process purpose* and the *necessary and sufficient outcomes* of achieving this purpose are generally identifiable. Each specific business operation consists of a set of interrelated business processes with allocated resources, specified product or service delivery requirements and schedules. Managing Operational Performance scenario is focusing on achievement of these - relatively - short term performance objectives, for example a unique product or service delivery based on a specific client's order. Most regulatory requirements (like health protection, safety, human rights, technical or accounting standards, etc.) might be also incorporated within activity goals and assured at this level.

Performance Reliability also refers to the above operational level with extended focus on additional aspects of performance. For repeating, parallel or extended cycles of operational processes, the operational management should establish longer term objectives such as customer retention and capacity utilization. At most cases these objectives are related to contractual or pay-off periods. For example customer satisfaction and capacity utilization rates are applicable measures for reoccurring business transactions for the monthly pay-off period of an outsourcing service.

Such as the Operational Performance instances drive the achievement of reliability objectives, the pay-off cycles based Reliable Performance drives to achieve entity (business unit) level *Operational Effectiveness* goals measured by profitability and agile resource allocation at business unit level for a quarterly or yearly reporting period. The business unit level effectiveness is also a driver to achieve objectives set by *Strategic Directions (Business Goals)*, like revenue targets and operational cash flow positions set for the strategic planning periods.

Managing Operational Risks sets risk tolerances (acceptable deviation from objectives) and risk appetites (affordable levels of uncertainties effecting objectives) for operational and organizational levels based on operational performance, reliability, effectiveness and strategic objectives. Each level's objectives have specific time-horizons, therefore the application of "traditional" consequence and probability metrics ("heat maps") for risk ratings and selecting or prioritizing the risk treatment options is reasonable only when operational or organizational levels and timescales of risk events are comparable.

Risk Management practices might show significant differences in details at SME or bigger company cases; however the same principles remain valid. Evidently a small entity or business unit also defines acceptable tolerance levels of its business targets, and establishes its governance structure adequately to affordable levels of internal and external uncertainties affecting these targets. Practically "affordable level" is different for a smaller entity with a few service or production lines than for a big multinational company with much more diversified activities.

The proposed Integrated Risk Management scenarios are applicable for all types of business entities and they use generic purpose governance frameworks for selecting those processes and practices which enable their business operations to achieve enterprise goals at adequately defined operational and organizational levels.

4.   Applying Governance SPICE Assessor Skills

For implementing Enterprise Governance the executive management and - if it exists - the supervisory board should follow scenarios to evaluate, direct and monitor business operation in alignment with the adapted governance objectives. In this term the "*Enterprise Governance*" is driven by the organization's specific business goals and enabling governance objectives instead of generic control or regulatory framework based "checklists". When ISO/IEC 15504 standard (SPICE) based **Governance Capability Assessment** [5] concept is applied, the evaluation of compliance will focus on how the capability profiles of the implemented core business and governance processes are aligned with the governance objectives customized for the specific enterprise goals. This customization keeps in mind three dimensions:

- the business operation (processes and activities) under scope,

- the applicable governance practices from recognized reference models and
- the capability level targets.

The governance processes defined by the Governance Model for Trusted Businesses are supported by selected processes from the **COSO**, **COBIT** [6] and **Enterprise SPICE** [7] reference models. These application areas associated with the process attributes defined in ISO/IEC 15504-2 provide a common basis for performing assessments of governance capability regarding Enterprise Governance and reporting of results by using a common rating scale. ISO/IEC 15504 (SPICE) offers not only transparent method for assessing performance of relevant governance processes, but also tools for assessing related risk areas based on the gaps between target and assessed capability profiles.

However traditional compliance-driven approaches have been facing to major problem as there is no evidence that compliance (to any model) really drives business success. On the contrary: all big failure companies of the last decades had been "equipped with" long list of compliance and excellence records for many years. The key problem is that managing compliance issues has only limited focus on lower level outcomes - like activity goals - without considering the overall success factors. Enterprise Governance should focus on wider internal and external contexts of risks defined as effects of uncertainties on enterprise objectives as referred by the *ISO 31000 Risk Management standard*. *Quantitative Performance Measurement* covering the overall governance structure is needed for establishing useful risk criteria for supporting management decisions at all organizational and operational levels.

Most of the metrics applied by Quantitative Performance Measurement, like those related to "Usefulness" and "Efficiency" *generic attributes* [8], are typically not interpretable for the ISO/IEC 15504 process capability levels. These metrics are applicable in business context of the processes by providing tool for defining and/or adapting economically meaningful base practices as process performance (level 1) indicators. There is no meaning to establish such metrics for the generic practices of higher capability level Process Attributes; however highlighting of the generic attribute metrics of those business or governance practices, which are identified enablers of higher capability level practices for all processes within the scope of the process assessment model, is more than reasonable. Those base practices adapted from control frameworks or reference models as performance (level 1) indicators of a governance process, are applicable to determine risk appetite at operational and organizational levels.

The **Governance Model for Trusted Businesses** and **Governance Capability Assessment** help the executive management and - if it exists - the board to look at compliance issues through customized governance objectives aligned with enterprise specific business goals and stakeholders' expectations. The Governance Model provides process descriptions and applicable practices for setting risk criteria over Enterprise Governance assuring achievement of specific enterprise goals according to stakeholders' needs and expectations. The **Governance SPICE Assessor** skills [9] are required to evaluate these management assertions (risk appetite statements) established by Integrated Risk Management scenarios implemented at different organizational and operational levels.

By using the terminology outlined in the ECQA skills definition model, the skills hierarchy for the job role "Governance SPICE Assessor" has been designed. The skill units and elements cover the relevant "Governance" domain specific knowledge (Governance, Risk and Controls), the principles of the Governance Model for Trusted Businesses (Governance Objectives), the basics of SPICE (Process Assessment) and the mapping of capability levels with Compliance, Reporting, Operations and Strategic objectives (Governance Capability). Next figure also presents the detailed list of the learning elements:
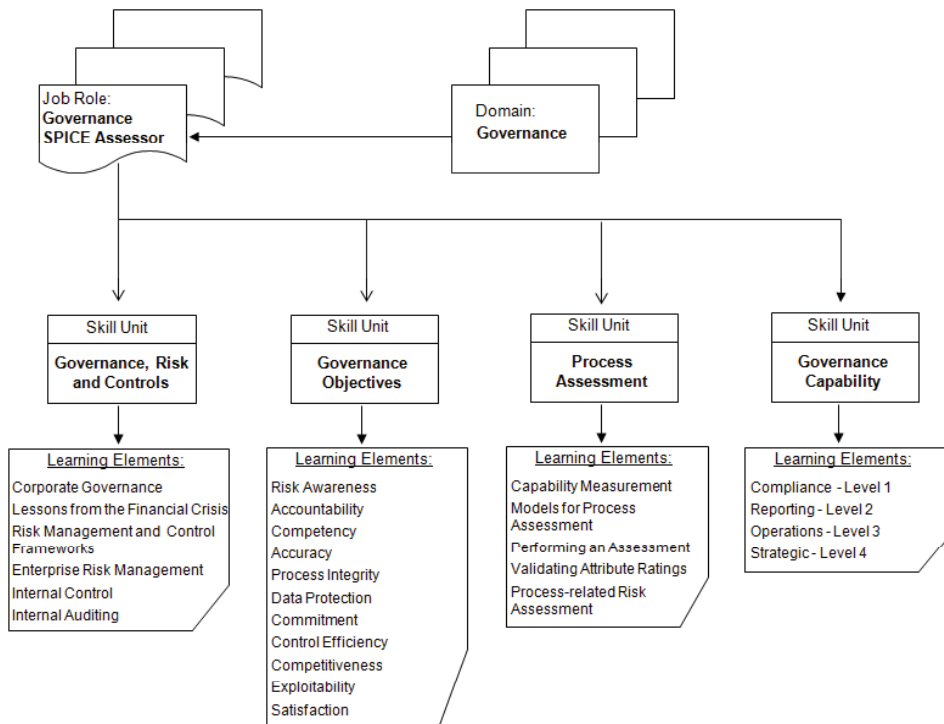
Figure 3: The ECQA certified Governance SPICE Assessor Skill Card

5. Conclusions

The **Integrated Risk Management scenarios** present how even local small business organization can efficiently implement compliant governance/control frameworks with respect of its real business needs and risks, and how the implementation results can be exhibited for external evaluation or audit in a cost effective way. Even in the case of SMEs the business environment might have internationally standardized (like **SOC 1** [10] and **SOC 2** [11]) requirements which should be carefully considered by small business companies providing local services to big firms or multinational clients, whose compliance managers, internal and external auditors are making great demands on local service providers and raising difficulties for these companies by increasing requested control and audit efforts and costs. At most cases these demands are driven by the multinational organizations' global compliance or audit requirements, so they are not really intended to be "customized" for local conditions.

By changing from the traditional *model based compliance* to *enterprise goals driven integrated risk management*, the management assertions (the links between business activities and governance practices) are implemented by applying significantly different scoping approach. The key business processes are viewed as instances of business performance at different operational and organizational levels, so the Integrated Risk Management scenarios are enhancing the meaning of "compliance" as *in what extent the model based governance/control practices are relevant for really supporting the achievement of enterprise goals within their acceptable tolerance levels*. The proposed Integrated Risk Management scenarios help to select and apply model based control practices by considering the operational and organizational performance levels and their adequate time-horizons for setting enterprise objectives. The term of "*Integrated Risk Management*" also refers to how the Governance Capability Assessment model is adapted, understood and used by the assurance providers of all organizational and operational levels, including the

oversight board (if exits), the executive and line management, the internal and external auditors, and other roles relevant in governance, risk management, control system and compliance related works.

Capability profiles of the business processes together with the enabling governance and control processes are representing "reverse", but well understandable measures of *management's risk appetite* as the higher capability levels indicate the more robust risk treatment for achieving relevant business objectives.

For all the 11 governance objectives the **Governance Model for Trusted Businesses** provides application practices with reference to governance processes offered by recognized reference models (COSO, COBIT, Enterprise SPICE) or generally accepted (e.g. privacy) principles. The full coverage of the governance objectives related processes - by implementing the **Integrated Risk Management** scenarios lets enterprises to qualify their business units. The qualification process of a business unit's **compliance to its unique governance objectives** - defined by the specific scoping of the governance practices from the Governance Model for Trusted Businesses - should cover all those business processes and information sources, which provide the sufficient evidences for management assertions (risk appetite statements) concerning to the effective and efficient implementation of enterprise risk management scenarios.

ISO/IEC 15504 process capability assessments (or similar audit approaches) are widely used in specific industries and sectors, like automotive, medical, space, finance, etc. Most of these assessments are performed only at operational levels aiming up to level 2 targets by using domain specific process assessment models adapting generic standards or recommendations, like ISO 12207, ITIL, COBIT, etc. The coverage of the 11 governance objectives referred by the enabling processes of the **Governance Model for Trusted Businesses** helps to use the industry and sector specific process assessment models by establishing the applicable organizational contexts of level 3 and level 4 process attributes concerning to the operational and supporting business processes.

Professionals having been acquiring and evidencing their **Governance SPICE Assessor** skills are able to provide unique consulting and assurance services for supporting enterprises in achieving well established business goals and targets at an affordable level of risk treatment costs and effect of uncertainties by assessing and evaluating enterprise governance processes.

6.   References

[1]   Governance Model for Trusted Businesses, BPM-GOSPEL Deliverable, 2011

[2]   ISO 31000:2009, Risk management – Principles and guidelines
       ISO Guide 73:2009, Risk management - Vocabulary

[3]   The Committee of Sponsoring Organizations of the Treadway Commission (COSO):
       • Internal Control - Integrated Framework (1992, 2013)
       • Enterprise Risk Management - Integrated Framework (2004)
       • Internal Control over Financial Reporting - Guidance for Smaller Public Companies (2006)
       • Enterprise Risk Management - Understanding and Communicating Risk Appetite (2012)

[4]   ISO/IEC 15504-1:2004 Information technology -- Process assessment -- Part 1: Concepts and vocabulary
       ISO/IEC 15504-2:2003 Information technology -- Process assessment -- Part 2: Performing an assessment
       ISO/IEC 15504-2:2003/Cor 1:2004
       ISO/IEC 15504-3:2004 Information technology -- Process assessment -- Part 3: Guidance on performing an assessment
       ISO/IEC 15504-4:2004 Information technology -- Process assessment -- Part 4: Guidance on use for process improvement and process capability determination
       ISO/IEC TR 15504-7:2008 Information technology -- Process assessment -- Part 7: Assessment of organizational maturity

[5]   J. Ivanyos, J. Roóz and R. Messnarz, Governance Capability Assessment: Using ISO/IEC 15504 for Internal Financial Controls and IT Management, in: The MONTIFIC Book, MONTIFIC-ECQA Joint Conference Proceedings, 2010

[6]   COBIT - Control Objectives for Information and related Technology, COBIT 4.1 © 2007 and COBIT 5 © 2012, IT Governance Institute

[7]   Enterprise SPICE® - An Integrated Model for Enterprise-wide Assessment and Improvement. Technical Report – Issue 1 September 2010. Copyright © The SPICE User Group 2010

[8]   C. Wells, L. Ibrahim and L. LaBruyere, New Approach to Generic Attributes, Systems Engineering, Vol. 6, No. 4, 2003. © 2003 Wiley Periodicals, Inc.

[9]   Governance SPICE Assessor and Internal Financial Control Assessor Skill Cards, ECQA Committees, 2011

[10]   Statement on Standards for Attestation Engagements (SSAE) No. 16, Reporting on Controls at a Service Organization. Copyright © 2010 American Institute of Certified Public Accountants, Inc. New York, NY 10036-8775

[11]   Reporting on Controls at a Service Organization Relevant to Security, Availability, Processing Integrity, Confidentiality, or Privacy (SOC 2). Copyright © 2011, American Institute of Certified Public Accountants, Inc. All Rights Reserved.

Leadership in Sustainability

Gabriela Fistis[1], Tomislav Rozman[2], Andreas Riel[3], Richard Messnarz[4]

[1]Universitatea Politehnica Timisoara, Romania
gabriela.fistis@denkstatt.ro
[2]DOBA Faculty Maribor, Slovenia
tomislav.rozman@bicero.com
[3]EMIRAcle c/o Grenoble Institute of Technology GSCOP UMR5272, France
andreas.riel@emiracle.eu
[4]ISCN LTD/GesmbH, Austria & Ireland
rmess@iscn.com

Abstract

In the last decades the innovation in engineering has been focused on producing reliable products and services. Also the international standards for work place safety, machine directive for safety and functional safety of a product have been considered. Usually such paradigm shifts in engineering mean that you must develop functions and services in these fields to sustain your leadership on the market. A new development in the last 10 years is the growing importance of social responsibility (Messnarz, 2014) based on the new published ISO 26000 standard (ISO 26000, 2010) for social responsibility. Based on that growing social awareness of industry and society new functions, features and services are developing which will form a large part of innovation in the next decade. This paper gives some outlook into that future.

Keywords

Fuzzy Front-End of Innovations, Idea Generation, Idea Management, Process Development, Innovation Management, Automotive Supplier Industry

Reference

Project valorisation through agility and catering for stakeholder expectations

Elli Georgiadou[1], Kerstin Siakas[2], Richard Messnarz[3]

[1] Middlesex University, School of Science and Technology, and SEEQMA Ltd, UK e.georgiadou@mdx.uk
[2] Alexander Technological Educational Institute of Thessaloniki, Greece
siaka@it.teithe.gr
[3] ISCN, Austria, rmess@iscn.com

Abstract

Project valorisation is paramount for gaining value in an increasingly competitive world. There is evidence that the majority of projects even when they are completed within budget and time fail to valorise (disseminate and exploit) their results so that they can deliver value to the organisation. Projects often have many stakeholders with different requirements and expectations. Identifying and understanding synergies, conflicts and changing requirements hold the key to project success. In this paper we discuss the challenges and failures of lack of valorisation from industry, government, academia and the European Union. Using the VALO project we demonstrate how the integration of the project plan, the quality plan and the sustainability plan started delivering value to a multiplicity of stakeholders throughout the project lifetime and beyond its completion. We propose a meta-framework for this integration taking into account the process maturity of an organisation for successful valorisation of projects.

Keywords

Valorisation, stakeholder expectations, agility

Reference

Going from the Idea to Enterprise in 4 EU Countries

Eva Homolová[1], Andreas Riel[2], Marek Gavenda[1], Ana Azevedo[3], Marisa Pais[3],
Jiří Balcar[1], Alessandra Antinori[4], Giuseppe Metitiero[4], Giorgos Giorgakis[5], Photis Photiades[5], Damjan Ekert[6], Richard
Messnarz[6], Serge Tichkiewitch[2]

[1] RPiC-VIP, Ostrava, Czech Republic
{gavenda, homolova, balcar}@rpic-vip.cz
[2] EMIRAcle c/o Grenoble Institute of Technology GSCOP UMR5272, Grenoble, France,
{andreas.riel, serge.tichkiewitch@grenoble-inp.fr},
[3] ISQ, Lisbon, Portugal
{aiazevedo, marisa.pais}@isq.pt
[4] CIRSES, Rome, Italy
ale.antinori@ale.antinori@cirses.it, g.metitiero@primaforma.net
[5] EUROSC, Nicosia, Cyprus
{george, photis}@eurosc.eu
[6] ISCN GmbH, Graz, Austria
{dekert, rmess}@iscn.com

Abstract

Innovation and entrepreneurship are among the top priority areas of the European Union in order to exit the economical crisis and assure sustainable and profitable growth and competitiveness on a global level. Although more and more entrepreneurship training and education programs exist in different EU countries, there is little cross-country cooperation and complementary among these activities. This paper introduces a European approach to a certified entrepreneurship training program that has been established in a consortium of several European training and education organisations. This program has been implemented around the long-term mission of empowering people to make ideas become real in the European context.

Keywords

Innovation, Entrepreneurship, Lifelong Learning

Reference

A Measurement Framework for a Transparent Certification using Structured Assurance Cases

Xabier Larrucea

Tecnalia. Parque tecnológico de Bizkaia 202. 48170 Bizkaia, Spain.
Xabier.larrucea@tecnalia.com

Abstract

Assurance cases are used in certification processes for developing safety critical applications. Each assurance case is composed by arguments and evidences that are used in order to demonstrate that a product or process it is safe. Safety critical products rely on this kind of judgments and assumptions. If we want to certify or to assess a system with respect to a reference model we can use the related assurance cases. This paper proposes a framework containing a set of metrics related to assurance cases in order to control its compliancy. These metrics make explicit a certification status based on arguments and evidences. The concept of transparency is introduced and it is related with the ability of providing a clear status and knowledge of an on-going certification process. In addition this paper describes briefly the implemented approach.

Keywords

Multimodel framework, quality models, process improvement

1 Introduction

While software engineering has quite well defined metrics [1] allowing to assess a software engineering project, such metrics are not well defined for e.g. safety engineering. However, we can reuse metrics concepts from the software domain as there is some similarity [2]. Software measurement methods [3] have been discussed in literature and they are required when putting into practice an improvement program, a software development process, an assurance infrastructure or just to provide a feedback [4] amongst others. One of the key elements is the collection of specific data [5] in order to provide a diagnosis about a system whether critical or not. This diagnosis is based on quantitative and qualitative attributes, and all of them represent some of the building blocks of a framework. This measurement framework is not a straightforward activity, and in fact: *"For safety indicators with their long-term effects, diagnosis is even more difficult"* [6].

Therefore there is a need for setting a measurement framework from a general point of view in safety critical systems such as in [7] where authors define a framework for measuring software safety risks. This is especially relevant during the certification process of safety critical systems [8]. Thus it is relevant to identify what are the most important elements in a certification process[9]. These elements will provide transparency to the certification process.

The main objective of this paper is to resolve the following research questions (RQ):

- RQ1. Can make explicit the certification process in safety critical contexts?
- RQ2. How can we make it explicit in assurance cases?
- RQ3: Is there any appropriate framework relating assurance cases and metrics from a metamodel point of view?

This paper is structured as follows. First an overview of the SACM metamodel and an overview of measurement frameworks are described. Second a conceptual framework for transparency certification is provided. Third section includes a description of its implementation. And finally a conclusion section ends this paper.

2 Research background

2.1 Structured Assurance Case Metamodel

Structure Assurance Case Metamodel released in February 2013 [10] is a notation for assurance cases widely used in safety critical systems. The main purpose of this metamodel is to make clearer claims and reasoning between claims. Additionally an assurance case is structured with arguments and evidences.
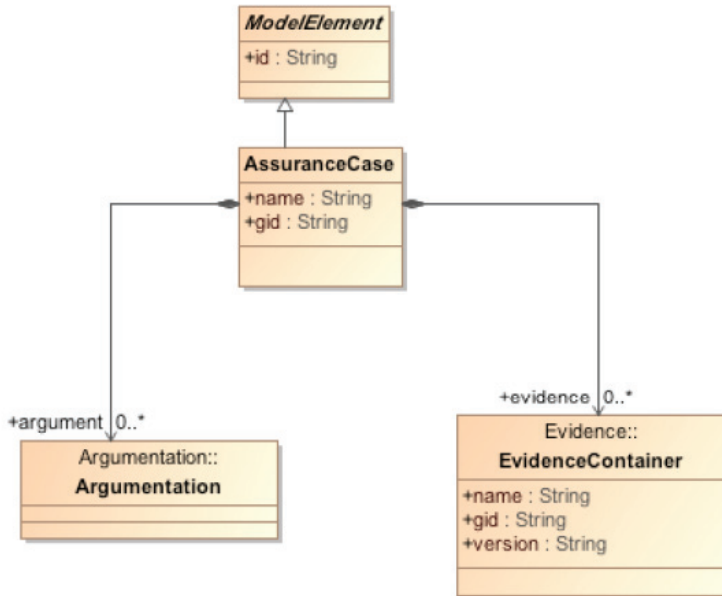
Figure 1: Assurance case metamodel administration diagram

Assurances cases have a main role when developing safety critical applications [11] because they are considered the main proofs for considering an artifact safe. In addition assurance cases have a clear role in certification processes [12]. Figure 1 describes, from a general point of view, the basic elements in an assurance case which inherit from a model element. This assurance case used to contain a set of argumentations and evidences validating safety goals.

2.2   Measurement framework

There are different measurement frameworks such as [13] where authors define precisely software measurements concepts and a life cycle. From a more general point of view, a Goal/Question/Metric approach[4], [14], or a Quality Function Deployment approach [15] have been used to organize approaches in the context of quality assurance. Both approaches consider using metrics, and these metrics should include quantitative as well as qualitative aspects.

From a qualitative point of view, safety critical systems includes a wide variety of options and conditions to be taken into account [16]–[20] in order to assure the safety of a system. Normally, these qualitative criteria are related to safety cases concepts [21], and different solutions include approaches such as MDE [20] or petrinets [22].

From a quantitative point of view, measurable metrics have been used in the context of safety critical systems [23]. In this sense we can point out which metrics are the most appropriate and related to the transparency of the certification process.

The notion of "Measurement" is the relationship between a measure and a "measurand" [24]. In our conceptual framework the "measurand" will be an element defined in an assurance case. "Measure" is a term referring to "metric" and provides a method to calculate a value with a set of potential constraints. The assigned value can be numerical or not depending on the domain and on the type of

measure. This value can be in a range of potential values. Each measure has a set of measurements. These measurements are calculated automatically or manually by using human interactions. Ideally an automated support [25] will be more appropriate. However we consider that users should be able to interact in order to calculate the appropriate values. On this process, users can make use of external tools. The measurement process as it is defined by the IEEE [26] they distinguish the following elements:

- *Indicator*: measure that provides an estimate or evaluation of specified attributes derived from a model with respect to defined information needs

- *Base measure*: measure defined in terms of an attribute and the method for quantifying it. A base measure is functionally independent of other measures.

- *Derived measure*: measure that is defined as a function of two or more values of base measures
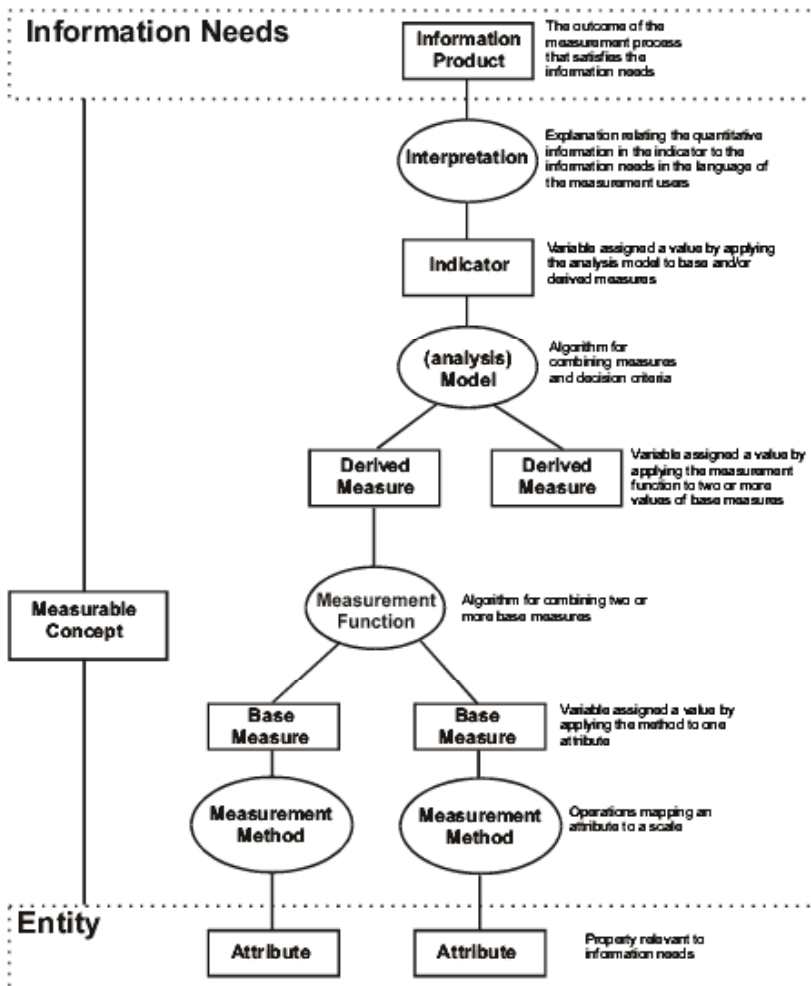
Figure 2: Elements involved in a measurable model [26]

These three main elements are defined as a kind of measure. In addition we are aware of the current misinterpretation of terms [27], and some stakeholders use the term "indicator" instead of "metric" or "measure"[28]. Relevant industrial efforts such as Structured Metrics Metamodel (SMM) [24] use "measure" term for referring to metrics and indicators. Therefore from a conceptual point of view, we are going to use the concept measure for representing any kind of measure.

In our case, a measurement process and framework [29] should be defined for safety-critical systems. This is especially relevant for achieving transparency across certification processes whereby we need to have a clear idea about the system's status at any point time in time.

2.3 Why do we need to measure?

Any human-intensive activity, without control, deteriorates over time. It takes constant attention and discipline to keep software acquisition and development processes from breaking down. If one does not measure, there is no way to know whether processes are on track or whether they are improving. Measurement provides a way to assess the status of the project to determine if it is in trouble or in need of corrective action and process improvement. This assessment must be based on up-to-date measures that reflect current program status, both in relation to the project's plan and to models of expected performance drawn from historical data of similar projects. If, through measurement, the project is diagnosed to be in trouble, on should be able take meaningful, effective remedial action (e.g., controlling requirements changes, improving response time to the developer, relaxing perform-ance requirements, extending the schedule, adding financial and other resources, or any number of options).

Measurement provides benefits at the strategic, program, and technical levels. A good measurement program is an investment in success by facilitating early detection of problems, and by providing quan-titative clarification of critical development issues. Metrics give the ability to identify, resolve, and/or curtail risk issues before they surface. In our certification process context measurements provide us a way to identify and evaluate the status of an assurance case assessment.

2.4 Approaches for metrics management

Ariane V launch failure, the losses of the Mars Polar Lander, the Mars Climate Orbiter, are some of the examples mentioned by John Knight in one of his seminal papers [30]. As mentioned [30] reliability is one of the major concerns for this kind of software-intensive systems but it is not enough for assur-ing safety in critical systems. Other factors should be considered such as defective software specifica-tions, software engineering and systems engineering interplay, development time and effort, and so forth. Therefore all these systems require setting up techniques for measuring and estimating failures. Its relevance is especially important for safety critical systems where "metrics are not always precisely defined, limiting the reproducibility of results, and lacking a directional quality" [31].

Therefore in our safety-critical context, we need to define an appropriate measurement framework[13] for not only managing metrics for assuring safety but especially metrics related to certification. Ideally, this topic covers a wide set of approaches and complex implementations [32]. A certification process is normally related to the certification of products, processes and personnel [33]. Our approach is fo-cused on products and processes, and a certification process in safety environments [12] is not a straightforward activity. We need to define and use a set of metrics or criteria appropriately. In fact our aim is to concretize a set of services in a measurement framework for metrics creation, configuration and maintenance.

Traditionally complexity metrics [34] can be divided on "static metrics" which measurements are taken at one particular point in time, and "history metrics" which include a set of measurements. Our safety related metrics are applied to products or/and processes from a general point of view [34], [4], and from a safety critical systems point of view [35]. On the process side, process metrics can have auto-mated support [25] such as complexity metrics in safety critical systems [36].

Metrics can be classified into three categories: product metrics, process metrics, and project metrics [1]. As stated previously and aligned to [33], our approach for certification process will be based on products and processes that both are included in an assurance case. Other metrics such as size, complexity, design features, performance and activity duration among others are some examples of metrics [37]. However these metrics are not defined for assurance cases.

All these metrics should be shown through an appropriate dashboard. Part of the existing approaches on metrics management such as FMEA approach [2], can be used for visualising metrics and measurements. Implementing a software metrics program such as in Nokia [38] it is not an easy task, and we even need to take into account architectural considerations on certification [39], and included in assurance cases.

3 Conceptual framework for achieving transparency

This section presents a conceptual framework for achieving transparency during the certification process in safety critical contexts. This framework enables users to define, use and relate metrics to assurance cases' artefacts used during a certification process. The following questions are some of the foundations for developing the proposed framework:

- RQ1: Can make explicit the certification process in safety critical contexts?
- RQ2. How can we make it explicit in assurance cases?
- RQ3: Is there any appropriate framework relating assurance cases and metrics from a metamodel point of view?

These questions require a deeper analysis on the type of information that it is required by each stakeholder to achieve a level of transparency and trust during the certification process. All this information leads us to identify the appropriate languages and metamodels to describe assurance cases and metrics, and at the same time to identify some metrics related to assurance cases for measuring the certification process.

3.1 Transparent Certification Approach

Transparency in the certification process[1] is needed in order to provide a clear status and knowledge of an on-going certification process. Some of the current problems encountered during process assessments or certification are amongst others [40]:

- auditors have a partial view of the certification process;
- a lack of objective criteria for determining the project's status
- a fuzzy or unclear assessment method with respect to a wide variety of standards[2].

From a naïve approach this certification process includes elements such as what it has been verified and validated versus what it is required to be verified and validated for an identified reference model. However a certification process is not so easy and auditors should verify all elements described in a product's associated assurance cases, and they require a defined framework.

3.2 Metrics Conceptual Framework

The approach is based on the current SMM metamodel [24] for representing and storing metrics.

---

[1] http://www.tuev-sued.de/uploads/images/1156764217583933590252/klima_goldstandard_e.pdf
[2] http://www.emci.nl/index.php/id_structuur/10593/general-information.html

Figure 3 represents conceptually the measurement and transparency package and highlights some interfaces for providing metrics information. IMetricsInfo is an interface for capturing the description of each metric. IMetric is an interface for managing metrics and provides a CRUD (Create, Read, Update, Delete) basic functionality. UIMetrics represents its user interface.
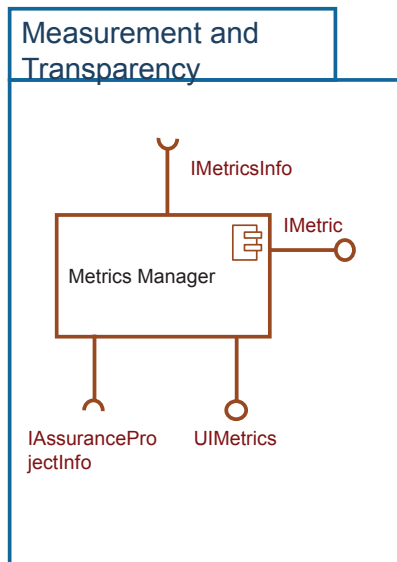
Figure 3: Measurement and transparency package

Figure 4 represents the four main conceptual classes to be considered and implemented. MetricLibrary shall contain a set of predefined metrics and measurements to be used and to be related to artefacts/evidences for transparent certification. MetricEngine shall provide the means for calculating values to each metric and controlling metrics at any stage. MetricInfo implements the interfaces for showing metrics and measurements. AssuranceProject contains information about a safety critical certification process.
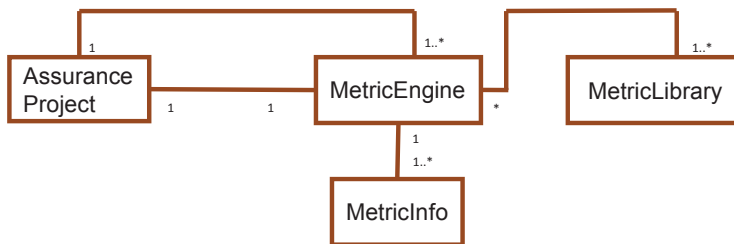
Figure 4: Conceptual classes and relationships among them

As our approach is based on SMM and Figure 5 describes SMM, SACM and Metrics manager elements relationships. It is worthy to note the role of the metric engine element which manages a set of metrics library, the relationship with SACM elements and it is able to manage the progress of the certification process.

Metrics Manager
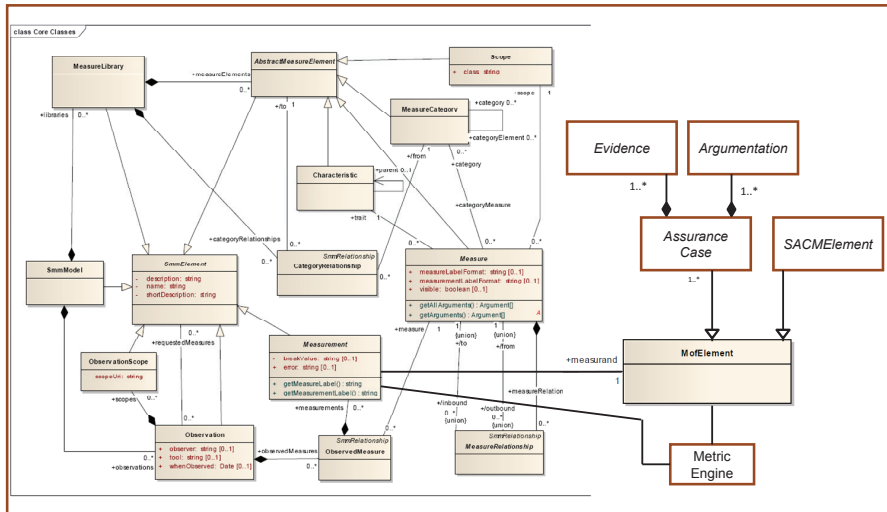
Figure 5: SMM, SACM and Metrics manager elements relationships

3.3 Structured assurance cases based metrics

Table 1 describes a set of predefined metrics associated to assurances cases in order to manage and control a certification process based on assurance cases. As stated previously evidences and arguments are the SACM elements.

Table 1. Structured assurance cases metrics for certification

Name	Description	SACM Elements
Number of Evidences	Number of evidences identified in an assurance case	Evidence
Number of Claims	Number of claims identified in an assurance case	Argumentation
Number of Assurance Cases	Number of assurance cases	Assurance Case
Density of Arguments	Ratio of arguments required by a goal	Argumentation
Density of Evidences	Ratio of Evidences required by a goal	Evidences
Density of Contexts	Ratio of Contexts required by a goal	Argumentation
Number of Contexts per Assurance Case	Number of Contexts per Assurance Case	Argumentation
Number of justifications	Number of justifications per claim	Argumentation
% of Identified Evidences	Percentage of Evidences identified versus the pending ones	Evidences

% of proved Claims	Percentage of Claims validated versus those that are pending to be validated	Argumentation
Number of Incomplete evidences	Number of evidences that should be identified their appropriate instance	Evidences
% of inappropriate evidences	Percentage of evidences which instance it is not appropriately identified	Evidences
Number of generic goals	Number of pending goals or subgoals[41]	Argumentation

4 Implementation

We have implemented the aforementioned framework as a Web based tool. The main purpose of this section is not to provide an exhaustive description of the current implementation. On the contrary we would like to show the main pieces of our current implementation for supporting our framework.

We have taken as a reference model the ISO26262 for the automotive domain. This reference model does not contain nor require a certification process. Instead stakeholders and the automotive domain are taking it as a reference and they are defining their own conformity process.

Figure 6 provides an overview of our tool for defining a set of metrics related to safety cases. These metrics depend on human tasks and it is associated to specific tasks of the ISO26262. These metrics are the basic information for the certification process.
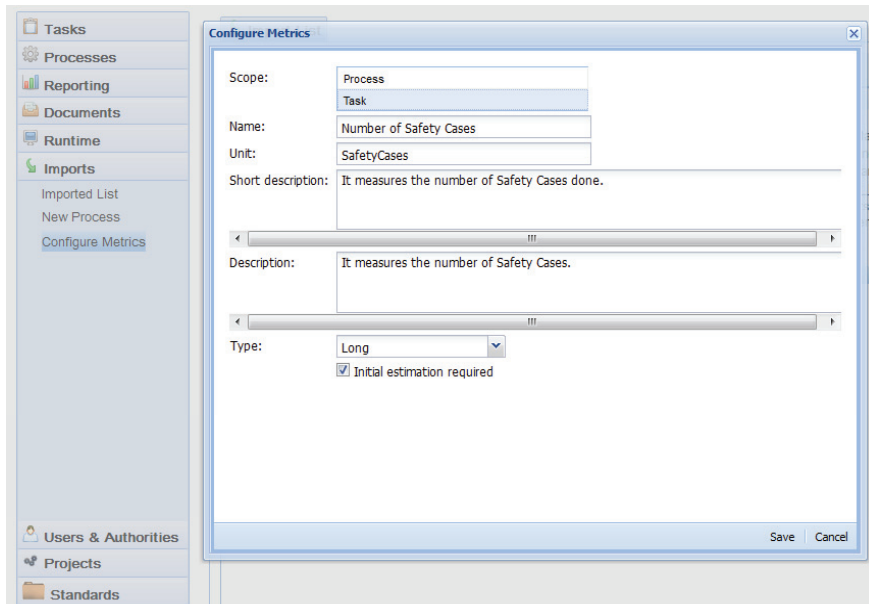
Figure 6: Configuring a metric related to safety cases

Our tool provides a predefined set of metrics included in a library shown in Figure 7 and described

previously in our conceptual framework (Figure 4). This tool does not just include our structured assurance cases metrics for certification. It includes also generic metrics such as Lines Of Code because this framework aims to be used as generic as possible.

Figure 7: Metrics library

All metrics for a specific process instance is managed from a scorecard (Figure 8). Each instance of the process contains different values for each measurement. This is a key aspect to evaluate processes behavior based on historical values. This functionality is going to be further elaborated in a future work.

Figure 8: Metrics scorecard

5 Conclusions

This paper is focused on the following research questions:

- RQ1: Can make explicit the certification process in safety critical contexts?
- RQ2. How can we make it explicit in assurance cases?
- RQ3: Is there any appropriate framework relating assurance cases and metrics from a metamodel point of view?

Assurance cases are used in certification processes in safety critical contexts. Each assurance case is composed by arguments and evidences. Once we have a well formed and complete assurance case we can argue that a specific part of a product or process it is safe. Safety critical products rely on this kind of judgments and assumptions. If we want to certify or to assess a system with respect to a reference model we can use the related assurance cases. This paper proposes a set of metrics related to assurance cases in order to control its compliancy. These metrics are the basis for the developed framework. Therefore using metrics are used to make explicit a certification status based on arguments and evidences. We have implemented at a metamodel level a set of relationships amongst these elements and it provides explicit means for controlling a certification process using assurance cases.

Our implementation is based on two standards from the OMG (Object Management Group): SMM for describing metrics and SACM for describing assurance cases. We have been using a standard approach in order to provide interoperability among different tools.

The concept of transparency has been introduced in this paper and it is related with the ability of providing a clear status and knowledge of an on-going certification process. As stated this approach has been developed for safety critical systems, but it can be discussed to be also applied to non-safety critical software applications. From a certification point of view, auditors or assessors required arguments and evidences for fulfilling with the reference models requirements. These metrics are defined taking into account these elements, so we can use them as a future work.

Currently we are applying it to an industrial scenario based on ISO26262 in order to refine these metrics. Additionally this work is planned to be extended with a COSMIC approach for Functional User Requirements3 approach identifying the appropriate metrics.

Acknowledgement

6 Literature

1. S. Kan, Metrics and Models in Software Quality Engineering , Second Edition. Addison Wesley, 2002, p. 560.
2. I. Bicchierai, G. Bucci, C. Nocentini, and E. Vicario, "Integrating metrics in an ontological framework supporting SW-FMEA," 2012 3rd Int. Work. Emerg. Trends Softw. Metrics, pp. 35–41, Jun. 2012.
3. J. Roche and M. Jackson, "Software measurement methods: recipes for success?," Inf. Softw. Technol., vol. 36, no. 3, pp. 173–189, Mar. 1994.
4. V. R. Basili, "Software modeling and measurement: the goal/question/metric paradigm," College Park, MD, USA, 1992.
5. V. R. Basili and D. M. Weiss, "A Methodology for Collecting Valid Software Engineering Data," IEEE Trans. Softw. Eng., vol. SE-10, no. 6, pp. 728–738, Nov. 1984.

---

[3] http://www.cosmicon.com/portal/public/COSMIC%20Method%20v3.0.1%20Measurement%20Manual.pdf

6. H. J. Pasman, B. Knegtering, and W. J. Rogers, "A holistic approach to control process safety risks: Possible ways forward," Reliab. Eng. Syst. Saf., vol. 117, pp. 21–29, Sep. 2013.

7. L. Layman, V. R. Basili, M. V. Zelkowitz, and K. L. Fisher, "A case study of measuring process risk for early insights into software safety," Proceeding 33rd Int. Conf. Softw. Eng. - ICSE '11, p. 623, 2011.

8. X. Larrucea, A. Combelles, and J. Favaro, "Safety-Critical Software [Guest editors' introduction]," IEEE Softw., vol. 30, no. 3, pp. 25–27, 2013.

9. D. Falessi, M. Sabetzadeh, L. Briand, E. Turella, T. Coq, and R. K. Panesar-Walawege, "Planning for Safety Standards Compliance: A Model-Based Tool-Supported Approach," IEEE Softw., vol. 29, no. 3, pp. 64–70, May 2012.

10. OMG, "Structured Assurance Case Metamodel," 2013.

11. R. Hawkins, I. Habli, T. Kelly, and J. McDermid, "Assurance cases and prescriptive software safety certification: A comparative study," Saf. Sci., vol. 59, pp. 55–71, Nov. 2013.

12. A. Wassyng, T. Maibaum, M. Lawford, and H. Bherer, "Software Certification : Is There a Case against Safety Cases ?," in Proceedings of the 16th Monterey conference on Foundations of computer software: modeling, development, and verification of adaptive systems, 2011, pp. 206–227.

13. N. Habra, A. Abran, M. Lopez, and A. Sellami, "A framework for the design and verification of software measurement methods," J. Syst. Softw., vol. 81, no. 5, pp. 633–648, May 2008.

14. R. van Solingen, V. Basili, G. Caldiera, and H. D. Rombach, "Goal Question Metric (GQM) Approach," in in Encyclopedia of Software Engineering, John Wiley & Sons, Inc., 2002.

15. L.-K. Chan and M.-L. Wu, Quality function deployment: A literature review, vol. 143, no. 3. 2002, pp. 463–497.

16. S. Lee, X. Bao, and T. Zhao, "A safety-critical software development strategy based on theory of diverse design," pp. 694–699, 2011.

17. X. Ge, R. F. Paige, and J. A. Mcdermid, "An Iterative Approach for Development of Safety-Critical Software and Safety Arguments," pp. 35–43, 2010.

18. J. B. Michael, S. Member, M. Shing, K. J. Cruickshank, and P. J. Redmond, "Hazard Analysis and Validation Metrics Framework for System of Systems Software Safety," vol. 4, no. 2, pp. 186–197, 2010.

19. J. R. Laracy, S. Member, and N. G. Leveson, "Apply STAMP to Critical Infrastructure Protection," pp. 215–220, 2007.

20. R. K. Panesar-Walawege, M. Sabetzadeh, and L. Briand, "Supporting the verification of compliance to safety standards via model-driven engineering: Approach, tool-support and empirical validation," Inf. Softw. Technol., vol. 55, no. 5, pp. 836–864, May 2013.

21. S. Linling, Z. Wenjin, and T. Kelly, "Do safety cases have a role in aircraft certification?," Procedia Eng., vol. 17, no. Isaa, pp. 358–368, Jan. 2011.

22. J. Guo, Z. Huang, and M. Liu, "Research on the Railway Safety Critical System with Petri nets," in 2006 6th International Conference on ITS Telecommunications, 2006, pp. 118–121.

23. M. Nanda, J. Jayanthi, C. S. Jamadagni, and M. V, "Quantitative metrics for improving software performance for an integrated tool platform," in 2013 IEEE International Systems Conference (SysCon), 2013, pp. 512–519.

24. OMG, "Structured Metrics Metamodel (SMM)," 2012.

25. L. Lavazza, "Automated support for process-aware definition and execution of measurement plans," in Proceedings. 27th International Conference on Software Engineering, 2005. ICSE 2005, pp. 234–243.

26. S. Engineering, S. Committee, and I. Computer, IEEE Std 15939TM-2008, IEEE Standard Adoption of ISO/IEC 15939:2007—Systems and Software Engineering—Measurement Process, no. January. 2009.

27. S. Web, "ONTOLOGY FOR SOFTWARE METRICS AND INDICATORS," vol. 2, no. 4, pp. 262–281, 2004.

28. V. R. Basili, L. C. Briand, W. L. Melo, and I. C. Society, "A Validation of Object-Oriented Design Metrics as Quality Indicators," vol. 22, no. 10, 1996.

29. J.-P. Jacquet and A. Abran, "From software metrics to software measurement methods: a process model," in Proceedings of IEEE International Symposium on Software Engineering Standards, 1997, vol. 3000, no. 1082, pp. 128–135.

30. J. C. Knight, "Safety Critical Systems : Challenges and Directions," in Software Engineering, 2002. ICSE 2002. Proceedings of the 24rd International Conference on, 2002, pp. 547–550.

31. A. Williams, S. Bradford, D. Liang, W. Post, U. Borkenhagen, and J.-M. Pomeret, "Common metrics framework for ATM performance assessment and monitoring," in The 23rd Digital Avionics Systems Conference (IEEE Cat. No.04CH37576), 2004, vol. 1, pp. 2.C.4–21–13.

32. Y. Yangyang and W. Barry, "A comparison of two safety-critical architectures using the safety related metrics," in Annual Symposium Reliability and Maintainability, 2004 - RAMS, 2004, pp. 621–627.

33. A. Kornecki and J. Zalewski, "Certification of software for real-time safety-critical systems: state of the art," Innov. Syst. Softw. Eng., vol. 5, no. 2, pp. 149–161, Jun. 2009.

34. V. R. Basili and D. H. Hutchens, "An Empirical Study of a Syntactic Complexity Family," IEEE Trans. Softw. Eng., vol. SE-9, no. 6, pp. 664–672, Nov. 1983.

35. N. Nagappan, T. Ball, and B. Murphy, "Using Historical In-Process and Product Metrics for Early Estimation of Software Failures," 2006 17th Int. Symp. Softw. Reliab. Eng., pp. 62–74, Nov. 2006.

36. J. Hill and R. Baggs, "Software safety risk in legacy safety-critical computer systems," in Proceedings 2007 IEEE SoutheastCon, 2007, pp. 229–232.

37. S. P. Kumar, P. S. Ramaiah, and V. Khanaa, "Identification of Software Safety Metrics for Building Safer Software based Critical Computing Systems," 2010.

38. T. Kilpi, "Implementing a Software Metrics Program at Nokia," no. December, 2001.
39. I. Bate and T. Kelly, "Architectural considerations in the certification of modular systems," Reliab. Eng. Syst. Saf., vol. 81, no. 3, pp. 303–324, Sep. 2003.
40. I. Dodd and I. Habli, "Safety certification of airborne software: An empirical study," Reliab. Eng. Syst. Saf., vol. 98, no. 1, pp. 7–23, Feb. 2012.
41. J. Spriggs, GSN - The Goal Structuring Notation. London: Springer London, 2012.

7 Author CV

Xabier Larrucea

Xabier Larrucea holds a Ph.D on software engineering, an Executive MBA and PMP (Project Management Professional) certification. He has more than 10 years of experience. Currently he is project leader at Tecnalia Research and Innovation leading the OPENCOSS project (www.opencoss-project.eu) and participating in TRIAL, TIER and FM-BIASED European projects in safety critical contexts. He has a broad experience in European projects going from FP5, FP6 to FP7. He has participated also in standardisation initiatives such as Software Process Engineering Metamodel 2.0 and SOAML at the Object Management Group. He is also part time lecturer at the University of Basque Country (Euskal Herriko Unibertsitatea).

TIPA for ITIL – From Genesis to Maturity of SPICE Applied to ITIL 2011

Alain Renault, Béatrix Barafort

Centre de Recherche Public Henri Tudor
29, avenue John F. Kennedy, L-1855 Luxembourg
Alain.Renault@tudor.lu
Beatrix.Barafort@tudor.lu

Abstract

The AIDA R&D project was launched in 2003 by Public Research Centre Henri Tudor with the aim to combine the ISO/IEC 15504 with the IT Infrastructure Library (ITIL) leading to the development of TIPA for ITIL as an open framework for the assessment of IT service management processes. As both the ISO standard and the best practices library have continued their evolution ever since, this paper reviews the developments brought to TIPA to keep it aligned with the state of the art in both disciplines, and the actions taken to experiment, validate and transfer the framework to the market.

Keywords

1   Introduction

TIPA® (Tudor IT Process Assessment) is an open framework for the assessment of IT processes. Its development took more than 10 years run in parallel of major improvement phases for ITIL and ISO/IEC 15504 (also known as SPICE[1]), and ending in March 2014 with the release of Classes of Assessment for TIPA.

1.1   Original motivation

In 2002, Tudor observed the first cases of adoption of ITIL® on the Luxembourg market. At the same period the ISO Joint Technical Committee 1, Sub-Committee 7 on Software and Systems Engineering was busy with the upgrade of the ISO/IEC TR 15504 Process Assessment standards series towards the creation of a generic standard for process assessment, enabling the assessment of any kind of process. Tudor being an active member of the ISO working group in charge of that transformation, the idea was born to apply and experiment this new generic assessment framework to both software and IT operation processes [3].

This led to the definition of the AIDA[2] R&D project, aiming at applying a common method for the definition and assessment of processes for both the software development and IT operations sides [2].

Indeed, as the impact of Information and Communication Technologies (ICTs) on company performance is becoming ever more critical, they shall be able to provide the various company sectors with efficient, reliable and effective services that enable users to reach all-round efficiency. ICTs must also act as an innovative factor adept at encouraging and supporting changes in these professional sectors. However most large organizations have split IT activities into two main sectors that cater to their dual mission as described above: IT design studies and development, and IT operations (also referred to as production).

As an answer to that situation, the original motivation of the AIDA R&D project [3] was to develop an integrated – development and operations – IT process assessment method.

The AIDA research project was thus defined in 2003 in order to develop an IT Service Management (ITSM) framework for assessing ITSM processes.

1.2   Development history

The AIDA project was officially launched in autumn 2003 and ended in 2008. The first Process Reference and Process Assessment models for ITIL v2 were developed and experimented during the years 2004-2005 in partnership with ITIL experts [2].

These works were shared with the ISO community in November 2005 at the ISO/IEC JTC1 SC7 interim meeting in Bari (Italy) where the ISO/IEC 2000-1 fast-track was discussed.

More experimentation took place the following years locally in Luxembourg and several other regions of the world [13] [8] following the example of the SPICE trial phases [18].

The success of the experimentations held worldwide with companies led to the initiation of a valorisation study in 2008 aiming at defining the best IP (Intellectual Property) and transfer strategies for the work products produced. The more visible output from that study has been the rebranding of AIDA into TIPA (Tudor's IT service management Process Assessment) [19] and the registration of this brand in several regions of the world.

---

[1] Software Process Improvement and Capability dEtermination
[2] AIDA stands for Assessment and Improvement integrateD Approach

These works led to launching the TIPA R&D project in 2009 (sill running). The TIPA project offered the frame for documenting whole the expertise gathered by the TIPA development team in a book published in December of the same year [1] thanks to a partnership with the international editor Van Haren Publishing. A second strategic partnership was signed the year after with ITpreneurs, worldwide leader in providing IT best practices trainings, for the development of a professional courseware for TIPA Assessors and Lead Assessors. The development of the training material done based on Tudor previous material ended mid-June 2011 and TIPA was redefined as Tudor IT Process Assessment opening the door to the integration of other process models than ITSM in the framework.

At the same time had the TIPA development team finished the revision of the process models and supporting toolbox to align them with ITIL v3, the latest version of ITIL. The TIPA for ITIL licensing scheme was officially launched in Luxembourg at that date with the first public TIPA for ITIL training for Assessors. The first TIPA for ITIL Lead Assessor happened two months later with the first training organized in Japan in September 2011.

Then came a major upgrade that took place in 2012-2013 to align the TIPA for ITIL framework with ITIL 2011. That upgrade has also been the opportunity to pay a particular attention to improve the overall quality of the framework based on the 8 years of experience by Tudor and historical users of the method.

The upgrade was done in February 2013 and was followed by the last big development impacting the core of the framework: the definition of classes of assessment aligned with the requirements of the upcoming ISO/IEC 3300x. The release of Classes of Assessment for TIPA was accompanied by the delivery of Assessment Certificates to organizations undergoing such assessments if reported in the benchmarking database.

2 The TIPA framework

TIPA is a standard-based, objective, repeatable and trustful method developed by Public Research Centre Henri Tudor for the assessment of IT processes. The TIPA framework (Figure 1) is the combination of a number of assets originally developed to support the use of the ISO/IEC 15504 assessment approach on the ITIL® processes.
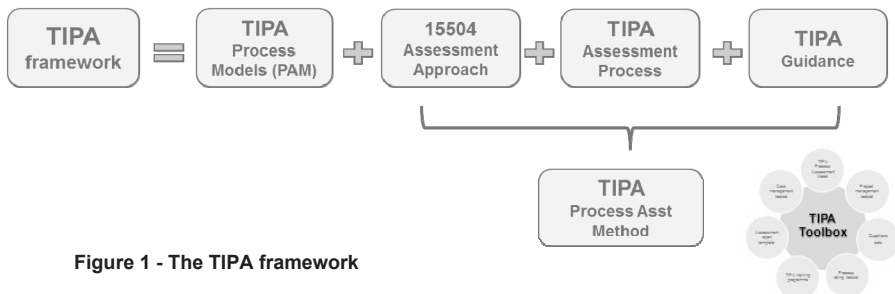
Figure 1 - The TIPA framework

The ISO/IEC 15504-2 requirements on performing assessments are structured and documented in the TIPA Assessment Process. Additional guidance, contextual advice and return of experience complete the TIPA Process Assessment Method and support the usability of the TIPA framework. The TIPA Process Assessment Method can then be used to assess the processes documented in the TIPA Process Models. The TIPA framework is supported by an exhaustive toolbox that provides templates and tools for every single step of the assessment process.

TIPA for ITIL (Figure 2) is the instantiation of the TIPA framework to the IT service management processes documented in the ITIL best practices. More precisely, the TIPA Process Assessment Method being mostly generic, TIPA for ITIL is the instantiation of the TIPA Process Model (PAM) and of the TIPA toolbox to that specific field of application.

Figure 2 - TIPA for ITIL

2.1 Design strategy and constraints

Though TIPA is known as targeting ITIL processes only, CRP Henri Tudor has been working on the development and experimentation of process models in various fields of activity for many years with the intent to have some of them embedded in TIPA. Examples of such process models are Digital Archiving [5] or Information Security Management [14], and further development currently taking place are targeting Management System processes. This strategy has highly influenced the architecture of the framework itself stressing the need to segregate the generic part of the framework from what is specific to an application domain, and the absolute necessity to simplify the activities required to customize the framework to a new domain. This is also reflected on the TIPA logo where the multiple fields of application are represented by multiple stars.

From that core design strategy, a number of design constraints were set to ensure the quality and consistency of the whole TIPA framework. These design constraints cover: 1) 15504-2 compliance; 2) Process Models quality; 3) Process Assessability; 4) Assessment approach effectiveness; and 5) Quality of assessment results.

1.   15504-2 compliance

The TIPA framework is using the structured approach for the assessment of processes defined in the ISO/IEC 15504 standard to understand the capability of the processes implemented by an organization ([10] 4.1).

The ISO/IEC 15504-2 sets out the requirements for performing assessments and for building process models. The former have directly influenced the TIPA Process Assessment Method whereas the latter have guided our design of the TIPA Process Models (PAMs).

a. Documented assessment process

"*The assessment shall be conducted according to a documented assessment process that is capable of meeting the assessment purpose*" ([10] 4.2.1).

The TIPA Assessment Process is detailing how to perform a TIPA assessment. It is completed with guidance to constitute the TIPA Process Assessment Method documented and made publically available through the publication of the book "ITSM Process Assessment Supporting ITIL" published with Van Haren Publishing in 2009 [1].

b. Roles and responsibilities

ISO/IEC 15504-2 ([10] 4.3) is defining roles and responsibilities for the Sponsor of the assessment, the competent assessor, and the assessor(s) whereas ISO/IEC 15504-1 was also defining the role of provisional assessor ([9] 3.49).

ISO/IEC 15504-2 states that an assessment shall be performed by a team of assessors from which at least one needs to have the necessary competence and skills to oversee the assessment. Assessor is here to be considered as a generic term.

TIPA is redefining these roles to make a clear distinction between the Lead Assessor (the one accountable for the assessment results), and the other members of the assessment team (Assessors or Observers). TIPA assessments can only be performed by duly trained and certified TIPA Assessors and led (or overseen) by certified TIPA Lead Assessors.

c. Assessment input

The TIPA Assessment Scope Agreement tool is documenting in details the *assessment input* required by ISO/IEC 15504-2 ([10] 4.4). This document is part of the TIPA toolbox.

d. Assessment output

"Information which is pertinent to the assessment and will support understanding of the output of the assessment shall be compiled and included in the assessment record for retention by the sponsor or their delegated authority" ([10] 4.5.1).

Different tools from the TIPA toolbox are used to collect, manage and keep the records of an assessment: the TIPA Assessment Report documents in details the assessment output, which is then summarized in the TIPA Assessment Result Presentation. The TIPA Assessment Report contains mandatory sections that ensure that the final deliverable is compliant with the requirements of the ISO/IEC 15504-2 [10].

e. Models for process assessment

ISO/IEC 15504-2 sets out the requirements that shall be met by process models used to support process assessment ([10] 6.1). *"The requirements for conformance of the Process Assessment Model enable comparison of outputs from assessments based upon the same Process Reference Model, using different Process Assessment Models"* ([10] 6.1). However this requires Process Assessment Models to comply strictly to the requirements of the standard. The TIPA process models have been designed using specific Requirement Engineering techniques [17] to ensure process models quality.

2. Process Models quality

The ISO/IEC 15504-2 standard sets the requirements that shall be met by the process reference and assessment models developed. However these requirements can be variously interpreted leading to a number of potential issues as no one can really assure the accuracy of a process model according to the specificities and constraints of the expected context of use (domain, type of organization…) [15].

Even if these requirements were strictly met, this would not ensure that the compliant process models would meet the concerns of the stakeholders: easy to use by the assessors, and easy to understand by the process users.

Aware of these limitations thanks to many years spent working on the design, experimentation an improvement of process models, the authors strove to improve the "quality" of the TIPA Process Models through the use of several techniques or tools:

- Requirement Engineering techniques (Goal trees) for process model management [4][15][17]

- *Specific practices* for more robust process descriptions [4][15]

- Traceability records between the components of the process models and the initial process descriptions [15]

- Guidance provided by ISO/IEC TR 24774 for more consistent and well-described process models [15].

The ultimate objective of improving the design and build of better-formed Process Assessment Models is to enhance the value of the process models from the users' perspective in order to make the use of these process models easier for both process users and assessors, for any application domain, encouraging interoperability. Process models are the basis for assessments. Better-formed process models make it easier for the assessor to conduct the assessment, to rate its observations, and to assess process capability. They also strengthen repeatability and comparability of assessments.

The *Transformation Process* applied to build the TIPA process models from the original ITIL content appeared to be very powerful in supporting process model quality [4].

3. Process assessability

Tudor made considerable efforts to come up with compliant, though usable Process Assessment Models. Developing such compliant but also usable Process Assessment Models turned to be a real challenge as the ultimate goal is not only to improve process model quality but also to balance it with "*process assessability*" [16].

TIPA is using the concepts, the capability levels, ordinal rating scale and process attributes from the measurement framework for process capability defined in ISO/IEC 15504-2. However, improvements made for strengthening the quality of the TIPA process models particularly reinforced the robustness and repeatability of the assessment method and improved its repeatability [15].

The content of the TIPA process models has been rearranged so that:

- each process has one single purpose

- process outcomes defined are necessary and sufficient to achieve the process purpose

- each process outcome is defined as a measurable objective

- base practices reflect the process purpose and outcomes

- process activities that contribute to higher levels of capability (than 1) are not base practices, but are defined as *specific practices* under a particular process attribute.

The combination of all these items has considerably improved *process assessability*. Particularly, the fact that in some specific cases, the original definition of process attributes has been further developed through adding contextual generic practices (also called "*specific practices*") for a particular process, has permitted avoiding base practices being polluted with activities that in fact contribute to higher levels of capability.

4. Assessment approach effectiveness

The TIPA Assessment Process and additional guidance have been designed based on concrete experience of the authors making assessments during many years in several fields of activity. The TIPA Process Assessment Method, described in details in [1] is documenting how to effectively run an assessment project. Putting more emphasis on the improvement of processes, TIPA recommends collecting process evidence through interviews mainly, as interviews enable to ask open questions to process actors, triggering a positive involvement in the assessment itself and initiating early improvement thinking.

The TIPA Toolbox provides all the tools to support effectiveness and efficiency both for the assessment team and the assessed organization. The structure proposed by the TIPA Process Assessment Method prevents the assessors from wasting time organizing the assessment themselves. The whole TIPA Toolbox is in fact supporting TIPA Assessors and Lead Assessors in their respective role. For example, the *TIPA workload estimation sheet* helps the Lead Assessor in estimating the cost of the evaluation on both sides.

The TIPA Classes of Assessments, the last evolution brought to the TIPA framework [5], improve even further the efficiency of assessment projects. Selecting the appropriate class of assessment allows to better size the effort to put on a TIPA assessment project and consequently to maximize its value based on the context, on the required level of confidence and on the available resources.

So beside of the original TIPA Assessment Process that can now be defined as a TIPA Class 2 assessment, it now possible to perform a TIPA assessment with an even higher level of confidence (TIPA Class 1) on one side or to perform much cheaper and lighter assessments with a lower level of confidence (TIPA Class 3) on the other side. These evolutions in the TIPA framework have been made possible thanks to the revision of the ISO/IEC 15504-2, and the definition of Classes of Assessments in the new ISO/IEC 33002 [11]. This was also a long waiting request from the TIPA user community.

5. **Quality of assessment results**

The last design constraint was to develop an assessment method that would deliver the best balance between cost and quality of the assessment results (and level of confidence on the results). The structure and content of the TIPA Process Assessment Method focus on producing a clear view of the process status (capability determination) while defining a reliable pathway for process improvement.

2.2   TIPA Assessment Process

To support the stated objectives were the requirements of ISO/IEC 15504-2 documented and organised on a six-sept lifecycle: the TIPA Assessment Process (Figure 3).
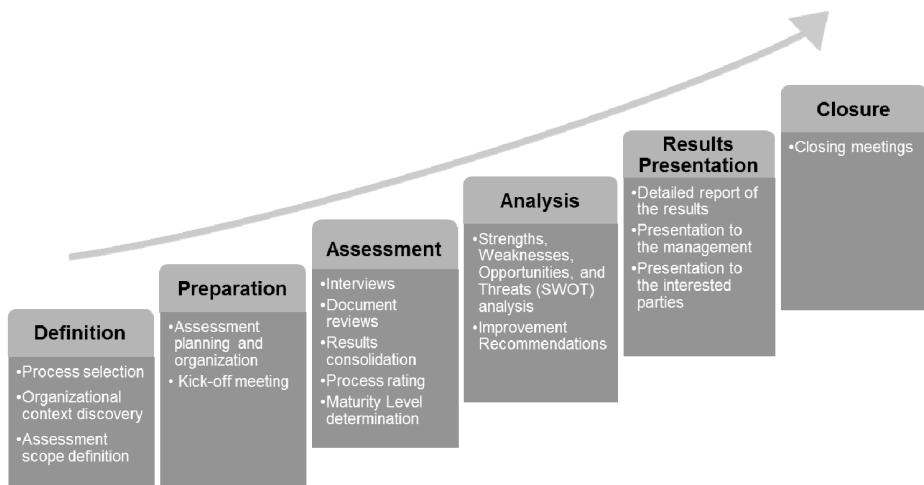
Figure 3 – The TIPA Assessment Process

The TIPA Assessment Process is guiding the TIPA Assessors and Lead Assessors all along the assessment project. It details the activities that need to be performed, highlighting the different roles involved, the resources needed, the inputs and the outputs. It also makes clear connections with the individual tools from the TIPA Toolbox that bring a dedicated support to one or the other activity or phase of the TIPA Assessment Process.

2.3   Toolbox

The TIPA Toolbox is a set of templates, spreadsheets and other documents (including their updates, new versions, translations or modifications) that can be used by the assessment team to perform a TIPA assessment. Once officially recognized as such, the TIPA Lead Assessor receives a license on the entire TIPA toolbox. The TIPA Assessor gets a license on the "TIPA Toolbox for Assessors", which is a subset of the TIPA Toolbox.

The TIPA Toolbox provides 20 tools to support performing a standardized, structured, objective, and repeatable assessment of IT processes, and so guarantees the quality of the assessment results whatever the Class of Assessment retained.
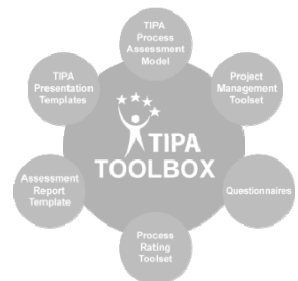
2.4　Alignment with other frameworks

The TIPA certification scheme has been designed to develop the knowledge and skills of individuals, and enable them to lead and/or participate in IT process assessments based on the TIPA framework. Two training courses and two levels of certification have been logically defined: TIPA Assessor and TIPA Lead Assessor. This gives the guarantee that TIPA assessments are led and ran by people having an adequate level of expertise for performing ISO/IEC 15504 assessment, together with sound ITIL knowledge and background.

In order to adhere to the highest quality standards for TIPA Assessments, the 'TIPA Lead Assessor' needs to demonstrate its experience in performing assessments in order to be recognized as a Certified TIPA Lead Assessor:

- Evidence of three TIPA assessments performed out of which two as Lead Assessor, covering a total of ten process instances

- Or evidence of two TIPA assessments performed as Lead Assessor if one of them is coached or supervised by a Certified TIPA Lead Assessor, covering a total of six process instances.

Candidates already holding one of the following qualifications do however not have to report additional evidence of experience and will be directly awarded the Certified TIPA Lead Assessor status: ScampiTM Lead Appraiser, IntacsTM certified Competent or Principal Assessor, IntRSATM certified Assessor or Principal Assessor.

3　Transfer strategy and adoption

Though the initial AIDA R&D project was launched back in 2003, the transfer of TIPA to the market started much later with the publication of the TIPA book in December 2009 and the first Certified TIPA Assessors and Lead Assessors in June and September 2011. However a number of ITSM or assessments experts worldwide had previously been "*grandfathered*" TIPA Certified Lead Assessors, in recognition of their role in the development, experimentation, validation or promotion of the framework.

Performing an assessment requires competences that one cannot get by just reading a book. For that reason is the training and certification programme playing the central role in the transfer strategy for TIPA. Both Assessor and Lead Assessor courses are interactive trainings, implementing serious games to simulate real-life situations and have candidates experience the assessment process and the benefits of implementing TIPA as it is recommended.

Beside the 21 "TIPA *Grandfathers*", 163 TIPA Assessors or Lead Assessors from 24 countries were trained and certified in 2 ½ years, and more than 360 individuals from 59 countries are following TIPA on LinkedIn[3]. This international adoption also reflects the variety of contexts of use (domain of activity) by internal or external process consultants.

4　Conclusion

This paper is providing a global picture of the R&D activities that have led to TIPA as an exhaustive framework for the assessment of IT processes using ISO/IEC 15504.

TIPA for ITIL can now be considered as a mature solution for ITSM consultants. The design constraints under which it was built helped to deliver a robust framework for the assessment of ITSM processes. The rigorous approach used to develop the process models increases "*process assessability*", bringing more value to both the assessors and organizations undergoing process assessments. The framework provided to TIPA certified Assessors and Lead Assessors is completed with an exhaustive toolbox supporting the whole assessment process to help ITSM consultants delivering

---

[3] https://www.linkedin.com/groups/TIPA-Tudor-IT-Process-Assessment-2005527

professional assessment services to organizations.

It has recently been further developed by defining three classes of assessment that increase even more the value of the framework through enabling the right-sizing of the effort based on the expected level of confidence of the results.

All these assets result from the smart combination of a number of design constraints that complemented the design requirements inherited from the ISO/IEC 15504-2. The TIPA Process Assessment Method has been continuously improved to benefit from almost 10 years of experience from its authors and from the ISO/IEC 15504 community.

Besides the activities that will be implemented to ensure continual alignment with the ISO/IEC33000 series of standard on one hand, and with ITIL on the other hand, further developments are planned in the future to support the study the support of other process models in TIPA, like Cobit5, Management Systems, Information Security or others.

5 Literature

1. Barafort B., Betry V., Cortina S., Picard S., St-Jean M., Renault A., and Valdés O. "ITSM Process Assessment supporting ITIL® - Using TIPA to Assess and Improve your Processes with ISO 15504 and Prepare for ISO 20000 Certification", Van Haren Publishing, The Netherlands, ISBN 978-90-8753-564-3, 2009

2. Barafort B., Di Renzo B., Lejeune V., Simon J-M, ITIL Based Service Management measurement and ISO/IEC 15504 process assessment: a win – win opportunity, SPICE'2005, Klagenfurt, Austria, 2005

3. Barafort B., Di Renzo B., Merlan O., Benefits resulting from the combined use of ISO/IEC 15504 with the Information Technology Infrastructure Library (ITIL), in: Proceedings of the International Conference PROFES'2002, Rovaniemi, Finland, 2002

4. Barafort B., Renault A., Picard M., Cortina S., A transformation process for building PRMs and PAMs based on a collection of requirements – Example with ISO/IEC 20000, in: Proceedings of the International Conference SPICE 2008, Nuremberg, Germany, 2008

5. Colet L., Renault A.; "Process Assessment for Digital Records Preservation and Information Security", 10th International Conference on Preservation of Digital Objects, iPRES-2013, Capability assessment and improvement workshop, Lisbon, Portugal, 2013.

6. Cortina S., Picard S., Betry V., Sizing the TIPA process assessment method by using classes of assessment, to be published in: Proceedings of the 18th European System & Software Process Improvement and Innovation Conference 2014. Luxembourg, June 25 - June 27, 2014

7. Cortina S., Picard S., Valdés O., and Renault A., "A Challenging Process Models Development: The ITIL v3 Lifecycle Processes", Proceedings of the International Conference SPICE 2010, Pisa, Italy, May 2010.

8. Hilbert R., & Renault A., Assessing IT Service Management Processes with AIDA - Experience Feedback. EuroSPI². Postdam, 2007.

9. ISO, ISO/IEC 15504-1: Information technology - Process assessment - Part 2: Performing an assessment, 2004

10. ISO, ISO/IEC 15504-2: Information technology - Process assessment - Part 2: Performing an assessment, 2003

11. ISO, ISO/IEC 33002: Information technology -- Process assessment -- Requirements for performing process assessment, 2014.

12. ISO, ISO/IEC TR 24774: Software and systems engineering -- Life cycle management -- Guidelines for process description, 2010

13. Jokela J., "Long Term Utilisation of SPICE in an IT Service Company", Proceedings of the International Conference SPICE 2009,Turku, Finland, May 2009.

14. Mangin O., Barafort B., Dubois E., Heymans P., Designing a Process Reference Model for Information Security Management Systems, in: Proceedings of the International Conference SPICE 2012, Mallorca, Spain, 2012.

15. Picard M., Renault A., Cortina S., "How to improve process models for better ISO/IEC 15504 process assessment", ", in Proceedings of 17th European Conference for Software Process Improvement EuroSpi, Grenoble, France, 2010.

16. Renault A., Barafort B., "TIPA: 7 years experience with SPICE for IT Service Management", Proceedings of the 18th European System & Software Process Improvement and Innovation Conference 2011. Roskilde, Denmark, June 27 - June 29, 2011.

17. Rifaut A., Goal-Driven Requirements Engineering for supporting the ISO 15504 Assessment Process. *European Conference for Software Process Improvement (*EuroSpi*) proceedings* (pp. 151-162). Springer 2005.

18. Rout T., El Emam K., Fusani M., Goldenson D., Jung H.: SPICE in retrospect: Developing a standard for process assessment. Journal of Systems and Software 80(9): 1483-1493, 2007.

19. St-Jean M., TIPA to keep ITIL going and going, in Proceedings of 16th European Conference for Software Process Improvement EuroSpi, Alcala, Spain, 200

Author CVs

Alain Renault

Alain Renault is Senior R&D Engineer with Centre de Recherche Public Henri Tudor in Luxembourg where he is leading the TIPA® R&D project. He returned to research after spending eight years as a software engineer and IT support. Since 1998 he is working on process improvement projects, firstly focusing on software development activities and then on IT Service Management.

Alain Renault is an ISO/IEC 15504 Assessor, a TIPA certified Lead Assessor, Vice-President of the Luxembourg chapter of itSMF and President of the ISO/IEC JTC1 SC7 National Mirror Committee for Luxembourg. He is a member of the ISO working groups on Process Assessment and IT Service Management.

Béatrix Barafort

Béatrix Barafort graduated as a Software Engineer in the "Conservatoire National des Arts et Métiers" (Lyon, France) and has worked in a software house in Lyon for development projects in banks and insurance companies prior to her current position. Since 1996 in CRP Henri Tudor, she has led R&D process assessment and improvement projects based on the ISO/IEC 15504 standard (Process Assessment), mostly in Software Engineering and IT Service Management. She is currently heading the "Business Process and Service Engineering" Research Unit encompassing the TIPA® initiative. She is actively involved in standardization activities in ISO/IEC JTC1 SC7 (Software and Systems Engineering) and in ISO/IEC JTC1 SC40 (IT Service Management and IT Governance). She is President of this latter for the National Mirror Committee for Luxembourg. She was editor of the published ISO/IEC 20000-4 standard for an IT Service Management Process Reference Model.

Development of a Process Assessment Model for Medical Device Software Development

Marion Lepmets, Paul Clarke, Fergal McCaffery, Anita Finnegan, Alec Dorling

Regulated Software Research Centre, Dundalk Institute of Technology, Dundalk, Ireland
{marion.lepmets, paul.clarke, fergal.mccaffery, anita.finnegan, alec.dorling } @dkit.ie

Abstract

Software that is incorporated into a medical device, or which is a standalone medical device in its own right, is of a safety critical nature and subject to regulation from various jurisdictions (principally the EU and the US). In order to satisfy jurisdictional regulations, developers of medical device software generally implement standards and guidance provided by international standards bodies and national administrative departments. However, the various standards and guidance documents are not developed as a single cohesive set but often as separate resources addressing distinct areas of concern. The result for medical device software developers is that integration of these various sources represents a challenging undertaking.

The aim of this paper is to describe the integration of the several process models and regulatory standards, first, into a process reference model and then into a process assessment model for medical device software development. The focus is on the integration of regulatory standards from the medical device domain with generic software development process models, resulting in a unified best practice framework for medical device software development. The process reference model for medical device software development is going to be published this year as IEC TR 80002-3, and the process assessment model for medical device software development is currently being validated through pilot studies in medical device industry.

This best practice framework will help small software developers in their adoption of regulations-compliant best practices while reducing the overhead associated with understanding the long list of regulations and standards they need to adhere to when developing software for medical devices. This framework will also help the manufacturers in selecting their software suppliers assuring that the suppliers have adopted the best practices and are compliant with the medical device standards and regulations.

Keywords

Process Assessment Model, Process Reference Model, Medical Device Software, Medical Device regulations, Best Practice Framework

1 Introduction

A basic requirement of the design of a medical device software process is that it satisfies the regulatory demands associated with the medical devices under construction; as failure to satisfy regulation in a particular region will mean that the device cannot be placed upon the market. In a sense, this is similar to the basic requirement of a software development process: that the process should fit the needs of the project [1]. However, in practice the realisation of this basic requirement has proven to be difficult to achieve in the general software engineering domain. Consequently, many software development approaches have been proposed and implemented, resulting in much discussion regarding the benefits and limitations of the various approaches [2]. It therefore seems likely that no single software approach should be universally prescribed for the general practice of software development. This is essentially owing to the complex interplay between people and the economic activity of commercial software development.

1.1 Medical device regulations and standards

A medical device can consist entirely of software or have software as a component of the overall medical device system. In order to be able to market a medical device within a particular region it is necessary to comply with the regulatory demands of that region. Two of the largest global bodies responsible for issuing and managing medical device regulation belong to the central governing functions of the US and EU. In the case of the US, the Food and Drug Administration (FDA) issues the pertinent regulation through a series official channels, including the Code of Federal Regulation (CFR) Title 21, Chapter I, Subchapter H, Part 820 [3]. Under US regulation, there are three medical device safety classifications: Class I, Class II and Class III. The medical device safety classification is based on the clinical safety of the device. Class I devices are not intended to support or sustain human life, and may not present an unreasonable risk of harm. Class II devices could cause damage or harm to humans. An example of a Class II medical device is a powered wheelchair. Class III medical devices are usually those that support or sustain human life, and are of significant importance in the prevention of human health impairment. An example of a Class III device is an implantable pacemaker. All implantable devices are Class III medical devices as the surgery required carries with itself additional high risks from anaesthesia and possible infections that go beyond the technical and engineering safety risks of the correct performance of the device.

In the EU, the corresponding regulation is outlined in the general Medical Device Directive (MDD) 93/42/EEC [4], the Active Implantable Medical Device Directive (AIMDD) 90/385/EEC [5], and the *In-vitro* Diagnostic (IVD) Medical Device Directive 98/79/EC [6] - all three of which have been amended by 2007/47/EC [7]. Although slightly different to the US safety classifications that are based on clinical safety of the device, the EU classifications essentially embody similar classifications and limitations, where Class I corresponds to Class I, Class IIa and IIb to Class II, and Class III to Class III. A further safety classification applies to the software in the medical device as outlined in IEC 62304, wherein the safety classification is concerned with the worst possible consequence in the case of a software failure (as compared with general medical device safety classification which is based on the difficulty of a regulator to determine if the device will be safe). Hence, some Class II medical devices can cause serious injury or even death, but they are Class II because they are similar (in clinical use and safety) to well understood devices that have been used before. Since IEC 62304 safety classifications are based on worse case failure of the software, it is possible that Class II medical devices can have Class III software.

In the medical device domain, ISO 13485:2003 (ISO 13485 from hereon) [8] outlines the requirements for regulatory purposes from a QMS perspective. ISO 13485, which is based on ISO 9001 [9], can be used to assess an organisation's ability to meet both customer and regulatory requirements. However, ISO 13485 does not offer specific guidance on medical device software development. IEC 62304:2006 (IEC 62304 from hereon) [10], which can be used in conjunction with ISO 13485, does offer a framework for the lifecycle processes necessary for the safe design and maintenance of medical device software. As a basic foundation, IEC 62304 assumes that medical device software is developed and maintained within a QMS such as ISO 13485, but does not require an organisation to be certified in

ISO 13485. Therefore, IEC 62304 can be considered to be a software development specific supplement to ISO 13485.

IEC 62304 is based on ISO/IEC 12207:1995 [11] which although a comprehensive standard for software development lifecycle processes has effectively been decommissioned following the publication of the more extensive ISO/IEC 12207:2008 [12]. Furthermore, other developments in the ISO and IEC communities for software development, such as ISO/IEC 15504 [13], have provided significant additional levels of software process detail to support ISO/IEC 12207:2008. IEC 62304 is currently being revised to better align with ISO/IEC 12207:2008. IEC 62304 is a critical standard for medical device software developers as it is the only standard that provides recommendations for medical device software implementations based on the worst consequences in the case the software failure causing hazards. Furthermore, for general medical device risk management, IEC 62304 is used in conjunction with ISO 14971 [14], with IEC 80002-1 [15] providing guidance on the application of ISO 14971 for software development. Additionally, as IEC 62304 considers a medical device system to consist of software as part of an overall medical device system, the system level requirements are not included within IEC 62304 but instead within the medical device product standard IEC 60601-1 [16]. Also it should be noted that due to the increasing importance of usability within the medical device industry organisations should also adhere to the medical device usability requirements outlined in IEC 62366 [17].

Numerous different medical device standards and regulations now exist, some of which are interlinked with generic software development standards and others which are inconsistent. The dominant medical device software standards such as IEC 62304 are not yet aligned with the approach adopted in the general software development standards community since the 1995 publication of ISO/IEC 12207. One significant change in this respect has been the introduction of a harmonised approach to process description (as defined in ISO/IEC 24774 [18]) which involves the identification of core process outcomes that can later be harnessed to develop a process assessment method. A further significant change relates to the movement in the general software development standards community (and in other safety-related domains) to include a software process improvement dimension that can be instrumental in guiding software development organisations towards the required process targets. In effect, the medical device standards have not kept up with the changes that have been made to the general software development standards. There are several reasons why the medical device standards lag the updates to the general software development standards, (perhaps) most importantly the IEC stability period during which adopted harmonised standards are not to be changed unless the proposed changes are necessary in terms of safety. With the expanding role of software in medical devices, there is a case to be made for introducing the accumulated up-to-date wisdom in the general software development standards into the medical device software development specific standards in a uniform fashion – and work in this direction should not wait for the IEC stability period to come to an end, but rather proceed in the interim period (such as the work reported upon in this paper).

In order to identify an appropriate architecture for introducing the significant body of general software process knowledge into the medical device process domain, an initial important step involves the building of a process reference model (PRM) for medical device software development. The approach used for the PRM development is described in the next section. We then illustrate the architecture and the challenging task of integrating various regulatory standards and informative guidance into a process assessment model (PAM) to allow consistent evaluation of medical device software development processes against the set of standards these organisations have to adhere to. Finally, we summarize the paper along with some concluding remarks.

2 Development of the PRM

A process reference model (PRM) describes a set of processes in a structured manner through a process name, process purpose and process outcomes. Process outcomes are the normative requirements the process should satisfy to achieve the purpose of the process. The PRM for medical device software development is based on various international medical device and generic software development standards as shown on Figure 1.

The development of the PRM was carried out in two steps. In the first step, the PRM for medical device software life cycle processes described in IEC 62304 was developed, entitled IEC 80002-3 which is currently under national ballot at IEC. This PRM was a result of an integration of requirements from ISO/IEC 12207 and IEC 62304 following the guidelines for process descriptions set forth in ISO/IEC 24774.
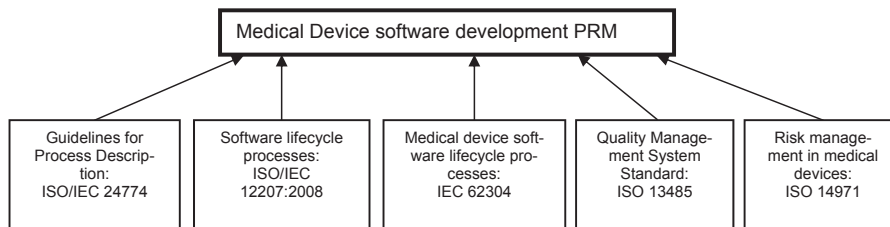
Figure 1. Building blocks of medical device software development PRM

The mapping of the requirements from the two different international standards aims at integrating the varying underlying requirements into a more abstract set of PRM-based requirements which can be applied in the development of a medical device software development PAM. In this section, we outline the approach to mapping those processes that have distinguishable one-to-one mappings from IEC 62304 to ISO/IEC 12207. With the exception of the IEC 62304 risk management process, the majority of the IEC 62304 processes are well mapped to the ISO/IEC 12207 software life cycle processes. The essential difference for many of these processes is that the safety critical activities are embedded throughout the IEC 62304 software life cycle processes.

In conducting process mappings for the directly corresponding processes, we applied the systematic approach of constant comparison and memoing as described by Grounded Theory. Constant comparison is an iterative process of data integration where the dimensions and the properties specific to data are specified [19]. This iterative process is supported by keeping memos that are a written record of analyses. Memoing "forces the analyst to think about the data and it is in thinking that analysis actually occurs" [20].

In this requirements integration activity, we conducted several iterations of constant comparison and memoing. In order to increase the validity of the mapping, a formal independent mapping of the individual processes was performed by two experienced software process researchers.

The first iteration took place when one of the researchers conducted an initial comparison of the two processes (ISO/IEC 12207 and IEC 62304). The result of this first iteration was a proposed mapping along with detailed memos that capture the reasoning behind the proposed process mappings. The second researcher then conducted a review of the first comparison and took notes for the underlying reasons for agreeing or disagreeing with the first researcher. The researchers then reviewed all the data together in the third iteration of constant comparison and they memoed the review results while finalising the requirements integration. Additional ideas and propositions of data integration that occured while the researchers finalised the comparison were again memoed and reviewed by both researchers. Iterative cycles of constant comparison were undertaken until the researchers had no conflict (or no new suggestions to the agreed requirements integration). Constant comparison is a systematic approach to analysing and integrating the process requirements that are written using different terminology and concepts [19]. Memos permit the tracking of justifications for process integration. The memos are also crucial in the validation of the PRM as the reviewers will be able to follow the data analysis and integration process in great detail.

At the time of submitting this paper, the PRM developed in the first step has been approved for publication by the IEC national bodies as IEC DTR 80002-3: Process Reference Model for Medical Device Software Life Cycle Processes (IEC 62304). IEC TR 80002-3 will be published in the middle of this year.

In the second step of the PRM development, IEC 80002-3 was extended with medical device regulatory standards that medical device software development organisations have to adhere to. The requirements from the international standards ISO 14971 and ISO 13485 where then analysed and the requirements that were not yet in the PRM were then integrated into it. ISO 13485 is a Quality Management System standard setting the requirements for regulatory purposes. ISO 14971 describes the application of Risk Management to medical devices. Both of these standards are mandatory for medical device software organisations. To have a comprehensive medical device software PRM, the relevant requirements from these two mandatory standards were included. The approach of integrating requirements from different standards was carried out similarly to the one described above with iterations of reviews by experts until there were no more contradictions in the experts' proposals for integration.

The resulting medical device software development PRM describes processes that could be grouped into three – the system life cycle processes, the software life cycle processes and the supporting processes. ISO 13485 requirements are primarily related to the system level processes which were derived from ISO/IEC 12207 in the first PRM development step. ISO 14971 maps mostly to the Software Risk Management process described also in IEC 62304.

ISO 13485 sets the requirements for Quality Management System (QMS) for Medical Devices. These requirements were integrated into the Medical Device Software Development PAM through relevant new Process Outcomes where no corresponding ones already exist, or as additional details in Base Practices where corresponding Process Outcomes already existed. Some of the QMS requirements target higher Capability Levels, in which case the requirements were related to Generic Practices PA 2.1 (e.g. on allocating resources) or to PA 2.2 (e.g. on documentation) on Capability Level 2. The outcomes or base practices derived from ISO 13485 were highlighted to visualize the source standard. This would then allow detailed feedback to companies in their compliance to the specific standards as a result of process assessment.

ISO 14971 distributes the risk management related requirements across all software life cycle processes. To avoid major duplication, the risk management requirements were kept only in the Software Risk Management process of the main body of the PAM. Instead of distributing these requirements across life cycle processes, a table was added in the Annex of the PAM that lists the specific risk management requirements for each software life cycle process. These requirements need to be added to the selected software life cycle processes for process assessment in the case where the process assessment will not include the Software Risk Management process. The table in the Annex provides the Outcomes from ISO 14971 and their corresponding Base Practice texts from IEC 80002-1 as well as the suggested location in the list of already existing Outcomes in each of the software life cycle processes.

As a result of the integration activities described above, the medical device software PRM consists of 25 processes in the three sets of software life cycle processes, software support processes, and system life cycle processes. In the following section we describe how this PRM was extended with additional elements for medical device software development process assessment model (PAM). This PAM will allow the evaluation of software and systems development processes against all the medical device standards mentioned above.

3 Development of the PAM

The aim of the Medical Device Software Development PAM is to provide a comprehensive model for assessing the software and systems development processes against the widely required medical device regulations, standards and guidelines that a software development organisation in the medical device section has to adhere to. Medical Device Software Development PAM has two dimensions - process dimension and capability dimension. Process dimension lists three groups of processes from various models and standards specified below, i.e. systems life cycle processes, software life cycle processes, and support processes. Each process is described in terms of a Process Name, Process Purpose, Process Outcomes, Base Practices, Work Products and Work Product Characteristics. Process Outcomes are the normative requirements within a process, the achievement of which will allow satisfying the Process Purpose statement. Base Practices are informative activities that illustrate one possible way (workflow) to achieve the corresponding Process Outcomes. Work Products are artefacts that are either produced or used by the Base Practices, both support the achievement of the Process Outcome. Each Work Product is further described in terms of its content called Work Product Characteristics. In the case of the medical device software development PAM, some of these Work Products are normative as they are based on the requirements derived from IEC 62304, ISO 14971 or ISO 13485. Similarly, their content may have been specified in these standards, and if that is the case, this information has been carried forward to Work Products Characteristics of the PAM.

Medical Device Software Development PAM includes information from the following standards and models. First, the baseline PAM is built upon the integrated model of IEC 62304 (Medical device software life cycle processes) and ISO/IEC 12207 (Software and Systems life cycle processes) – the PRM for IEC 62304, i.e. IEC 80002-3 and the QMS and Risk Management requirements added in the second step of the PRM development from ISO 13485 and ISO 14971, respectively. The Process Outcomes were derived from these four standards resulting in the PRM for Medical Device Software Development as described in the previous section. This PRM was extended with corresponding Base

Practices, the process implementation steps that result in the achievement of the outcomes, from IEC 62304, ISO/IEC 15504-5 and IEC 80002-1 (for ISO 14971).

As mentioned previously, IEC 80002-1 provides guidance on the application of ISO 14971 (Risk Management) to Medical Device Software. This guidance information was added to the Base practices to correspond to the requirements from ISO 14971 described in the normative part of the model illustrating a way for achieving these requirements. Most of the information from IEC 80002-1 was integrated into the PAM through Software Risk Management process Base Practices.

Additional base practice information was then derived from the FDA guidance documents on Pre-market Submission [21], Software Validation [22], and Off-The-Shelf (OTS) [23] software. In most cases, this guidance was integrated into the existing processes through adding onto the existing Base Practices and Work Products or adding new Base Practices. Software Validation is not described in a separate process in IEC 62304 but as the FDA guidance provides the current best practices for software development in medical device domain, this area should also be considered. Additional processes of Software Validation, Software Installation and Software Acceptance Support were therefore added from ISO/IEC 12207 to satisfy the requirements of the FDA Guidance Document on Software Validation. Best practices from the FDA guidance documents were then analysed and iteratively integrated into these three processes.

Figure 2 below describes the different sources for the medical device software development PRM and PAM. The medical device software development PRM is based on IEC 62304, ISO/IEC 12207, ISO 14971 and ISO 13485. The PAM then extends this PRM with base practices and work products, some of the latter also being normative as they are described in IEC 62304, ISO 14971 or ISO 13485 as requirements. Where process outcomes are derived from ISO/IEC 12207, their corresponding base practices and work products are derived from ISO/IEC 15504-5. Where process outcomes are derived from ISO 14971, their corresponding base practices are derived from IEC 80002-1. In addition to these sources, FDA guidance on premarket submissions, software validation and off-the-shelf software have been added to the informative base practices where appropriate.
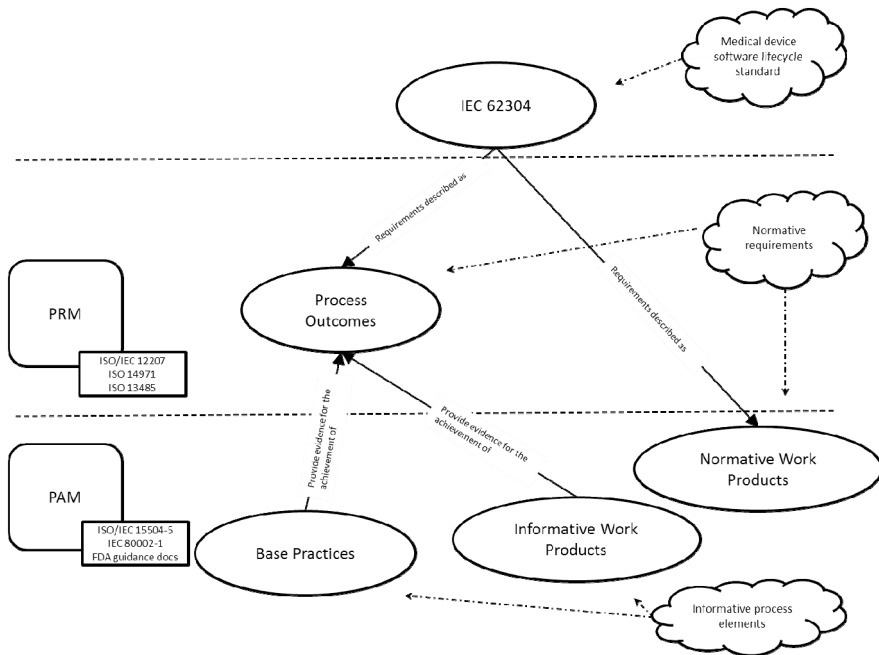
Figure 2. Normative and informative elements of the medical device software development PRM and PAM

Capability dimension of the Medical Device Software Development PAM is derived directly from ISO/IEC 15504 together with the Capability Levels, Process Attributes, Generic Practices, Generic Resources and Generic Work Products.

4   Discussion

One of the biggest challenges in integrating requirements and informative information from various sources is to do with the structure and terminology used in standards. Although ISO/IEC 24774 describes a uniform structure for process models, this has not been widely adopted quite yet. Some of the standards list their requirements in activities distinguishable as requirements only by the verb used in that sentence, e.g. "shall", "should" or "are".

Natural language offered another challenge in the integration of requirements from various sources. Terms like "define", "identify", "document", "record", and "establish" have each got a different meaning in some standards but in other standards they can be all be summed up under the term "establish". It took many iterations of constant comparison before reaching a result that both reviewers were satisfied with.

The terminology of risk management in medical device standards has a different meaning from risk management in the generic software development standards where the risks are mostly related to project budget and schedule. In generic software development standards, safety engineering and safety management correspond to the product risk management central to any safety critical domain software development. Deciding on the terminology to adopt in the single best practice framework has been a great challenge. On one hand, the terminology should be the one that the safety critical domain experts use every day in their work as they will be applying the framework. At the same time, the terminology should be comprehensible for software developers across different domains allowing the latter to move into a safety critical software development.

Another significant challenge in integrating various requirements from different sources is the interface between systems and software levels. This is a challenge often faced in embedded software development. In our PAM, there are both system and software life cycle processes and for embedded software development, the interface between these levels should be very strong. One way to strengthen the interface is to trace requirements from users throughout the development to validation ensuring that the user gets exactly what he or she needed. In order to further strengthen the interface between the software and system level development processes, this traceability should be bilateral, meaning that not only should you be able to trace the user requirements through system and software requirements specification to the final product, but you should also be able to trace every feature of the product and software item back to the user needs.

5   Conclusions and Future Works

The medical device software development domain is filled with regulations that software development organisations need to comply with in order to market their products. In this paper we have described these regulatory standards together with the generic software development standards, further detailing how we have integrated requirements from both into a best practice framework.

This framework can be used for medical device software development process assessments on both system and software level. The framework provides visibility of compliance of the organisation's processes with the requirements from all the source standards of the framework. This will allow the medical device manufacturer to select the supplier that focuses on the improvement of their medical device software development and adopts the best practices derived from the set of required standards.

At the time of submitting this paper, the framework is being validated in industry. The framework has been piloted in the first medical device companies with the aim to gather feedback from medical device software developers and to improve the framework based on this feedback before its official launch later this year.

6   Literature

1.   Feiler, P. and W. Humphrey *Software Process Development and Enactment: Concepts and Definitions*. 1992.

2.   DeMarco, T. and B. Boehm, *The agile methods fray.* IEEE Computer, 2002. **35**(6): p. 90-92.

3.   FDA. *Chapter I - Food and drug administration, department of health and human services subchapter H - Medical devices, Part 820 - Quality system regulation*. [cited 2013 15.05]; Available   from: http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfcfr/CFRSearch.cfm?CFRPart=820.

4.   European Commission, *Directive 93/42/EEC of the European Parliament and of the Council concerning medical devices*, in *OJ o L 247 of 2007-09-21*. 1993: European Commission, Brussels, Belgium.

5.   European Commission, *Council directive 90/385/EEC on active implantable medical devices (AIMDD)*. 1990: Brussels, Belgium.

6.   European Commission, *Directive 98/79/EC of the european parliament and of the council of 27 october 1998 on in vitro diagnostic medical devices*. 1998: Brussels, Belgium.

7.   European Commission, *Directive 2007/47/EC of the European Parliament and of the Council concerning medical devices*, in *OJ no L 247 of 2007-09-21*. 2007, EC: Brussels, Belgium.

8.   ISO, *ISO 13485: Medical Devices - Quality Management Systems - Requirements for Regulatory Purposes*. 2003, ISO: Geneva, Switzerland.

9.   ISO, *ISO 9001:2000 - Quality Management Systems - Requirements*. 2000: Geneva, Switzerland.

10.   IEC, *IEC 62304: Medical Device Software - Software Life-Cycle Processes*. 2006, IEC: Geneva, Switzerland.

11.   ISO/IEC, *ISO/IEC 12207:1995 - Information Technology - Software Life-Cycle Processes*. 1995, ISO/IEC: Geneva, Switzerland.

12.   ISO/IEC, *ISO/IEC 12207:2008 - Systems and Software Engineering - Software life cycle processes*. 2008, ISO/IEC: Geneva, Switzerland.

13.   ISO/IEC 15504-2, *Information technology - Software process assessment - A reference model for processes and process capability*, in *15504*. 2004.

14.   ISO, *ISO 14971 - Medical Devices - Application of Risk Management to Medical Devices* 2009, ISO: Geneva, Switzerland.

15.   IEC, *IEC TR 80002-1 - Medical Device Software - Part 1: Guidance on the Application of ISO 14971 to Medical Device Software*. 2009, IEC: Geneva, Switzerland.

16.   IEC, *IEC 60601-1 - Medical electrical equipment – Part 1: General requirements for basic safety and essential performance* 2005, IEC: Geneva, Switzerland.

17.   IEC, *IEC 62366 - Medical devices - Application of usability engineering to medical devices*. 2007, IEC: Geneva, Switzerland.

18.   ISO/IEC, *ISO/IEC 24774 - Systems and Software Engineering - Life Cycle Management - Guidelines for Process Description*. 2010: Geneva, Switzerland.

19. Clarke, P. and R.V. O'Connor, *The situational factors that affect the software development process: Towards a comprehensive reference framework.* Information and Software Technology, 2012. **54**(5): p. 433-447.

20. Corbin, J. and A. Strauss, *Basics of Qualitative Research*. 2008, Thousand Oaks, CA, USA: Sage Publications.

21. FDA, *Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices*. 2005. p. 20.

22. FDA, *General Principles of Software Validation; Final Guidance for Industry and FDA Staff*. 2002. p. 43.

23. FDA, *Guidance for industry, FDA reviewers and compliance on - Off-The-Shelf Software Use in Medical Devices*. 1999. p. 26.

Author CVs

Dr. Marion Lepmets

Dr. Marion Lepmets is a Senior Research Fellow at the Regulated Software Research Centre in Dundalk Institute of Technology and a member of the Irish Software Engineering Research Centre (Lero). Her PhD and post-doc research focused on the impact of process model implementation and of process assessment on process improvement success, respectively. Her current research is about the development of the process reference and process assessment models for medical device software development. She is also involved in the development of software engineering standards in the International Standardization Organization (ISO/IEC JTC1 SC7).

Dr. Paul Clarke

Dr. Paul Clarke is a Research Manager in the Regulated Software Research Centre based at DkIT and a Research Fellow with Lero – the Irish Software Engineering Research Centre. Primary among Paul's present research interests is the establishment of a best practice framework for medical device software development.

Dr. Fergal McCaffery

Dr Fergal Mc Caffery is a Science Foundation Ireland (SFI) Principal Investigator. He is a Senior Lecturer in the Department of Computing and Mathematics, Dundalk Institute of Technology (DkIT). He is Director of the Regulated Software Research Centre in DkIT and the Medical Device Software Engineering competency area in Lero. He has been awarded SFI funding through the Stokes Lectureship, Principal Investigator and CSET Programmes to research the area of medical device software. He has published over 150 peer-reviewed conference and journal papers and is on the editorial board/programme committee for a number of leading software engineering conferences and journals.

Anita Finnegan

Anita Finnegan is a Senior Software QMS Researcher in the Regulated Software Research Centre based at DkIT and a member of Lero – the Irish Software Engineering Research Centre. Anita's primary focus is establishing the relationship between QMS requirements and software development requirements. Her PhD research addresses the development of a framework for the security assurance of medical devices using cybersecurity assurance cases.

Alec Dorling

Alec Dorling is a member of the research team at the Regulated Software Research Centre in Dundalk Institute of Technology and a member of the Irish Software Engineering Research Centre (Lero). He is the ISO convener of the ISO/IEC 15504 series and ISO/IEC 330xx family of standards on Process Assessment, and the international project leader of SPICE (Software Process Improvement and Capability dEtermination). He has held key positions in software engineering at both national and European levels at IVF's Centre for Software Engineering in Sweden, IT Research and Technology Transfer Centre at the University of Boras in Sweden, European Software Institute in Spain and the National Computing Centre in the UK. He is on the editorial board/programme committee for a number of leading software engineering conferences and journals.

Sizing the TIPA process assessment method by using classes of assessment

Stéphane Cortina, Michel Picard, Valérie Betry

Centre de Recherche Public Henri Tudor
29, avenue John F. Kennedy,
L-1855 Luxembourg
{ stephane.cortina ; michel.picard ; valerie.betry }@tudor.lu

Abstract

This paper describes how the TIPA framework has been enhanced with the addition of three new classes of assessment. These brand new TIPA classes of assessment comply with the requirements coming from ISO/IEC 33002. This paper provides, for each class, a detailed description and a clear definition in terms of purpose and recommended use cases. Finally, the paper explains how to select the appropriate class of assessment, according to the context, in order to better size the effort to put on a TIPA assessment project and consequently to maximize its value.

Introduction

When performing a process assessment, whatever the domain assessed, the level of confidence in the assessment results is directly related to the level of rigor of the assessment method. Up to now, when the capability of some processes was assessed with the help of the Tudor's IT Process Assessment (TIPA®) framework, the level of confidence in the assessment results was always the same. As the TIPA practitioner community increases in numbers, the difficulty of TIPA to fit specific contexts of use is becoming a brake on its market adoption. This paper describes how, by taking the advantage of an alignment with the upcoming new ISO/IEC 33000 series of standards, the TIPA[1] framework has been enhanced to propose a sizeable process assessment method. This article also explains how this new process assessment method, based on three classes of assessment, better fits the different contexts and purposes of various process assessments.

This paper is structured as follows: section 2 presents the current state of the TIPA framework and its sole process assessment method based on well-defined criteria, such as the number of evidence collected, their sources, or the composition of the assessment team. Section 3 presents the generic concept of classes of assessment and describes two different instantiations of this concept. Then, section 4 presents in detail the three brand new TIPA classes of assessment. Finally, section 5 brings some guidance on how to select the appropriate class according to the context and purpose of one assessment. To conclude, section 6 summarizes the findings of this article.

1 Previous state of the TIPA framework

TIPA is a generic process assessment framework (developed by the Public Research Centre Henri Tudor[2]) that can be used to evaluate the capability of processes in an objective and repeatable way [1]. It combines a standard description of processes, which are organized in a process assessment model (PAM), with a well-defined process assessment method (Figure 1).

Up to now, whatever the context on which it was applied, the TIPA framework was invariably based on the same *TIPA Process Assessment Method* that strictly complies with all the requirements defined in the part 2 of ISO/IEC 15504 [2], the international standard for process assessment.
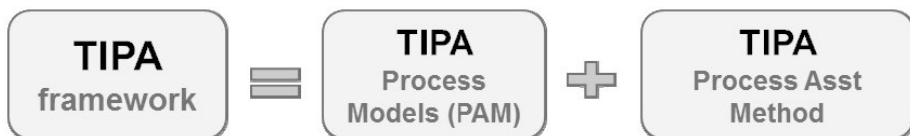
Figure 1 - The TIPA generic framework

TIPA for ITIL is the instantiation of this generic framework to the IT Service Management (ITSM) domain. As visible on Figure 2Figure 2, it combines the *TIPA Process Assessment Method* with the *TIPA PAM for ITIL* to evaluate the capability of (a subpart of) all the 26 processes defined in the latest version of ITIL®, published in 2011. [3][4][5][6][7]

---

[1] www.tipaonline.org
[2] www.tudor.lu

Figure 2 - TIPA for ITIL

TIPA for ITIL is globally recognized and has now been used for several years [8][9][10] to assess the capability of ITIL processes within multiple companies around the world [11][12][13][14]. In 2013, 64 TIPA Assessors and Lead Assessors have been trained, bringing the total number of certified assessors to 173, distributed over more than 20 countries[3].

Up to now, TIPA for ITIL, and particularly the *TIPA Process Assessment Method,* was recommending to follow the same steps whatever the purpose of the assessment, whatever the size of the assessed organization, and whatever the level of confidence required by the assessment results. Thus, by following this standardized assessment method, criteria such as as minimum numbers of assessor, minimum numbers of process instance to evaluate, as well as minimum numbers of evidence to collect, stayed unchanged from one assessment to another. In 2013, after several years of utilization by users of various origins and of various sectors, the TIPA development team received a strong request from the TIPA practitioners' community for a lighter (and cheaper) assessment method. Beside this, an increasing number of TIPA beneficiaries (i.e. assessed organizations) asked for TIPA certificates that could be publicly displayed for communication and marketing purposes. These two requests from the market led the TIPA development team to consider the possibility to enhance the TIPA framework with the definition of 3 classes of TIPA assessment.

2 Classes of assessment...

2.1 ... in general

In the domain of process assessment, defining classes of assessment is a way to size the assessment method. An assessment method is usually characterized by a set of activities (sometimes grouped by phase as illustrated by **Figure 3**)

---

[3] Source: ITpreneurs
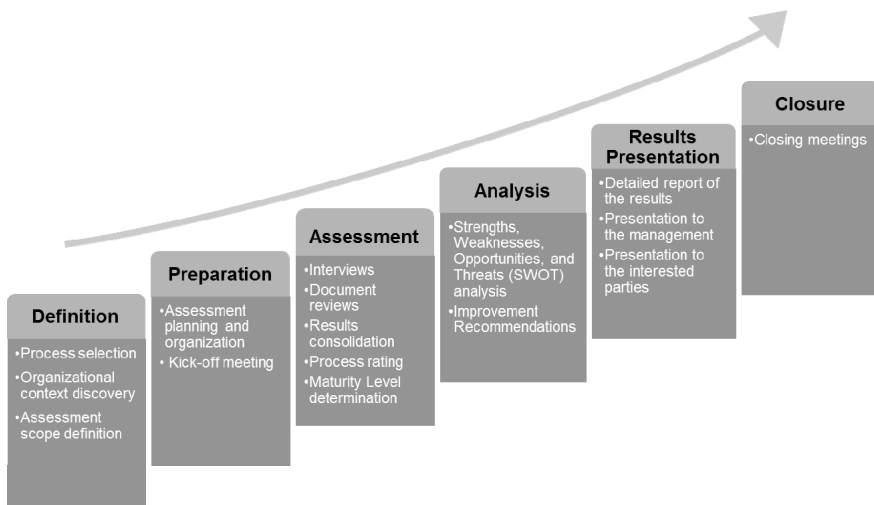
Figure 3 - Typical activities composing an assessment method

Defining classes of assessment consists in setting the requirements related to each of the key activities composing the assessment method at more or less restrictive levels. Thus, if the objective is to define a really robust class leading to objective and comparable assessment results, the class of assessment should require that the activity of conducting interviews is performed by a pair of qualified and independent assessors. On the contrary, if the objective is to define an assessment class that allows performing quick and rapid assessments, then this class should require that only one internal assessor conducts the interviews.

2.2   ... according to the Software Engineering Institute

A family of assessment methods that satisfy a defined subset of requirements in the Assessment Requirement for CMMI (ARC). These classes are defined so as to align with typical usage modes of assessment."

Source: ARC v1.0 [15]

In 2000, the Software Engineering Institute (SEI[4]) published the first version of its 'Appraisal Requirements for CMMI' (ARC) [15] that specified three appraisal method classes (A, B, and C), intended for use with Capability Maturity Model Integration (CMMI). This document defined three different classes of assessment, or according to the vocabulary used by SEI, three different appraisal classes. Each class is distinguished by the degree of rigor associated with the application of the method. Class A is the most rigorous. Class B is slightly less rigorous and class C is the least rigorous.

The latest version of ARC (v1.3)[16], published in 2011, defines the requirements for appraisal methods intended for use with CMMI and People CMM reference models. It still contains the same concepts of classes of assessments (A, B, and C) and the requirements related to each of these classes are visible in the Table 1Table 1 below.

---

[4] www.sei.cmu.edu

Table 1 - Classes characteristics according to SEI

Characteristics	Class A	Class B	Class C
Amount of objective evidence gathered	High	Medium	Low
Rating generated	Yes	No	No
Team size	Large	Medium	Small
Data sources (artifacts and affirmations)	Requires two data sources	Requires two data sources	Requires only one data source
Appraisal team leader requirement	Certified Lead Appraiser	Person trained and experienced	Person trained and experienced

SCAMPI is the family of appraisals, created by SEI and based on the Appraisal Requirements for CMMI (ARC).

SCAMPI A is a rigorous method and the only one that can result in a rating. It is used for performing a formal benchmark.

SCAMPI B is a less formal method that usually helps an organization to understand, with a relatively high degree of confidence, its progress toward a target CMMI maturity level or target capability profile.

SCAMPI C is a shorter, more flexible, and cheaper appraisal method that is used to assess the adequacy of planned approaches to process implementation and to provide a quick gap analysis between the organization's processes and CMMI practices.

2.3   ... according to the International Organization for Standardization

The Class of assessment will determine a level of rigour for which an assessment is to be performed."
Source: ISO/IEC 15504-7:2008 - clause 6.2.1[17]

In 2008, the International Organization for Standardization (ISO[5]) published the part 7 of ISO/IEC 15504 [17]. This technical report was the first document representing an international consensus about the concepts of classes of assessment. It defined three classes of process assessment (with their specific requirements) and their application for the rating of organizational maturity levels. In 2014, the new ISO/IEC 33000 series of standards ([18][19][20][21][22]) is emerging, in order to replace the ISO/IEC 15504 family. This set of generic standards describes how to define process models, how to define measurement frameworks and how to perform process assessment with the purpose of evaluating any kind of process quality characteristics (for example capability, safety, or sustainability). Particularly, the ISO/IEC 33002 [19] proposes a definition of three classes of assessment. In this normative document, specific requirements related to assessment planning, data collection and validation, and process attribute rating and reporting are identified and associated to each of these classes (see Table 2 below).

---

[5] www.iso.org

Table 2 - Classes characteristics according to ISO/IEC 33002

Characteristics	Class 1	Class 2	Class 3
Amount of objective evidence gathered	High	Medium	Low
Rating generated	Yes	Yes	No
Minimum number of instances (per process)	4	2	1
Data sources (interviews and documents)	Requires two data sources	Requires two data sources	Requires only one data source
Number of assessors (per process)	Two assessors, one of whom shall be a Lead Assessor independent of the organizational unit being assessed.	One assessor who shall be a Lead Assessor.	One assessor who shall be a Lead Assessor.

3 The new TIPA classes of assessment

As mentioned in section 2, the TIPA development team decided in 2013 to enhance the TIPA framework in order to better address the needs coming from the market for a sizeable assessment method. Indeed, as stated in [23], to deal with resource limitations, small companies need short light assessments. Moreover, at the same period, the ISO/IEC 15504 standard, on which the *TIPA Process Assessment Method* was previously based, had started to be progressively replaced by the ISO/IEC 33000 series.

The TIPA development team hence has taken the opportunity to re-align the TIPA framework with this new set of standards and to enhance TIPA with the addition of classes of assessment.

For that, the TIPA development team first started by mapping the previous *TIPA Process Assessment Method* to the ISO/IEC 33002 requirements to find that this assessment method can be qualified as a Class 2. Then, the TIPA development team defined two additional classes, one more exhaustive and one lighter.

In the end, the use of such classes enables organizations to size the effort they are willing to put on an assessment depending on the level of confidence required for the results and/or on the amount of resources available to perform the assessment.

These new TIPA classes of assessment are compliant with the requirements defined in ISO/IEC 33002 [19]. However, in some cases, the new TIPA classes of assessment define requirements that are stronger than those in ISO/IEC 33002. This choice has been made on purpose by the TIPA owners, in order to, on one hand ensure a smooth transition from the previous version of the assessment method, and on the other hand reinforce the soundness of the new method.

For each new TIPA class, a purpose, a list recommended use cases, and a description are detailed in the following paragraphs.

3.1 TIPA Class 1

Purpose

The purpose of a TIPA Class 1 assessment is to provide a sound assessment of individual process capability enabling determination of organizational maturity and trustworthy comparison between organizations.

Recommended use cases

The recommended use cases for a TIPA Class 1 assessment are:

- Comparison across different organizations
- Comparison in order to select suppliers
- Benchmarking with other Class 1 assessment results
- Sound rating of process capability to determine organizational maturity
- Public display of assessment results

Description

Compared to the previous version of the *TIPA Process Assessment Method*, TIPA Class 1 is a more exhaustive assessment method. It provides the highest levels of objectivity and reliability in the assessment results because it requires more interviews (a minimum of 4 for each assessed process) and two sources of evidence are necessary to confirm the established ratings. In addition to interviews, systematic reviews of the process inputs and outputs are conducted. Moreover, a certified TIPA Lead Assessor is required to manage a TIPA Class 1 assessment project and each assessed process requires a pair of 2 certified TIPA assessors (one of them can be the project's Lead Assessor). Furthermore, the assessment team must be independent of the organization being assessed.

3.2 TIPA Class 2

Purpose

The purpose of a TIPA Class 2 assessment is to provide a sound assessment of individual process capability, enabling a structured improvement and allowing comparisons over time or between internal units.

Recommended use cases

The recommended use cases for a TIPA Class 2 assessment are:

- Sound rating of process capability
- Determination of detailed gap and SWOT analysis
- Definition of sound improvement recommendations
- Determination of the capability of one supplier
- Comparison over time of the same organizational unit
- Comparison across different organizational units within the same organization
- Determination of organizational maturity (for internal use only)
- Benchmarking with other Class 2 assessment results
- Display of assessment results inside the organization

Description:

TIPA Class 2 corresponds to the previous *TIPA Process Assessment Method*. Compared to Class 1, Class 2 is more focused on one organization, assessing its processes for internal purpose or comparing its internal units. Compared to the minimum set of requirements for a Class 2 coming from the ISO/IEC 33002, the TIPA Class 2 is more rigorous. Indeed, during a TIPA Class 2 assessment, a minimum of three interviews should be conducted and rated for each assessed process (whereas the ISO/IEC 33002 only recommend two interviews per process). In addition to assessment interviews, a TIPA Class 2 requires reviewing process inputs or outputs when additional supporting evidence is necessary to make sure that the rating is objective and reflects the reality of the process execution.

Moreover, as for a TIPA Class 1 assessment, a TIPA Class 2 assessment requires a certified TIPA Lead Assessor to manage the assessment project and a pair of two certified TIPA Assessors for each assessed process (one of the assessors can be the Lead Assessor). Again, this requirement for a pair of assessors is stronger than what is required by the ISO/IEC 33002 standard. Finally, in a TIPA Class 2, even if the assessment team is not required to be independent of the organization being assessed, a separation of responsibility from the personnel being interviewed is required.

3.3   TIPA Class 3

Purpose

The purpose of a TIPA Class 3 assessment is to provide a quick overview of process capability enabling the monitoring of both process implementation and improvement.

Recommended use cases

The recommended use cases for a TIPA Class 3 assessment are:

- Quick rating of process capability
- Light strength and weakness analysis
- Monitoring of process implementation and improvement
- Light definition of improvement suggestions

Description:

TIPA Class 3 assessment is a lighter version of the previous *TIPA Process Assessment Method*. It is the quickest and the cheapest class of the three. It requires a certified TIPA Lead Assessor for the management of the assessment project but each process is assessed by a single certified TIPA Assessor. Only 2 interviews for each assessed process are required and it is possible to provide only one rating for these two interviews. There are no requirements regarding the independence of the assessment team, which makes it possible for an assessor to conduct assessments within his/her own organizational unit. As in Class 1 and Class 2, the Lead Assessor can play the role of an assessor. The goal of TIPA Class 3 assessment is to focus on the rating of the processes and to quickly get a view on the current status of the assessed processes. Less time is spent on side results. Interviews are the unique source of evidence required and no additional review of the process inputs and outputs is necessary.

3.4   Summary of the three TIPA classes of assessment

The characteristics of the three TIPA classes of assessment are summarized in the Table 3 below.

Table 3 - TIPA classes characteristics

Characteristics	TIPA Class 1	TIPA Class 2	TIPA Class 3
Amount of objective evidence gathered	High	Medium	Low
Rating generated	Yes	Yes	No
Minimum number of instances (per process)	4	3	2
Data sources (interviews and documents)	Requires two data sources	Requires two data sources	Requires only one data source
Number of assessors (per process)	2 certified TIPA Assessors (one can be the project's Lead Assessor)	2 certified TIPA Assessors (one can be the project's Lead Assessor)	1 certified TIPA Assessor (who can be the project's Lead Assessor)

4 Selecting the appropriate class of assessment

First of all, there is not one TIPA class that is the most appropriate in all cases.

Indeed, it is not necessary to use a restricting assessment method when monitoring the internal implementation of an improvement plan? On the contrary the use a light assessment method is probably not the better choice to obtain a trustable benchmarking. In fact, the selection of the appropriate TIPA class of assessment depends on many factors such as: the objectives of the assessment, the level of exposure required for the assessment results, the resources available, or the expected outputs.

In other words, considering the context of the assessment, to choose the appropriate class of assessment, the Lead Assessor should compare the objective, the description and the use cases associated with each class. For example, if the objective of the assessment is to make a quick check-up of internal ITIL processes to measure the progress of their implementation, the Lead Assessor should choose a TIPA Class 3. On the opposite, if the main objective is to publicly display the results of the TIPA assessment project on a corporate website, he should select a TIPA Class 1. Beside the contextual considerations, the target capability level also affects the selection of the assessment class. For instance, when performing assessments targeting the highest capability levels (e.g. level 4 or 5), the Lead Assessor should prefer using a TIPA Class 1 or TIPA Class 2 (due to their inherent level of rigor) instead of using a "lighter" class (e.g. TIPA Class 3).

In the early stages of a TIPA assessment project, a Lead Assessor has to explain the differences, the benefits and the limits of each TIPA class to the Sponsor (i.e. the one paying for the assessment). This way, whatever the goal of the assessment project, they should be able to agree on the TIPA class that will best suit their particular context.

5 Conclusion

Although the soundness of the TIPA framework was the main reason for making TIPA adopted all around the world, the economic and business realities have suggested a refinement of the *TIPA Process Assessment Method* in order to make it sustainable and so ensure the lasting quality of the TIPA framework.

Indeed, the previous *TIPA Process Assessment Method* was often considered as too inflexible and consequently unusable in some but frequent specific contexts of use. On one hand, always requiring a pair of assessors was sometimes seen as too heavy and too costly (for example on the context of internal assessments regularly performed to control the progress of a process implementation project). On the other hand, in certain context, some organizations considered that a higher level of rigor in the assessment method was justified (for example when they wanted to publicly display their assessment results on a certificate, for marketing or communication purpose).

In order to take into account the feedback provided by the TIPA practitioners' community and to respond to an increasing demand for a sizeable assessment method, the TIPA framework has been enriched with three classes of assessment. These new TIPA classes of assessment enable organizations to size the effort they are willing to put on a TIPA assessment, depending on the level of confidence required on the results and on the amount of resources available to perform the assessment. Thus, by using the appropriate class according to its context, an organization is able to maximize the value of each TIPA assessment project.

Even if they are inspired by the new ISO/IEC 33000 series of standards, these new TIPA classes of assessments are a little bit more demanding than the classes described in ISO/IEC 33002. This choice has been made deliberately, based on the experience brought by all TIPA assessments performed in the past 10 years. Indeed, requiring an additional assessor (in TIPA Class 2) or an additional instance (in TIPA Class 2 and Class 3) brings, *in fine,* much more value than the costs implied by these additional requirements. Thus, using more demanding classes of assessment contributes to enhance the quality of the SWOT analysis, the pertinence of the improvement recommendations, and the confidence in the assessment results. Particularly, requiring an additional instance (compared to the ISO requirements) is highly valuable when performing a TIPA Class 3 assessment targeting the capability level 3 (which, among other things, consists in assessing how a standard process is deployed within the organizational unit assessed). Furthermore, preserving a higher level of requirements in the TIPA classes allows to keep the new TIPA Class 2 aligned with the previous *TIPA Process Assessment Method*, but also to ensure that the results of both assessments (i.e. TIPA Class 2 and previous TIPA) are still comparable.

Moreover, the authors of this paper are convinced that each new TIPA class of assessment represent, in its own context of use, the best possible balance between the effort required and the quality of the assessment results. The latter assertion implies that the TIPA Lead Assessor, in conjunction with the assessment sponsor, is able to select the right class of assessment for a given context or specific needs. That's why, in order to provide the market with well-trained and well-equipped Lead Assessors, the TIPA development team is now focusing on the improvement of the existing Lead Assessor courseware and on the diffusion of an up-to-date TIPA toolbox.

It is only in this way that the new TIPA classes of assessment will effectively bring their added value to the market.

6 Literature

1. Barafort, B., Betry, V., Cortina, S., Picard, M., St-Jean, M., Renault, A., & Valdés, O. (2009). ITSM Process Assessment Supporting ITIL - Using TIPA to assess and improve your processes with ISO 15504 and prepare for ISO 20000 certification. ISBN: 978 90 8753 564 3: Van Haren Publishing.
2. ISO/IEC 15504-2: Information technology -- Process assessment -- Part 2: Performing an assessment. (2003). Information Technology - Process assessment - Part2: Performing an assessment.
3. The Cabinet Office. (2011). IT Infrastructure Library - Service Strategy. The Stationery Office Edition.
4. The Cabinet Office. (2011). IT Infrastructure Library - Service Design. The Stationery Office Edition.
5. The Cabinet Office. (2011). IT Infrastructure Library - Service Transition. The Stationery Office Edition.
6. The Cabinet Office. (2011). IT Infrastructure Library - Service Operation. The Stationery Office Edition.
7. The Cabinet Office. (2011). IT Infrastructure Library - Continual Service Improvement. The Stationery Office Edition.
8. Hilbert, R., & Renault, A. (2007). Assessing IT Service Management Processes with AIDA - Experience Feedback. EuroSPI². Postdam.
9. Jokela, J. (2009). Long Term Utilisation of SPICE in an IT Service Company. SPICE 2009, (pp. 117-122). Turku.
10. Renault, A., & Barafort, B. (2011). TIPA: 7 years experience with SPICE for IT Service Management. EuroSPI² 2011 workshop.
11. Fullilove, M. (2012). ITIL Process Maturity Assessment done right ! -- using the TIPA framework. Retrieved from Brighttalk: https://www.brighttalk.com/webcast/845/58087
12. McIntyre, J. (2012). Operational Excellence - Transforming IT using TIPA Assessments to Drive Change. A Third Sky Inc. white paper.
13. Tremblay, S. (2012). Process Maturity Assessment Done Right. Paper presented during Session 308 of itSMF USA Conference (Fusion' 12). Dallas, USA.
14. Worthington, J. (2013). Business-Driven IT Transformation - The Business Value of Process Assessment. A Third Sky Inc. white paper.
15. SEI. (2000). ARC, Assessement Requirement for CMMI, Version 1.0.
16. SEI. (2011). ARC, Assessment Requirements for CMMI, Version 1.3.
17. ISO/IEC 15504-7: Information technology -- Process assessment -- Part 7: Assessment of organizational maturity. (2008).
18. ISO/IEC 33001: Information technology -- Process assessment -- Concepts and terminology. (2014).
19. ISO/IEC 33002: Information technology -- Process assessment -- Requirements for performing process assessment. (2014). Information Technology — Process assessment — Requirements for performing process assessment.
20. ISO/IEC 33003: Information technology -- Process assessment -- Requirements for process measurement frameworks. (2014). Information Technology — Process assessment — Requirements for process measurement frameworks.
21. ISO/IEC 33004: Information technology -- Process assessment -- Requirements for process reference, process assessment and maturity models. (2014). Information Technology — Process assessment — Requirements for process reference, process assessment and maturity models.
22. ISO/IEC 33020: Information technology -- Process assessment -- Process measurement framework for assessment of process capability. (2014).
23. McCaffery, F., Taylor, P. And Coleman, G., 'Adept: A Unified Assessment Method for Small Software Companies', *IEEE Software*, Jan/Feb 2007, pp. 24-31.

Author CVs

Stéphane Cortina

Stéphane works at the Public Research Centre Henri Tudor since 2007. This R&D Engineer is involved on research projects related to normalization, IT service management, business continuity management, and process assessment and improvement. He holds a Master degree in Software Engineering from the University Joseph Fourier of Grenoble. He is also a certified ITIL Expert, an ISO/IEC 15504 Assessor, a TIPA certified Lead Assessor, member of Luxembourg chapter of itSMF, and member of ILNAS network (The Luxembourg Institute for Normalization, Accreditation, Product Safety and Quality).

Michel Picard

Since 2004, Michel works at the Public Research Centre Henri Tudor in Luxemburg as R&D engineer for the "Business Process & Service Engineering" Unit (PROSERV). He works on projects related to the modelling and assessment of the Operational Risk Management (Basel II). Michel also works on projects related to normalization, IT service management and processes assessment and improvement. He obtained his Master degree in Computer Science from the Facultés Universitaires Notre-Dame de la Paix (FUNDP) in Namur (Belgium). He is also a certified ITIL Expert, an ISO/IEC 15504 Assessor, a certified TIPA Lead Assessor, member of Luxembourg chapter of itSMF, and member of ILNAS network (The Luxembourg Institute for Normalization, Accreditation, Product Safety and Quality).

Valérie Betry

Project Manager at CRP Henri Tudor (Luxembourg) since 2003. She graduated from ICN Business School and holds a Master's Degree in Information System Design from the University of Nancy2. She has worked as a developer, and IT business analyst within the Arthur Andersen internal IT development team prior to her current position. At CRP Henri Tudor, she is involved as a project manager on various research projects related to process assessment and improvement. She leads and takes part in consulting assignments for clients on IT process assessment, process improvement, project management method implementation, and software package selection. Her major competences include project and portfolio management, quality management, process assessment and improvement, as well as software package selection and implementation.

Where does all this waste come from?

Wolfgang Raschke[1], Massimiliano Zilli[1], Johannes Loinig[2],
Christian Steger[1] and Christian Kreiner[1]

[1] Institute for Technical Informatics, Graz University of Technology, Austria
{wolfgang.raschke, massimiliano.zilli, steger, christian.kreiner}@tugraz.at
[2] NXP Semiconductors Austria GmbH, Gratkorn, Austria
johannes.loinig@nxp.com

Abstract

Agile development processes are more flexible than conventional ones. They emphasize iterative development and learning over feedback loops. Nevertheless, we experienced some pitfalls in the application of agile processes in dependable software systems. We present here the experiences we gathered in the construction of high-quality industrial software. Moreover, we will digest our experiences into a conceptual model of waste creation. This model will be refined to a case study where we take appropriate measurements in order to provide empirical evidence for it. Finally, we discuss the implications of the developed model, which helps to estimate the trade-off between agile and traditional software processes.

Keywords

Software Process Improvement, Innovation, Agile Processes

1 Introduction

Today's dependable systems face the challenge of being built as safe, secure and reliable systems. The construction of such systems utilizes quality assurance methodologies such as code reviews, testing and static code analysis. Industry-relevant frameworks for process-based quality assurance exist for process maturity (SPICE [1]) and security (Common Criteria [2]). A higher capability level in such a framework implies more processes accomplished at a higher level of maturity. Generally, a higher level of assurance comes with a higher development effort.

We investigated the development process of a highly secure software system. This project is developed with agile methodologies: Test-Driven Development [3], Continuous Integration [4] and Scrum [5]. These agile processes have considerable benefits in the industrial software project. Nevertheless, we observed some inconsistencies with regards to waste. Waste is an agile terminology, which denotes unnecessary work, which does not add to customer value [6]. We believe that it can be traced back to the agile way of working in the context of dependable systems.

In order to build a suitable model for software development in dependable systems, we discuss a pattern of evolution, which has been observed in diverse industries. First, we apply this pattern to a general model for dependable systems. Second, we refine the dependable systems pattern in order to create a case study, which will give us the opportunity to provide evidence.

We will discuss the implications which our model has on various levels of granularity and explain them regarding the agile development model and the V-model. Moreover, we show how our model can be used as framework of considering software development processes on a high level of abstraction. Finally, we conclude our findings and provide guidelines for scoping agile versus traditional processes.

2 Experience From an Industrial Project

We gained experience from an industrial software project during the development of a highly secure system. It is imperative that evidence be provided for the purposes of security via certification. The so-called *Common Criteria* [2] is a documentation-based approach. To guarantee high levels of assurance, this evidence is the result of carefully designed and accomplished processes. A great amount of effort has been put into composing the appropriate documentation for these processes.

We investigated the development process through interviews with software architects and developers. As a reference model, we utilized the V-model. Within the reengineered V-model, we identified two agile iterative cycles. Our observations are outlined in Figure 1.
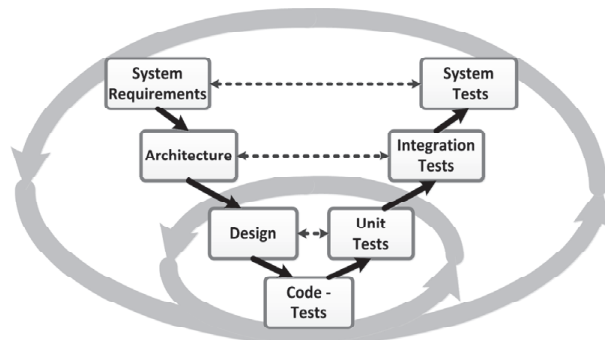
Figure 1: Iterative cycles in architecture and on component level.

The inner cycle is an iterative loop that represents the agile development process that mainly focuses on the design and implementation of user stories. Since we are applying Test-Driven Development (TDD), writing tests belongs to the code-and-test phase. Conversely, the execution of the tests is part of the unit test phase. The design comprises of the specification of the component and its interface at component level. It is allowed to change interfaces and connections between components in this inner cycle. It pushes the outer architectural cycle. This is usually regarded as a bottom-up design methodology.

The outer cycle represents the development of system requirements, architecture, integration tests and system verification (certification) in the V-model. As mentioned above, the outer cycle is driven by the inner cycle, which means that the architecture is first established incrementally at code and design level and then explicitly defined in the architecture phase. The system requirements do not evolve directly out of the inner circle but the linkages between them and the architecture have to be permanently maintained.

We found that there is a friction between both cycles, namely in the form of synchronizing implementation level artifacts (code and design) with architectural artifacts. Such a friction occurs in several manifestations which we describe in the following.

First, changing interfaces affects testing. Thus, unit tests and integration tests have to be adopted according to the new interfaces. We will show this effect later in more detail.

Second, changing interfaces have an impact on security evaluation because interface descriptions are part of the respective documentation.

Third, some interfaces have to be known to link the components together because different programming languages are utilized. Hence, interfaces have to be reworked in the product integration, as well.

Finally, we conclude that changing an interface causes rework in at least three obvious cases. The real cost cannot be measured reliably.

3 Model of Change Impact

In order to investigate potential causes for waste, we introduce a conceptual model of an innovation lifecycle. We apply the pattern described in [7] because it is simple and allows mapping to diverse circumstances.

3.1 Product Innovation vs. Process Innovation

Abernathy [7] describes a lifecycle, where at the beginning of a new development, product innovation is accomplished at a high pace and process innovation on a relatively low one. Gradually, product innovation decreases its rate whereas process innovation increases. There is a lag between product innovation and process innovation. It is important to understand the meaning of *product* and *process* innovation, which we will clarify in the following.

3.2 Product Innovation

Product innovation is the way a product is altered. Generally, product innovation is the change of *what* we build. Henderson [8] defines a model of innovation, where the rate of changing links between components is an important indicator. The change of links can be seen as the change of interfaces. In the following, we will use the term architectural evolution to label the change of interfaces

3.3   Process Innovation

Process innovation defines *how* we build products. This how usually includes all processes that embrace quality engineering, such as reviews, tests and static code analysis. It is likely that a change in an interface causes rework in the quality processes. A common practice is to increase the rate of process improvement when the interfaces are more or less stable [7].

3.4   Mapping to Software Terminology

In order to facilitate the construction of a hypothesis, which can be evaluated empirically in a software project, we map the previously outlined terminology to software engineering methodologies.

Product innovation is the sum of all activities that immediately affect a product. In a typical software project, these activities are: architecture, design and coding.

We assume architectural evolution as a quantifiable indicator for product innovation. We define architectural evolution here as the rate of interface changes between at least two components. For example, in the architecture phase of the V-model, the rate of interface changes is high. In design and coding, this rate is, in comparison, low.

Process Innovation is the aggregate of all activities that indirectly affect a product. In a software project these activities are usually: unit testing, integration testing, system testing, code review and static code analysis. In order to facilitate the interpretation we assume that process innovation is equal to all quality activities in a software project. The measurable quantity is the amount of quality activities at a certain point in time.

3.5   Definition of Waste

Ikonen [6] defines waste as "basically everything that does not add to the customer value of a product". This viewpoint regards everything except coding as waste, because it does not directly add value to the customer. This expectation is too narrow in our opinion, because testing and quality assurance add value indirectly. So, we regard waste as activities that neither *directly* nor *indirectly* add value.

3.6   Relation between Waste, Product and Process Artifacts

Our observations in the industrial case study suggest that waste is proportional to the evolution of the architecture in terms of changed interfaces *dA*. The definitions of the terms are given in Table 1. The following formula states the generic model of waste creation:

$$W = \sum_i c_i \cdot dA = Q \cdot dA$$

Definition	Interpretation	Short
Architectural evolution	Change of component interfaces	dA
Waste	Effort with no additional value	W
Quality processes effort	Total effort for *all* quality activities	Q
Constant	Coupling factor *no. i*	c_i

Table 1: Terminology of the generic model of waste creation.

4 Evaluation With Empirical Data

The empirical evaluation has been conducted in cooperation with the company *NXP Semiconductors*. We evaluated a software project which is dedicated to the implementation of a highly-secure embedded system. About 100 developers participate at the project. The project includes several development teams, a testing team and a dedicated security team. The number of tests is typical for a high-quality embedded system of medium size: approximately 80 modules are tested by several thousand tests (unit tests, integration tests and acceptance tests).

In order to apply the previously mentioned model to real data, we refine it in the following. We measured the number of failing test cases W. In addition, we recorded the number of changed interfaces *dA* on a daily basis. We provide the explanation of the terms in Table 2. Finally, we can construct a hypothesis, which can be evaluated with empirical data:

$$W_t = c_t \cdot dA$$

Definition	Interpretation	Short
Architectural evolution	Change of component interfaces	dA
Waste	Number of failing test cases	W
Constant	Coupling factor	c_t

Table 2: Terminology of the refined model which is evaluated empirically.

Change of component interfaces: this is the accumulated number of changes in interfaces per day. This number has been measured on a daily basis, seven days a week. This information was retrieved from the source code repository.

Number of failing test cases: the number of failing test cases has been measured on a daily basis, seven days a week. The tests have been automatically executed each day on a continuous integration server. All test results have been logged.

We apply pre-processing to the data, which describes the changing of interfaces for the following reasons: First, there are remarkably few changing of interfaces on Saturdays and Sundays. Hence, there is a fundamental oscillation on a weekly basis. Second, the time span between altering an interface to the effect on the number of failing tests is not constant. This variation of time from cause to effect exists, because each and every interface is not immediately integrated into the tested product.

In order to mitigate the mentioned effects, we apply a staged pre-processing to the recorded data, which is described in the following:

Step 0: Raw Data

For the raw data, the Cross-Correlation Coefficient (*CCC*) equals 0.16.

Step 1: Constant Moving Average Filter

In order to smooth the vector *dA*, we apply a simple moving average with a length of *N=7*:

$$dA_{avg}(n) = \frac{1}{N} \sum_{j=0}^{N-1} dA(n+j)$$

The *CCC* equals 0.51 and thus is remarkably higher, as before. We explain this by the filtering out of peaks that occur on weekends.

Step 2: Vector Norm

Both vectors *W* and *dA* have considerably different scales. In order to apply vector normalization, we divide both vectors by their length and obtain the respective unit vectors.

$$dAnorm = \frac{dAavg}{\|dAavg\|} \quad , \quad Wnorm = \frac{Wt}{\|Wt\|}$$

The normalization does not affect the Cross-Correlation Coefficient. The normalized signals are shown in Figure 2.
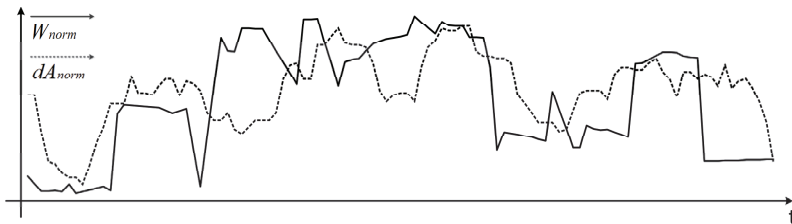
Figure 2: *Wnorm* and *dAnorm* after the Normalization.

Step 3: Dynamic Time Warping

In order to cope with the time variation, we apply *dynamic time warping* [9, 10], a non-linear signal processing algorithm. Dynamic time warping compares two signals and locally stretches and compresses the time axis in order to find an optimal alignment between both. A good alignment is characterized by a high similarity of both data series. The warping path *p* is a timely mapping between *dAavg* and *Wnorm* which is computed by the dynamic time warping algorithm:

$$p = dtw(dAavg, Wnorm)$$

The time mapping p allows the reconstruction of the signal *dArec*:

$$dArec(t) = dAnorm(p(t))$$

The *CCC* equals 0.87 and shows that there is a high correlation between the pre-processed data. The reconstructed signal *dArec* and the reference signal *Wnorm* now show a remarkable visual similarity, as can be seen in Figure 3. Thus, we can assume a proportional relation between both signals.
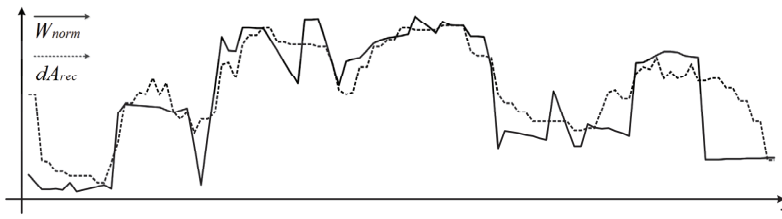
Figure 3: After data processing step 3, the signals *Wnorm* and *dArec* are obviously similar.

5 *Implications*

In the presented study, there is correlation between the evolution of the architecture and failing test cases. This undermines the general hypothesis that the evolution of the architecture causes rework in quality processes. The implications can be applied to the V-model and agile development processes.

5.1 Implications for the V-Model

The V-model reflects Abernathy's pattern of innovation [7] where in a cycle of experimentation, learning and refinement, an architecture is defined. In this architecture phase, there are a high amount of changing interfaces (see dotted line in Figure 4 a) but quality processes are not unfolded to full maturity (see continuous line in Figure 4 a). So, the product of both curves is in the middle range (see Figure 4 b).

Certainly, there occurs a point in time, where architectural activities slow down. In the V-model this is usually the phase of implementation, which is the pivotal element of the V and thus has connections to architectural activities *and* to quality processes. In this phase, architecture *and* quality processes are assumed to operate at a medium velocity. The product of both is high in this phase. In the third phase, the interfaces are mostly stable and quality assurance is at its highest level of the whole lifecycle. The product of both curves is in the middle range.

For a better illustration of the curves in Figure 4, we apply sample values in Table 3. The sum of a column represents the architectural evolution (*dA*), effort for quality processes (*Q*) and the resulting total waste (*W*) of the model.
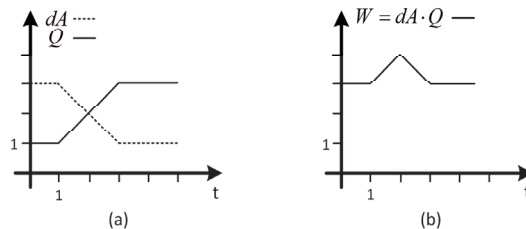
Figure 4: Coherence between waste *W*, changing interfaces *dA* and total effort for quality processes *Q* in the V-model.

5.2 Implications for the Agile Model

In agile processes, the sequence of process steps is abandoned. Rather, the activities are accomplished concurrently in iterative cycles. In such an iterative cycle, each activity (architecting, coding, testing and verification) has to be performed. We assume for architectural evolution (see dotted line in Figure 5 a) and for quality activities (see continuous line in Figure 5 a) a constant and medium velocity. As can be seen in Figure 5 b, the product of both is constantly very high. Therefore, the interface changes create huge effort because all quality activities in the current iteration are immediately affected.
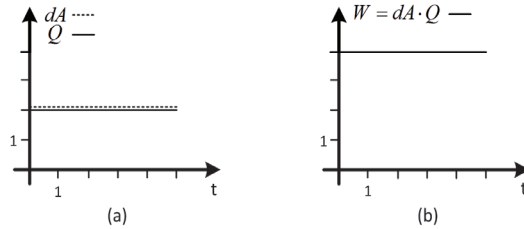
Figure 5: Coherence between waste *W*, changing interfaces *dA* and total effort for quality processes *Q* in the V-model.

For a better illustration of the curves in Figure 5, we apply sample values in Table 3. The sum of a column represents the agile evolution (*dA*), the agile effort for quality processes (*Q*) and the resulting total waste (*W*) of the model.

6 Conclusion

In Table 3, the effort for architecture and processes is equal (Sum = 10). This allows a comparison of the resulting waste: for agile working practices, the model predicts a higher resulting waste compared to the V-model. Of course, the presented model of waste creation is an abstraction. In reality there is some amount of approximation in the application of such a model. Nevertheless, the theoretical model is suitable for deliberation. Such deliberation suggests the following conclusion: agile methods are more appropriate for software projects with low demand for quality and possibly only acceptance tests. Traditional processes, like the V-model process is more appropriate in software projects with a high demand for quality and respective activities.

Time	V-model *dA*	V-model *Q*	V-model *W*	Agile *dA*	Agile *Q*	Agile *W*
0	3	1	3	2	2	4
1	3	1	3	2	2	4
2	2	2	4	2	2	4
3	1	3	3	2	2	4
4	1	3	3	2	2	4
Sum	10	10	16	10	10	20

Table 3: Sample data for the model of waste creation for the V-model and agile processes.

7 Acknowledgment

8 Literature

1. Messnarz R., et al. Integrated Automotive SPICE and safety assessments. Software Process: Im-provement and Practice, 14(5):279–288, 2009.

2. Common Criteria. Common Criteria for Information Technology Security Evaluation - Part 1-3. Version 3.1 Revision 3 Final, 2009

3. Janzen D., Saiedian H. Test-driven development: Concepts, taxonomy, and future direction. Computer Science and Software Engineering, 2005.

4. Duvall P., Matyas S., Glover A. Continuous integration: improving software quality and reducing risk. Pearson Education, 2007.

5. Cohn M. Succeeding with Agile. Pearson Education, 2009.

6. Ikonen M., Kettunen P., Oza N. Exploring the sources of waste in Kanban software development pro-jects. Conference on Software Engineering and Advanced Applications, SEAA 2010.

7. Abernathy W. J., Utterback J. M. Patterns of Industrial Innovation. Technology review, 64: 254-28, 1978.

8. Henderson R., Clark K. Architectural innovation: the reconfiguration of existing product technologies and the failure of established firms. Administrative science quarterly, 9-30, 1990.

9. Giorgino T. Computing and visualizing dynamic time warping alignments in R: the dtw package. Journal of statistical Software, 31(7):1-24 , 2009.

10. Müller M. Dynamic Time Warping. Information retrieval for music and motion, 69–84, Springer, 2007

9 Author CVs

Wolfgang Raschke

He completed his studies in telematics in 2012. Since February 2012 he works as a PhD student at the Graz University of Technology. His current research interests are: software product lines for secure systems, innovation and evolution of software and software models.

Massimiliano Zilli

He received his Master's degree in Electronic Engineering at the University of Udine in 2007. From 2008 to 2012 he worked as a software developer for embedded systems. Since February 2012 he works as a PhD-Student at the Graz University of Technology. His research focuses on optimization techniques for embedded systems.

Johannes Loinig

received the Dr.techn. degree (PhD) in electrical engineering with focus on Secure Embedded Systems from Graz University of Technology in 2012. Currently he works as software and security architect in several projects at NXP Semiconductors Austria and is in the lead of several research projects with focus on HW/SW codesign and information security.

Christian Steger

received the Dipl.-Ing. degree (M.Sc.) 1990 and the Dr.techn. degree (PhD) in electrical engineering from Graz, University of Technology, Austria, in 1995, respectively. He is key researcher at the Virtual Vehicle Competence Center (VIF, COMET K2) in Graz, Austria. From 1989 to 1991 he was software trainer and consultant at SPC Computer Training GmbH, Vienna. Since 1992, he has been Assistant Professor at the Institute for Technical Informatics, Graz University of Technology. He heads the HW/SW codesign group at the Institude for Technical Informatics. His research interests include embedded systems, HW/SW codesign, HW/SW coverification, SOC, power awareness, smart cards UHF RFID systems, multi-DSPs.

Christian Kreiner

graduated and received a PhD degree in Electrical Engineering from Graz University of Technology in 1991 and 1999, respectively. From 1999 to 2007 he served as the head of the R&D department at Salomon Automation, Austria, focusing on software architecture, technologies, and processes for logistics software systems. He was in charge to establish a company-wide software product line development process and headed the product development team. During that time, he lead and coordinated a long-term research programme together with the Institute for Technical Informatics of Graz University of Technology. There, he currently leads the Industrial Informatics and Model-based Architectures group. His research interests include systems and and software engineering, software technology, and process improvement.

EXPERIENCE REPORT: OPPORTUNITIES AT LACK OF SOFTWARE QUALITY MEASUREMENT

Onur Kaynak[1], Ünal Aksu[2]

[1]*Innova Information Technologies, Quality Assurance Team Leader*
[2]*Innova Information Technologies, Quality Assurance Consultant*
okaynak@innova.com.tr, uaksu@innova.com.tr

Abstract

This paper explains the experience of an IT Solutions Provider company about how the lack of quality measurement creates opportunity to improve its quality management system and quality management infrastructure. As a result of costs due to lack of quality analysis, JIRA platform based Quality Assurance System was developed. The collected metrics were used to deploy ISO / IEC 15504 SPICE based Process Improvement Projects.

Keywords

Software Process Improvement, Cost of Quality, ISO / IEC 15504, SPICE, Lack of Quality Measurement

1 Introduction

INNOVA Information Technologies is the Turkey's one of the greatest IT solution provider which develops IT oriented software products and services for the national and international customers. INNOVA IT Solutions consist of various business directorates such as; Kiosk & Automation, Financial Transactions and Applications, IT Operations, Enterprise Resource Management, Public Solutions, Information Technology Solutions and Telecommunication. All of these directorates give services to national & international customers.
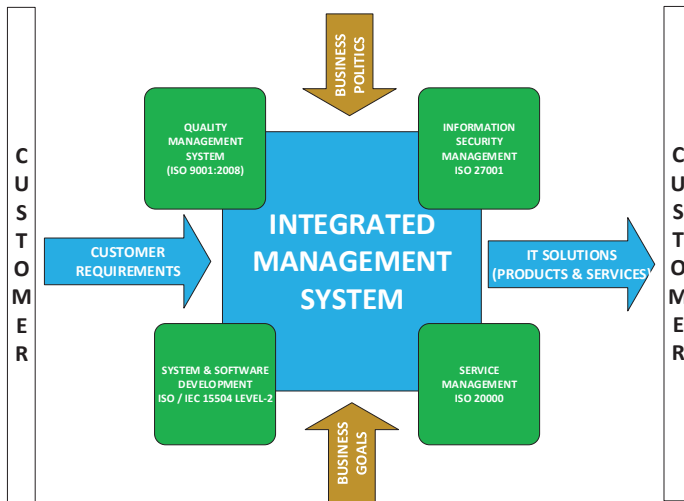
Figure 1: INNOVA Integrated Management System Diagram

INNOVA IT Solutions; plans, applies, checks and improves various management system standards. Those standards were directly related with company's directorates working strategy such as; ISO 9001:2008 [3] quality management system for whole company, ISO / IEC 27001 [2] information security management for IT Operations directorate, ISO / IEC 20000 [4] service management for IT Operations and IT Solutions directorates, ISO / IEC 15504 Level 2 [1] system and software development lifecycle related directorates. As a result; in order to handle those standards' different point of views, they were combined to form the integrated management system.

1.1 Motivation

The integration of ISO / IEC 20000, ISO / IEC 27001, ISO / IEC 15504 Level-2 and ISO 9001:2008 management system comes with the difficulty at process improvement phase of the plan-do-check-adjust cycle. Although each standard has its own specific processes, there is an intersection between other standards processes as well. In order to measure the processes' efficiencies and how to improve them, an integrated process improvement program has been initiated at 2012.

Besides, at the end of 2013, the company strategy revised as; "lower the costs and increase the profitability". But in order to achieve that, it is important to identify the inefficient process and initiate process improvement projects.

The best way to respond those challenges quality management decides to measure the costs due to

lack of software quality. Lack of software quality is calculated in terms of man-hour and man-hour transform into money easily. As the companies' point of view income and expense oriented it will be easy to supply the necessary support for the process improvement projects from the executives. But there were certainties at how to measure the lack of software quality, which metrics were needed to be collected and how those metrics were analyzed, what kind of infrastructure was needed to collect and store the metrics and at last how those measurements transform into process improvement proposals in a software process improvement project.

In order to measure lack of software quality, JIRA platform based Quality Assurance System project was initiated and the opportunities that were identified assessed under directorate based process improvement projects.

2 INNOVA Cost of Quality Project

2.1 Scope of the Project

INNOVA IT Solutions has ISO /IEC 15504 Level 2 maturity. In 2013 during a recertification audit meeting the QA team presented a trend analysis report based reviews, findings and non-conformances. In the report there are some outliers on the graph. At that points corrective and preventive actions applied and trend of non-conformances started decreasing. Based on that trend analysis the lead auditor advised the QA team to spread that trend analysis to company and enrich it as a cost of quality program. Then QA team planned a cost of quality project to initiate software process improvement project in the company. In order to initiate a process improvement project and get commitment from top management measurement is essential.. As Watts S. Humphrey has mentioned, "If you don't know where you are, a map won't help" [5]. In order to determine the organization's position and get commitment from management, lack of quality measurement project is performed. To institutionalize measurement and improvement a branch is selected to perform the project. During the project failure costs are collected and reported as evidence to show improvement is needed.

2.2 Phases of the Project

2.2.1 Plan the Lack of Quality Measurement

Lack of quality measurement project is planned to calculate failure costs in 2013 in Telecommunication branch. Appraisal and prevention costs calculation will be future work of this project. Failure costs will be collected from JIRA used in the company for tracking issues. Both internal and external defects are managed by JIRA. Type and size of the projects were same in the branch because there is only one customer. That's why collect in pilot projects and estimate for others based on pilots' average strategy is selected for the branch. Project is planned as two phases the first one is to select pilot projects and modify defect workflow to collect failure metrics in pilot projects. The second phase is to estimate other projects based on pilot projects' metric averages.

2.2.2 Define the Metrics

Failure metrics are collected during this work. Failure metrics are divided in two categories as internal and external. Internal failure costs are the cost of defects which are identified and fixed before releasing the project into live. External failure costs are the cost of defects which are identified and fixed after the project is in live. During the cost of quality project, failure metrics will be collected based on defect severity.

2.2.3 Define the Method How to Collect Metrics

Failure metrics are collected from JIRA issue tracking application. A separator called security level defined in JIRA to identify whether a defect is occurred in production or before... Security level has two option categories; the first one is Customer and the other one is Project. Customer level defects will be analyzed as external and Project level defects will be internal.

Effort spent on a defect is collected based on entered effort values at defect workflow execution which is indicated below. The workflow defines the lifecycle of a defect which goes from creation to closure. The workflow tracks the defect with respect to its states and transitions. During each state transition the spent time is recorded.
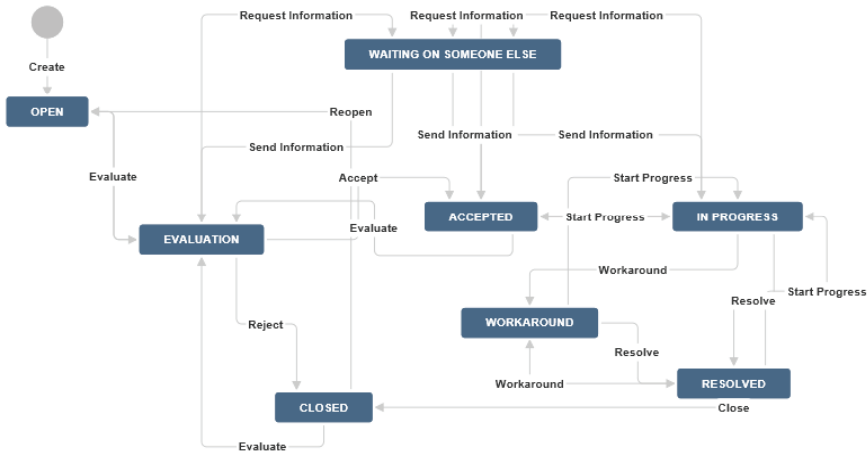
Figure 2: Defect Workflow

2.2.4 Collect the Metrics & Analyze

Pilot projects' internal and external defects are analyzed according to severity level and resolution category by Quality Assurance team. Firstly, defects are grouped by severity levels. Then defects in a severity level distributed according to resolution type criteria. Average time spent for each resolution type and severity level is calculated in pilot projects. Based on pilot projects' average time metrics, cost of defects in the branch is estimated. Table 2 shows defect fix costs by severity levels. In table 2, severity 1 defects mean that blocks the running of the software. Severity 2 means that it causes crash, data loss or memory leak problem. Severity 3 refers to major functionality is not working. Severity 4 defects are minor feature problems or cosmetic issues.

Analyzed defect numbers in pilot projects and total in Telecommunication branch are shown in Table 1.

Table 1: Defect Fix Costs by Severity Levels

Metrics Collected in Projects	# of Internal Defects	# of External Defects
Pilot (2 Projects)	3120	1767
Telecommunication Branch (25 Projects)	5436	2974

Table 1 shows that 57% of internal defects and 59% of external defects distributed in two pilot projects for Telecommunication branch.

Table 2: Defect Fix Costs by Severity Levels in Pilot Projects

Defect Category	Defect Severity	Defect Cost (man-hour)
Internal	Severity 1	19.04
Internal	Severity 2	17.23
Internal	Severity 3	15.23
Internal	Severity 4	9.06
External	Severity 1	42.34
External	Severity 2	34.67
External	Severity 3	30.78
External	Severity 4	24.81

Table 2 shows that cost of and external defect is more than two times of an internal failure defect.

Based on pilot projects Telecommunication branch projects failure costs are estimated. Estimated failure costs by security levels are shown in Table 3.

Table 3: Defect Fix Costs by Severity Levels in Telecommunication Branch Projects

Defect Category	Defect Severity	Defect Cost (man-hour)
Internal	Severity 1	586.2
Internal	Severity 2	715.22
Internal	Severity 3	31566.92
Internal	Severity 4	2558.56
External	Severity 1	4396.58
External	Severity 2	7664.5
External	Severity 3	12915.668
External	Severity 4	1154.331

Table 3 shows that 89% of time spent for internal defects is in Severity 3 level type issues. That means project teams spent most of time for Severity Level 3 issues in Telecommunication branch. . In addition to that, 49% of time spent for external defects is for Severity 3 level ones.

Total failure cost of defects are calculated and converted to man-year metric. Man-year metric conversion is done with the assumption; there are 8 hours in a working day and 22 working days in a month. Results are shown below in table 4.

Table 4: Total Failure Costs in Telecommunication Branch Projects

Defect Category	Defect Cost (man-year)
Internal	16.77
External	12.37
Total	29.14

Table 4 shows that 58% of total failure cost is for internal failure costs. Internal failure costs are more than external failure costs for Telecommunication branch. That shows the branch is detecting defects at internal activities and fixing before production but it needs improvement.

2.3 Opportunity 1: Conduction of JIRA Quality Assurance System

Cost of quality is not the price of quality sauce that added while preparing a product or a service. It is the cost of poor quality of the product or service. Total cost of quality is the total of quality costs occurred during investigating prevention of non-conformance to requirements, appraising a product or service to check conformance to requirements and failures to meet requirements.

Failure costs are calculated using the lack of quality measurement project. Prevention and appraisal costs need be calculated to see the total cost of quality for Telecommunication branch. In order to calculate prevention and appraisal costs automatically and correctly a quality assurance activity management application (JQAS) is planned which will be executed as a plugin in JIRA. Technical Review, Non-Conformance Management, Corrective and Preventive Action (CAPA) Management and Internal Audit processes will be performed on JQAS.

In order to perform internal audits and technical reviews' there will be checklists including control item and assessment scale for each type of audit. Checklists will be loaded based on selected reference standard on audit issue create.. Auditor will ask and control according to control item in the checklist and write her/his finding and select performing level from the scale in audit meeting. At the end of the audit, auditor will select control item/s and click create CAPA button and JQAS will show new CAPA form having filled fields according to the audit issue and findings on the control item/s in checklist. JQAS will allow to create a CAPA from more than one control item findings based on measurement scale. In CAPA form auditor will select responsible, fill other fields and click create CAPA button. After the CAPA created it will be shown as a link near related control item/s.

Review process will nearly same as audit process in JQAS. There will also checklists but it will be populated according to work item type which will be reviewed. At the end of review workflow a NCR will be created and assigned to work item owner.

Checklists and assessment scales will be configurable in JQAS. New checklists and assessment scales can be created via a management interface. That will allow the company to adopt easily to any changes in control items and new reference standards.

In JQAS, CAPA workflow is designed as shown in Figure 3. It will make easy to track changes and states. While executing the flow time spent for status changes measurement will be collected. Also, there will be escalation to management option to get commitment.
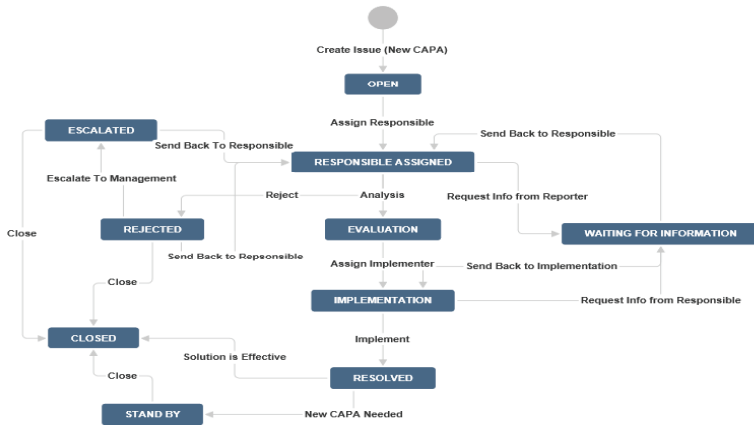
Figure 3: Corrective and Preventive Action Workflow

JQAS will have creating CAPA from non-conformance record (NCR) feature. If there is any reoccurring non-conformance than a CAPA will be created to make root cause analysis and provide and apply permanent solution. As same in CAPA, there will be escalation mechanism. NCR flow is shown in Figure 4.
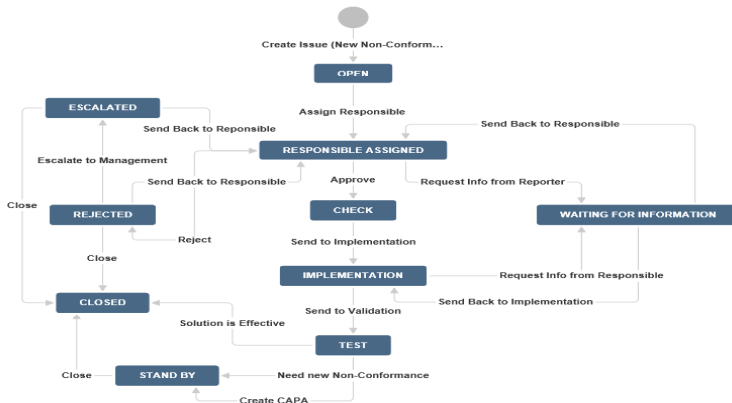
Figure 4: Non-Conformance Record Workflow

JAQS will have dashboard to track quality assurance system instantly. The dashboard will include metrics and their trends in graphical reports listed below:

- Time spent for CAPAs
- Time spent for NCRs
- Time spent for audits
- Time spent for reviews
- Number of CAPAs (and also by state)

- Number of NCRs (and also by state)
- Number of CAPAs per audit control item
- Number of NCRs by review control item
- Trend of CAPAs
- Trend of NCRs

2.4   Opportunity 2: Telecommunication Branch Based Software Process Improvement Project

Software process improvement is like changing the tires of a moving car. You should keep on going the right way and the right lane without any stop. To be alive and successful in these conditions software process improvement must planned as a project. As shown above from quality measurement results software process improvement is inevitable in Telecommunication branch. After publishing results to the top management at the end of 2013, the top management gave commitment to start a software process improvement (SPI) project.

A software process improvement project is started in Telecommunication branch referencing ISO/IEC 15504 (SPICE). The SPI project is planned as two phases. In the first phase engineering and management processes' as-is execution with current software infrastructure will be defined. Secondly, defined processes and software infrastructure will be improved based on ISO/IEC 15504 Level 3 processes. In parallel with the SPI project, cost of quality metrics will be collected and tracked as dashboard of the SPI project.

Processes in the scope of the SPI project is are listed in below.

- Requirements Management
- Software Design
- Software Construction
- Software Integration
- Testing
- Quality Assurance

- Configuration Management
- Change Request Management
- Problem Resolution Management
- Project Management
- Risk Management
- Measurement and Analysis

The SPI project will take 377 calendar days. And, there will 24 team members taking different roles during the project. The SPI project will end with and internal audit at the end of the second phase.

3 Conclusions

In this paper we shared our experience about using failure costs to initiate a cost of quality project and later software process improvement project. We identified opportunities collecting metrics due to the lack of quality measurement. We listed cost of quality project steps and the opportunities that we realized. JAQS is presented as a solution to collect appraisal and prevention costs automatically. As more projects start to be tracked on JQAS, there will be more valuable and consistent data to assess the cost of quality. Besides, the accurate cost of quality data will be beneficial for software process improvement projects efficiency which will be initiated on other branches of the INNOVA IT Solutions.

4 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] Humphrey Watts, "Managing the Software Process", SEI, 1989

5 Author CV's

Onur KAYNAK, MSc.

Onur KAYNAK is currently Quality Assurance Team Leader at INNOVA IT Solutions. He holds BSc. degree from Bilkent University Computer Technology and Information Systems department and MSc. degree from Middle East Technical University Software Management program. He is still studying at Middle East Technical University Information System Ph.D. program. He took part at projects as software quality engineer/quality assurance consultant in defense and telecommunication industry. He is leading the company wide process improvement initiatives and quality assurance activities. He is interested in Quality Management, Process Improvement, Configuration Management, System Engineering and Cost of Quality.

Ünal AKSU, MSc.

Ünal AKSU is currently Quality Assurance Consultant at INNOVA IT Solutions. He holds MSc. degree from Middle East Technical University Software Management program and BSc. degree Ege University Computer Engineering department. He is still studying at Middle East Technical University Information System Ph.D. program. He took part in many projects as application development consultant and quality assurance consultant. He is interested in Process Improvement, Quality Assurance, Requirements Management, Object Oriented Software Development and Cost of Quality.

Identifying Correlations of Findings for Building Process Improvement Packages Using Graph Clustering

Su-jin Choi[1] and Dae-Kyoo Kim[2] Sooyong Park[1]

[1] *Sogang University, Seoul, South Korea,*
{sujinchoi, sypark}@sogang.ac.kr
[2] *Oakland University, Rochester, MI 48309, USA,*
kim2@oakland.edu

Abstract

Software process improvement (SPI) begins with process assessment based on a process reference model such as CMMI. Process improvement action items in SPI are determined according to the identified strengths and weaknesses of the current practice. Therefore, given that a list of assessment findings has been identified, it is important to analyze correlations of findings and identify relevant findings for building improvement items. However, correlation analysis requires expertise and considerable efforts, which makes it difficult for practitioners to perform it in process improvement projects. In this work, we present a CMMI-based method for identifying correlations of findings and building improvement packages using graphs clustering techniques. We evaluate the method using industrial data.

Keywords

Software process improvement, Process improvement items, Process correlation analysis, Decision support, Over-lapping clustering

1 Introduction

Software process improvement (SPI) is pervasive in software industry for its proven effects on product quality and productivity [1–4]. SPI starts with improvement planning where the current practice of software development is understood and the improvement goals to achieve are determined [5]. A software development process is assessed to identify its strengths and weaknesses based on a process reference model (e.g., Capability Maturity Model Integration (CMMI)) [6–8]. The results of assessment are then used as the base for defining improvement items.

Process assessment usually focuses on individual activities to identify strengths and weaknesses. However, identifying effective process improvement items requires not only individual activities, but their relationships to be analyzed [9-11]. Activity relationships capture dependencies of activities which should be considered in defining improvement items. For instance, the requirements specification activity is highly related to the test case specification activity in that the former provides inputs to the latter.

Analyzing activity relationships requires expertise and considerable efforts, which makes it difficult for practitioners to carry out. There are some studies (e.g., [9–12]) on identifying relationships of activities and improvement items. The existing work, however, largely depends on practitioner's expertise and manual work.

In this paper, we present a graph-based approach for identifying correlations of findings and grouping them into improvement packages in an SPI project. In the approach, we define an initial process correlation model specifying correlations of practices based on CMMI [14], a widely used process reference model. The initial model developed is refined using a graph-based similarity analysis technique [19]. The model is then used to identify correlations of given findings. Based on identified correlations, a set of improvement packages are built using a graph-based clustering algorithm. We evaluate the approach by applying it to an industrial SPI project and comparing the results to manual outputs.

The paper is organized as follows. Section 2 gives an overview of related work, Section 3 details the proposed method, Section 4 presents evaluation results, and Section 5 concludes the paper with the future work.

2 Related Work

Several researchers studied on identifying relations of activities and improvement items. Calvo-Manzano *et al*. [15] present an implementation sequence of process areas at maturity level 2 in CMMI for Acquisition (CMMI-ASQ) which is a reference model for providing guidelines for acquiring products and services [13]. Dependencies between process areas are obtained by analyzing the formal CMMI-ASQ documentation and then the obtained dependencies are represented in a directed graph. Based on the analysis of dependencies, the process areas are grouped into clusters and implementation sequence alternatives are provided. Arcilla *et al*. [16] present a similar work to Manzano *et al*.'s work, but their work is based on *Information Technology Infrastructure Library (ITIL)* which is a set of best practices for IT Service Management [18]. Monteiro *et al*. [17] identify dependencies among CMMI process areas to understand their impact on the implementation at the maturity level 2. They also consider process areas from maturity level 3. While, the existing work provides a guidance on how to elicit dependency information from a reference model, they focus on the process area-level with little attention to the practice-level from which practices are derived.

Chen *et al*. [11] introduce a practice dependency model for CMMI using six process areas at level 2. Dependencies between practices are identified via the flow of work products between practices based on a textual analysis of the CMMI specification. Their work is similar to our work in that our work also uses CMMI as the base. However, we further employ graph clustering techniques for packaging relations.

Choi *et al*. [10] propose a method for building a CMMI-based process correlation model based on his-

torical data on findings correlation analysis. To apply the method, empirical data should be available as a prerequisite, which may limit the application in practice.

Gorscheck and Wolin [9] present DAIIPS, a method for packaging and prioritizing improvement items by analyzing their dependencies. Their work focuses on improvement items rather than assessment findings. Dependencies of improvement items are determined by votes of participants, which heavily relies on human involvement and consequently is subjective.

Villal´on *et al.* [12] propose an improvement items specification template called Action Package for describing organizational, technical, and management aspects of process improvements. The template consists of twelve items such as policy, training, and metrics. While the template provides general factors to be considered in improvement identification, it does not reflect assessment findings which are the fundamentals in SPI. Also, the template is designed for experts, providing little details as to how the items should be filled out, which makes it difficult for the less experienced to use.

3   Approach

In this section, we describe a method for identifying correlations of findings and building improvement packages using graph clustering techniques. Figure 1 shows an overview of the method. We first define a process correlation model based on CMMI and refine it through similarity analysis [19]. The model is then applied to a set of assessment findings to identify their correlations. Based on identified correlations, findings are grouped to build improvement packages (IPs) using clustering algorithms [20,21].
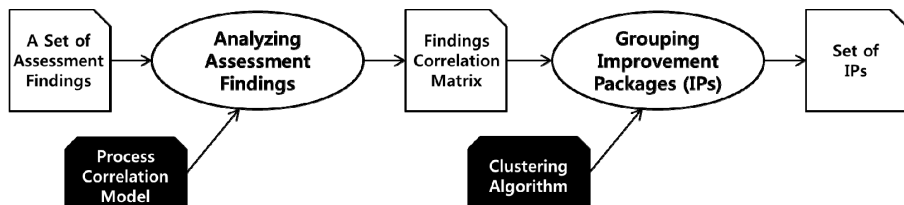
Figure 1: Overview of Method

3.1   Defining Process Correlation Model

CMMI, a widely practiced process assessment model, provides guidelines for assessing a software development process in terms of maturity and capability levels. It describes process elements (e.g., process areas, practices) and how they are structured and related over software development. We use CMMI to define a process correlation model that allows one to systematically identify correlations of assessment findings. Due to the inherent generality of CMMI, the proposed model can be used for different projects in various domains.

The correlation model is defined based upon the following principle – Two practices are correlated if 1) one practice provides work products as inputs to the other practice or 2) one practice specifies a reference to the other practice. CMMI provides reference information in the "*Related Process Areas*" section per process area and in "*refer to*" statements in the description of process area, which are sometimes inconsistent. In this work, we use "*refer to*" statements describing relationships of practices on which this work focuses. Figure 2 illustrates identifying correlations of practices from the CMMI specification and their representation in a matrix. In this work, we consider 590 of 9,730 pairs of 140 specific practices in 18 process areas in CMMI. Figure 2(a) shows that the two practice CM_SP1.1 and CMSP2.2 are correlated via work products and MA_SP1.1 and PMC_SP1.1 are correlated by the reference information specified as 'refer to' statements. The information on correlations of practices is depicted as a symmetric matrix as shown in Figure 2(b).

(a) CMMI specification

(b) Process Correlation Matrix
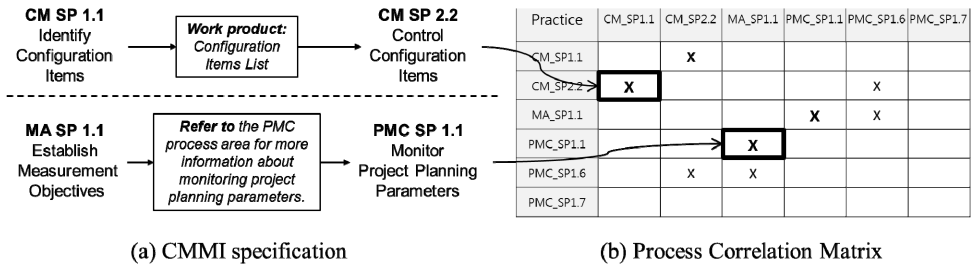
Figure 2: Building Initial Process Correlation Matrix

The correlation model is refined through similarity analysis of practices. In the analysis, the correlation matrix in Figure 2(b) is represented in an undirected correlation graph where vertices represent practices and edges represent relations. Figure 3 shows an example. The graph in Figure 3(a) shows five practices and their correlations. The graph is represented as an adjacency matrix in Figure 3 (b) where 1 represents the existence of relation while 0 represents no relation.

(a) Graph $G = (V, E)$

(b) Adjacency Matrix of G
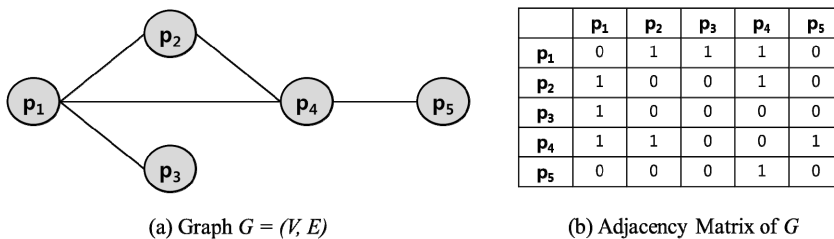
Figure 3: Correlation Graph

The graph is then refined by analyzing similarity between practices using their adjacency information [16]. The similarity of two practices is measured by the degree to which their neighborhoods overlap in the graph. The following defines the metric for measuring similarity

$$s(p_1, p_2) = \frac{\mid N(p_1) \bigcap N(p_2) \mid}{\mid N(p_1) \bigcup N(p_2) \mid}$$

where $N(p_i)$ denotes the set of adjacent practices of a practice p_i. The metric produces a value ranging from 0 to 1 where zero represents that the two practices have no adjacent vertices shared while one represents that all of their adjacent vertices are shared. For example, in Figure 3(a), the vertices $p2$ and $p3$ have similarity of 0.5 as they have one vertex ($\{p1\}$) shared out of two vertices ($\{p1, p4\}$). Figure 4(a) shows the symmetric similarity matrix of the adjacency matrix in Figure 3(b). A threshold may be used to refine the results. The similarities in bold box in Figure 3(a) are identified additionally by the threshold set to 0.5. The lower the threshold is, the higher the number of correlated practice pairs is found. Figure 4(b) shows the refined graph where the bolded edges capture the additionally identified similarity.

	p₁	p₂	p₃	p₄	p₅
p₁	1.00	0.25	0.00	0.20	0.33
p₂	0.25	1.00	0.50	0.25	0.50
p₃	0.00	0.50	1.00	0.33	0.00
p₄	0.20	0.25	0.33	1.00	0.00
p₅	0.33	0.50	0.00	0.00	1.00

(a) Similarity Matrix of G (b) Refined Graph $G' = (V, E')$
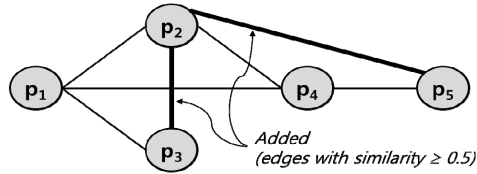
Figure 4: Refining Correlation Graph of Figure 3(a)

A major advantage of the correlation model is its generality which allows the model to be used in various contexts. The model is defined at the practice level and can be instantiated by mapping practices to findings.

3.2   Analyzing Assessment Findings

The correlation model in Figure 4(b) is used to identify correlations of assessment findings, which helps prioritizing findings. Findings that have more related findings have a greater impact and thus, may be given a higher one-to-one mapping and thus, the mapping is rather straightforward. As an example, consider the mapping in Figure 5(a). Given the mapping, the model in Figure 4(b) is instantiated as shown in Figure 5(b). Note that the vertex *p2* is not instantiated as there are no findings mapped to the practice. Also note that findings *{f1,f2}* are mapped to the same practice *p1* and thus, they have the same set of edges. Since they are mapped to the same practices, they can be considered as related each other. Figure 5(c) shows a findings correlation matrix corresponding to the graph in Figure 5(c).

Finding	Practice
f₁	p₁
f₂	
f₃	p₃
f₄	P₄
f₅	p₅

	f₁	f₂	f₃	f₄	f₅
f₁	0	1	1	1	0
f₂	1	0	1	0	0
f₃	1	1	0	0	0
f₄	1	0	0	0	1
f₅	0	0	0	1	0

	f₁	f₂	f₃	f₄	f₅
f₁	3	0	0	0	0
f₂	0	2	0	0	0
f₃	0	0	2	0	0
f₄	0	0	0	2	0
f₅	0	0	0	0	1

(a) Finding-Practice Mapping table   (b) Findings Correlation Graph   (c) Findings Correlation Matrix   (d) Findings Degree Matrix

Figure 5: Correlation Analysis of Findings

Figure 5(d) shows the diagonal degree matrix representing the number of edges for each vertex in the graph in Figure 5(b). In the matrix, it is observed that *f4* has the highest number of edges which implies a large impact and thus, should be considered with a high priority. Note that an assessment finding is an instance of a practice and thus, relationships of findings are often more specific than relationships of practices.

3.3   Grouping Improvement Packages

Based on the graph resulting from Subsection 3.2, findings are grouped to build improvement packages (IPs) using clustering techniques. An improvement package (IP) is a set of correlated assessment finings, helping practitioners define improvement items.

By the inclusive nature of IPs, the same finding may belong to multiple IPs. To accommodate the nature, we use star clustering which groups elements into star-shaped clusters with overlaps [20]. The vertex having the highest degree in a graph lends itself as the center of a star and its adjacent vertices are clustered. The same is carried out for the next highest degree vertex that has not been covered and it repeats until all vertices are covered. Figure 6 illustrates finding clustering. The correlation model in Figure 4(b) is used to identify correlations of assessment findings, which helps

(a) Set of star centers = $\{f_1\}$ (b) Set of star centers = $\{f_1, f_5\}$ (c) Set of star centers = $\{f_1, f_4\}$
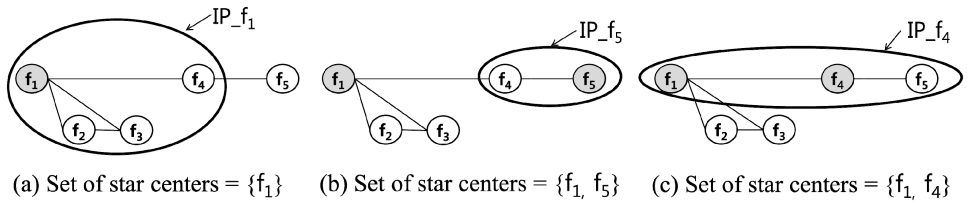
Figure 6: Star Clusters

Figure 6(a) shows the formation of an IP *IP f1* with f1 as the star center denoted in gray which has the highest degree in Figure 5(d). Note that f5 in Figure 6(a) has not been covered. In the next iteration, *f5* is chosen as the star center of another IP *IP f5* in Figure 6(b) which covers every vertex. However, this is not the only solution for the complete coverage. Figure 6(c) shows an alternative where *f4* is used as the star center. We recommend using as the star center a finding that belongs to a different process area than those of the previous star centers. Per the recommendation, Figure 6(b) is preferred to Figure 6(c) since if *f1* and *f4* belong to the same process area although the degree of *f4* is higher than f5.

IPs are defined sequentially and even if the next IP that does not add new coverage to the previous coverage, the IP is still considered as valid as long as it is not a subset of previously defined IPs and the star center (practice) of the IP has not been used as a star center of another IP. To reduce overlaps, we prioritize star centers by the degree and the relative density of a vertex [21]. The relative density of a vertex measures the degree to which the vertex is related to other vertices that have a lower degree than or equal to the target vertex. The relative density RD of a vertex v is measured as follows:

$$RD(v) = \frac{\mid AdjLowerDegree(v) \mid}{\mid Adj(v) \mid}$$

where *Adj(v)* is the set of adjacent vertices of *v* and *AdjLowerDegree(v)* denotes the set of adjacent vertices of *v* that have a degree less than or equal to the degree of *v*. A higher relative density implies that the vertex has a higher likelihood of increasing the coverage if it is chosen as the star center. The following constraints are enforced on defining IPs:

a. A finding that has no adjacent findings forms itself an IP and becomes the star center of the IP.

b. An IP which does not increase the coverage can be selected unless it is not the subset of the previously created IPs. This condition allows the selection of IP.

c. A finding that belongs to a different PA from those of previously selected center findings has priority.

d. A finding that has a higher degree is given a higher priority.

e. A finding that has a higher relative density is given a higher priority.

Algorithm 1 describes the clustering algorithm implementing star clustering. The goal is to define IPs that together cover all findings with less overlaps while having its own stand point.

```
Algorithm 1 Grouping Improvement Packages
1   Input:    G=(F,E) where  F is the set of findings and  E is the set of edges
2             FindingDescSet (id, processArea, findingDescription)
3   Output: ImprovementPackageSet
4   Let ImprovementPackageSet be the empty set;
5   Let coveredList be the set of findings as marked  'Uncovered';
6   buildImprovementPackage (G, FindingDescSet): ImprovementPackageSet {
7       Compute the degree of each finding in F; Compute the density of each finding in F;
8       /* First round   */
9       Add the degree and the density of each finding;
10      Store the results in baseList in descending order;
11      For each finding f in baseList {
12          if (isCoveredPA( f) = false) AND (isSubset( f) = false)  {
13              append f and f.adjacent into ImprovementPackageSet;
14              update f and f.adjacent as 'Covered' in coveredList; }
15      }
16      /* Second round       */
17      IF isAllCovered(coveredList) = False {
18          Multiply the degree by the density of the uncovered finding in coveredList.
19          Store the results in tempList in descending order;
20          /* Do while non-covered finding exists in tempList */
21          For each finding  f in tempList {
22              if( f has no adjacent finding) OR
23                 ( f has at least one adjacent finding which is not covered yet) {
24                  append f and f.adjacent into ImprovementPackageSet;
25                  update f and f.adjacent as  'Covered' in coveredList;
26                  delete f and f.adjacent from tempList; } }
27          }
28      return improvementPackageSet;
29  }
30  /* isCoveredPA(finding) returns true if the PA of the finding is covered */
31  /* isSubset(finding) returns true if the finding-centered group is a subset
32       of any of already identified improvement packages */
33  /* isAllCovered(coveredList) returns true if every finding in coveredList
34       is marked as  'Covered' */
```

The algorithm takes a correlation graph as an input and produces improvement packages as defined in line 1-3. Line 7 computes the degree and the relative density of each finding in the input graph. Lines 8-15 create an improvement package per PA as the first iteration. To prioritize findings, the degree and the density are added stored in *baseList* in the descending order in line 9-10. In the case where two findings have the same degree, the one that has a higher density is selected. Line 11-14 fetch each finding from *baseList* and form an improvement package with the adjacent findings of the fetched finding if the process area of the fetched finding has not been covered and the formed IP is not a subset of the previously defined IPs. The second iteration starts in line 16-26 to identify any additional IPs from those findings that do not belong to any IPs. Relative density is considered heavier by multiplying the degree by the relative density line 18-19 in order to increase coverage and decrease overlaps. Findings are then prioritized by the value of the multiplication. Lines 21-27 fetch each finding from *tempList* and forms an IP if the fetched finding has at least one adjacent finding that has not been covered. In case of no adjacent findings, the fetched finding itself becomes an IP.

Priority may be assigned to identified IPs based on their scope. For instance, An IP covering more findings and process areas may be given a higher priority for implementation since it is concerned with cross-cutting issues in the organization.

4 Evaluation

In this section, we describe the results of the evaluation where the presented method is applied to a data set collected from a completed SPI project. The data set include findings, improvement packages, and improvement items. Findings were drawn from the assessment targeting CMMI Maturity Level 2 and correlation-based improvement packages were manually identified based on the findings. Using the identified improvement packages, improvement items were built and implemented. The implementation of improvement items were led by the guidance provided by two external process experts having about 18 - 20 years of experience. The target organization in the project is a small-sized software organization of 15 employees developing automotive black box systems and lane departure warning systems. The SPI project identified 18 weaknesses in assessment from six process areas including Requirement Management (REQM), Project Planning (PP), Project Monitoring and Control (PMC), Measurement and Analysis (MA), Configuration Management (CM), and Product and Process Quality Assurance (PPQA). Table 1 shows the findings.

Table 1: Assessment Findings

Finding	Finding Description	Practice
CM-01	Identify configuration items, components, and related work products to be placed under configuration management.	CM_SP1.1
CM-02	Establish and maintain a configuration management and change management system for controlling work products.	CM_SP1.2
CM-03	Control changes to configuration items.	CM_SP2.2
CM-04	Perform configuration audits to maintain the integrity of configuration baselines.	CM_SP3.2
MA-01	Establish and maintain measurement objectives derived from identified information needs and objectives.	MA_SP1.1
PMC-01	Monitor actual values of project planning parameters against the project plan.	PMC_SP1.1
PMC-05	Monitor the management of project data against the project plan.	PMC_SP1.4
PMC-02	Periodically review the project's progress, performance, and issues.	PMC_SP1.6
PMC-03	Review the project's accomplishments and results at selected project milestones.	PMC_SP1.7
PMC-04	Review the project's accomplishments and results at selected project milestones.	PMC_SP1.7
PMC-06	Manage corrective actions to closure.	PMC_SP2.3
PP-02	Establish a top-level work breakdown structure (WBS) to estimate the scope of the project.	PP_SP1.1
PP-03	Establish and maintain estimates of work product and task attributes.	PP_SP1.2
PP-01	Estimate the project's effort and cost for work products and tasks based on estimation rationale.	PP_SP1.4
PPQA-01	Objectively evaluate selected performed processes against applicable process descriptions, standards, and procedures.	PPQA_SP1.1
REQM-01	Manage changes to requirements as they evolve during the project.	REQM_SP1.3
REQM-02	Manage changes to requirements as they evolve during the project.	REQM_SP1.3
REQM-03	Maintain bidirectional traceability among requirements and the work products.	REQM_SP1.4

From the findings, six IPs were identified manually in the project as shown in Table 2. As shown in the table, about 13 to 16 findings belong to each IP and description represents the improvement items from the six process areas.

Table 2: Manually Identified IPs

IP	Num. of findings	Description	IP	Num. of findings	Description
IP_a	16	Improve configuration management	IP_d	13	Improve project progress management
IP_b	16	Establish standard project management process	IP_e	13	Establish measurement and analysis
IP_c	15	Establish quality assurance process	IP_f	13	Improve requirements traceability change management

The process correlation model produced by the presented method is applied to the finding set of the project to identify correlations of the findings and the identified correlations are used for defining IPs by applying the clustering algorithm. We compare the resulting IPs are to the manually built six IPs in the project with comments from the two experts who participated in the project.

Table 3: Process Correlation Models

Models (similarity)	Number of correlated pairs	Ratio of correlated pairs	Difference (Mn - Mn-1)
M0 (initial ver.)	590	6%	590
M1 (≥0.7)	604	6%	14
M2 (≥0.5)	651	7%	47
M3 (≥0.3)	765	8%	114
M4 (≥0.2)	957	10%	192
M5 (≥0.)	1601	16%	644
M6 (≥0.05)	1601	16%	0

Table 3 shows produced process correlation models - M0 to M6 varying by similarity threshold. Models are built upon an analysis of 140 practices from 18 process areas at CMMI maturity level 2 and 3. In the table, M0 is the base model on which the similarity analysis is conducted. It has 590 correlated pairs out of total 9,730 pairs, which accounts for 6%. M1 refines M0 using a similarity threshold 0.7 and it defines 14 additional correlated practice pairs to M0. We evaluate the precision, recall, F-measure, and accuracy of the models using industrial data. In the evaluation, the output correlations of the models are compared to those that are identified manually by experts. Table 4 shows the results of the evaluation for models M0 to M5. M6 is not included in the evaluation it has the same number of identified correlations as M5 (see Table 3).

Table 4: Evaluation Results for Process Correlation Models

Models	Precision		Recall		F measure		Accuracy	
	average	variance	average	variance	average	variance	average	variance
M0	0.64	0.06	0.25	0.01	0.36	0.02	0.67	0.02
M1	0.65	0.06	0.27	0.01	0.37	0.01	0.67	0.02
M2	0.64	0.05	0.29	0.01	0.39	0.02	0.67	0.02
M3	0.64	0.05	0.30	0.01	0.41	0.02	0.68	0.02
M4	0.62	0.04	0.37	0.01	0.46	0.02	0.68	0.02
M5	0.59	0.05	0.46	0.01	0.50	0.01	0.67	0.02

The table shows that M1 has the highest precision of 65% which means that 65% of model-identified correlations are also identified by the manual analysis. On the other hand, M5 has the highest recall of 46% which means that 46% of manually-identified correlations are also identified by the model. F-measure is the harmonic mean of precision and recall. 68% accuracy of M3 and M4 means that 68% of correlation decisions by the models are the same as the manual decisions.

(a) Findings Correlation Matrix

(b) Findings Correlation Graph

Figure 7: Findings Correlation Analysis

We use M4 in Table 4 to demonstrate identifying correlations using the findings in Table 1 where 18 findings are mapped to 16 practices across six process areas. M4 with similarity threshold 0.2 is chosen for its good evaluation results in terms of accuracy and F-measure. Figure 7(a) shows the resulting correlation matrix which is represented in a graph in Figure 7(b) where the findings grouped in the circle represent an IP with CM-03 as its star center. Based on the resulting correlation matrix in Figure 7, six IPs are identified using the clustering algorithm in Subsection 3.3. Table 5 shows the identified IPs and their description with the star center and the degree and relative density of the star center.

Table 5: Method-Identified IPs

IP	Center finding	Degree (A)	Relative Density (B)	A+B	Subject
IP_1	PMC-02	6	1.0	7	Improve project progress management
IP_2	MA-01	6	1.0	7.0	Establish measuremnet and analysis
IP_3	CM-03	6	1.0	7.0	Improve configuration management
IP_4	REQM-01	5	1.0	6	Improve requirements traceability change management
	PP-03	4	0.5	4.5	
IP_5	PPQA-01	0	0	0	Establish quality assurance process
IP_6	PMC-06	3	0	3	Improve issue management

In the table, IP_1 is composed of the star center PMC-02 from the project monitoring and control process area and other six adjacent findings. In the first round of clustering, a finding with the highest value of degree + relative density is selected per process area and becomes the star center as long as its IP is not a subset of previously defined IPs. Given that, IP_2 through IP_4 following IP_1 are defined for each process area. For the planning process area, PP-03 has the highest value (4.5). However, it is not a valid center because its IP is a subset of IP 2. In IP_5, the degree of the center finding PPQA-01 is zero and the center finding itself constitutes the IP. By the end of the first round, all the findings have been covered except PMC-06. In the second round, PMC-06 which does not belong to the IPs identified in the first round is considered as the start center of the next IP. PMC-06 is found to have degree 3 and thus, the finding and its adjacent three findings form another IP (IP_6).

Finding	Method-identified IPs						Manually Identified IPs					
	IP_1	IP_2	IP_3	IP_4	IP_5	IP_6	IP_a	IP_b	IP_c	IP_d	IP_e	IP_f
CM-01			X	X			X	X	X	X		X
CM-02			X	X			X	X	X		X	X
CM-03	X		C	X			C	X	X	X	X	X
CM-04			X	X			X	X				X
MA-01	X	C					X	X		X	C	X
PMC-01	X	X					X	X	X		X	
PMC-02	C	X	X				X			X		
PMC-03	X					X	X	X	X	C	X	X
PMC-04	X					X	X	X	X		X	
PMC-05	X					X	X	X	X			X
PMC-06						C		X	X	X		
PP-01		X					X	X	X	X		
PP-02		X						X	X	X	X	X
PP-03		X					X	C	X	X	X	X
PPQA-01					C		X	X	C	X		X
REQM-01			X	C			X	X	X	X	X	X
REQM-02			X	X			X	X	X	X	X	X
REQM-03		X		X			X	X	X	X	X	C

* C: center finding, X: adjacent finding

Figure 8: Comparing Involved Findings of IPs

Figure 8 compares involved findings of the IPs identified using the presented method to those of the IPs identified manually in the project where C denotes a center finding and X indicates an involved finding. From the comparison, the following observations are made: 1) the improvement items identified in both practices are similar and in each practice, one or two improvements are identified for each process area. 2) The average number of involved findings in the method-driven practice is 5.5 while the corresponding number in the manual practice is 14.3. This may imply that either the method-driven practice has missing correlations or the manual practice overly analyzed correlations. We observe both. Each IP in the manual practices covers 79% of findings in average, which is too inclusive. On the other hand, the method-driven practice identifies PMC-06 as a separate IP (IP_6) while the same finding is considered to be part of a multiple-finding IP (IP_d) which seems to be more appropriate in semantics. Such a semantic delicacy is hard to be addressed in a mechanical method and can be easily reflected in a review. 3) With respect to IP overlaps, a finding in the method-driven practice belongs 1.7 IPs out of 6 IPs while a finding in the manual practice belongs to 4.8 IPs out of 6 IPs. This implies that the stand point of the IPs in the manual practice is not as clear as the IPs in the method-driven practice.

5 Conclusion

We has presented a method for identifying correlations of findings and clustering them to build IPs. The correlation model in the method takes as an input a set of assessment findings and produces a matrix of finding correlations which is input to the clustering algorithm to produce IPs. In the evaluation, the presented method produces less-overlapped IPs each having its own stand point compared to manually identified IPs. While IPs produced by the method may not address all the semantic delicacies in finding correlations, which is unavoidable, they provide a constructive basis for identifying improvement items. In this work, we observe that the CMMI specification is ambiguous in describing practice relationships which might be intended for generality. In the future, we plan to study on use of field data together with CMMI in building the correlation model and how field data may complement CMMI.

6 Acknowledgements

7 Literature

1. Clarke, P., O'Connor, R.V.: An empirical examination of the extent of software process improvement in software SMEs. Journal of Software: Evolution and Process 25(9) (2013) 981–998

2. Staples, M., Niazi, M.: Systematic review of organizational motivations for adopting CMM-based SPI. Information and Software Technology 50(7–8) (2008) 605–620

3. Swinarski, M., Parente, D.H., Kishore, R.: Do small IT firms benefit from higher process capability? Communications of the ACM 55(7) (July 2012) 129–134

4. Unterkalmsteiner, M., Gorschek, T., Islam, A.K.M.M., Cheng, C.K., Permadi, R.B., Feldt, R.: Evalua-tion and measurement of software process improvement - a systematic literature review. IEEE Transactions on Software Engineering 38(2) (2012) 398–424

5. Humphrey, W.: Introduction to Software Process Improvement. Technical Report CMU/SEI-92-TR-007, Software Engineering Inst., Carnegie Mellon Univ. (1992)

6. Herbsleb, J., Carleton, A., Rozum, J., Siegel, J., Zubrow, D.: Benefits of CMM-based software process improvement: Initial results. CMU/SEI-94-TR-013, Software Engineering Inst., Carnegie Mellon Univ. (1994)

7. Kitson, D., Humphrey, W.: The role of assessment in software process improvement. Technical Report CMU/SEI- 89-TR-003, Software Engineering Inst., Carnegie Mellon Univ. (1989)

8. McFeeley, B.: IDEAL: A user's guide for software process improvement. Technical Report CMU/SEI-96-HB-001, Software Engineering Inst., Carnegie Mellon University (1996)

9. Gorschek, T., Wohlin, C.: Packaging software process improvement issues: a method and a case study. Software: Practice and Experience 34(14) (2004) 1311–1344

10. Choi, S., Kim, D.K., Park, S.: Process Correlation Analysis Model for Process Improvement Identification. The Scientific World Journal (2014)

11. Chen, X., Staples, M., Bannerman, P.L.: Analysis of Dependencies between Specific Practices in CMMI Maturity Level 2. In: EuroSPI. (2008) 94–105

12. Calvo-Manzano Villal´on, J.A., Cuevas Agust´ın, G., San Feliu Gilabert, T., De Amescua Seco, A., Garc´ıa S´anchez, L., P´erez Cota, M.: Experiences in the application of software process improve-ment in SMES. Software Quality Control 10(3) (November 2002) 261–273

13. CMMI Product Team: CMMI for Acquisition, Version 1.3, Technical Report CMU/SEI-2010-TR-032, Software Engineering Inst., Carnegie Mellon University (2010)

14. CMMI Product Team: CMMI for Development, Version 1.3. Technical Report CMU/SEI-2010-TR-033, Software Engineering Inst., Carnegie Mellon University (2010)

15. Calvo-Manzano, J., Cuevas, G., Mejia, J., San Feliu, T., Sanchez, A.: Cmmi-acq: A formal implementation sequences of the processes areas at maturity level 2. In: Electronics, Robotics and Automotive Mechanics Conference, 2008. CERMA '08. (Sept 2008) 212–217

16. Arcilla, M., Calvo-Manzano, J.A., Agust´ın, G.C., G´omez, G., Ruiz, E., Gilabert, T.S.F.: A solution for establishing the information technology service management processes implementation sequence. In: EuroSPI. (2008) 106–116

17. Monteiro, P., Machado, R., Kazman, R., Henriques, C.: Dependency analysis between CMMI process areas. In: Product-Focused Software Process Improvement. (2010) 263–275

18. Arraj V.: ITIL: the basics. (2013) http://www.best-management-practice.com/gempdf/ITIL_The_Basics.pdf

19. Schaeffer, S.E.: Graph clustering. Computer Science Review 1(1) (2007) 27–64

20. Aslam, J.A., Pelekhov, K., Rus, D.: Static and Dynamic Information Organization with Star Clusters. In: CIKM. (1998) 208–217

21. Su´arez, A.P., Trinidad, J.F.M., Carrasco-Ochoa, J.A., Medina-Pagola, J.E.: OClustR: A new graph-based algorithm for overlapping clustering. Neurocomputing 121 (2013) 234–247

8 Author CVs

Su-jin Choi

She is a PhD candidate in software engineering at the Graduate School of Sogang University and a software quality engineering specialist in Printing Solution Division, Samsung Electronics. Her primary research interests and experience areas are software and systems process improvement, requirements engineering and product-line engineering. Choi received her masters degree in software engineering from the Graduate School of Information and Technology of Sogang University. Contact her at sujinchoi@sogang.ac.kr.

Dae-Kyoo Kim

He is an associate professor of the Department of Computer Science and Engineering at Oakland University. He received the Ph.D. in computer science from Colorado State University in 2004. He worked as a technical specialist at NASA\Ames Research Center in 2003 and as a senior software engineer at McHugh Software International from 1997 to 2000. His research interests include UML modeling, smart grid, design pattern formalization, model refactoring, software architecture modeling, access control modeling, and software process. He is a senior member of the IEEE Computer Society.

Sooyong Park

He is a professor in Sogang University's Computer Science Department. His research interests include requirements engineering, self-adaptive software for unmanned systems, and dynamic software product line engineering. Park received his PhD in information technology from George Mason University. He's the president of the National IT Industry Promotion Agency in Korea and a steering committee member of the Asian-Pacific Software Engineering Conference. Contact him at sypark@sogang.ac.kr.

Change Strategy for ISO/IEC 33014: A multi-case study on which change strategies were chosen

Jan Pries-Heje[1] and Jørn Johansen[2]

[1] Roskilde University, Roskilde, Denmark
janph@ruc.dk
[2] DELTA, Hørsholm, Denmark
joj@delta.dk

Abstract

In the newly published ISO/IEC 33014.2013 [1] standard there is a strategic activity called "Identify the overall change strategy" that includes selecting a change strategy "from among a myriad of available change models". The book [2] on the ImprovAbility model describes a framework of how to select change strategy. There are 10 different change strategies to choose from. But which ones are chosen in practice? To answer that we have analysed data from 49 assessments in 44 organizations that have used the framework. We give a ranking of strategies chosen and we analyse how they adapt the change strategy to their specific conditions. We conclude that the most often recommended organizational change strategy is Optionality followed by three other strategies: Socializing, Learning-driven, and Specialist-driven.

Keywords

Organisational change, change strategy, ISO/IEC 33014, process improvement

Reference

Experience with teaching and applying Process Appraisals at a University

Jozef De Man Ghent University, Alcatel-Lucent Belgium

Abstract

Software Development mainly happens in small and very small enterprises. Establishing process improvement initiatives in such organizations is challenging because prevailing software improvement frameworks are perceived to be expensive and oriented towards larger companies. Much research has been done in adapting the standard frameworks to make them more suitable for small organizations. Light-weight techniques have been developed to reduce the cost of appraisals. In this paper we report experience with teaching and applying a light-weight appraisal approach in a university context. Focus has been on educational aspects but we believe that the approach can also be deployed in small and very small enterprises.

Keywords

CMMI, SCAMPI, Software Process Improvement, Appraisals

1 Introduction

Most software development is estimated to happen in very small enterprises [1]. Implementing a process improvement program in such organizations presents particular challenges. That has sparked considerable research in developing light-weight appraisals [2][4][5] and in demonstrating the applicability of the prevailing improvement frameworks in smaller entities using agile development techniques [7]. As a matter of fact, the application to smaller organizations has become an important consideration in the evolution of the standard frameworks. To facilitate the introduction of process improvement practices in small (start-up) companies it is important that these practices are taught in the software curricula of software engineering schools.

In this paper we report on experience with teaching and applying a light-weight appraisal method based on the Standard CMMI Appraisal Method for Process Improvement (SCAMPI) [6] using the Capability Maturity Model Integration (CMMI®).

2 Organizational Context

The training in software appraisals is part of a one-semester Software Management course at Ghent University. Because we want students to acquire the skills to perform appraisals the training is performed by means of a practical exercise in which they perform a real appraisal on a real organizational unit.

Most students who follow the Software Management course also follow a Software Development Project course in which they gain experience with the project management aspects of software development. The projects are executed under guidance of university staff and using the university infrastructure. The projects apply Scrum for planning, monitoring and control. They are formally organized with a project manager and various other roles allocated to members of the team. Every two weeks they have to formally report progress to the supervising professor.

That set of projects and their organizational context is the subject of the appraisal exercise. Although most students follow the two courses mentioned above that is not a prerequisite for successfully participating in the appraisal exercise. The exercise is organized in such a way that students who are not involved in a project can still take part in the appraisal without any problems.

3 Background

We use the Capability Maturity Model Integration for Development (CMMI-DEV 1.3) as global improvement framework in the course because it provides a structure to understand the relationship between many other frameworks, methods and techniques [8].

The CMMI-DEV 1.3 identifies 22 Process Areas. Process Areas are classified in four Categories (Process Management, Project Management, Engineering and Support) and in four Maturity Levels (2 - Managed, 3 – Defined, 4 – Quantitatively Managed and 5 – Optimizing). A Process Area involves Specific and Generic Goals with their associated Specific and Generic Practices. The Practices define requirements for the processes implementing the Process Areas. They specify *what* needs to be done, not *how*. Specific Practices are specific for each Process Area and occur only once. Generic Practices are common to all Process Areas. The Generic Practices contribute to the so-called institutionalization of the process ensuring it becomes engrained in the organization and part of its culture.

The appraisal method is based on the Standard CMMI Appraisal Method for Process Improvement (SCAMPI) [6]. SCAMPI comes in three classes of which Class C is the least restrictive and least expensive. We use a tailored Class C SCAMPI approach.

Improvement frameworks can be classified as inductive or prescriptive [4]. Inductive frameworks focus on improvements based on an understanding of the current strengths and weaknesses of the organi-

zation. Prescriptive frameworks include a predefined set of practices that must be satisfied by a development process. Although it has been argued [4] that CMMI is prescriptive we believe that the implementation of the Organizational Process Focus Process Area using e.g. the SCAMPI Class C appraisals provides an excellent balance between induction and prescription. Inductive approaches tend to be less valuable in low-maturity organizations where evaluations will typically suggest implementation of the practices associated with Maturity Level 2 Process Areas. As the maturity of the organization increases CMMI/SCAMPI can be used for objective-driven performance improvement as well [3].

4 Appraisal Steps

The appraisal method employed in the course is a tailoring of the SCAMPI C [6]. It involves the following steps.

- Training and preparation
- Managed discovery of artifacts and creation of draft findings
- Interview sessions
- Creation of final findings

4.1   Training and Preparation

In preparation of the appraisal exercise students are educated in software improvement frameworks and the Capability Maturity Model Integration in particular. They are trained in the appraisal method and receive a template for the draft and final findings.

Because we obviously want to train the entire class, the appraisal team includes all students (30-40) who follow the class. This is an unusually large team but it enables us to divide the work and reduce the effort that must be contributed by each student. The team is split in mini-teams of 3-4 persons that must each cover a process area. This requires each student to gain an in-depth understanding of only part of the model with a relatively modest investment of time.

The scope of the appraisal includes all Maturity Level 2 process areas except Supplier Agreement Management and a selection of Maturity Level 3 process areas: Organization Process Focus and Organizational Training, Risk Management and the engineering process areas. Risk Management is included because it is also emphasized in the Software Development Project course as an instrument for pro-active project management.

All the projects executed in the context of the Software Development Project course that is taught in parallel with the Software Management course are included in the scope. Most students participate in both courses facilitating the flow of information. Students can freely subscribe to mini-teams. In some cases students not following the project course join in the same mini-team, in other cases they do not. Both approaches seem to work.

4.2   Managed Discovery of Artifacts and creation of draft Findings

We follow the approach of "Managed Discovery" to identify artifacts that provide objective evidence of the implementation of process area practices. This approach is advocated in the SCAMPI MDD [6] as being more efficient and effective than the "Discovery" or "Verification" approaches. In a discovery approach objective evidence is discovered during interview sessions with members of the organization. The approach requires relatively modest effort but is not very rigorous. In the verification approach, artifacts are collected as more reliable objective evidence in preparation of the interviews during which the evidence is verified. That approach is more rigorous but there is a risk of spending a lot of effort on collecting artifacts that may in the end not be relevant for the appraisal.

With the Managed Discovery approach artifacts are collected incrementally.

- We start with a first collection of artifacts that are expected to have a large coverage of the practices of the process areas in scope of the appraisal. It includes the project plan, development plan, test plan, quality assurance plan, configuration management plan, copies of the Scrum or Kanban boards, burn-down charts, presentations of review meetings. The project teams have to publish these artifacts in a folder on the university forum to ensure adequate visibility to all appraisal mini-teams and in particular to students who do not follow the Software Development Project course.

- The mini-teams are requested to establish a good understanding of the practices in there scope so that they can evaluate the degree of implementation by a single scan through the documents. They are not expected to give a rigorous rating but a risk level (high, medium, low) for each practice resulting in a first iteration of the draft findings.

- Artifacts are added to the collection based on the feedback of the mini-teams to incrementally improve the coverage of the draft findings.

The draft findings presentation starts with a short description of the purpose of the process area and a description of its scope in the terminology of the processes that are used for its implementation. For each practice (specific and generic) the implementation is briefly described with a reference to the corresponding artifacts.

The mini-teams must enrich the draft findings with questions they intend to ask during the interview sessions. Questions should be open-ended and be of the form: How do you perform <practice>, What training did you receive in <practice>, etc.

During this phase, all students are also requested to respond to the two dream questions:

- If you would be allowed to change one element in the organization, what would that be?
- If you would be allowed to keep one element in the organization, what would that be?

The "element" can be a procedure, way of working, tool, organizational infrastructure, etc. These questions are typically asked at the end of the interview sessions but that would in our case have taken too much valuable time. A response must therefore be submitted in the drop box of the university forum.

4.3   Interview Sessions

The draft findings developed by the mini-teams are shared with the entire appraisal team to enable preparation of the interview sessions. All students should come to the sessions with an understanding of the purpose of each process area and with answers to the questions prepared.

The interview sessions are actually plenary sessions with the entire appraisal team. Each mini-team in turn takes the lead. One member of the mini-team asks the questions and other members take notes. Answers can be given by any member of the other mini-teams. The goal is to promote information flow from the class to the mini-teams. We allocate about 20 minutes for each process area.

These sessions are actually a combination of the traditional interview sessions with the feedback sessions of the draft findings as defined in SCAMPI. By involving the entire class at once the findings are consolidated between the teams.

4.4   Creation of Final Findings

Based on the feedback received during the interview sessions including the answers to the dream questions, the mini-teams update the findings presentation and create a summary slide with the strengths of the organization, the opportunities for improvement and some suggested actions.

Each mini-team contributes the data for one column in an aggregated presentation of the appraisal results as shown in Figure 1. Example Scoring Matrix.

	CM	MA	PMC	PP	PPQA	REQM	SAM	DAR	IPM	OPD	OPF	OT	PI	RD	RSKM	TS	VAL	VER	OPP	QPM	CAR	OPM
SP 1.1	L			L	H	L					L	M	L	L	L	L	M	L				
SP 1.2	L			L	H	L					L	M	L	L	L	L	M	L				
SP 1.3	L			L		L					L	M	L		L		M	L				
SP 1.4				L							M											
SP 1.5						L																
SP 1.6																						
SP 1.7																						
SP 2.1	M			L	M						L	M	L	L	L	L	M	L				
SP 2.2	M			L	M						L	M	L	L	M	L	M	L				
SP 2.3				L								M		M			L	L				
SP 2.4				L													L					
SP 2.5				L																		
SP 2.6				L																		
SP 2.7				L																		
SP 3.1				L							L		L	L	M	L		L				
SP 3.2				L							M		L	L	M	L		L				
SP 3.3				L							M		L	L								
SP 3.4											M		L	L								
SP 3.5																		L				
GP 2.1	H	H	H	H	H	H					H	H	H	H	H	H	H	H				
GP 2.2	M	L	L	M	H	M					L	L	L	L	L	L	L					
GP 2.3	M	L	L	M	H	M					L	L	L	L	L	L	L					
GP 2.4	M	L	L	M	H	M					L	L	L	L	L	L	L					
GP 2.5	M	M	L	M	H	M					L	L	M	M	M	M	M					
GP 2.6	M	L	L	M	H	M					L	L	L	L	L	L	L					
GP 2.7	M	L	L	M	H	M					L	L	L	L	L	L	L					
GP 2.8	M	M	M	M	M	M					M	L	M	M	M	M	M					
GP 2.9	H	H	H	H	H	H					H	L	H	H	H	H	H					
GP 2.10	L	L	L	L	L	L					L	L	L	L	L	L	L					
GP 3.1																						
GP 3.2																						

Figure 1. Example Scoring Matrix

5 Lessons Learned

Feedback from students after completing the course is positive. They find the practical experience in the appraisal exercise a necessary complement to the classroom training in process improvement frameworks which is perceived to be too theoretical on its own.

Students following both the software management and project courses experience particular benefits in that they can apply the process understanding they gain already in their software project. It happens they make mistakes in the project and realize they would not have made those mistakes with a more rigorous application of the process framework practices. Some have argued that the software management course should therefore proceed the project course but that would not allow the projects to be used as object of the appraisal exercise. As a matter of fact, students should be allowed to make mistakes in their educational journey because it reinforces retention of the remedies. For that reason we have kept the two courses in parallel.

The effort to be invested in the exercise is modest. The time spent in training and the interview sessions is small and predictable. The main variance is in the data collection phase leading up to the interview sessions. We have learned that students should be better guided in this process. It is not sufficient to explain the managed discovery process and leave the students on their own in applying it. For that reason we have made the managed discovery iterations explicit and track progress ensuring more consistency in the outcome of the mini-teams. In a way we are applying an agile scrum-like approach where the draft findings are developed incrementally and issues can be discovered before reaching the interview sessions. Efficiency of this phase was improved by publishing the artifacts in a common repository separate from the data management platforms used by each of the projects. It ensure visibility to all team members and keeps focus on the set of documents that is considered to be relevant.

In the past the interview sessions were not very efficient nor effective because participants were exposed to the draft findings and questions for the first time at the sessions. As a consequence mini-teams had to spent too much time educating the audience on the purpose of the process areas and explaining the questions. By sharing the draft findings some time before the interview sessions they can gain a better understanding of the scope and issues up front. That ensures a better communica-

tion flow from the audience towards the mini-teams and makes the sessions more effective, also from an educational perspective.

The application of their recently gained knowledge in a real example brings misunderstandings and misconceptions about the framework to the surface during the course so that they can be corrected before the examination and be used to improve the course. Some commonly encountered issues are listed below. They are not much different from what we have experienced in appraisals in a commercial setting. E.g.

- Students have difficulties elaborating the generic practices in the context of their process area. E.g. "Plan the Process" and "Monitor and Control the Process" are immediately associated with planning, monitoring and controlling the *project* and not with the process implementing their process area.
- Mini-teams to which organizational process areas are assigned (e.g. Organizational Training, Organizational Process Focus) tend to look for evidence in the projects and not in the organization that is supporting the projects. The relevance of university courses and the appraisal exercise itself tend to be overlooked as evidence. That issue is addressed by collecting all relevant artifacts in a dedicated repository.

The projects apply Scrum and this presents additional challenges. It must be emphasized repeatedly that the CMMI defines *what* needs to be done (requirements) and not *how*.

- The process aspect of Process and Product Quality Assurance must be interpreted and not necessarily associated with a separate Quality function. The role of "Scrum Master" can satisfy some of the requirements.
- The engineering process areas should not automatically be associated with a waterfall approach but can very well be used to evaluate an iterative approach as well.

6 Next Steps

The main purpose of this exercise is educational. The goal of the exercise is to acquire the skills to perform an appraisal but the actual results of the appraisal are of secondary importance. So we considered efficiency and effectiveness in terms of the educational value but not the actual results.

In the future we want to explore how this approach can be improved to become part of a larger software process improvement framework where the outcome of the appraisals can be used to improve the software development process of the university. We believe the value of the approach extends beyond its educational purpose. In small organizations it is not possible to allocate resources for a full-time process team. With our approach the effort for process improvement is distributed to all members of the organization. It also ensures that responsibility is shared and the entire organization is empowered to participate in the process improvement process.

The university promotes an entrepreneurial spirit and encourages students to set-up small companies to commercialize ideas they have developed after they have completed their study. The appraisal exercise gives them the skills and tools to apply light-weight process improvement in that context. It needs to be investigated whether the approach can also be applied in a commercial context with development teams that are not as homogeneous in terms of education and experience.

We have also started now to implement the improvement actions in the organization of the projects. The results are shared with the staff that is supporting the projects to establish a recommended process and tools environment.

7   Conclusion

In this experience report we have outlined an effective and efficient approach to teaching process appraisals by executing a real appraisal on a set of projects executed by university students. The exercise is primarily intended for training and structured with that purpose but the results can be used as a basis for improving the processes applied in those projects.

8   Literature

[1]   M. E. Fayad, M. Laitinen, and R. P. Ward, "Software Engineering in the Small," *Communications of the ACM*, vol. 43, pp. 115-118, 2000.

[2]   A. Cater-Steel, Aileen "Low-rigour, rapid software process assessments for small software development firms". In: 15th *Australian Software Engineering Conference (ASWEC 2004)*, 13-16 April 2004, Melbourne, Australia.

[3]   C. Ebert, J. De Man: Effectively utilizing project, product and process knowledge. *Information & Software Technology 50(6)*: 579-594 (2008)

[4]   F. Pettersson, et al. "A practitioner's guide to light weight process assessment and improvement planning", Journal of Systems and Software vol 81, issue 6, 2008.

[5]   F. McCaffery and G. Coleman "Lightweight SPI Assessments: What is the Real Cost?", *Softw. Process Improve. Pract.* 14: 271–278 (2009).

[6]   SCAMPI Upgrade Team, (2011). *Standard CMMI Appraisal Method for Process Improvement (SCAMPI) A, Version 1.3: Method Definition Document* (CMU/SEI-2011-HB-001).

[7]   Glazer, H., Dalton, J., Anderson, D., Konrad, M., & Shrum, S. (2008). *CMMI or Agile: Why Not Embrace Both!* (CMU/SEI-2008-TN-003). Retrieved May 06, 2012, from the Software Engineering Institute, Carnegie Mellon University

[8]   J. De Man, "Multi Model Performance Improvement in Action*", Improving Systems and Software Engineering Conference, Proceedings*. p.31-37 (2012)

9   Author CV

Jozef De Man

Jozef De Man is a Distinguished Member of Technical Staff at Alcatel-Lucent, based in Antwerp, Belgium. He is also part-time professor at Ghent University, department of Information Technology. He has more than 20 years experience in process improvement using the CMMI and its predecessors. He received a Master Degree in Electrical Engineering from Ghent University (Belgium) and has a Doctoral Degree in Computer Science from the University of Leuven (Belgium).

IT systems and services procurement: a framework-based support for buyers in retrospect

Samuel Renault

Public Research Centre Henri Tudor, Luxembourg
samuel.renault@tudor.lu

Abstract

This paper relates the on-going development and deployment of a framework for the purpose of easing the procurement of IT systems and services from a buyer perspective. The lifecycle of the framework is outlined with the point of view of a service system design science method. The framework is built on scientific proposals from the Requirements Engineering community, on standards' procurement process models and on practical experiences of IT procurement projects in public and private organizations. The transfer and dissemination of the framework at a regional level in Luxembourg and Belgium and at a European level through the European Certification and Qualification Association is detailed. Then the usage of the framework is analysed and discussed through 58 procurement projects covering a 12 years period. Last, some perspectives for further development of the framework are proposed.

1 Introduction

Procurement or acquisition are commonly agreed as the *process used by organisations in order to acquire goods or services from third parties* (usually a supplier) [1], [2], [3]. In the Information Technology (IT) context, procurement is one of the alternatives for setting-up a new Information System (IS), the other alternative being the in-house development (traditional „*make or buy*" decision). While the processes for software/system development (covering the *"make"*) are well explored by numerous methods and standards, the procurement of information systems (covering the *"buy"*) is proportionally far less covered by standards and frameworks.

Due to its Research and Technology Organisation position in Luxembourg, the Public Research Centre Henri Tudor (called "the Centre" in the remaining of the paper) has always been requested for neutral and fact-based point of view on technology acquisitions. In the beginning of the 2000's with the increased spreading of business support information systems (Enterprise Resources Planning – ERP, Customer Relationship Management – CRM, Enterprise Content Management – ECM…), the Centre has been more frequently requested for such pre-acquisition point of view. This growth of demands led to the identification of a field for developing an innovative service for supporting such demands. Our initial intend was to develop a set of processes, guidelines, and templates to ease IT procurement projects from the procurer side. We based our IT procurement framework development on proposals from academics works in the domains of Requirements Engineering and Project Management and we experimented our development on real project before transferring the framework on the Luxembourg and Belgium markets. Around 10 years after the first deployment we analyse the usage of the framework.

This paper is structured as follows: section 2 is a review of existing wide-spread frameworks related to IT procurement. Section 3 explains the motivation, the construction and the dissemination of a framework for specific IT procuring organisations. Section 4 analyses the data collected on the usage of the framework and draws conclusions on its utility and usage context. Section 5 discusses the results presented in other sections. Section 6 concludes and explores future works in relation with the domain.

2 Frameworks related to IT procurement

2.1 ISPL

ISPL® stands for Information Services Procurement Library. It is a framework dedicated to describing the processes, activities and deliverables for the management of IT procurement. ISPL® is the result of a European project combining public and private companies. ISPL® is described in a series of 5 volumes edited by EXIN in 1999 [4]–[8]. Based on Euromethod and on practices of public and private organisations, ISPL® describes 3 main process groups covering the initiation of the acquisition, the execution of the acquisition (that ISPL calls „procurement"), and the closing of the acquisition. Practically, ISPL® divides the acquisition execution (the „procurement") into an initiation process and a closure one.

Describing the whole process chain from a neutral point of view, ISPL® addresses both acquirers (customers) and vendors (suppliers). In both cases, ISPL® addresses well-structured organisations performing large-scale IT acquisitions.

ISPL® has been supported by an association the Information Service Procurement Group, which doesn't seem to be active since very few references can be found for this association and all in the Netherlands, and also since the website is not reachable anymore. It is therefore difficult to identify the spreading and the level of use of this framework.

2.2   eSCM

The eSourcing Capability Model (eSCM) is another framework developed in the beginning ot the 2000's by the Carnegie Mellon University. In comparison to more generic capability models like CMMi[9] or the ISO/IEC 15504 exemplar process assessment model [10], eSCM can be seen as a specialised capability model focused on sourcing and procurement. In addition to process description under a lifecycle view (initiation, delivery, completion), eSCM comes with a breakdown between capability areas (10) and capability levels (5).

Designed as a capability model, eSCM intends to be used for evaluating either supplier or customer capabilities to work in a sourcing mindset. Thus in practice eSCM is available in two versions: eSCM-SP[11] for IT services provider and eSCM-CL[12] for IT services customers (i.e. consumers). Each version comes with its own certification schema based on the capability model. Both versions share a common backbone of processes, activities and deliverables.

Like ISPL®, eSCM is mainly oriented towards mature and well-structured organisations with frequent strategic procurement processes. One can note that while some IT services providers are publicly eSCM-SP certified at various capability levels, no customer has gone through the certification for the eSCM-CL version, according to ITSqc [13], the Carnegie Mellon University's spin-off created to promote the models.

2.3   ISO/IEC 12207 & ISO/IEC 15288 proposals on procurement

The ISO/IEC standards for software[2] and system[14] processes lifecycle define acquisition-related processes. Both are structured the same way and seven acquisition activities are described: acquisition preparation, acquisition advertisement, supplier selection, contract agreement, agreement monitoring, acquirer acceptance, closure.

These two standards are generic by nature and the processes and practices description is succinct. The deliverables (outputs) to be produced or used within the processes are named but their contents are not described in details.

These standards are mainly known by software or systems companies or important IT departments of large organisations.

3   An IT procurement framework for buyers

We will outline the main steps of the framework lifecycle from the perspective of a Science-based Sustainable Service Innovation Process model (S3IP)[15]. As represented on figure 1, this model states that building innovative services should follow 6 main processes, not necessarily sequentially: service value (identification of the innovation opportunity), service design (requirements for the service), service exposition (promotion towards the interested stakeholders), service engineering (development of the services itself), service operation (practical exploitation of the service) and service monitoring.

Figure 1: The Science-based Sustainable Service Innovation Process model (S3IP)

3.1   Framework construction

3.1.1 Value of an IT procurement supporting service in small business units

The Centre first looked at determining the value for an IT procurement supporting service. The usual context for support request addressed to the Centre is a business unit of an organisation, or the executive management for smaller ones, which has a business strategy in which the acquisition of an Information System is planned. For simplification in the remaining of the paper we call such demanding organisations "the organisation(s)" whatever their size, status or core business.

Usually the organisations cannot dedicate internal IT skills on the procurement project. Either because the organisation lacks IT-management skilled staff (which is often the case for SMEs) or, if the IT-management skilled staff is present (in larger organisations) it is not available for the project of the Business Unit. Then organisations lack IT technical knowledge and also lack knowledge of the business support information systems available on their market.

Regarding their procurement processes, the organisations may or may not have established procurement processes. When these processes are established, they are aimed at the acquisition of strategic and recurrent services or goods acquisitions. In the context of these organisations, purchasing a large scale business support IS is strategic for their business but it is not recurrent and frequent enough to have them established specific acquisition processes. Due to their lack of IT and market knowledge, the organisations cannot precisely define their acquisition criteria (requirements). Last, due to the lack of market knowledge the organisations cannot target suppliers of solution that could meet their requirements.

3.1.2 Framework engineering and validation

From section 2, we have seen that while not the most developed part of IT processes initiatives, some frameworks have been developed for the purpose of IT procurement. However these frameworks address organisations that are mature regarding their IT processes and IT organisation. These frameworks were thus not suited to the organisations lacking IT skills and knowledge and lacking adapted procurement processes. The Centre decided to develop a practice-oriented but science-based framework with the main assumption that an external actor, with both IT and management competencies, supports the organisation in its procurement project. Such actor can be of various profiles: an IT consultant in the traditional case, but also a business or procurement expert, a R&D engineer / researcher in some cases. For simplification we will name it the supporting third party in the remaining of the paper.

The framework was developed from 2002 to 2004 as a set of six processes, 18 deliverables models and tools starting from the definition of the acquisition purpose up to the contractual agreement with the IS supplier (the detail of the framework is explained in [16] and the framework itself can be found on [17]). The framework focuses on:

- Techniques and processes for collaborative requirements elicitation and priorisation,

- Requirements formalization in structured natural language (use of "shall" statements, black box description),

- Iterative and competitive selection processes (RFI, RFP),

    o A specific sub-process for objective comparison of bids relevance to requirements combined with a total cost of ownership analysis (covering both capital and operational expenditures)

In parallel, the Centre developed a software prototype supporting requirements management and call for tenders analysis: bids comparison based on requirements compliance (the scope of the tool is explained in [18]).

This two development steps can be classified as the engineering of the service according to the S3IP model.

During its development, the Centre validated both the framework and the software prototype on real procurement projects from private and public organisations. The experiments feedbacks were positive from the academic point of view and from the beneficiaries' perspective.

3.2 IT Procurement Framework exposition

3.2.1 Local transfer: the CASSIS network of independent consultants

The validation of the framework demonstrated the market need for neutral profile helping organisation in formalizing and conducting their IT procurement and the value of providing services based on this framework. Just after the validation of the framework, the Centre had the opportunity to build a certification schema for independent IT consultants for SMEs. This certification schema was named CASSIS and was deployed in Luxembourg and French speaking Belgium regions. Based on the IT Procurement framework, the Centre designed a two-days training module, an exam questionnaire (based on multiple choice questions) and a set of 53 requirements to assess the compliance of a consultant's mission to the framework processes. From the S3IP point of view, this last artefact corresponds to the design of the service.

Between 2004 and 2007, the Centre exposed the service to the market and trained around 30 consultants in Belgium and Luxembourg. The Centre attempted to train French consultants, but the results were mitigated (only 2 were trained and none went to the certification), maybe due to the cultural difference and lack of recognition of a foreign certification schema.

Among the trained Consultants, about one third were coached for their first mission and their compliance to the processes was formally recognised according to the full certification process. This tenth of consultants were acknowledged capable to perform IT procurement supporting missions and started to operate services based on the framework since then.

3.2.2 Wider transfer: the European Certification and Qualification Association

From 2006 to 2010, the Centre was involved in two European projects[19], [20] that lead to the building of the European Certification and Qualification Association (ECQA). Within these projects, in parallel of the generic contribution in the building of the association's business models, the Centre adapted the CASSIS training, exams and requirements to the ECQA certification processes and to the wider audience (European scale) [21].

During the projects the Centre had the opportunity to train a few more practitioners based on the European certification model. However after the two projects, to comply with its business strategy, the Centre did not exploit the ECQA certification, but still focused on its local and preceding certification schema. Unfortunately, the ECQA certification has not been exploited by other ECQA members so far.

We can then analyse from the S3IP point of view, that additional iterations were performed on service design and service exposition steps.

4 12 years of IT procurements projects in retrospect

4.1 Data collection from service operation

After the transfer of the IT procurement framework, thanks to the certification monitoring and renewal rules, the Centre kept in touch with certified Consultants on their IT procurement projects. In parallel, the Centre carried on to perform punctual missions, but only outside of the Consultants' scope (i.e. not for SMEs and not for the procurement of ERP systems) and for the purpose of improving the framework. Thus from S3IP point of view, while the Centre performed service value, service design and

service engineering iterations on the framework, the Consultants performed the service operation.

By monitoring its own experiences and the consultants' services provisions, the Centre has collected summary data covering 58 IT procurement projects using the framework between 2001 and 2013 (see the yearly distribution in figure 2). For Consultant's feedback on their service provision, due to the gathering process, the Centre usually collects data about projects with a delay of 18 months in average. This explains the significant decrease in 2012 and 2013.
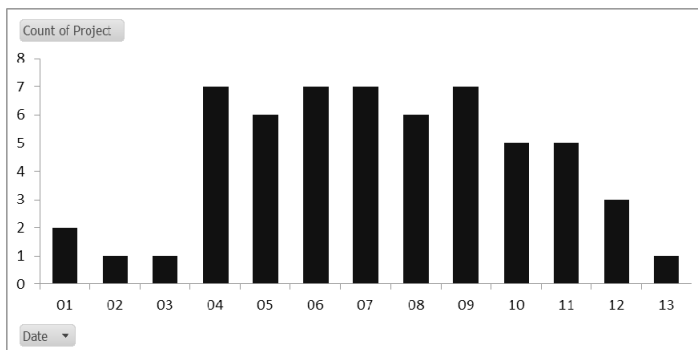
Figure 2: Yearly distribution of the IT procurement projects (N=58)

These data cover

- the customer's profile: type of organisation (SME, large private enterprise, public organisation, not-for-profit association), localisation, activity domain (based on NACE), workforce,

- the supporting third party's profile: certification status, seniority regarding the framework, price charged for support,

- the IT procurement project's profile: type of Business Support IS (ERP, CRM, ECM …), project duration, number of suppliers invited to tenders, price of the awarded contract.

For confidentiality reasons between the Consultant and their client organisations, the Centre has not been able to collect all the data for all the projects. In addition, the Centre has lost the contact with some certified or trained consultants. Therefore we can consider that the number of projects counted in the data collection is a lowest threshold. But we cannot estimate with confidence the number of additional projects.

4.2   Analysis of procurement projects

About the effectiveness of the transfer, the data analysis on figure 3 shows that starting from the end of the first consultants' certification the number of projects performed by consultants oversteps the number of projects performed by the Centre (represented as "Tudor" on the figure) and does not go under since 6 years. We can conclude that the framework has been effectively taken over by external consultants.
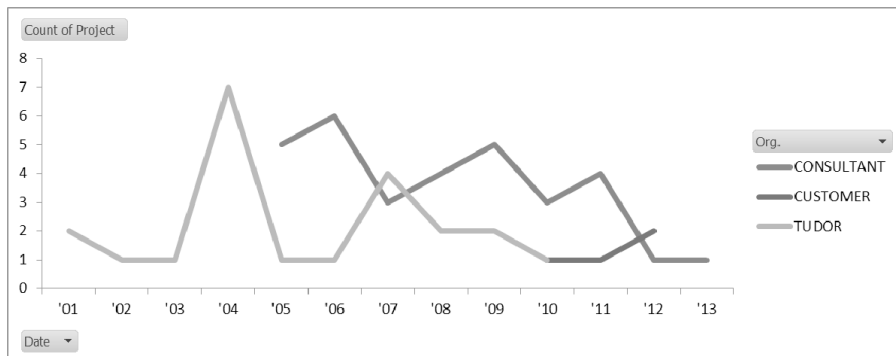
Figure 3: Yearly distribution of projects by third party profile (N=58)

It is also important to notice that, starting from 2010, a marginal way of transfer has been put in place where some organisations (represented as "customer" on the figure) take over the framework and apply it by themselves with limited third party support or without any external support. We have not explored the causes of such behaviour, but the financial crisis effect and the decrease of consultancy spending of organisations could be one of the reasons.

Another evidence of the framework transfer is the localisation of the organisations performing a procurement project following the framework. Indeed half of the projects (50%) are initiated by Belgian organisations, while Luxembourgish organisations "only" counts for 43% and French organisations for the remaining 7%. Given the fact that Luxembourgish, Belgian and French consultants have been trained and certified, and that the Centre only supports Luxembourgish organisations, we can conclude that the Centre only perform a minority of projects, whereas the majority of the projects are performed by external entities whose the framework has been transferred (consultants or organisations by themselves).

Regarding the kind of procurement project in which the framework is applicable, considering the type of information systems targeted by the procurement, we can see in figure 4 that the top 3 information systems acquired by organisations are ERP (covering a large set of processes in the organisation), ECM (covering mainly the processes that need to build, exchange and consume business information) and CRM (covering mainly sales and marketing processes). The framework is then generic enough to be applied in multiple situations where the procurement of a business information system is required.
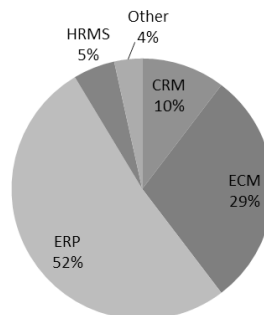
Figure 4: Scope of the procured Information System (N=58)

Considering the size of the project, we can measure it from the point of view of the organisation's spending. Figure 5 shows the split of external costs supported by the organisations (procurement support costs and Information System purchase costs represented as "awarded contract"). One can notice that the average Information System costs are related to the organisation profile (SME, large enterprise, public organisation), whereas the supporting costs are quite independent. Projects of large organisations are more expensive than projects of SMEs and public organisations. However the finan-

cial figures show that the framework is mainly applicable in consequent projects with a purchase budget over 100 K€. For projects of all types, the median Information System purchase cost is around 150 K€ and the average is 300 K€. When measuring the project from the number of people impacted by the system procurement, the average workforce of the purchasing organisation is around 405 persons (but only 86 when considering only SMEs).
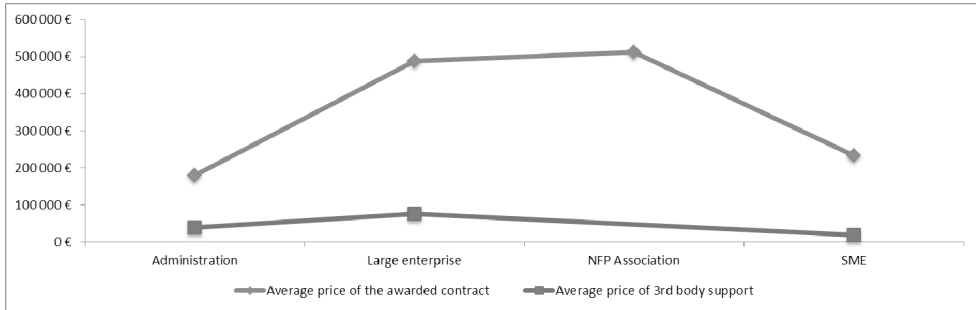
Figure 5: External costs of the procurement by organisation profile (N=38)

Projects performed by not-for-profit (NFP) associations are a bit specific. First, all projects of this category have been performed by the organisations themselves either with a limited support of a third party, or without any support (which explains the lack of value for 3rd body support price). Next all of these projects are related to ERP systems (usually the most expensive kind of information systems) while other projects mix all kind of information systems (expensive and cheaper ones). Last the number of related projects (4) is not statistically relevant to be fully compared with other projects.

5 Discussion

Data about the usage of the framework have not been collected for the initial purpose of statistical analysis. Therefore the statistical validity of the data set is questionable and could be a threat to validity. However this paper does not relate a full scientific study but rather the case study of a framework's deployment over a relatively long period. In addition the analysis has to be considered with the local context in mind, i.e. covering Luxembourg and partially French speaking regions of Belgium. The findings cannot be generalised directly in other contexts.

The conclusion on the effectiveness of the framework lacks a comparison with equivalent projects but not using the framework. Due to the difficulty to collect this kind of data with a sufficient level of confidence, we have not yet had the opportunity to set-up a data collection campaign for these kinds of projects. On the other hand three facts give us confidence in the effectiveness of the framework: 1. The fact that the framework is used by more third parties than by its creator, 2. the number of projects covering a wide range of organisations in size, in type and in business, 3. the relatively low rate of projects that have not achieve their goal (procuring an IS).

Last we can notice that a tenth of consultants continue to use the framework and provide regular feedback on the projects they perform ten years after their first training. We consider that this demonstrates indirectly the usefulness of the framework though this reasoning is certainly not as strong as a formal qualitative analysis of the framework's usefulness. But we have not had the opportunity to realise such qualitative analysis so far.

6 Conclusion

This paper has presented the deployment and usage of a framework composed of process guides and

templates designed to ease IT procurement projects.

Unlike other frameworks for IT systems and services procurement such as ISPL®, eSCM or acquisition processes described in ISO/IEC 12207 and ISO/IEC 15288, our framework has been designed for organisations that lacks IT expertise and procurement processes knowledge. The framework has been transferred at a regional level to independent consultants in Luxembourg and Belgium with quite good results, while the opportunity for a European-wide dissemination, notably through the ECQA, has not yet been fully investigated.

Feedback from projects using the framework up to eight years after its first transfer show that the framework is efficient and useful for small organisations (the initial target) but also in some cases large enterprises and public organisations. The medium-term overview also shows that the initial model intended for the framework usage (a third party supporting the purchasing organisation) is changing since more and more organisations take over the framework for applying it by themselves without any third party support or with a limited support.

Our future works will be focused on exploring additional services value for capitalizing and reusing knowledge acquired in these projects, as we already started with the proposal of patterns based requirements engineering [22]. In parallel we are closely collaborating with a start-up that is acquiring our software prototype for the call for tenders' management and bids comparison. We expect that a new Web version of such tool will give us more data on procurement projects either performed according to our framework or not. A larger analysis of procurement practices will help in understanding how organisations perform their procurements projects and in proposing innovations.

7 Literature

[1] 'An introduction to public procurement', Office of Government Commerce, London, UK, 2008.

[2] 'Systems and Software Engineering — Software Life Cycle Processes', International Standards Organization, Standard ISO/IEC 12207, 2008.

[3] Project Management Institute, *A guide to the project management body of knowledge (PMBOK Guide).*, 4th ed. Newtown Square, Pa.: Project Management Institute, 2008.

[4] EXIN, Ed., *Introduction to ISPL*, vol. 1, 5 vols. The Hague: SDU, 1999.

[5] EXIN, *Managing acquisition processes*, vol. 2, 5 vols. The Hague: SDU, 1999.

[6] EXIN, *Specifying deliverables*, vol. 3, 5 vols. The Hague: SDU, 1999.

[7] EXIN, *Managing risks and planning deliveries*, vol. 4, 5 vols. The Hague: SDU, 1999.

[8] EXIN, *Dictionary*, vol. 5, 5 vols. The Hague: SDU, 1999.

[9] M.-B. Chrissis, M. Konrad, and S. Shrum, *CMMI : guidelines for process integration and product improvement*. Addison-Wesley, 2006.

[10] 'Information technology — Process assessment — Part 5: An exemplar Process Assessment Model', International Standards Organization, Standard ISO/IEC 15504-5:2006(E), 2006.

[11] E. B. Hyder, K. M. Heston, and M. C. Paulk, *ESourcing Capability Model for Service Providers (eSCM-SP)*. Van Haren Publishing, 2010.

[12] B. Hefley and E. A. Loesche, *ESourcing Capability Model for Client Organizations (eSCM-CL)*. Van Haren Publishing, 2010.

[13] 'ITSqc: Certification: eSCM-CL Certified Organizations'. [Online]. Available: http://www.itsqc.org/certification/certified-cl.html. [Accessed: 30-Jan-2014].

[14] 'Systems and Software Engineering — System Life Cycle Processes', International Standards Organization, Standard ISO/IEC 15288, 2008.

[15] E. Dubois and A. Rousseau, 'Service Science: A Service System Design Science Research Method?', in *Exploring Services Science*, J. F. e Cunha, M. Snene, and H. Nóvoa, Eds. Springer Berlin Heidelberg, 2013, pp. 100–113.

[16] M. Krystkowiak and B. Bucciarelli, 'COTS Selection for SMEs: a report on a case study and on a supporting tool', in *Proceeding of the 1st International Workshop on COTS and Product Software: Why Requirements are so Important*, Monterey, California (USA), 2003.

[17] 'Introduction to the Cassis software selection methodology'. [Online]. Available: http://tinyurl.com/cassis-IT-procurement. [Accessed: 22-Apr-2014].

[18] M. Krystkowiak, V. Betry, and E. Dubois, 'Efficient COTS selection with OPAL tool', in *International Workshop on Models and Processes for the Evaluation of COTS Components (MPEC'04)*, Edinburgh, Scotland (UK), 2004, pp. 23–27.

[19] 'EQN project page on ADAM', *ADAM - Leonardo da Vinci Projects and Products Portal*. [Online]. Available: http://www.adam-europe.eu/adam/project/view.htm?prj=2022&page=1. [Accessed: 27-Feb-2014].

[20] 'European Certificates Innovative Online Training Campus - Final Report', Education, Audiovisual and Culture Executive Agency, Final Report - Public Version 135691- LLP-1-2007-1-AT-KA3-KA3MP, 2008.

[21] S. Renault, B. Barafort, E. Dubois, and M. Krystkowiak, 'Improving SME trust into IT consultancy: a network of certified consultants case study', in *EuroSPI 2007 Industrial proceedings*, Potsdam, Germany, 2007, pp. 10.1 – 10.8.

[22] S. Renault, O. Mendez-Bonilla, X. Franch, and C. Quer, 'A pattern-based method for building requirements documents in Call-for-tender processes', *Int. J. Comput. Sci. Appl.*, vol. 6, no. 5, pp. 175–202, 2009.

Author CVs

Samuel RENAULT

Engineer in Information System Architecture in 2003, Samuel Renault has joined the Public Research Center Henri Tudor a Research and Technology Organisation (RTO). He is now product manager of a set of services dedicated to the management of procurement and sourcing of IT systems and services.

Samuel has co-developed methods, tools and training dedicated to requirements engineering and business IT solutions and services procurement. These methods and tools have been used in more than 50 IT procurement projects of public and private organisations in Luxembourg and Belgium.

Samuel has trained independent consultants in Luxembourg, Belgium and France to these methods through a local certification schema (the CASSIS network) and in Europe through the European Certification and Qualification Association (ECQA). He is still trainer, coach and manager of the CASSIS service: consulting in software selection, as well as head of the ECQA job role committee « IT consultant for SMEs ».

On his themes, Samuel works both on the development of innovation in partnership with researchers and on the application in business context in partnership with professionals.

Compliance and Rigour in Process Assessment for Safety-Critical Domain

Timo Varkoi[1] and Risto Nevalainen[2]

[1] *Spinet Oy, Finland*
[2] *Finnish Software Measurement Association – FiSMA ry, Finland*
{timo.varkoi@spinet.fi, risto.nevalainen@fisma.fi}

Abstract

Safety-critical systems are increasingly affecting our lives and welfare. New approaches are being developed to evaluate the abilities related to development of these systems. Process assessment can be applied to increase our trust in safety related systems development. Importance of meeting the requirements of existing safety standards and regulations has increased, but also the quality of the process assessments needs to be ensured. Important features include assessment rigour, and compliance to standards and regulatory requirements. In this paper we discuss the challenges in process assessment with highest safety-criticality and present an approach to manage the assessments by a classification of relevant assessment types. The outcome is evaluated with a domain specific example. We conclude that process assessment has significant limitations in its capability to verify safety requirements, and especially regulatory requirements. On the other hand, process assessments are applicable to cer-tain purposes, like supplier selection, and they can be developed to include a wider coverage of evidence important to the safety-critical domain.

Keywords

Process Assessment, Safety, Safety-critical, Software Process

Reference

Intangible Assets Management in the Software Industry

Alejandro Ruiz-Robles[1], Maria-Isabel Sánchez-Segura[2], and Fuensanta Medina-Dominguez[2]

[1]Information Systems Department, University of Piura, Peru
alejandro.ruiz@udep.pe
[2]Computer Science Department, Carlos III University of Madrid, Spain
{misanche, fmedina}@inf.uc3m.es

Abstract

The most important resource for software development companies is their intellectual capital, which is comprised of all the intangible assets that contribute to the delivery of products and services. For a software development company, a subset of intangible assets is especially important, the software process assets. These assets are related to describe, implement and improve software processes, and their quality is a critical factor for any software process improvement initiative. This paper points out the importance of having a strategic view of companies' intangible assets and performing an assessment of them. By doing this, a software development company would have a comprehensive view of its software process assets and their relationship with its software processes and business goals. This is the base to make informed decisions regarding what and how software process assets need to be improved and which software processes and business goals would be affected.

Keywords

Intellectual Capital. Software Process Assets. Software Process Improvement.

1 Introduction

Like any organization, software development companies depend on three main vital resources: Physical Capital, e.g. buildings or computers; Financial Capital, e.g. money or credit, and Intellectual Capital, e.g. non-tangible resources like processes or knowledge [1]. Intellectual Capital is the most important resource for knowledge intensive companies [2], this is the case of software development companies [3].

Intellectual Capital is comprised of all the non-tangible resources, i.e. intangible assets, which contribute to the delivery of products or services. Some examples of intangible assets are talents and skills of individuals and groups, knowledge and experience of people, patents, copyrights, methods or procedures [1], [2].

Several studies have proven the importance of intellectual capital in organizations of different sizes, different industries, included the software industry, and located in different countries. Intellectual capital has been positively related with the improvement of productivity, profitability, innovation capability, growth, and market value of companies [4]–[13].

Between all of the intangible assets that could be found in a software development company, there is a subset of them which are critically important for software processes, the software process assets [14].

Software process assets are intangible assets that relate to describe, implement and improve software processes. These assets exist within companies and are developed or acquired in order to meet companies' business goals. Any initiative of software processes deployment or improvement depends on the company's software process assets and their quality [14].

The quality of software process assets is a critical success factor for any software process deployment or improvement initiative, and for meeting companies' business goals [14]. Therefore, to assess a software process asset is paramount in determine if such asset has an adequate quality for improving a process or meeting a business goal, and is the cornerstone in answering the next three useful questions for any software development company:

- What software process assets need to be improved?
- What the company should expect if a determined software process asset is improved?
- Through the improvements of which software process assets can the company improve a software process or meet a business goal?

Besides, as software process assets are intangible assets, and part of companies' intellectual capital, their assessment will be reflected in the determination of the companies intellectual capital value, and therefore in the companies' market value.

In this paper, the authors want to draw attention to the fact that the software industry, although is one of the most knowledge intensive industries in the world, needs to manage and assess its software process assets as a mean to improve its performance, intellectual capital, and market value. And due to the relationship between software process assets and software processes, this goal will be reflected directly in the improvement of software processes.

2 Looking for a Strategic View of Software Development Companies Core Software Process Assets

Let's take a look at three intangible assets. Could a software development company survive in the time if its brand is being undermined right now? Could a company grow in the mid-term if its processes do not allow satisfying its clients? Could a company adapt to new requirements and technologies if its employees' knowledge and skills are non-optimal?

Knowing that intangible assets are the main vital resources for software development companies; assessing, managing, and evolving these assets are paramount activities to guarantee the companies' success in the mid and long term. Although a company could be profitable right now, how do we know if that situation is sustainable in the time? The answer lays in the company's intellectual capital.

To manage and assess intangible assets is not simple, their intangible nature makes this hard to accomplish. For instance, companies struggle when they need to put a value to their processes or employees' knowledge.

Although not simple, companies need to assess their intangible assets in order to make decisions to meet their business goals [1], this is the same for software development companies. By focusing specifically on software processes assets, software development companies should understand the value of these assets in order to make decisions that affect positively the software process improvement initiatives and contribute to meeting business goals [14].

To achieve this, companies must identify all of their software process assets, and relate those assets with their software processes and business goals; this is, to move from a non-organized partial view of their software process assets, Figure 1, to a strategic view of them, Figure 2.
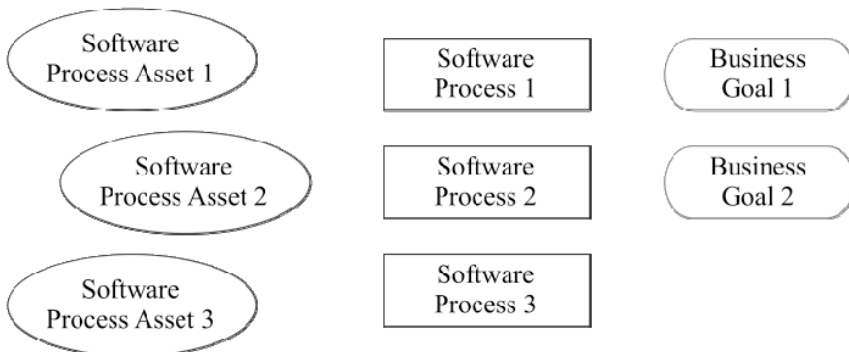
Figure 1. A non-organized partial view of software process assets
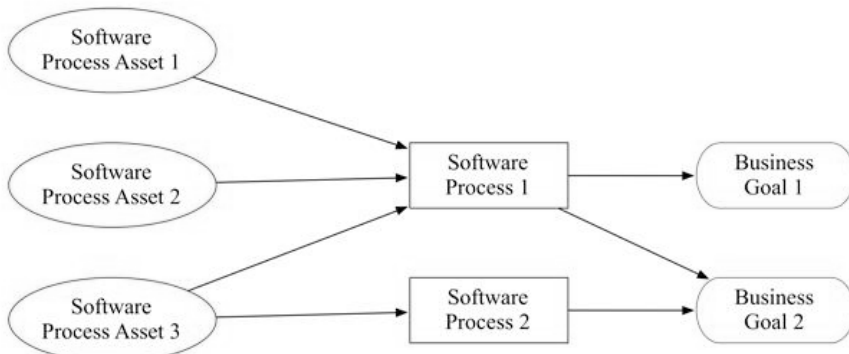
Figure 2. A strategic view of software process assets

By having a strategic view of its software process assets, a company will be able to relate them with its business goals, which is key in making informed decisions affecting specific business goals [15]–[17], and due to the relationship between software process assets and software processes, informed decisions regarding how to deploy and improve software processes through the use and improvement of software process assets [14].

3 Software Process Assets Assessment

Having a strategic view of a company's software process assets is the base upon which the assessment of such assets should be performed. Any effort of assessing an intangible asset without taking into account its relationship with the company's business goals will be worthless [1], [15]–[17]. The reason to perform such assessment is to make informed decisions regarding which software process assets need to be improve and how, in order to improve the software processes and the meeting of business goals.

By performing the assessment, a software development company should know (a) if their software process assets have the adequate quality to describe, implement or improve their software processes. This quality is determined through the evaluation of the relevant assets characteristics associated with the description, implementation or improvement of the linked software processes. And (b) if their software process assets are having the expected impact in their software processes. This impact is determined through the evaluation of the expected results in the description, implementation or improvement of software processes.

Below, an example of how the final assessment should be is presented in Table 1. This example is an extract of a real case application in an IT company specialized in the provision of a learning management system (LMS) to large and medium-sized companies in the form of software as a service, and in the development of online learning contents.

In this example, the traceability between the software process assets value and the business goals is explicit. It is possible to relate software process assets with software processes and business goals. This allow software development companies to answer the questions previously stated, they will be now capable of determine: What software process assets need to be improved, what should be expected if a determined software process asset is improved, and through the improvements of which software process assets can improve a software process or meet a business goal.

For instance, through this assessment, this company was able to understand that the Communication skills with clients at a technical and managerial level, was undermining the performance of the Online learning content development process and was not contributing to managing the market position of the company. And that the Kick off meeting process needed to be improved because it was not efficient.

Besides, the software process assets are classified as intellectual capital, and the results of the assessment could be used as an input in the assessment of a company's intellectual capital. For instance, the asset Communication skills with clients at a technical and managerial level, needs to be improved, this means that the Human Capital of the company needs to be improved due to this is affecting the online learning content development processes and the company's market position.

As a complement to the traceability presented above, the software process assets assessment can be analyzed in another way. The software process assets can be classified in one of the next four categories,

Figure 3, based on their quality and their impact in the company's processes and business goals.

Table 1. Software Process Assets Assessment

Business Goal	Process	Intellectual Capital Classification	Process Asset	Indicator	Indicator Value	Process Asset Assessment
Managing market position	Online learning content development process	Structural Capital: Knowledge Documents	Kick of meeting process	Requirement Elicitation efficiency	Medium	Inadequate Quality
				Requirement Elicitation efficacy	Complete	
				Client's Learning of the kick off process	Medium	Relevant Impact
				Client's conformance with Gantt Diagram	Agree	
				Approved proposals	High	
		Structural Capital: Tools	Proprietary web project management system	Use easiness	High	Adequate Quality
				Proposals validation time	Adequate	Insufficient Impact
				Number of iterations per project	High	
		Human Capital: Competences and Skills	Communication skills with clients at a technical and managerial level	Client's knowledge process	Partial	Inadequate Quality
				Clients perform activities within the process	Normal	Insufficient Impact
				Clients perform activities of further steps in the process	Sometimes	
		Human Capital: Knowledge and Experience	Course development process experience	Client trust in project manager indications	High	Adequate Quality
				Number of iterations per project	Adequate	Relevant Impact

Figure 3. Software process assets assessment classification

In this view, a company can identify the:

- Warning Software Process Assets, those assets needing to be improved firstly due to their low quality and low impact in the company,

- The Replaceable Software Process Assets, those assets with high quality but that are not having a relevant impact in the organization, and that maybe need to be replaced because it is not possible to improve their quality,

- The Evolving Software Process Assets, those assets that are having a relevant impact, but that can still have a better quality, and could contribute more to the company, and

- The Stable Software Process Assets, those assets with high quality and high impact, the most valuable of all the assets.

Also, this offers a general view of the software process assets. A company can easily know if most of its, for instance, three hundred software process assets are stable, or are in a warning situation. This, from the intellectual capital perspective means that the health of the intangible assets is good or not, which is related to the productivity, profitability, innovation capability, growth, and market value of the company, and an indicator of the mid and long-term success of the company.

4 Conclusions

The importance of intangible assets in software development companies has been pointed out. Some of these assets, the software process assets, are especially important due to their relationship with the software process improvement and the meeting of business goals.

Having a strategic view of software process assets will provide a comprehensive view of the relation-ships between these intangible assets and the company's business goals, which is the basis for per-forming an assessment of this type of assets.

The assessment of software process assets and its traceability with the company's business goals, allow software development companies to make informed decisions regarding the improvement of their assets, and the expected effect of software processes and business goals.

Finally, the assessment and disclosure of companies' intellectual has proven to be related with its productivity and market value. The assessment of software process assets contributes in this aim; the results obtained by performing such assessment can be used as an input for the assessment of a company's intellectual capital, and can contribute to increase its productivity and market value.

5 Literature

[1] B. Marr, "Impacting Future Value : How to Manage your Intellectual Capital," The Society of Management Accountants of Canada, the American Institute of Certified Public Accountants and The Chartered Institute of Management Accountants, 2008.

[2] T. Stewart and C. Ruckdeschel, "Intellectual capital: The new wealth of organizations," Perform. Improv., vol. 37, no. 7, pp. 56–59, 1998.

[3] F. O. Bjørnson and T. Dingsøyr, "Knowledge management in software engineering: A systematic review of studied concepts, findings and research methods used," Inf. Softw. Technol., vol. 50, no. 11, pp. 1055–1068, Oct. 2008.

[4] S. A. Mosavi, S. Nekoueizadeh, and M. Ghaedi, "A study of relations between intellectual capital components, market value and finance performance," African J. Bus. Manag., vol. 6, no. 4, pp. 1396–1403, Feb. 2012.

[5] S. Kehelwalatenna, "An Empirical Investigation on Intellectual Capital Performance : Evidence from Banking Sector," 2007.

[6] A. L. Ferreira, M. C. Branco, and J. A. Moreira, "Factors influencing intellectual capital disclosure by Portuguese companies," Int. J. Account. Financ. Report., vol. 2, no. 2, pp. 278–298, Dec. 2012.

[7] M.-S. Abdulai, "Intellectual Capital and Firm Performance : An Empirical Study of Software Firms in West Africa," African J. Inf. Syst., vol. 4, no. 1, p. 31, 2012.

[8] M. Ghorbani, B. Mofaredi, and S. Bashiriyan, "Study of the relationship between intellectual capital management and organizational innovation in the banks," African J. Bus. Manag., vol. 6, no. 15, pp. 5208–5217, Apr. 2012.

[9] M. Cordazzo, "Intellectual capital disclosure in the UK biotechnology IPO prospectuses," J. Hum. Resour. Costing Account., vol. 16, no. 1, pp. 4–19, 2012.

[10] D. Branswijck and P. Everaert, "Intellectual capital disclosure commitment: myth or reality?," J. Intellect. Cap., vol. 13, no. 1, pp. 39–56, 2012.

[11] J. K. Ngugi, P. R. W. Gakure, W. M. Susan, P. K. Ngugi, and C. Robinson, "The Influence of Intellectual Capital on the Growth of Small and Medium Enterprises in Kenya," J. Bus. Manag. Corp. Aff., vol. 1, no. 1, pp. 11–19, 2012.

[12] T. Yosano, C. Nielsen, and G. Rimmel, "The effects of disclosing intellectual capital information on the long-term stock price performance of Japanese IPO 's," 2003.

[13] A. C. Tonini, M. D. M. Spinola, J. Manuel, C. Medina, V. L. Gava, and R. F. Gonçalves, "Relationship of Intangible Assets and Competitive Advantage for Software Production : A Brazilian Companies Study," pp. 3604–3612, 2012.

[14] Software Engineering Institute, "CMMI® for Development, Version 1.3," 2010.

[15] J. L. Thompson, Strategic management :awareness and change, 2nd Ed. London: Chapman & Hall, 1993, p. 757.

[16] G. G. Dess, G. T. Lumpkin, and M. L. Taylor, "Strategic Management : Creating Competitive Advantages," in Strategic Management : Text and Cases, McGraw-Hill, 2004, pp. 4–33.

[17] C. W. L. Hill and G. R. Jones, Strategic Management: An Integrated Approach, Eighth Edi. USA: Houghton Mifflin, 2008.

6 Author CVs

Alejandro Ruiz-Robles

Alejandro Ruiz-Robles has an MSc in Computer Science from Carlos III University in Madrid and is a faculty member of the University of Piura. His research interests include software engineering and knowledge management. Contact him at alejandro.ruiz@udep.pe

Maria-Isabel Sanchez-Segura

Maria-Isabel Sanchez-Segura has a PhD in Computer Science from Madrid Technical University. She is a faculty member of the Carlos III University of Madrid. Her research interests include software engineering with a focus on processes and intelligent organizations. Contact her at misanche@inf.uc3m.es

Fuensanta Medina-Dominguez

Fuensanta Medina-Dominguez has a PhD in Computer Science from Carlos III University where she is assistant professor. Her research interests include software engineering and software process reuse. Contact her at fmedina@inf.uc3m.es

Finding Threats with Hazards in the concept phase of product development

Masao Ito

Nil Software Corp., Tokyo, Japan
nil@nil.co.jp

Abstract

In this paper, we present an approach to find threats together with hazards. We've already presented the hazard identification approach in [1]. In this paper, it is elaborated and extended to identify threats too. The basic approach is the same as the previous paper and has four steps. First of all, we roughly describe the static structure and dynamic behaviour. Then using the goal-oriented approach, we depict the goal tree of a system. The top goal of the tree is the most abstract representation of a system and we will divide it repeatedly. If S is a sentence as a description of each goal, we can make the new sentence S-* by applying the guideword of HAZOP [2] (when we adopt the NO guideword, we name the new sentence S-NO, asterisk means the meta-character here). S is a desirable goal; S-* is an undesirable goal (i.e. anti-goal [3]). Using the previous static structure and dynamic behaviour, we then consider whether it is possible to create this negative situation caused by the malfunction of each node or attack to a relation between nodes. The exhaustiveness is important for finding hazards and threats. In our methods, we check them in two ways. One is the checking of the sentence of the goal description using the guideword; the other covers every structural and dynamic elements of a target system.

Keywords

Security, safety, hazards, threats, goal model, ISO 26262

Reference

Integrating Functional Safety, Automotive SPICE and Six Sigma – The AQUA Knowledge Base and Integration Examples

Richard MESSNARZ[1], Christian Kreiner[2], Andreas Riel[3], Serge Tichkiewitch[3], Damjan Ekert[1], Michael Langgner[4], Dick Theisens[5]

[1]ISCN LTD/GesmbH, Schieszstattgasse 4, A-8010 Graz, Ireland and Austria
Tel: +43 316 811198, Fax: + 43 316 811312, Email: rmess@iscn.com
[2] Graz University of Technology, Austria
[3]EMIRAcle c/o Grenoble Institute of Technology GSCOP UMR5272, Grenoble, France,
[4]Automotive Cluster Austria, Austria
[5]Symbol BV, Netherlands

Abstract

AQUA stands for Knowledge Alliance for Training Quality and Excellence in Automotive. The AQUA project is financially supported by the European Commission in the Leonardo da Vinci part of the Lifelong Learning Programme under the project number EAC-2012-0635. This paper extends the EuroSPI 2013 publication [4] which discussed (based on the EU project AQUA) how the core elements of three complementary approaches (Automotive SPICE, Functional safety, Six Sigma) and standards can be integrated into one compact skill set with training and best practices to be applied. In this paper we describe the modular knowledge base which was elaborated and highlight some aspects where the integrated use of all three methods can be demonstrated. The results of the project are disseminated to Automotive industry in partnership with a set of European Automotive associations.

Keywords

Automotive SPICE, Functional Safety, Lean Six Sigma, Integrated Approach for Engineering, AQUA - Automotive Knowledge Alliance

Reference

Medical Usability Engineering and Use Risk Management

Zsófia Béky, Máté Bocz, and Sándor Dolgos

B. Braun Medical Kft., Budapest, Hungary
B. Braun Avitum AG, Active Medical Device R&D, Location Budapest
{zsofia.beky, mate.bocz, sandor.dolgos}@bbraun.com

Abstract

In product design it is an essential task to provide safe and error tolerant usage. It is especially true in the field of medical devices since an unintended or unforeseeable action may cause harm not only to the patient but to the user as well. Identifying and eliminating hazardous situations is therefore literally a matter of life and death. Incorporating usability engineering in the development process is the key to understand the human characteristics, behaviour and skills that lead us to better assess and mitigate use related risks. While the ultimate goal of medical usability engineering is to ensure safety, one should not forget about other goals: to design effective, efficient and easy to learn user interfaces. Derived from multiple standards, guidances and our own experience this paper describes a usability engineering process with a strong focus on IEC62366 compliance and managing use related errors.

Keywords

Usability engineering, risk management, use risk, human factors, medical device, user interface, IEC 62366

1 Introduction

In our everyday lives we use hundreds of different devices and we do not have the time or energy to learn how to use each and every one of them. Usability engineering strives for designing products that require minimum effort to be used. Usability is a metric; it is a set of characteristics of the interaction between the user and the device. These characteristics can express for example effectiveness, efficiency, satisfaction, or learnability. Usability engineering aims for higher usability therefore the goal is to optimize the user-device interaction. This interaction is done through the user interface which is the bridge between the user and the device. User interface can be a display, a button, a mouse, handles, a speaker, or even sounds and lights. The accompanying documents, such as instructions for use or service manuals are also considered user interfaces and thus shall be subjected to usability activities. The graphical user interface (sometimes mistakenly the only focus of usability engineering) is the set of graphical elements on the display, but the user interface can also be a cable or hook on the device. Usability engineering deals with these hardware user interface elements, as well.

1.1 Medical Usability and Use Risk

While general usability engineering focuses on such aspects as efficiency or user satisfaction, the medical usability strives for providing safe use and reducing risk arising from use errors[1]. Risk can arise from two sources, either the device itself can generate a hazardous situation (e.g. by a loose screw) or by use errors: when the device works safely but the user causes unforeseeable or unintended situations. It is however not the user to blame for such errors. These errors can occur because there is room for them to occur. With better identification of hazardous situations the device can be designed not to allow the users to make mistakes. The users of medical devices usually work in a stressful environment while they need to make fast decisions, their attention is divided, and the fear of making a mistake makes them even less confident. Recent medical devices are very complicated and are designed by engineers in a way that is not always easily understandable by medical professionals, let alone patients (who can be users as well). It is therefore fundamental to correct design flaws once it is known how the users interact with the devices and the weak points are identified.

The essence of usability engineering is the extended knowledge about human factors such as skills, behavior, motivation, etc. This knowledge is then applied throughout the whole process. This user-centered approach is crucial in dealing with use errors. Interleaved in the usability engineering and the whole design process, use risk management is involved in the following tasks:

1. identify hazards and hazardous situations connected to use
2. describe how these situations can occur
3. provide user interface related control measures to minimize (eliminate if possible) the risk related to these hazards
4. implement these control measures
5. repeatedly seek unanticipated hazardous situations (steps 1-4)
6. prove that the risk control measures are effective and the device is free from any unacceptable risk

Even though these steps are mentioned and required by usability engineering standards, they are more elaborated in other regulations related to risk management. The following chapters interlace these two processes.

2 Medical usability engineering standards

Medical device manufacturers are required to establish and follow a usability engineering process in order to ensure safe use of the device by complying with various standards. The IEC 60601-1 stand-

---

[1] While focusing on safety, with the same effort usability engineering can also be used to assure pleasant user experience, to create competitive advantage, or to reduce production costs.

2   *Medical usability engineering standards*

Medical device manufacturers are required to establish and follow a usability engineering process in order to ensure safe use of the device by complying with various standards. The IEC 60601-1 standard family for medical electrical equipment provides a framework for a usability engineering process (IEC 60601-1-6, Medical electrical equipment –Part 1-6: General requirements for basic safety and essential performance – Collateral standard: Usability [1]) which is further elaborated and extended in the IEC 62366 standard (Medical devices – Application of usability engineering to medical devices [2]). Both of these standards focus on the process and the documentation requirements[2]. They are supplemented by a draft guidance issued by the US Food and Drug Administration (FDA) (Applying Human Factors and Usability Engineering to Optimize Medical Device Design [3]), which rather puts an emphasis on the methodology and the user-centered approach. HE 74 [4], issued by the Association for the Advancement of Medical Instrumentation (AAMI) is synchronized with the IEC 60601-1-6 process standard, while HE 75 [5]– issued by the same association – provides very detailed design guidelines. To ensure the safety related approach the ISO 14971 (Medical devices – Application of risk management to medical devices [6]) standard issued by the International Organization for Standardization is often referred in the IEC 62366 standard. ISO 14971 provides guidelines for use risk analysis being an essential part of the usability engineering process. Therefore the usability engineering and the risk management activities cannot be dealt with separately.

This set of standards and guidances cover a wide range of issues in the usability engineering process and are a great help in the manufacturers' usability activities. The next chapter describes a process which has been established by combining requirements from the above mentioned regulations and our own experience.

3   *Medical usability engineering process*

The usability engineering process in general is an iterative process, which means that the activities cannot be simply performed one after the other. Some of the steps are running parallel or are done multiple times. As Figure 1 shows, the main phases are:

1. analyzing device use, (3.1 Use analysis)
2. defining requirements,(3.2 Requirements)
3. digging deeper and adding details to the design, implementing it by creating prototypes and finally the product itself (3.3 Design and implementation)
4. prototypes, requirements and scenarios are repeatedly evaluated during formative tests, and the implementation of the requirements is checked during verification (3.4 Usability verification)
5. when ready, the final user interface is validated by a usability validation (summative) test and by a checklist of all activities in the usability engineering process, (3.5 Usability validation)
6. and after market launch post-market data is collected and evaluated regularly throughout the lifetime of the machine. (3.6 Post-market surveillance)

---

[2]   Documenting usability activities is the only way to prove the undertaken effort to manage use risk.
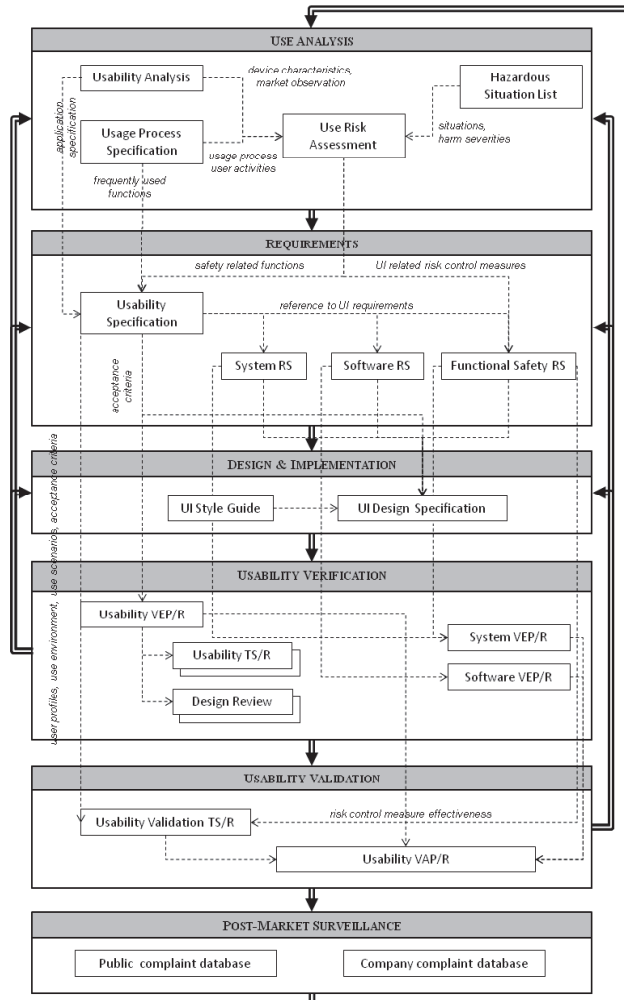
Figure 11 Overview of the medical usability engineering process

When talking about medical usability engineering, these steps are saturated with the risk-centered thinking, which results in even more iterations and deflection towards the risk management process. Evidently, the magnitude of the usability engineering process depends largely on the nature (risk level) of the device itself.

B. Braun Medical Kft. is involved in the development of dialysis machines and other blood purification devices. The next section elaborates on our usability engineering activities and deliverables while providing theoretical background derived from the standards. Our division has a standard operating procedure for all medical devices in terms of usability activities, which defines each step of the general process to follow, inputs, outputs and responsibilities of the usability engineering team. It furthermore provides document templates for a standardized documentation practice throughout all project teams.

At the start of the project we have planned our usability activities and described them in the Usability Engineering Plan document. Figure 1. represents our approach and documentation strategy. Aligned with the activities we have included a project schedule and defined the roles in the team. Also, the used tools and methodology for the project has been decided. The Usability Engineering Plan ensures the right track of activities and provides guidelines for many practical issues. All documents that are the outputs of the usability process, or are in connection to any of the steps are collected in the Usability Engineering File, which is a folder listing and referencing these documents.

In this chapter we go through all process steps identified in Figure 1. and describe our project in details. Our main focus is the compliance with the IEC 62366 standard, as well as the corresponding FDA guidances.

3.1. Use analysis

As being a user-centered approach, the usability engineering process has to put much effort in user research. In this phase all kind of information is gathered and analyzed related to the use of the device. Three main parts of this step are: conducting user research and collecting information for the Application Specification, specifying use scenarios, and identifying and analyzing hazardous situations during use.

Usability Analysis

The Usability Analysis document collects all information related to the use of the device. Input information comes from the marketing department, from experience with previous (own and competitor) devices, and from contextual inquiries in various use environments. This knowledge is included in the application specification chapter as required by the IEC 62366.

The Usability Analysis also contains a summary of known use problems with previous models and similar devices based on public company complaint databases. Furthermore it identifies all device characteristics associated with usability that could impact safety. Even though such informations are collected in the Usability Analysis, they will be further considered during the use risk assessment activity.

Application Specification

As major part of the Usability Analysis, according to the IEC 62366 the application specification identifies the followings:

1. The purpose of the medical device,

The purpose of the devices determines whether it is appropriate for the treatment of a specific patient. Our device is intended to perform all intermittent and continuous acute blood purification treatments for patients with acute renal failure, intoxication and diseases where plasma therapy is indicated. The patients can have serious injuries and can be operated on or examined several times during the therapy.

2. Patients and body parts involved in the therapy,

Anyone can be a patient with acute renal failure, intoxication or a disease where plasma therapy is indicated. There are no age restrictions, but the minimum patient weight is 30kg.

3. Users and user profiles,

The user of a device is not only the one who is using it for treatment. Everyone who interacts with it is considered a user, e.g. maintaining or cleaning staff. Based on common demographic, cultural, professional or other characteristics the users can be divided into user profiles. Founded on on-site (hospital ICU) contextual inquires and interviews with field experts, two distinct user groups were identified with different sets of tasks and context of use. The device will be used by medical professionals (e.g. doctors, nurses to perform therapy) and technical personnel (to manufacture, maintain and repair the device).

4. The context of use,

It is important to note every information about the circumstances of use. Our device can be used in Intensive Care Units (ICU) or renal wards. These are hectic environments, where interruptions and distraction of the user shall be anticipated. Due to the stressful and tiring situations in an ICU, the user's cognitive abilities can decrease leading to harder decision making, more time needed to process information, less sensitivity to audio and visual effects and decay in short-term memory. These condi-

tions imply the need for short and easily understandable texts, unambiguous graphical elements, rigorous parameter checks, intuitive mechanical operation and distinguishable audio signals in the user interface design.

5. The operating principle of the device.

The mechanism of device operation or the physical principles utilized for attaining the medical benefits shall be described. Our device provides multiple therapy types. In all cases, the device pumps the blood of the patient through the vascular access (typically central venous catheter) and a set of disposable tubing into the filter and then back to the patient. In most cases the excess body water is slowly and continuously filtered out along with uremic toxins. Convection and diffusion (osmosis) principles are utilized to remove toxins inside the filter. Dialysis fluid can be used to induce diffusive clearance and substitution solutions can be used to facilitate convection and replace part of the removed fluid volume.

Our device follows the state of the art architecture of acute blood purification devices. The user interface includes an LCD touch screen for displaying and editing therapy parameters, performing actions and following instructions. Hardware and disposable interfaces require manual operation to achieve the intended use, e.g. opening doors, closing clamps, connecting tubes or hanging solution bags on hooks. The machine is a mobile device having wheels to be moved between different rooms. The user interface includes accompanying documents such as Instructions for Use.

Use scenarios

Use scenarios describe the user actions, the way they follow each other and the connections between them. Task analysis method was used to understand the flow of the user activities and to identify essential and frequent functions. This is depicted in a workflow diagram which better represents the relationship between the activities, and is described in the Usage Process Specification. The process contains the main tasks such as preparing the device for therapy, connecting patient to the device, running, maintaining and ending the therapy. Further details are identified as subtasks and are later used for building up use scenarios, e.g. changing bags or handling alarm situations (recognize the alarm, identify the source, solve the problem, continue therapy). When these tasks and subtasks are identified, it is important to identify which are the so called Frequently Used Functions[3], because they will be part of the Primary Operating Functions, as mentioned in the later chapters.

At this point the high level early concept of the user interface shall be developed. This high level concept can consist of drawings of the device or a verbal description of it.

User Risk Assessment

As identifying and understanding all the hazards and hazardous situations is a complex task, it requires an interdisciplinary team: risk management engineers, clinical specialists, system engineers, marketing experts and usability engineers. They all have different fields of expertise and that ensures a wider range for the findings. Usability engineering is a user-centered process, meaning that it focuses on human factors and the use related characteristics. These aspects are then integrated in the risk management process. To make sure that the risk assessment covers all use errors, it is important for usability and risk engineers to strongly collaborate.

According to ISO 14971, risk analysis is the first part of the risk management process, which includes collecting information on anticipated and unanticipated hazards and hazardous situation. Several methods can be used. A good way is to start with analytic approaches – such as literature review, task analysis, Failure Mode and Effect Analysis (FMEA), etc. – for the anticipated use scenarios, then to use empirical studies – formative usability test (see 3.4 Usability Verification), contextual inquiry, etc. – where the information is extracted from using the device. These empirical studies can explore such scenarios and can lead to such situations that were unanticipated before. As their strengths and weaknesses complement each other, these different approaches are best used together. In the first iteration of the usability process we have used only analytical studies. As explained later, the formative evaluation tests have provided additional information that were added to the hazardous situation list and were evaluated from risk point of view.

Inputs for the Use Risk Assessment are collected from different sources:

1. Acquiring information on known problems with similar devices or previous versions of the device (market observation) that can be gathered from public databases or the complaints arriving from

---

[3] As a general definition, we categorize those function as frequent that are used at least once every occasion in order to achieve the medical benefits of the device

users. Various databases have been searched for known issues, such as MedSun [7] and Maude [8]. Some examples of the findings:

- Incorrect User Response to Hardware Instruction
- Lack of indication of battery status on screen.
- Incorrect adapter connection
- User unable to clear alarms. Override and continued treatment.
- User fails to respond to alarm correctly.
- Incorrect dose / flow rate settings, Overriding limits, misunderstanding of parameter settings
  These findings are listed in our Usability Analysis document and if they are considered relevant to our device they are imported to the Use Risk Assessment document.

2. Collecting all product characteristics that can have an impact on safety. Information gathered during the user research can help make a full list, and the IEC 62366 Annex E provides a list of questions that can be used to identify device characteristics associated with usability that could impact safety. This list and the corresponding answers for our project can be found in the Usability Analysis document. These answers then form an input to the Use Risk Assessment document.

3. Identification of hazards and hazardous situations which are known or can be foreseen. A hazard is a potential source of harm, leading to an injury or even death of the user, the patient or other people. A hazardous situation is a combination of events where a person is exposed to a hazard. All possible harms, hazards and hazardous situations have to be identified and described how they can occur. This is usually done by the guidance of medical experts who have experience with these situations and can estimate the severity of a harm. These are collected in a table format in the Hazardous Situation List document.

4. All the main and subtasks (activities) from our Usage Process Specification were analyzed and the corresponding anticipated hazardous situations were collected considering information collected in the application specification. Some examples:

- user uses incorrect disposable kit that is not appropriate for the selected therapy modality
- user selects incorrect therapy modality
- user fails to open clamp on anticoagulation line
- user connects pressure line to incorrect port
- user fails to connect venous line to patient when entering therapy and leaving it on the priming waste bag
- user fails to remove air from venous line
- user fails to change empty anticoagulation syringe

Once the harms, hazards and hazardous situations have been collected, the next step is to estimate and to evaluate the corresponding risk level based on severity and probability of occurrence[4]. This was done according to the process described in the ISO 14971 standard. The use risk analysis is structured to the same tasks and subtasks identified in the Usage Process Specification. The risks were placed in a so called risk matrix that shows which of these tasks and subtasks have an unacceptable risk and are therefore safety relevant. These tasks are considered critical and are in the centre of attention throughout the whole usability process. They form the second part of the Primary Operating Functions list. In the previous section we have identified the Frequently Used Functions and that list has now been extended with the Safety Relevant Functions. It is important to note that some of these tasks (functions) are both frequent and safety relevant. The following Table 1. shows some examples of our Primary Operating Functions.

Table 11. Examples for the Primary Operating Functions

Tasks, subtasks	Frequently Used Function	Safety Related Function
Connect venous line to patient	YES	YES
Change syringe containing anticoagulation solution	NO	YES
Remove air from venous line	NO	YES
Change solution bags	YES	NO
Switch on machine	YES	NO

---

[4] The severity is derived from the Hazardous Situation List document established by clinical experts. The probability of occurrence is in general considered frequent (the highest probability in our risk management matrix) for use errors given that we do not have reliable statistics on probabilities.

3.2 Requirements

The purpose of this phase is to provide testable requirements that will guide the design of the user interface. User interface design is an iterative process, which provides design solutions for the user needs. These needs are the outcome of the previous phases and they are translated into user interface requirements. The use analysis phase can reveal latent needs from the users, marketing professionals can provide market requirements in order to have a competitive advantage. Usability concerns can extend this list and as also mentioned before, safety considerations can invoke requirements for risk mitigation or control measures.

Usability Specification collects all requirements concerning the user interface elements. In our project some of the requirements are stored in other documents and only referenced here.

Major parts of the Usability Specification:

1. The System Requirement Specification (System RS) contains requirements regarding hardware and disposable user interfaces and also about accompanying documents. These requirements are referenced by the Usability Specification and used for implementation verification (e.g. *The user interface shall provide a control to set fluid temperature.).*

2. The Software Requirement Specifications (Software RS) include requirements for the look and behavior of the Computer User Interface (graphics, audio and visual signals, etc.). These requirements are referenced by the Usability Specification.

3. The Functional Safety Requirement Specification (Functional Safety RS) contains requirements for risk mitigation that originate in the use risk assessment. Following the ISO 14971 standard, the following risk mitigation techniques are used:

   Inherent safe design: the devices is designed in a way that the hazardous situation can not occur at all. E.g. designing round edges on sheet metal parts to avoid service technicians cutting their fingers.

   Protective measure: if the hazard cannot be eliminated, the risk shall be controlled. e.g. to avoid the user to get his fingers hurt by a rotating pump, a protective pump door is required to cover the rotating parts. In case the door is open, the pump is stopped.

   Information for safety: this includes all kinds of labeling and written information, such as labels on the device or the Instructions for Use. This can be for example color-coding the different tubing lines or giving detailed guidance with images for the user to follow the correct steps during patient connection. Sometimes the design cannot be further improved and the user's mental model has to be altered by training or requiring new skills. In our case, both user groups (clinical and technical staff) will be trained for using the device, and these trainings will cover the most important safety issues that the users might face as they are using the device.

   These requirements are referenced by the Usability Specification.

4. The Usability Specification contains the list of Primary Operating Functions (according to IEC 62366). As mentioned before, the Primary Operating Functions is the sum of the Frequently Used Functions and the Safety Related Functions.

5. The Usability Specification contains best practices and heuristics from user interface design experts. Items in ANSI/AAMI HE:75 provide the most important design guidelines concerning medical products, so relevant points were used.

6. The Usability Specification includes marketing requirements to ensure competitive advantage (e.g. *"All users shall be able to finish the preparation process in 15 minutes."*).

7. The Usability Specification describes the frequent and reasonably foreseeable worst case use scenarios built up from Primary Operating Functions. These scenarios will form the tasks of the participants on the Usability Validation Test (see 3.5 Usability Validation). Corresponding acceptance criteria are formulated, either quantitative (e.g. *90% of the users shall be able to complete the syringe change procedure.*) or better qualitative (e.g. *"The user shall successfully perform a bag change procedure without external help."*) to meet FDA expectations.

3.3 Design and implementation

As the interface design evolves from concepts and sketches to high fidelity working prototypes, the requirements become more specific and detailed. In this phase design alternatives compete with each other and this is how the final product is created.

Two main documents capture the details of the evolving user interface design. The User Interface Style Guide shared between similar devices within the company drives the general appearance and behavior of user interface elements thus enhancing brand recognition. The User Interface Design document includes specific details such as 3D models of the device and the disposable kit, screenshots of the graphical user interface screens, defines the structure and hierarchy of these screens along with navigation paths.

Some aspects of the user interface design[5]:

– location and positioning of user interface elements based on anthropometric data related to the intended user group, for example the display on the front of the device is positioned in a way that all nurses can see it properly without any distortion in the image regardless of their height;
– control panel layout: position and arrangement of controlling elements and displays;
– hierarchy, navigation: the structure of the interface, how the screens are connected, and how they can be reached; - this shall be understandable for the user when interacting with the graphical user interface;
– screen layout: how the different screens are built up, what kind of areas and elements appear and how they are positioned, for example the control element to set up pump flows shall be grouped together and shall be separated from actual flow values;
– user interaction modes: deciding how the elements should behave upon user interaction, whether a toggle button shall control the status of the fluid warmer or a radio button (on/off switch);
– fonts, style, colors, icons: aesthetics is an important factor in providing a clear, understandable design which improves usability and is a good means to guide user's attention. A nicely designed user interface suggests reliability and ensures the correct mental model.

The design and implementation activities can be done parallel. The design does not have to be very detailed before it is implemented. The sooner one starts creating prototypes and getting feedback on them (see 3.4 Usability Verification), the sooner the design flaws can be corrected. Early prototypes can be hand-drawn sketches of screens, or foam mock-ups showing very little details - they are called low-fidelity prototypes. High-fidelity prototypes are interactive, working versions of the product that are more realistic. We have created prototypes with little functionality but more focus on the structure and content of the screens; concepts for various graphical elements in a non-working screenshot to decide between alternative looks and high-fidelity working prototypes that run in a simulated environment. Some of these prototypes were created either in a regular graphic design software, in a presentation creating tool or in an integrated software development tool. They were used for either design reviews or for formative usability tests (see 3.4 Usability Verification).

3.4 Usability Verification

Given that the Usability Specification consists of many different types of requirements, it is not an easy task to verify its whole content. Depending on the verification method, we have identified two categories: one for the user interface requirements in the requirements specifications and one for usability related information. Therefore usability verification consists of two main parts.

Verifying user interface requirements:

These requirements reside in different requirement specification documents (System RS, Software RS, etc.) which are all referenced in the Usability Specification, and the verification activity means that their correct implementation is checked by e.g. functional tests. This is documented in the corresponding Verification Plans and Reports (System VEP/R, Software VEP/R, etc.).

Formative evaluation of the user interface:

One of the core principles of the usability engineering is to constantly assess the design and correct it according to the findings as soon as possible. The changes then need to be assessed again and this

---

[5] Not only related to the graphical user interface.

leads to an iterative process where the steps are not linear and distinct. Design solutions are repeatedly changing while they are being assessed. In some cases this iteration points back to the Use Analysis or the Requirement phases, if the core assumptions about the user profile or the use scenario have to be revised or the user interface requirements need to be modified (see the backward arrows starting from Usability Verification step pointing to all previous steps on Figure 1.)

Depending on the scheduling in the timeline of the design process we can talk about two types of usability evaluation: the formative (during the development phase, forming the design) and the summative (at the end of the development, validating the design).

The formative tests take place while the device is still being designed and implemented (when the device can still be changed) and the goal is to assure that the development is on the right track. It can be used for several goals:

- to identify development opportunities, design strength and flaws of the user interface and usability issues. Usability issues can involve anything that a user does in a different way than expected, e.g. not following the workflow, making a mistake, assuming something is correct when it is not or being confused or lost.
- check whether the usability requirements are attainable,
- to compare competing design solutions
- and nevertheless to identify unanticipated hazardous situations which are then tracked back to the Use Risk Analysis for assessment and risk mitigation.

During the formative evaluation we have focused on the design flaws and investigated whether unanticipated hazardous situations have occurred. Since the design is not finished the test is conducted on a preliminary concept or prototype. It is beneficial to conduct several formative tests since it is a great way to refine the concept and avoid surprise (and failure) at the end. The formative tests are less formal, the rules are less strict and there are usually no acceptance criteria for the performed tasks.

There are many ways of categorizing the various evaluation methods. Similarly to use risk and hazard analysis (see 3.1 Use Analysis), the main difference is whether the intended user is involved in the evaluation: analytical (e.g. design review) and empirical methods (e.g. usability test).

Analytical methods do not require the participation of users, but are carried out by human factor or domain experts. They break down the user interface systematically and analyze it thoroughly. They focus on design heuristics[6] and domain specific characteristics. These supplement empirical methods, since they can be used to identify such design flaws and usability issues that would not occur during empirical research. Most common methods are the heuristic evaluation, the design review and the cognitive walkthrough evaluation. We performed design reviews where we checked the compliance with usability design heuristics, design best practices of the ANSI/AAMI HE:75 and other applicable standards. We have identified flaws in our use scenarios while conducting a cognitive walkthrough and we have made design decisions on graphical appearance when comparing design alternatives. Analytical methods may be too theoretical and hence may miss important problems that only arise during use, but they provide valuable input to use risk analysis and the user interface design.

The most common empirical usability evaluation method is a usability test. These tests are conducted in either artificial or real environment with the aim of measuring user performance on typical tasks while a representative user interacts with a prototype of the product. As a formative usability test is less formal and strict than the summative usability validation test (see 3.6. Usability Validation), it does not necessarily cover all tasks and all user interface elements, and the number of participants can also be less.

We have conducted two formative usability tests. The tests involved a participant performing a task on the device, while commenting on his actions[7]. During the evaluation session one or more evaluators were present, who gave tasks to the participant and interacted with him (moderator), or just watched the course of the test without intervention and noted down findings based on the behavior of the participant (observer).

The first test covered only some part of the use scenarios, focusing on the graphical user interface only, while the second test included some manual tasks with the hardware part of the device, too. In total 8 and 5 test participants took part in the first and test, respectively. In average 50 usability issues have been identified both times. Around 70% of these issues were related to the graphical user interface, 10% to the hardware and disposable elements, the rest was about the device behavior at the software level. Some examples of the unanticipated use errors and the design modifications to avoid such errors in the future:

---

[6] Design heuristics are well-established design rules and principles based on years of experience and research.
[7] This is called the thinking out loud or think-aloud technique.

– Some of the users were confused by the terminology used on the screen so we updated the wording according to current medical terminology.
– The structure of the GUI was not understandable in some cases when users were looking for a specific function on another part of the user interface. We have either duplicated the functions for both screens or placed it on another.
– Entering time settings was not easy, because the minutes could have been mistaken to hours, so we have redesigned the user interface.
– The screens were found to be too crowded and the most important information was not prominent enough so we redesigned the information displayed on the screen.

The overall impression was very good with the device and the users were satisfied with its usability. The good task completion rates indicated that we are on the right track.

3.5 Usability Validation

When the design is considered to be finalized, no further usability issues can be identified and the user interface requirements have been verified, the validation process can begin. We have distinguished two steps during this phase:
- Usability Validation Test or Summative Usability Test
- Usability Validation.

Usability Validation Test

It is also called Summative Usability test, because it takes place when the design is ready. The aim of the summative test is to validate that the product meets the user needs and provides safe use for the user. By this time all risks have been mitigated with risk control measures. Summative test can only be a usability test, because that is the only method that involves the users and gives valid data on whether the product meets their needs. The usability validation test has strict acceptance criteria (coming from the Usability Specification), it is conducted with 15-20 participants from all user profiles and in the use environment which is specified in the Usability Analysis.

At this point it is important to state that IEC 62366 requires the details (methods, participant selection, etc.) and the acceptance criteria for the validation test to be written in the Usability Validation Plan. We captured these details in a dedicated Usability Validation Test Specification (TS) instead, as Usability Validation Plan document is used for a different purpose in our project.

We are in preparation for the usability validation test at the moment, creating the test specification. The Usability Validation TS describes that both user groups (medical professionals and technical personnel) will be involved in the validation test – – and we plan to have 15 test participants in each group. The two groups will be tested separately, both in time and space, and they will have different tasks to perform as their use scenarios greatly differ. The tests will take place in a simulated environment where the audio, video and screen capture will be provided. This will give us the possibility to later evaluate the participants' actions based on the recordings. After the test we will conduct one-on-one interviews with each of the participants to gather more information on their experience. During the test they will not be asked to formulate their thoughts so we will interact with them as little as required when giving the tasks – this makes the test more realistic. We will focus on the critical tasks (e.g. alarm handling, etc.), but also try to cover as much of the machine's functionality as possible. The use scenarios for the test are the ones defined in the Usability Specification: covering both frequent and worst case scenarios. The main goal is to prove that all risk control measures related to usability are effective, thus the design is free from unacceptable risks. We plan the test sessions to last for 4 hours. After the sessions every usability issue will be analyzed and decide if the acceptance criteria are met or not. In case of a failure, another iteration is necessary (through design, requirement or use analysis phases), with a repeated validation test.

Usability Validation

The usability validation process is more than conducting the summative usability test. It is a confirmation by examination and provision of objective evidence that device specifications conform to user needs and intended use(s), meaning that the requirements for a specific intended use or application have been fulfilled. Usability validation ensures the safe and effective use of the device. These activities contain:

- verifying implementation of user interface requirements including risk measures (e.g. System Verification Report (VER)),
- proving risk measure effectiveness (e.g. Usability Validation Test Report (TR)),
- ensuring compliance with internal usability engineering plans, standard operating processes and with international regulations and standards (e.g. Design Reviews),
- checking the completeness of the Usability Engineering File, meaning that all required documents are created and are correct (e.g. IEC 62366 Test Report File).

Only after the usability validation is finished can we state that our device is safe and can be used for the intended medical use by the intended users.

3.6 Post-market surveillance

Usability activities are not finished when the product reaches the market: gathering clinical data on usage and user behavior becomes useful in identifying strengths and weaknesses of the product. Post market surveillance and vigilance reporting are part of the design transfer and design change processes (they are barely mentioned in the usability standards), but are supported by the user-centered thinking of the usability engineering process. In our company the customer complaints are collected in a database and are then assessed and categorized whether they are usability issues or not. Also, FDA databases, such as MedSun, are regularly checked for incident reports on own and competitor devices. As mentioned in 3.1 Use Analysis chapter, these information are then collected and analyzed during the development of the next version of the device or a new model.

4. Conclusion

After giving a brief introduction to what usability is, the main focus of medical usability has been defined: the usability engineering process must be carried out together with the risk management process as they are connected in multiple steps. We have collected all international and US standards related to use risk management. With a strong focus on the IEC 62366 standard, we have extended the usability engineering process with our experience in developing a blood purification device. The importance and necessity of risk-related activities have been emphasized while describing each step in our process. During Use Analysis the main information on device use are collected, the complaint databases are searched for previous incidents and the list of hazardous situations along with device characteristics that impact use risk is created. These information augment the risk analysis process with a human factors point of view. By the end of the Use Analysis, the frequent and critical user actions are identified (Primary Operating Functions). During the Requirement step the user interface related requirements are established (along with the risk control measures) and the frequent and worst case scenarios are built from the Primary Operating Functions. The Design and Implementation of the user interface is done based on the requirements in an iterative fashion. Usability Verification tests whether the user interface requirements have been correctly implemented and formative usability tests seek for unanticipated hazardous situation and usability issues. After all design flaws are corrected and no further unanticipated use errors can be found, the Usability Validation begins. This includes a summative usability validation test which proves that all risk control measures are effective and the device meets the user needs. The Usability Validation process provides evidence that the manufacturer has done all the required activities in order to conclude that the use of the medical device is safe. Post-Market Surveillance reports customer complaints from the market after device launch.

We have provided an integrated framework of process steps, input and output documents and connection with the risk management process. Applying this framework makes our everyday work easier and allows us to develop safe and usable devices.

References

1. IEC/ISO 60601-1-6:2010, *Medical electrical equipment – Part 1-6: General requirements for basic safety and essential performance – Collateral standard: Usability,* International Electrotechnical Commission. Geneva.
2. IEC/ISO 62366:2007, *Medical devices – Application of usability engineering to medical devices,* International Electrotechnical Commission. Geneva.
3. Food and Drug Administration (FDA) draft guidance, *Applying Human Factors and Usability Engineering to Optimize Medical Device Design*, 2011:
4. Association for the Advancement of Medical Instrumentation (AAMI) HE74::2001, *Human factors design process for medical devices*
5. Association for the Advancement of Medical Instrumentation (AAMI) HE75:2009,*Human factors engineering – Design of medical devices*
6. ISO 14971:2007, *Medical devices – Application of risk management to medical devices,* International Organization for Standardization. Geneva.
7. Medical Device Safety, http://www.fda.gov/MedicalDevices/Safety/default.htm
8. Manufacturer and User Facility Device Experience (MAUDE) database,
   https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfmaude/search.cfm

Linguistic Analogy for Software Process Innovation

Kouichi Kishida

Software Research Associates, Inc., Tokyo, Japan
e-mail: k2@sra.co.jp

Abstract

There are many useful metaphorical notions in linguistics to think about soft-ware process innovation. Continuous evolutionary change is the essential nature of any software system. Human languages also change over time. In this paper we will investigate some notions in linguistic study to apply issues of software process innovation.

Keywords

Conceptual Modelling, Language Change, Innovation Factors

Reference

The Need for a Structured Approach towards Production Technology Roadmaps in Innovation-driven Industries

Martina Flatscher[1], Andreas Riel[2], Tobias Kösler[1]

[1]ZF Friedrichshafen AG, Friedrichshafen,Germany,
email: {martina.flatscher, tobias.koesler}@zf.com
[2]EMIRAcle c/o Grenoble Institute of Technology GSCOP UMR5272, Grenoble, France,
email: andreas.riel@grenoble-inp.fr

Abstract

As innovation cycles are becoming shorter and technological progress faster, the need for reliable decision support for product and production planning is rapidly gaining crucial importance. To this aim, strongly innovation-driven industries like automotive use roadmaps relating products and technologies to a timeline from a specific company's viewpoint. The roadmapping process, however, is typically neither systematic nor transparent. Furthermore, there is a lock of integration of product roadmaps and production technology roadmaps, although these cover complementary and mutually dependent aspects. This paper investigates the motivation and necessity for systematic and integrated roadmapping with a specific focus on production industries, and introduces a related automotive supplier industry research project that aims at designing and implementing a holistic approach to integrated technology roadmapping.

Keywords

Innovation Management, Technology Management, Technology Planning, Process Innovation, Ideation

Game Changing Beliefs for the Product Developing Organization

Morten Elvang

DELTA, Venlighedsvej 4, DK-2970 Hørsholm
meg@delta.dk

Abstract

How can you improve as a product developing organization? Are there principles that you as an organization can chose to believe in, which will then help change the odds of success in your favour? This paper will argue for a positive answer to this question and further attempt to suggest a useful set of such beliefs – I call them 'Game Changing Beliefs'

Keywords

Effectiveness, Success, Value based leadership, Agile, Lean Product Development

Reference

Functional roadmap - scheduling in complex mechatronic projects

Damjan Ekert[1], Bernhardt Steger[2], Dr. Richard Messnarz[1]

[1]ISCN Ges.m.b.H., Austria
[2] Performing Facilities Software & ConsutlingGes.m.b.H., Austria
dekert@iscn.com, b.steger@performing-facilities.com, rmess@iscn.com

Abstract

This article describes best practices for planning complex mechatronic project. The development, coordination, integration and testing of each component, especially software is a challenge itself. The technical project leader must find a balance between a manageable project schedule and the level of detail to be able to react to delays and deviations in the schedule and deliveries to the customer. The article will present the concept of a functional roadmap, the structure of a typical mechatronic schedule with all its challenges and introduce how tasks and activities between all involved stakeholders can aligned to have as a result a coherent and consistent project schedule.

1 Motivation

Planning a complex mechatronic project is more often a tough job. On one side there are a lot of different stakeholders with different views and needs increasing the complexity of scheduling. On the other side the administrative effort for a lean schedule should be as small as possible to free key members of the development team to handle technical issues. Complicating a lot of project leader handle different subsystems, very often technical needs clash with economic constraints. Many cooks spoil the soup – and also different planning tools (e.g. MS Excel vs. MS Project) and different planning focuses (e.g. industrialisation vs. software development) make things worse.

But how can we minimize the (administrative) effort for creating and maintaining the project schedule but still have a consistent and coherent planning? Is there no other way than handle up to 9000 lines in a project plan?

Out of our long time experience (as project leader ourselves and also consultant for project management) we developed a different approach – an approach that sets the focus on the function to be delivered.

2 Project Interfaces & Roles
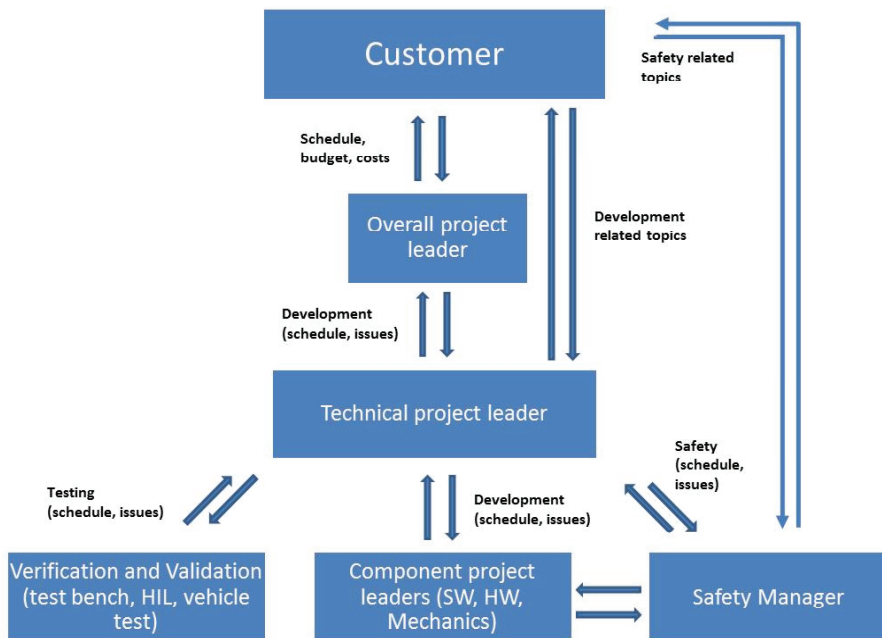
Figure 1: Project Interfaces

In most mechatronic projects the following roles are identified

Overall project leader / program manager (PL)

The overall project leader is the representative of the company to the customer. He/she is responsible for schedules, budget, costs, agreements, change management and the communication with the customer. He/she has a milestone plan where all relevant project schedules from the customer, departments of development, purchase, industrialization, quality, sales, logistics etc. are planned and tracked. Any changes in the milestone plan are communicated immediately to the customer.

Technical project leader (TPL)

The technical project leader work at system level and is responsible for the development of the integrated mechatronic system. He/she is responsible that all customer (functional/non-functional) requirements are developed and delivered on time to the customer. The technical project leader is coordinating the components project leaders, the testing department and the safety manager to work efficient together and that all changes to the schedule are communicated and adjusted accordingly. All changes related to the project milestones are discussed with the overall project leader

Component project leader (CPL)

The component project leaders are responsible for the development of their components (SW, ECU, Sensors, and Mechanics) according to the requirements in the agreed time and budget. They have to ensure that the technical project leader is regularly informed about any deviations from the plan (scheduled dates, technical issues or costs). The technical project leader must update the component project leaders if there are any technical changes to the product, schedule or costs.

Safety Manager (SaM)

The safety manager is responsible for all activities concerning the functional safety in the project. He/she plans all safety related milestones and tasks and works closely with the technical project leader and the component project leaders (mainly SW and Hardware). The safety manager is usually also the contact person to the customer in case of queries regarding the functional safety.

Verification and validation (STM)

The testing departments are responsible for all testing activities in the project on all levels (SW, HW, system). They ensure that all customer requirements are tested. All testing activities on system level are planed with the technical project leader, on the component level with the component project leader. The technical project leader is informed about all testing activities.

3 (Functional) Roadmap – Content Delivery Plan

At the start of the project the customer has already a very strict timeline regarding the maturity process of his vehicle, i.e. important dates for vehicle integration steps and also SOP are given and must be fulfilled by the supplier. Therefore the project leader (PL) sets up a master schedule containing the integration steps of the customer, e.g.
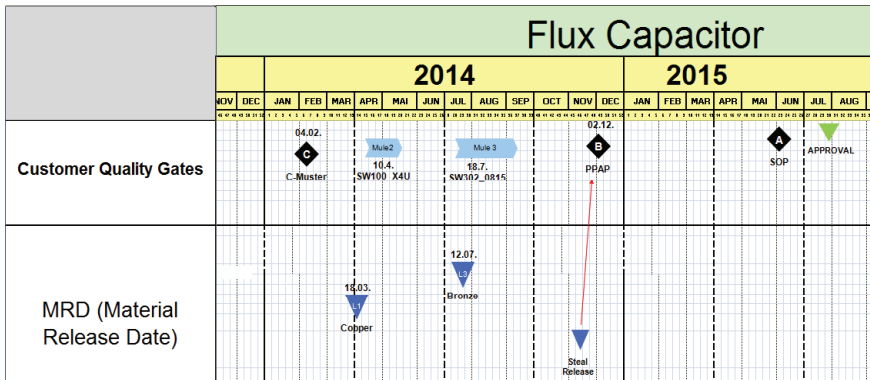
Figure 2: Milestone plan

In reality this schedule is also enriched by important development dates like customer engineering approval (PPAP), approvals of technical constructions and so on. Additionally a raw maturity planning of the system to be delivered is added to this master schedule, e.g. milestones, when the customer expects full functionality of the system, availability of system diagnosis etc. This is the customer road map the supplier must fulfil.

The technical project leader uses this plan to discuss with his component project leaders and test manager the possible content delivery plan (CDP) of the system to meet the required milestones within time, budget, material and last but not least human resources. The proposed CDP is discussed with the customer to obtain the coordinated development road map for the project. These milestones are entered into the technical development schedule using the labels of the customer for better discussion and coordination between customer and supplier. So, the general development road map contains the following milestones:

- delivery dates to the customer

- content delivery plan per integration step / delivery

- additional quality gates of the customer

- approvals / clearance of the customer needed by the supplier

- material release dates (e.g. for vehicles to test the vehicle integration)

- quality gates of the supplier
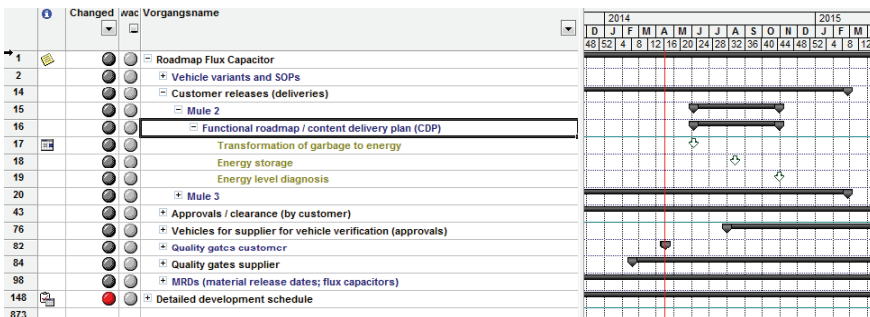
Figure 3: (Functional) Roadmap

The content delivery plan is the centrepiece of the schedule. For future releases it is used as a forecast for budget and material, for the actual release the CDP offers a guard, whether the desired feature will be delivered on time.

Of course also every component project leader creates his own schedule and road map, e.g., the software road map contains the following milestones:

- material release dates (e.g. for mechanic and/or hardware parts)
- content delivery plan, i.e. software functions being delivered
- requirement freeze date
- resign freeze date
- application freeze date
- software integration end date
- quality reviews
- software delivery date(s)
- software approvals

4 Level of Detail

To manage the project not only the given road map including the content delivery plan may be used, but also more tasks must be planned (and tracked) in more details. Therefore the development of the system is divided into the maturity process of each component of the system. For better coordination within the development team the detailed planning uses the labels of the supplier (e.g. A, B, C for hardware; SW100_X4U, SW302_0815 for software, and so on).
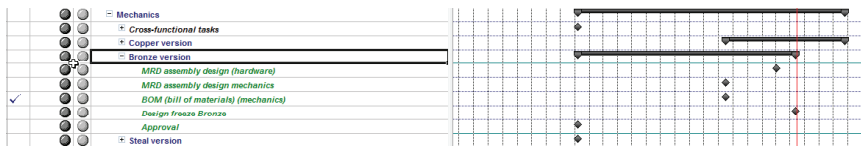
Figure 4: Maturity planning (e.g. mechanics)

Additionally the different component releases are summed up to the system release to be delivered regarding the road map. Also the content delivery plan of the road map is divided into the features of the component and for each component each feature is calculated regarding time, budget, material, and human resources.
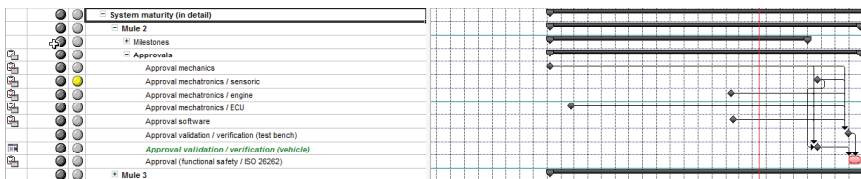
Figure 5: System release package

The detailed planning of each feature of each component is done in the component development schedule by the component project leader. The technical project leader plans the overall forecast for each feature and controls deviations. Therefore the technical project leader tracks periodically the

milestones and efforts with the component project leaders. To keep the planning lean the require-ments to fulfil are not planned in detail by the technical project leader but the features. For monitoring the technical requirements in all its particulars the technical project leader groups them to the features specified in coordination with the system engineer / requirement manager. As the requirements are also connected to a required release date the particular content of a feature to be delivered can be easily determined and monitored.

Requirement ID	Requirement text	System feature	Required release date
SysPH_4711	The energy level must be checked every 10ms.	Energy level diagnosis	Mule 2
SysPH_4712	If energy level is lower than 65% more garbage must be filled in.	Energy level diagnosis	Mule 2
SysPH_4713	If energy level is lower than 40% a warning beep rings up and warning bit is send on communication channel via message STAT_FLUX_CAPA.ENERGY_LEV_STATUS.	Energy level diagnosis	Mule 2

Figure 6: Mapping between (system) requirement, function and release date

Of course, not only the development of the components is planned within the technical schedule but also the testing and validation of the system.
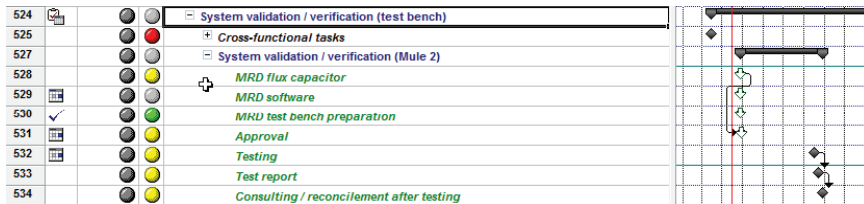
Figure 7: Verification / validation schedule

As mentioned before the specific implementation of a feature by a component is part of the detailed component schedule. E.g. software project leader may distinguish between major releases (to be de-livered to customer) and minor releases (being tested internally be the supplier). Each minor release may contain the change requests connected to the feature to be delivered and additional tasks like:

- technical clarification

- specification

- design

- implementation

- (implementation) test

- cross functional tasks like software integration etc.
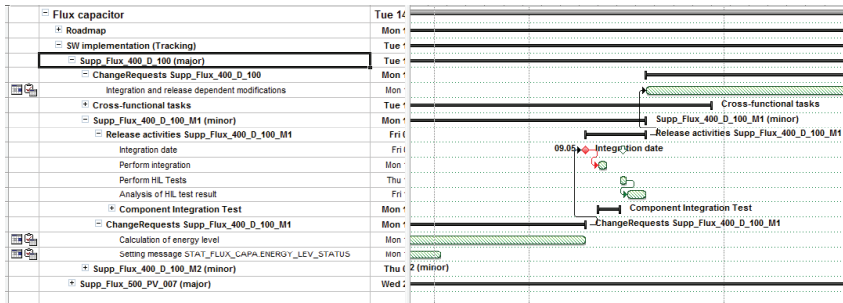
Figure 8: Project Interfaces

All tasks are planned in detail and connected either to the responsible team leader or to the appropriate human resource. By using special attributes and filter the efforts spent and estimated may be coordinated easily between the technical and the component project leaders.

5 Cross-functional Tasks

In each project there are numerous tasks and activities which are not planed within the customer release and delivery phase, they are rather performed during a certain period in the project or planed according to internal milestones (e.g. quality gates). The cross functional tasks include in most cases activities related to the supporting and management processes from the HIS SCOPE of Automotive SPICE (quality assurance, change management, configuration management etc.). The following types of activities were identified and planned:

- Development of central planning documents (quality plan, configuration management plan, test plan…)

- Regular and repeating activities (customer and internal meetings, workshops, reviews)

- Exceptional activities, where the project team members are blocked for more than two days (assessments, reviews, improvement workshops)

- Tasks related to functional safety, requirements managements, FMEA etc.

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Release A	▬▬▬▬											
Release B			▬▬▬▬									
Release C					▬▬▬▬▬▬							
Release D										▬▬▬▬▬		
Quality Assurance			◇		◇		◇	◇		▬▬		
Project Management	▬▬	▬▬	▬		◇	◇		◇				
					▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬							
FMEA	▬▬▬▬											
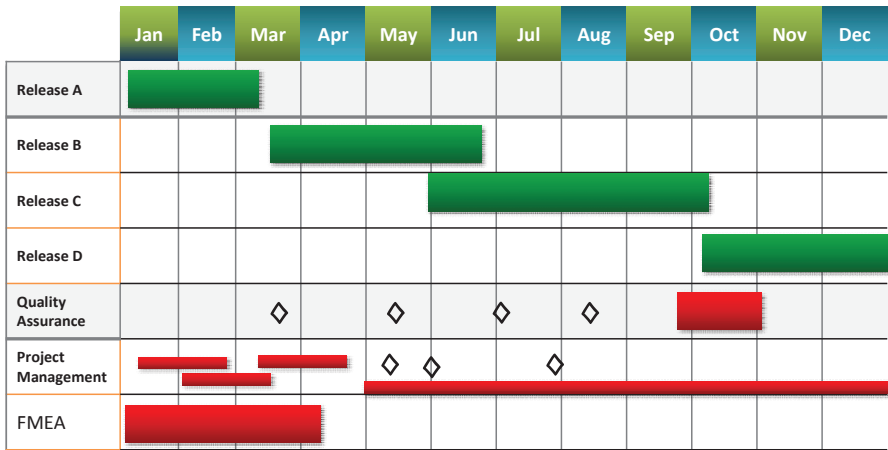
Figure 9: Cross-functional tasks – milestone plan

Figure 10: Cross-functional tasks – detailed plan

6 Project Control and Matrix Organisation

Each project leader has to deal very often with different human resources to be planned: on one side the project leader can directly coordinate human resources being attached to the project, on the other side the project leader must hand some development tasks over to departments not knowing who will really work (or deliver) on the task.

For all human resources attached to the project the project leader creates specific project calendar depending on the availability of the team members. If the project team contains members with different work contracts (e.g.: 40 hours or 20 hours) for each different type of contract a specific calendar is created keeping also in mind the Pareto principle (i.e., if e.g. someone has a 20 hours' work contract he may work only 16 hours efficiently for project). This is done also for team members not working full time for the project.
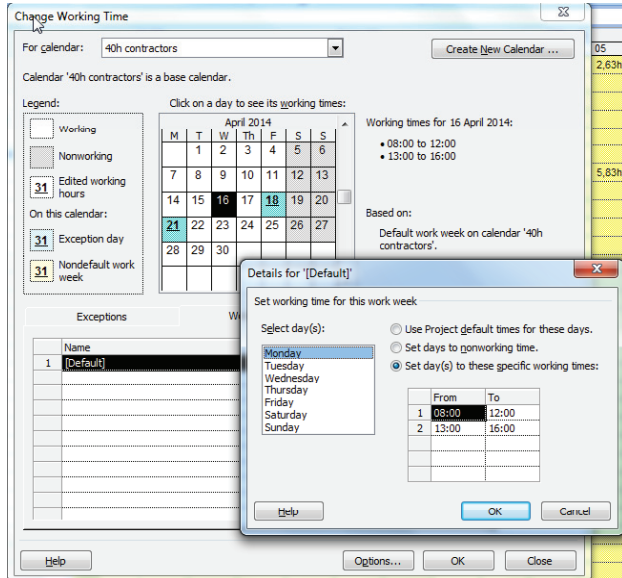
Figure 11: Example calendar

The specific calendars are afterwards attached to the human resources.
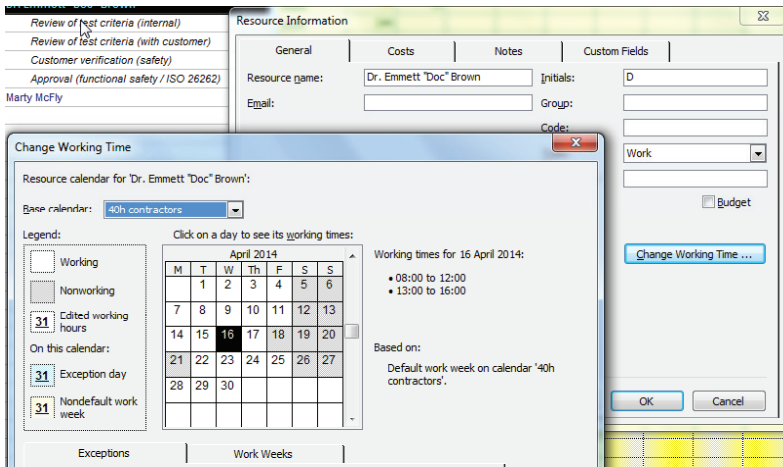
Figure 12: Human resource with connected calendar

Also public and individual holidays are entered so that the project leader can track the workload of each team member he is responsible for.

Figure 13: Workload of human resources

All tasks in the development schedule that cannot be attached to project team members are handed over to the responsible department leader for further planning and execution. The department leader uses his own (department) schedule to control the work load of each department member cross different projects.

Any conflicts must be stated, discussed and if needed also escalated between project and organisation.

7 Project Schedule Alignment

In order to track and communicate all changes to the schedule (top-down and bottom-up) the schedules of all involved parties need to be aligned. Rather than using tool automated update functions a personal (face-to-face) schedule adjustment was much more in favour. On the system level, each week the technical project leader organised a meeting, where all changes in the schedule were discussed with all involved parties.

The same approach as on the system level was also adapted on the component level, especially in the software development where a schedule alignment between the software project leader, the developers, architects and integration and testing departments was required.
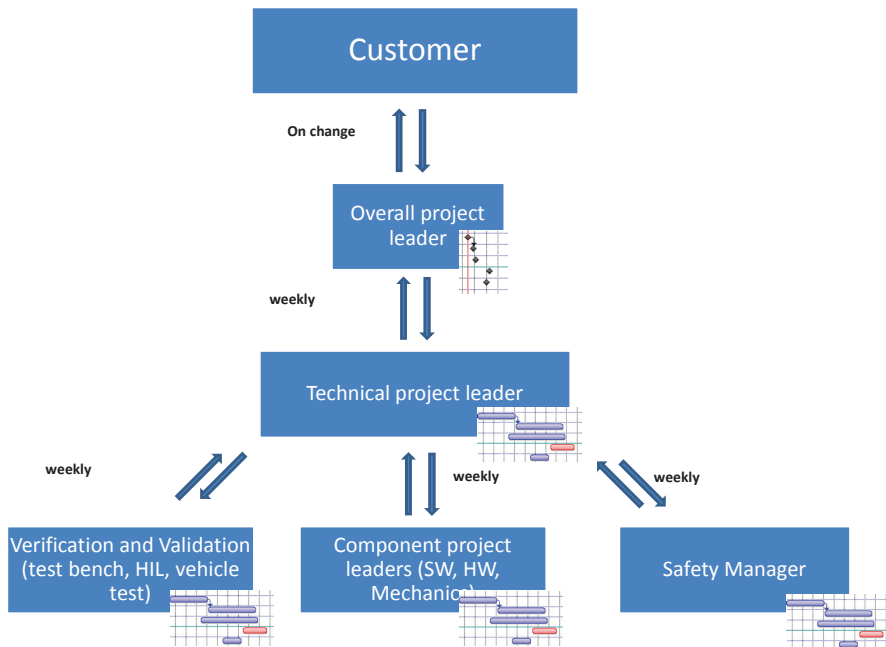
Figure 14: Project schedule alignment

8 Implementation Experiences and Best Practices

The approach described above proved to have several advantages

- Automatic exchange between schedule and used tool for tracking software change requests decreases the planning effort and increases the monitoring possibilities.

- Each schedule could have their own format and structure; therefore the teams were not limited to single software solution (e.g. MS Excel vs. MS Project).

- Easier handling: several smaller plans than one big plan with many thousands of tasks.

- Easier handling: each schedule is divided into two different parts – the road map and the detailed plan; therefore the coordination effort could be minimized as all important milestones and tasks were grouped.

- Better communication: the functional road map used the labels of the customer whereas the detailed planning used the labels of the supplier.

- Dependencies between tasks and activities were quicker identified.

- Deviations from the plan were discussed and communicated to all involved parties.

- Solutions to the deviations and rescheduling took part in the team.

Additional best practices increased the performance of projects sticking to this approach:

- There should be a translation map between function labels of the customer and function labels of the supplier to support communication between customer and supplier development team.

- Using additional attributes and filter speeded up communication (by focusing on milestones and/or functions for specific stakeholders).

- Using planned target dates for milestones and linking releases (and/or function) efforts to the specific milestone leads to lean guards regarding the fulfilment of milestones.

- Using a generic approach to system (and/or component) specification and connecting the requirements to given (supplier) functions lead to a raw but very quick schedule of the content delivery plan.

- Only the actual and the next release were planned in detail in the schedule. Releases already passed (and finished) were deleted out of the schedule to keep the schedule as lean as possible. As a version control is mandatory a later check of already passed releases was still possible.

- The working hours were uniformly distributed throughout the week and only if the workload regarding a week's effort was exceeded the responsible (T/C)PL had to do a re-planning to solve the problem.

9 Conclusion and Outlook

The approach using a functional road map for planning leads to a tight schedule and improves the traceability of functions and releases to the customer from system to software (and back). Implementing periodical project schedule alignment meetings helps identifying quicker impacts of re-scheduling and involves all stakeholders. The functional road map provides more freedom in planning whereas the schedule still remains stable because of a good abstraction layer without missing the necessary details. Detailed lessons learned regarding forecasts and deviations are also supported by this approach of scheduling, especially because the identification and planning of cross-functional tasks leads to a realistic workload of resources and improves the process performance.

10 Literature

[1] Automotive SPICE©, Version 2.3, September 2007, ISSN 0943-9412,
http://www.automotivespice.com/
[2] ISO/IEC 15504-5:2006 Information technology -- Process Assessment -- Part 5: An exemplar
Process Assessment Model. International Organization for Standardization (2006)
[3] Höhn,H.,Sechser, B.,Dussa-Zieger, K.,Messnarz, R., Hindel, B. (2009) Software Engineering nach
Automotive SPICE: Entwicklungsprozesse in der Praxis: ein Continental-Projekt auf dem Weg zu Le-
vel 3, dpunkt Verlag, ISBN-10: 3898645789

11 Author CVs

Dipl.Ing. Damjan Ekert

Dipl.-Ing. Damjan Ekert studied Telematics at the University of Technology, Graz, Austria. He
finished his studies with distinction. Since 2001 he works for ISCN as SW project manager
and SW integrator. He is a certified e-Security Manager and European Project Manager Train-
er. For the last three year he has been working as a consultant in the field of Software Process
Improvement, he is member of the S2QI Initiative, certified ISO15504 Assessor, member of
the European Certification and Qualification Association and vice- president of the ISECMA
e.V. (International IT Security Management Board).

Dipl.Ing. (ETH) Bernhardt Steger

Dipl.-Ing. (ETH) Bernhardt Steger studied computer science at the Swiss Federal Institute of
Technology, Zurich, Switzerland. During the 90ies he worked as a software engineer for AVL
List GmbH. Between 2004 and 2008 he worked for MAGNA Powertrain AG & Co KG as a
software project leader. He is an iNTACS™ certified ISO/IEC 15504 Provisional Assessor for
AutomotiveSPICE®. Since 2008 he works as a consultant for automotive companies, such as
ZF, ZF Lemförder, ZF Boge, ZF Lenksysteme, MAGNA Powertrain, etc in the field of system
and software process improvement as well as functional safety, and also for telecommunica-
tion companies such as T-Mobile Austria in the field of software architecture and requirement
management.

Dr. Richard Messnarz

Dr. Richard Messnarz (rmess@iscn.com) is the Executive Director of ISCN LTD. He studied at
the University of Technology Graz and he worked as a researcher and lecturer at this Univer-
sity from 1991 - 1996. In 2 European mobility projects (1993 and 1994) he was involved in the
foundation of ISCN, and he became the director of ISCN in 1997.
He is the editor of a book "Better Software Practice for Business Benefit", which has been
published by IEEE (www.ieee.org) in 1999 (the leading research publisher in the USA). He is
the chairman of the EuroSPI initiative and chair of the programme committee of the EuroSPI
conference series.
He is author of many publications in e-working and new methods of work in conferences of the
European Commission (E-2001 in Venice, E-2002 in Prague), and in the magazine for soft-
ware quality (Software Quality Professional) of the ASQ (American Society for Quality).
He is a lead ISO 15504 assessor. He has worked as a consultant for many automotive firms,
such as BOSCH, ZF TE, ZF N, Continental TEMIC, Audi/VW, etc. He is a founding member of

the INTACS (International Assessor Certification Scheme) accreditation board, a founding member of the Austrian Testing Board, a founding member of the Configuration Management Board, and he is the technical moderator of the SOQRATES initiative ([www.soqrates.de)](www.soqrates.de).

The Impact of Fuzzy Requirements on Medical Device Software Development

Martin McHugh, Abder-Rahman Ali and Fergal McCaffery

*Regulated Software Research Centre, Dundalk Institute of Technology,
Dundalk Co. Louth, Ireland
(Martin.McHugh, Fergal.McCaffery)@dkit.ie,
Abder-Rahman.A.Ali@ieee.org*

Abstract

Any software development project can experience difficulties with unclear or vague requirements. Unfortunately, this problem can be experience two fold in regulated environments such as the medical device software development industry. In the medical device software development industry, development organisations must contend with vague or "fuzzy" both the customer and regulatory bodies. As new requirements are introduced they can have a knock on effect on other requirements. These requirements should be analysed to determine if they are conflicting, cooperative, mutually exclusive and irrelevant. Only when the requirement is classified can a clear method be established as how to integrate that requirement with previous ones. Medical device software organisations could benefit from understanding the impact of fuzzy requirements as it could result in reduced rework at a later stage in the project.

Keywords

Requirements Engineering, Medical, Fuzzy Requirements, FDA

1   Introduction

Every software development project consist of a Requirements phase. It is at this phase it is established what is to be development. Experience suggests that requirements are the biggest software engineering problem for the developers of large, complex systems. Many decades after the invention of computer programming, software practitioners still have raging debates about exactly what a "requirement" actually consists [1]. A software requirement can be defined as: *"a software capability needed by the user to solve a problem or to achieve an objective"; "a software capability that must be met or possessed by a system or a system component to satisfy a contract, standard, specification, or other formally imposed documentation"* [2]

   It is generally agreed that the goal of the requirements phase is to establish what the software must do without describing how to do it. Most authors agree in principle that requirements should specify "what" rather than "how". In other words, the goal of requirements is to understand and specify the problem to be solved rather than the solution. The most basic reason for this is that a specification in terms of the problem, captures the actual requirements and does not over constrain the subsequent design or implementation. Also, solutions are typically more complex, more difficult to change, and harder to understand than a specification of the problem [3].

   Obtaining good software requirements is a crucial step towards building reliable and usable software systems. Studies show that one of the main reasons for software project failures is due to poor requirements [4]. It is extremely desirable to detect errors in the requirements before the design and development of the software begins. Due to the nature of the requirements specification phase, there is a lot of room for misunderstanding and committing errors, and it is quite possible that the requirements specification does not accurately represent the client's needs [5].

2   Medical Device Software Development

Medical device software is typically developed in accordance with the V-Model [6]. When developing software in accordance with the V-Model each stage of development is completed sequentially. Unlike other plan driven software development life cycles such as the Waterfall model, testing is planned in conjunction with each stage of development. The V-Model is typically followed as it produces the necessary deliverables required when seeking regulatory approval. However, there is a shift towards more agile development techniques in the medical device software development industry [7-9]. Agile methods appear to solve an often faced problem when following a plan driven SDLC, i.e. accommodating changing requirements once the requirements phase has been completed. However, this flexibility can create problems in itself. When following a plan driven approach the requirements are heavily refined before development begins, this includes resolving issues where requirements are unclear. When following agile methods, requirements are subject to change at any point in a software development project, therefore the process of understanding fuzzy requirements is need throughout a software development project. Medical device software development organisations who wish to market their device for use must conform to the regulations within that region. For example, medical devices marketed for use must beer the CE mark, showing conformance, and those marketed for use within the United States (US) must provide evidence of conformance to the Food and Drug Administration (FDA)

3   FDA & IEC 62304 stance on Requirements

The FDA regulations impose stringent requirements on the process by which software systems used in medical devices are developed. These requirements translate into various software artefacts that must be made available for the software to be FDA compliant [10] and, for medical device software, the FDA is responsible for assuring that the device utilizing the software is safe and effective [1].

   FDA requires medical device manufacturers to submit their device requirements before beginning development. System and software requirements are taken from the FDA medical device Quality System Regulation [11]. FDA regulations cover all aspects of the medical device product lifecycle, and

the FDA requires medical device manufacturers to submit evidence of product safety and efficacy for FDA review and clearance before the manufacturer can market, sell, or distribute the product [1]. Thus, it is critical to obtain information from the FDA on the requirements applicable to the proposed device [5].

Validation compares the final product to the original specifications [3], and is closely related to the requirements specification. You can validate the user's requirements; this is where ambiguity reigns most of the time and where formal methods, through the use of specification languages, have the biggest strides. There is still a wide gap between what the user wants and what the developer understands that the user wants. Very often this is where one of the causes of initial system failures can be found [12]. Software validation is the confirmation that all software requirements have been met and that all software requirements are traceable to the system requirements, provided that it is not possible to validate software without predetermined and documented software requirements [13]. There are two major types of validation that come into play with medical devices - design validation and process validation. Design validation means establishing, by objective evidence, that device specifications conform to the user's needs and the device's intended uses. Process validation, on the other hand, means establishing, by objective evidence, that a process consistently produces the desired result or a product meeting the predetermined specifications [14]. The FDA requires medical device manufacturers to submit their device specifications before beginning development. Thus, validation could come at early stages of development if the user's requirements could be precisely defined, and which from them the rest of the development derived [15]. Ideally, validation work would be accomplished while the requirements are being written [12]. Any safety and regulatory requirements for medical devices necessarily call for rigorous software development methods to ensure reliability and to protect public health. In addition to that, requirements and specifications based on medical practice are needed to help ensure that devices will perform appropriately [16].

The regulatory bodies request that medical device software development organizations clearly demonstrate how they follow a software development life cycle without mandating a particular life cycle. In order to comply with the regulatory requirements of the medical device industry, it is necessary to have clear linkages to traceability from requirements through the different stages of the software development and maintenance life cycles. Traceability is central to medical device software development and essential for regulatory approval. Software traceability refers to the ability to describe and follow the life of a requirement in both forward and backward direction [17]. FDA for instance states that traceability analysis must be used to verify that a software design implements all of its specified requirements [18]. Thus, traceability is particularly important for medical device companies, as they have to demonstrate this in order to achieve FDA compliance [19].

IEC 62304:2006 [20] is harmonized with the European Medical Device Directive (MDD) [16] and is approved for use by the FDA. As with guidance documents, adherence to IEC 62304:2006 is not mandatory, however, if a manufacturer chooses not to follow it, they would need to provide a sufficient explanation behind not following it. IEC 62304:2006 does not address software development lifecycle models; instead, it defines processes, which consist of activities that should be conducted in each medical device software development project [21]. As with the QSR, initial reading of IEC 62304:2006 would appear to suggest it should be followed in accordance with a sequential lifecycle model such as Waterfall Model. The publishers of IEC 62304:2006 observed that the standard appeared to mandate following the Waterfall Model and added the following to remove any ambiguity;

> *"it is easiest to describe the processes in this standard in a sequence, implying a "waterfall"*
> *or "once through" life cycle model. However, other life cycles can also be used"*

4   Fuzzy Requirements

Requirements are sometimes not specified and documented in detail in many software development projects, which makes software validation and maintenance very difficult. One challenge is that many product requirements are fuzzy in nature. Actually, customers usually describe their requirements in fuzzy terms such as *good*, *high*, *very important*, *etc*. Translating such fuzzy terms into design specifications that will accurately create the desired product is difficult [12].

There are two important goals in requirements engineering: (a) acquiring requirements that are satisfactory to their customers; and (b) generating feasible requirements. These two goals often

compete with each other. To achieve both goals, the requirements often need to be refined many times [12].

4.1 Fuzzy Requirements & Fuzzy Sets

In the medical device software domain, fuzzy requirements may emerge. An example of such requirements is:

R: the software system should fully support the clinician

The constraint imposed by the fuzzy requirement R can be represented as a satisfaction (membership) function, denoted as Sat_R, which maps an element of $R's$ domain D to a number in the range $[0,1]$, which represents how well the requirement is satisfied [7]:

$$Sat_R : D \rightarrow [0,1] \qquad (1)$$

Let us assume that the type of medical device software used is a medical imaging system. The elasticity of R can be captured using the satisfaction function, and corresponds to the membership function of the fuzzy set *FULLY* in the requirement R [7].

Examples of the characteristics which should be available in order for the software system to be considered as a support for the clinician are as follows:

C_1: load medical image
C_2: view medical image
C_3: segment medical image
C_4: save medical image
…
C_n

A membership function of the fuzzy set *FULLY* that can be used, is Zadeh's *S-function*, defined as follows:

$$\mu_F(x) = \begin{cases} 0, x \leq a; \\ 2\left(\dfrac{x-a}{c-a}\right)^2, a \leq x \leq b; \\ 1-2\left(\dfrac{x-c}{c-a}\right)^2, b \leq x \leq c; \\ 1, x \geq c \end{cases}$$

(2)

Where $\mu_F(x)$ is the degree of membership of the requirement x (represented in terms of the numbers of characteristics achieved, provided that the weights of importance of the characteristics is assumed to be the same) in the fuzzy set *FULLY*, where the value evaluates to the range $[0,1]$, such that, if no characteristics are available, $\mu_F(x) = 0$. a is the *minimum* number of characteristics, c is the maximum number of characteristics, and b is any value between a and c. The S-function can be plotted as shown in figure.1.
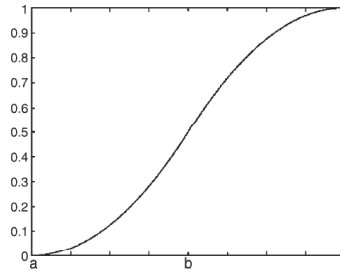
Figure 1. Membership Function of the Fuzzy Set F (support)

5 Relationship Classification

There are four types of significant relationships between requirements: (a) conflicting; (b) cooperative; (c) mutually exclusive; and (d) irrelevant. The classification is determined by how satisfying one requirement impacts the satisfaction degree of another requirement [12].

Two requirements are *conflicting* if raising satisfaction in one requirement *often* decreases the other's level of satisfaction. If it *always* decreases the satisfaction degree of the other, they are said to be *completely conflicting*. Figure 2 shows an example of completely and partially conflicting requirements [22].
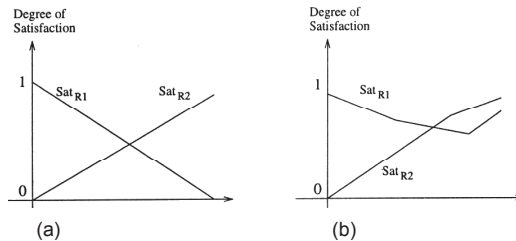
Figure 2. (a) completely conflicting requirements; (b) Partially conflicting requirements

Fuzzy conflicting relationships can relax the conditions of the crisp conflicting relationships using fuzzy terms such as *strong*, *medium*, *weak*, *etc*. Thus, one can define terms such as *strong conflict*, *medium conflict*, and *weak conflict* using satisfaction functions. Figure 3 shows an example of fuzzy conflicting relationships [22], where it can be noticed that when two requirements have the conflicting degree 0.5, we are very sure that they are weak conflicting, since their satisfaction degree in the membership function *Weak Conflict* is 1.0, and are not strong conflicting since their degree of satisfaction in membership function *Strong Conflict* is 0. These two requirements are somewhat medium conflicting since their degree of satisfaction in membership function *Medium Conflict* is 0.6.
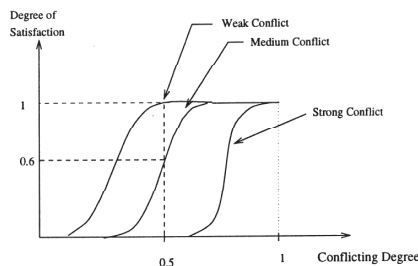
Figure 3 Fuzzy conflicting requirements

Two requirements are *cooperative* if increasing the satisfaction in one often increases the degree in the other. If the rise in satisfaction of one always increases satisfaction in the other, they are *completely cooperative*. Figure 4 shows an example of completely and partially cooperative requirements [22]. Fuzzy cooperative relationships can relax the conditions of the crisp conflicting relationships using fuzzy terms such as *strong*, *medium*, *weak*, *etc*. Thus, one can define terms such as *strong cooperative*, *medium cooperative*, and *weak cooperative* using satisfaction functions.

(a)   (b)
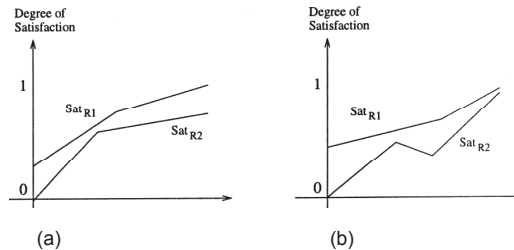
Figure 4 (a) completely cooperative requirements; (b) Partially cooperative requirements

Sometimes, two requirements cannot be satisfied at the same time, such that, if one fuzzy requirement is satisfied, the other is not satisfied at all. Those requirements are referred to as *mutually exclusive requirements [12]*.

6 Implicit Relationships Detection

In large scale software systems for instance, many conflicts are implicit, and thus, difficult to identify. Therefore, it helps to have techniques that can aid in identifying implicit conflicting and cooperative relationships between requirements. In this case, several heuristics can be used to infer relationships between requirements based on the identified relationships [22, 23]:

> **Heuristic rule 1 (infer relationships from cooperative requirements):** *Let D be a domain shared between three requirements R1, R2, and R3. If requirement R1 is completely cooperative with R2 in D, R2 is completely cooperative with R3 in D, and they are not irrelevant, then R1 is completely cooperative with R3 in D.*

> **Heuristic rule 2 (infer relationships from conflicting and cooperative requirements):** *Let D be a domain shared between three requirements R1, R2, and R3. If requirement R1 is completely cooperative with R2 in D, R2 completely conflicts with R3 in D, and they are not irrelevant, then R1 is completely conflicting with R3 in D.*

> **Heuristic rule 3 (infer relationships from conflicting requirements):** *Let D be a domain shared between three requirements R1, R2, and R3. If requirement R1 completely conflicts with R2 in D, R2 completely conflicts with R3 in D, and they are not irrelevant, then R1 is completely cooperative with R3 in D.*

It can be noticed from heuristic rule 1 that a completely cooperative relationship in a domain is transitive. Whilst heuristic rule 3 indicates that a completely conflicting relationship in a domain is *not* transitive.

7   Results & Discussion

Suppose that we are planning to develop some medical device software, MEDSYS. Such software to be used in or as a medical device is subject to user requirements. However, unlike unregulated software, medical device software must meet both the user's requirements and the requirements of the regulatory body (*i.e.* FDA) of the region into which the software will be marketed [10]. Thus, we are expected to comply with both user requirements and regulatory requirements.

Examples of user requirements for MEDSYS are:

R_1: The medical device software shall *fully* support the clinician
R_2: The medical device software shall be developed in *short* time

Examples of IEC 62304:2006 requirements for MEDSYS are:

R_3: The manufacturer shall retain *sufficient* records to permit the test to be repeated
R_4: The manufacturer shall establish procedures to ensure that the released software product can be *reliably* delivered to the point of use without corruption or unauthorized change
R_5: The manufacturer shall consider potential causes including, as appropriate, *reasonably* foreseeable misuse

In the above requirements, the fuzzy terms have been written in italics. Such fuzzy terms can be characterized by fuzzy sets, and thus, represented by a membership function.

Figure 5 shows the relationships between the requirements as given by a requirements analyst and a customer, where "-" denotes a conflictive relationship, and "+" denotes a cooperative relationship. Here, we assume that the conflictive and cooperative relationships are *complete* (Figure 2(a) and Figure 4(a)).

	R_1	R_2	R_3	R_4	R_5
R_1			+	+	+
R_2			-		
R_3					
R_4					
R_5					

Figure 5 Initial relationships as specified by a requirements analyst and a customer

Using the heuristics in section 6, more relationships (shown with a green background box) could be infered as shown in figure 6.

	R_1	R_2	R_3	R_4	R_5
R_1		-	+	+	+
R_2			-		
R_3		-		+	+
R_4			+		+
R_5		-	+	+	

Figure 6 Inferring relationships between requirements

From Figure 6, we can notice some interesting relationships being inferred. Since most of the time we may be interested in trying to manage between the user requirements and the requirements of the IEC 62304:2006 standard (regulatory requirements), it is thus necessary to find out where such requirements would not meet (*i.e.* conflict). For instance, it can be noticed that the user requirement R_2 and the IEC 62304:2006 requirement R_5 cannot be achieved at the same time, since they completely conflict with each other.

8 Conclusions

Vague or unclear software requirements also known as "Fuzzy Requirements" can have a detrimental effect on a software development project. Often what is finally delivered to the customer is not what they asked for, rather what the software development organization perceived them to need. This problem can be exacerbated in the medical device software development industry where there are two customers, the end user and the regulatory bodies. Regulatory bodies impose strict controls to ensure the safe and reliable performance of medical devices. However, these regulations and associated development standards introduce requirements which can be deemed as fuzzy. By fully understanding fuzzy and categorizing them they can be accommodated better in a software development project and therefore the potential for a project being deemed a failure can be reduced.

References

[1]   R. C. Fries, *Reliable Design of Medical Devices* vol. 3rd ed. Boca Raton FL: CRC, 2012.
[2]   T. H. Faris, *Safe And Sound Software: Creating an Efficient And Effective Quality System for Software Medical Device Organizations*: Asq Press, 2006.
[3]   M.S. Sivakumar, V. Casey, F. McCaffery, and G. Coleman, "Verification and Validation in Medi SPICE," presented at the The 11th International SPICE Conference, Dublin, 2011.
[4]   F. Pavese and A. B. Forbes, *Data Modeling for Metrology and Testing in Measurement Science*: Birkhäuser, 2009.
[5]   J. K. Shapiro, "The Pathway to Market for your Medical Device: A Primer on Obtaining Information from FDA " *FDLI,* vol. Update May/June, 2008.
[6]   F. McCaffery, D. McFall, P. Donnelly, F. G. Wilkie, and R.Sterritt, "A Software Process Improvement Lifecycle Framework for the Medical Device Industry," presented at the Proceedings of the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS'05) 2005.
[7]   R. Rasmussen, T. Hughes, J. R. Jenks, and J. Skach, "Adopting Agile in an FDA Regulated Environment," presented at the Agile Conference, 2009. AGILE '09. , Chicago, IL 2009.
[8]   P. A. Rottier and V. Rodrigues, "Agile Development in a Medical Device Company," presented at the Proceedings of the 11th AGILE Conference. AGILE '08., Toronto, 2008.
[9]   J. Schmidt and K. Weyrauch, "Getting'Agile'with Medical Device Development," *Biomedical Instrumentation & Technology,* vol. 47, pp. 221-223, 2013.
[10]   H. Mehrfard, H. Pirzadeh, and A. Hamou-Lhadj, "Investigating the Capability of Agile Processes to Support Life-Science Regulations: The Case of XP and FDA Regulations with a Focus on Human Factor Requirements," presented at the In Proceedings of SERA (selected papers), 2010.
[11]   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," ed: *U.S. Department of Health and Human Services*, 2007.
[12]   J. O. Grady, *System Requirements Analysis* Amsterdam: Elsevier Academic, 2006.
[13]   D. Farb and B. Gordon, *Pharmaceutical Computer Validation Introduction Guidebook*: UniversityOfHealthCare, 2005.
[14]   C. T. DeMarco, *Medical Device Design and Regulation*: Asq Press, 2011.
[15]   A. Dasso and A.Funes,*Verification,Validation and Testing in Software Engineering*:Idea Group Pub., 2007.
[16]   *Directive 2007/47/EC of the European Parliament and of the Council of 5 September 2007*, 2007.
[17]   V. Casey and F. McCaffery, "Med-Trace: Traceability Assessment Method for Medical Device Software Development.," presented at the European Systems & Software Process Improvement and Innovation Conference, (EuroSPI). Roskilde, Denmark, 2011.
[18]   J. Cleland-Huang, A. Czauderna, M. Gibiec, and J. Emenecker, "A machine learning approach for tracing regulatory codes to product specific requirements," presented at the Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 1, Cape Town, South Africa, 2010.
[19]   F. McCaffery and G. Coleman, "The need for a software process improvement model for the Medical Device Industry," *International Review on Computers and Software,* vol. 2, pp. 10-15, 2007.
[20]   AAMI, "ANSI/AAMI/IEC 62304, Medical device Software - Software life cycle processes," ed. Association for the Advancement of Medical Instrumentation, 2006.
[21]   M. McHugh, F. McCaffery, and V. Casey, "Standalone Software as an Active Medical Device " presented at the The 11th International SPICE Conference Process Improvement and Capability dEtermination, Dublin, 2011.
[22]   C. Temponi, J. Yen, and W. Amos Tiao, "House of quality: A fuzzy logic-based requirements analysis," *European Journal of Operational Research,* vol. 117, pp. 340-354, 1999.
[23]   X. F. Liu, "Fuzzy Requirements," *IEEE Potentials,* pp. 24-26, 1998.

Characteristics of a medical device software development framework

Paul Clarke[1], Marion Lepmets[1], Fergal McCaffery[1], Anita Finnegan[1], Alec Dorling[1], Sherman Eagles[2]

[1] Regulated Software Research Centre, Dundalk Institute of Technology, Dundalk, Ireland
[2] SoftwareCPR, USA
{paul.clarke; marion.lepmets; fergal.mccaffery; anita.finnegan; alec.dorling} @dkit.ie
and seagles@softwarecpr.com

Abstract

This paper aims to describe the software development settings of medical device domain focusing on the demands of the safety critical software processes. Medical device software developers have to adhere to a number of regulations and standards. This paper addresses the most important characteristics of a software development framework that could support medical device software developers in their efforts to comply with these regulations as well as to improve their software development processes.

Keywords

Process model, medical device software development, safety critical software process

1 Introduction

Software engineering has been around as an academic field and as an industry domain long enough to not be called novel anymore. It has been taught in universities for several decades. As with any domain that has moved from innovation to commonplace, the research questions have changed along with it. Software has become central to how we live and for many other domains whose knowledge is built on top of software engineering applications. The critical questions for software engineering have now shifted from the fundamental issues of how to develop software into an easier adoption and automation, adjustment and tailoring of these software development tasks. Software development in safety critical domains is one of these critical questions as an increasing amount of software is being embedded to medical devices, cars and airplanes.

A stable body of knowledge for software engineering exists in the world today which describes how to design and develop software. Two main philosophies of software development have emerged: *prescriptive development* and *agile development* [1]. Prescriptive development along with prescriptive process models are often associated with the development of detailed process definitions, followed by the application of process activities and tasks in accordance with the process definition. The intention of prescriptive process models is to improve the product quality by reducing the number of errors that are made and by supporting the achievement of delivery dates, budget constraints. In the case of medical device software development, the primary goal is to create safe and effective medical devices. Agile development, on the other hand, seeks to reduce the levels of bureaucracy associated with prescriptive process models and promote project agility. Increased agility allows development to respond rapidly to changing requirements and relies more on human-centric skills; thus empowering individuals to make decisions that best support emerging demands as opposed to strictly following an extensively defined and heavily prescribed process. As to which of these two philosophies is best (prescriptive or agile), Pressman points out that although an emotional debate has raged, it is really a question of trade-offs [1]. Or put another way, the selection of a software process philosophy (and indeed a software process itself) must first consider the benefits bestowed by different approaches and thereafter, identify the process characteristics that best service the demands of the given software development domain/environment [2].

This paper aims to describe how the generic software development approaches fit to the software development in highly regulated medical device domain. The next section describes prescriptive process models and their importance in medical device domain. The focus then shifts to the medical device software development - the medical device regulations and standards, and the current medical device software development practices. The authors then describe the demands of and appropriate approaches to medical device software development that lead to the characteristics that a medical device software development process model should have. Finally, the paper presents some concluding remarks and the future works.

2 Overview of Prescriptive Process Models

Quality Management System (QMS) standards provide various prescriptive process models that have long been established in software development (and earlier in other industrial sectors) as an approach to process management. A QMS is essentially concerned with the design and management of a suite of processes that support the achievement of consistent levels of quality in the delivery of products and services. Typically, a QMS standard will have an associated audit system whereby the performance of the QMS can be evaluated. The outcome of a QMS audit is either pass or failure, with instances of failure having an associated list of non-conformances with the standard. Audits can be conducted either internally by organisational personnel or by an external party. When the outcome of a third party audit is successful, it is generally possible to obtain official certification of compliance with the QMS. Perhaps the most widely adopted QMS is the ISO 9001 standard [3], which is applied in software development settings via the guidance provided in ISO 90003 [4]. Although ISO 9001 can theoretically be implemented in any software development setting, it has been suggested that the benefits are greater for large rather than small organisations [5]. It has also been reported that ISO 90003 may not be sufficiently rich in software development know-how [6]. Despite the guidance for performance improvement provided in ISO 9004, the primary aim of QMS standards like the ISO 9000

series is to evaluate the organizational conformance necessary for regulation – appropriate in the medical device sector because of the need to have consistent and equal application of the pertinent legislation. Continuous improvement of processes is the cornerstone of alternative prescriptive models that are referred to in this paper as the Capability Maturity Frameworks (CMFs). The underlying notion is that when the processes with higher capability are applied in the organization they will also advance the overall organizational maturity [7].

CMFs accept that process implementations vary greatly, ranging from complete disorganisation to extensive process implementation and management. CMFs provide a roadmap for process improvement that is based on the extensive experience of a large number of previous implementations over many years. It is also possible to undertake an assessment of process implementation using the infrastructure of a CMF. However, the outcome of a CMF-related assessment is different to the outcome of a QMS audit, in that it is concerned with the identification of the process capability when measured against established best practice for a process (as opposed to the pass/failure scenario associated with QMS audits).

Two of the best known and most widely adopted software development CMFs are ISO/IEC 15504 [8] and CMMI-DEV [9]. Software development CMFs address the need for process improvement in software development settings, and are noted to provide benefits, such as a reduction in the cost of quality [10], improved customer satisfaction and project performance [11]. However, software development CMFs are not entirely without criticism. As with QMSs, it has been noted that CMFs may be more challenging to implement in smaller software development companies, which are often confronted with customer pressures and a general lack of time and resources [12].

To date, a number of different software development standards and guidance documents have been developed for use in the medical device sector. These contemporary standards are presented in the following section, along with a brief outline of the medical device regulations. Thereafter, we examine the suitability of the two different software development philosophies (prescriptive process models and agile process models) to medical device software development.

3 Background to Medical Device Software Development

The medical device industry is focusing increasingly on software quality as more and more software is integrated into medical devices. According to data from the U.S. FDA [13], "software failures were behind 24% of all the medical device recalls in 2011" resulting in "gearing up the FDA labs to spend more time analysing the quality and security of software-based medical instruments and equipment." Although the domain is heavily controlled by regulations, the regulation itself is satisfied in practice through the implementation of appropriate process and quality standards. Therefore, it is critically important that the medical device software process and quality standards, presented in Figure 1, adopt the expertise accumulated in the generic, best practice software process standards (that have proven to be the foundation of developing high quality software).

3.1 Medical Device Regulations

A medical device can consist entirely of software or have software as a component of the overall medical device system [14]. In order to be able to market a medical device within a particular region it is necessary to comply with the regulatory demands of that region (Figure 1). Two of the largest global bodies responsible for issuing and managing medical device regulation belong to the central governing functions of the US and EU.

In the case of the US, the Food and Drug Administration (FDA) issues the pertinent regulation through a series of official channels, including the Code of Federal Regulation (CFR) Title 21, Chapter I, Subchapter H, Part 820 [15]. Under US regulation, there are three medical device safety classifications: Class I, Class II and Class III. The medical device safety classification is based first and foremost on the clinical safety of the device. Class I devices are not intended to support or sustain human life, and may not present an unreasonable risk of harm. Examination gloves are an example of a Class I medical device. Class II medical devices are those devices for which Class I general controls alone cannot assure human safety and effectiveness. Class II devices could cause damage or harm to humans. An example of a Class II medical device is a powered wheelchair. Class III medical devices are usually

those that support or sustain human life, and are of significant importance in the prevention of human health impairment. An example of a Class III device is an implantable pacemaker. All implantable devices are Class III medical devices as the surgery required carries additional high risks from anaesthesia and possible infections that go beyond the technical and engineering safety risks of the correct performance of the device.
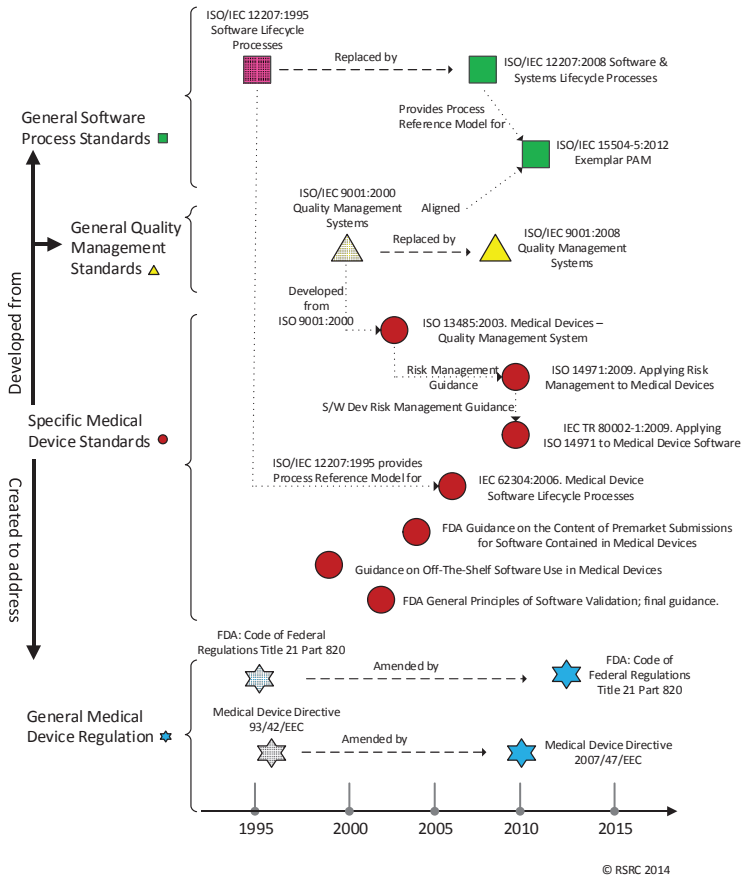
Figure 1. Medical Device Standards and Regulation

In the EU, the corresponding regulation is outlined in the general Medical Device Directive (MDD) 93/42/EEC [14], the Active Implantable Medical Device Directive (AIMDD) 90/385/EEC [16], and the *In-vitro* Diagnostic (IVD) Medical Device Directive 98/79/EC [17] - all three of which have been amended by 2007/47/EC [18]. Although slightly different to the US safety classifications that are based on clinical safety of the device, the EU classifications essentially embody similar classifications and limitations, where Class I corresponds to Class I, Class IIa and IIb to Class II, and Class III to Class III. A further safety classification applies to the software in the medical device as outlined in IEC 62304, wherein the safety classification is concerned with the worst possible consequence in the case of a software failure (as compared with general medical device safety classification which is based on the difficulty of a regulator to determine if the device will be safe). Hence, some Class II medical devices can cause serious injury or even death, but they are Class II because they are similar (in clinical use and safety) to well understood devices that have been used before. Since IEC 62304 safety classifica-

tions are based on worse case failure of the software, it is possible that Class II medical devices can have Class III software.

In the medical device domain, ISO 13485:2003 (ISO 13485 from hereon) [19] outlines the requirements for regulatory purposes from a QMS perspective. ISO 13485, which is based on ISO 9001 [3], can be used to assess an organisation's ability to meet both customer and regulatory requirements. However, ISO 13485 does not offer specific guidance on software development.

IEC 62304:2006 (IEC 62304 from hereon) [20], which can be used in conjunction with ISO 13485, does offer a framework for the lifecycle processes necessary for the safe design and maintenance of medical device software. As a basic foundation, IEC 62304 assumes that medical device software is developed and maintained within a QMS such as ISO 13485, but does not require an organisation to be certified in ISO 13485. Therefore, IEC 62304 can be considered to be a software development specific supplement to ISO 13485. IEC 62304 is based on ISO/IEC 12207:1995 [21] which although a comprehensive standard for software development lifecycle processes has effectively been decommissioned following the publication of the more extensive ISO/IEC 12207:2008 [22]. Furthermore, other developments in the ISO and IEC communities for software development, such as ISO/IEC 15504, have provided significant additional levels of software process detail to support ISO/IEC 12207:2008. IEC 62304 is currently being revised to better align with ISO/IEC 12207:2008 (Figure 1). IEC 62304 is a critical standard for medical device software developers as it is the only standard that provides recommendations for medical device software implementations based on the worse consequences in the case where the software failures lead to hazards.

Furthermore, for general medical device risk management, IEC 62304 is used in conjunction with ISO 14971 [23], with IEC 80002-1 [24] providing guidance on the application of ISO 14971 for software development. Additionally, as IEC 62304 considers a medical device system to consist of software as part of an overall medical device system, the system level requirements are not included within IEC 62304 but instead within the medical device product standard IEC 60601-1 [25]. Also it should be noted that due to the increasing importance of usability within the medical device industry organisations should also adhere to the medical device usability requirements outlined in IEC 62366 [26].

3.2 Alignment between general software development and medical device standards

One of the more obvious examples of the gap that has emerged between the general software development standards and IEC 62304 (incl. ISO 14971 and IEC 80002-1) is the inconsistency in the use of language and terminology. For example, the Risk Management process that is present in ISO/IEC 12207:2008 (as opposed to ISO/IEC 12207:1995 upon which IEC 62304 is based) is concerned with project-level risks. In effect it is a project level process aimed at identifying and controlling general project risks of budget and schedule. However, the Risk Management process in IEC 62304 is concerned largely with product safety issues (i.e. addressing only negative outcomes) and how these might be reduced through robust process implementation throughout the entire software development lifecycle. Given that the medical device sector, and its many related standards, tends to term safety engineering as risk management, it is appropriate that IEC 62304 should adopt this language. In contrast, process safety issues are dealt with separately in ISO/IEC 15504 (in the Part 10 extension for Safety Critical software development [27]). This has resulted in different process related concepts for medical device software development as compared with generic software development – the Risk Management process. Many additional gaps also exist, and these extend beyond language and terminology, permeating the very architecture and design of the standards themselves. The major difference between these two standards is based both on their different design and purpose.

Figure 1 above presents numerous different medical device standards and regulations that exist today, some of which are interlinked and others which are inconsistent. Although the border between these two domains is potentially difficult to navigate (medical device development is focused on product safety management while general software development has a broader software development mandate), there are some shortcomings in the presently available approaches.

The dominant medical device software standards such as IEC 62304 are not yet aligned with the approach adopted in the general software development standards community since the 1995 publication of ISO/IEC 12207. One significant change in this respect has been the introduction of a harmonised approach to process description (as defined in ISO/IEC 24774 [28]) which involves the identification of core process outcomes that can later be harnessed to develop a process assessment method. A fur-

ther significant change relates to the movement in the general software development standards community (and in other safety-related domains) to include a software process improvement dimension that can be instrumental in guiding software development organisations towards the required process targets. In effect, the medical device standards have not kept up with the changes that have been made to the general software development standards. There are several reasons why the medical device standards lag the updates to the general software development standards, (perhaps) most importantly the IEC stability period during which adopted harmonised standards are not to be changed unless the proposed changes are necessary in terms of safety. With the increasing use of software in medical devices, there is a case to be made for introducing the accumulated up-to-date wisdom in the general software development standards into the medical device software development specific standards in a uniform fashion – and work in this direction should not wait for the IEC stability period to come to an end, but rather proceed in the interim period (such as the work reported upon in this paper).

In order to identify an appropriate architecture for introducing the significant body of general software process knowledge into the medical device process domain, an initial important step involves an evaluation of the general software development methodology to fit with the specific demands of medical device software development. The next section outlines the results of an evaluation of the general software development methodology with the specific demands of medical device software development.

4 Characteristics of a medical device software development framework

The primary observation in relation to the identification of the demands of medical device software development is that a large degree of variation is evident from a development process perspective. Although agile software development approaches are increasingly adopted in industry, the medical device software development still needs to comply with several standards indicating the need for a more disciplined software development approach. This variation in the demands presents a significant challenge to any general framework for medical device software development. Any such framework should be capable of supporting both an agile software development philosophy while also addressing the very high levels of process rigour associated with the more disciplined software development process philosophies [29]. Since these opposing software development philosophies are essentially discordant, it is not surprising that a general framework supporting both philosophies does not presently exist. In developing the medical device software framework, it is useful to first identify the characteristics that such a framework should ideally exhibit:

Characteristic 1: The framework should support the development of software for all medical device safety classifications.

Characteristic 2: The framework should offer greater levels of detail on the implementation of software development processes than presently exists in the dominant medical device software development standards.

Characteristic 3: The framework should identify a roadmap that companies can follow in order to implement the process improvements required in order to progress towards both regulatory compliance and best practice.

The three characteristics highlighted above can be summarised as follows: a general framework for medical device software development should be capable of supporting a spectrum of process implementation, it should offer a high level of detail regarding software process implementation, and it should facilitate software process improvement. A purely agile software development methodology would be unsuited to these characteristics since without adaptation; it can lack a significant investment in up-front requirements elicitation and formal documentation (such as is required for Class C medical device software). Equally, a wholly prescriptive process approach would also be unsuitable for the identified characteristics, since the levels of bureaucracy associated with such approaches would not be ideally aligned with the needs of Class A medical device software development. Therefore, an approach should be designed that supports agility for some types of development, while also providing a prescriptive process for other types of development. Considering that a spectrum of process implementations is required in order to satisfy the characteristics identified above, it is proposed herein that of the contemporary general software development approaches, a CMF offers the most natural fit.

However, some caution should be applied to this judgment as CMFs are more challenging to implement in smaller companies [12, 30-34]. Plus, rapid product innovation may be an important survival characteristic for Class A medical device software developers. Therefore, we must examine if a CMF exists that supports the needs of medical device software development – and if such a CMF does exist, can it be adapted for use in organisations that are engaged in the development of highly innovative, though regulated, Class A medical device software. One possibility would be to adapt the ISO/IEC 15504 CMF (and the underlying ISO/IEC 12207:2008 process activities and tasks) to harmonise with the explicit requirements of IEC 62304 – an approach which (owing to the large amount of descriptive text required) the authors intend to address in a separate publication. A further adaptation of this framework could include the regulatory requirements from FDA and QMS that the medical device products need to be compliant with prior to be placed on the market.

5 Conclusions and Future Works

Medical device domain is heavily controlled by regulations. Regulations are satisfied in practice through the implementation of appropriate process and quality standards. It is therefore critically important that the medical device software process and quality standards adopt the expertise accumulated in the generic, best practice software process standards (that have proven to be the foundation of developing high quality software). As outlined in this paper, this is not presently the case in the medical device software development sector – with IEC 62304 being based on the now withdrawn ISO/IEC 12207:1995. The result is that medical device software development standards are no longer up to date with the acknowledged best practice or process definitions set forth in the international standardisation community (particularly with respect to ISO/IEC 27447, ISO/IEC 12207:2008 and ISO/IEC 15504).

The goal of the work presented in this paper was to describe the landscape of software development in medical device domain focusing on the demands of safety critical software processes. There is a myriad of regulations and standards that need to be applied in medical device software development and the authors presented the most important characteristics of a capability maturity framework that could support medical device software developers in their efforts to adhere to these regulations as well as improving their software development processes.

The work is currently underway integrating the generic software development process requirements with the specific medical device regulatory requirements and guidelines into a comprehensive medical device software development capability maturity framework.

6 Literature

1. Pressman, R., *Software Engineering - A Practitioner's Approach*. 2005, Boston, MA: McGraw-Hill.
2. Dyba, T., *Contextualizing Empirical Evidence.* IEEE Software, 2013. **30**(1): p. 81-83.
3. ISO, *ISO 9001:2000 - Quality Management Systems - Requirements*. 2000: Geneva, Switzerland.
4. ISO, *ISO 90003:2004 - Software Engineering - Guidelines for the Application of ISO 9001:2000 to Computer Software*. 2004: Geneva, Switzerland.
5. Stelzer, D., W. Mellis, and G. Herzwurm, *A critical look at ISO 9000 for software quality management.* Software Quality Journal, 1997. **6**(2): p. 65-79.
6. Yang, Y.H., *Software Quality Management and ISO-9000 implementation.* Industrial Management & Data Systems, 2001. **101**(7): p. 329-338.
7. Humphrey, W.S., *A Discipline for Software Engineering*. 1994: Addison-Wesley.
8. ISO/IEC, *ISO/IEC 15504: Information Technology - Process Assessment*. 2005: Geneva, Switzerland.
9. SEI, *CMMI® for Development, Version 1.3*. 2010, Software Engineering Institute.

10. Gibson, D., D. Goldenson, and K. Kost, *Performance results of CMMI-based process improvement*. 2006, Software Engineering Institute, Carnegie Mellon University: Pittsburg, PA, USA.

11. Wegelius, H. and M. Johansson. *Practical experiences on using SPICE for SPI in an insurance company*. in *EuroSPI 2007 Industrial Proceedings*. 2007.

12. Laporte, C.Y., et al., *Initiating software process improvement in small enterprises: Experiments with micro-evaluation framework*. Proceedings of the International Conference on Software Development, 2005: p. 153-163.

13. FDA. *FDA News on Software Failures Responsible for 24% of all Medical Device Recalls*. 2012 [cited 2013 12.04]; Available from: http://www.fdanews.com/newsletter/article?articleId=147391&issueId=15890.

14. European Commission, *Directive 93/42/EEC of the European Parliament and of the Council concerning medical devices*, in *OJ o L 247 of 2007-09-21*. 1993: European Commission, Brussels, Belgium.

15. FDA. *Chapter I - Food and drug administration, department of health and human services subchapter H - Medical devices, Part 820 - Quality system regulation*. [cited 2013 15.05]; Available from: http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfcfr/CFRSearch.cfm?CFRPart=820.

16. European Commission, *Council directive 90/385/EEC on active implantable medical devices (AIMDD)*. 1990: Brussels, Belgium.

17. European Commission, *Directive 98/79/EC of the european parliament and of the council of 27 october 1998 on in vitro diagnostic medical devices*. 1998: Brussels, Belgium.

18. European Commission, *Directive 2007/47/EC of the European Parliament and of the Council concerning medical devices*, in *OJ no L 247 of 2007-09-21*. 2007, EC: Brussels, Belgium.

19. ISO, *ISO 13485: Medical Devices - Quality Management Systems - Requirements for Regulatory Purposes*. 2003, ISO: Geneva, Switzerland.

20. IEC, *IEC 62304: Medical Device Software - Software Life-Cycle Processes*. 2006, IEC: Geneva, Switzerland.

21. ISO/IEC, *ISO/IEC 12207:1995 - Information Technology - Software Life-Cycle Processes*. 1995, ISO/IEC: Geneva, Switzerland.

22. ISO/IEC, *ISO/IEC 12207:2008 - Systems and Software Engineering - Software life cycle processes*. 2008, ISO/IEC: Geneva, Switzerland.

23. ISO, *ISO 14971 - Medical Devices - Application of Risk Management to Medical Devices* 2009, ISO: Geneva, Switzerland.

24. IEC, *IEC TR 80002-1 - Medical Device Software - Part 1: Guidance on the Application of ISO 14971 to Medical Device Software*. 2009, IEC: Geneva, Switzerland.

25. IEC, *IEC 60601-1 - Medical electrical equipment – Part 1: General requirements for basic safety and essential performance* 2005, IEC: Geneva, Switzerland.

26. IEC, *IEC 62366 - Medical devices - Application of usability engineering to medical devices*. 2007, IEC: Geneva, Switzerland.

27. ISO/IEC, *ISO/IEC 15504-10 - Information Technology - Process Assessment - Part 10: Safety Extension*. 2011, ISO/IEC: Geneva, Switzerland.

28. ISO/IEC, *ISO/IEC 24774 - Systems and Software Engineering - Life Cycle Management - Guidelines for Process Description*. 2010: Geneva, Switzerland.

29. M. Mc Hugh, et al., *Balancing Agility and Discipline in a Medical Device Software Organisation*, in *13th International SPICE Conference on Process Improvement and Capability dEtermination*. 2013, Springer: Bremen, Germany.

30. El Emam, K. and A. Birk, *Validating the ISO/IEC 15504 Measure of Software Requirements Analysis Process Capability*. IEEE Transactions on Software Engineering, 2000. **26**(6): p. 541-566.

31. Jung, H.-W., et al., *Findings from Phase 2 of the SPICE trials*. Software Process: Improvement and Practice, 2002. **Vol.6**(no 4): p. 205-242.

32. Staples, M., et al., *An exploratory study on why organizations do not adopt CMMI* Journal of Systems and Software, 2007. **80**(6): p. 883-895.

33. Saastamoinen, I. and M. Tukiainen. *Software process improvement in small and medium sized software enterprises in Eastern Finland: A state-of-the-practice study*. in *Proceedings of the 11th European Conference on Software Process Improvement*. 2004.

34. Khurshid, N., P. Bannerman, and M. Staples. *Overcoming the first hurdle: Why organizations do not adopt CMMI*. in *Proceedings of the International Conference on Software Process*.

Author CVs

Paul Clarke

Dr. Paul Clarke is a Research Manager in the Regulated Software Research Centre based at DkIT and a Research Fellow with Lero – the Irish Software Engineering Research Centre. Primary among Paul's present research interests is the establishment of a best practice framework for medical device software development.

Marion Lepmets

Dr. Marion Lepmets is a Senior Research Fellow at the Regulated Software Research Centre in Dundalk Institute of Technology and a member of the Irish Software Engineering Research Centre (Lero). Her PhD and post-doc research focused on the impact of process model implementation and of process assessment on process improvement success, respectively. Her current research is about the development of the process reference and process assessment models for medical device software development. She is also involved in the development of software engineering standards in the International Standardization Organization (ISO/IEC JTC1 SC7).

Fergal McCaffery

Dr Fergal Mc Caffery is a Science Foundation Ireland (SFI) Principal Investigator. He is a Senior Lecturer in the Department of Computing and Mathematics, Dundalk Institute of Technology (DkIT). He is Director of the Regulated Software Research Centre in DkIT and the Medical Device Software Engineering competency area in Lero. He has been awarded SFI funding through the Stokes Lectureship, Principal Investigator and CSET Programmes to research the area of medical device software. He has published over 150 peer-reviewed conference and journal papers and is on the editorial board/programme committee for a number of leading software engineering conferences and journals.

Anita Finnegan

Anita Finnegan is a Senior Software QMS Researcher in the Regulated Software Research Centre based at DkIT and a member of Lero – the Irish Software Engineering Research Centre. Anita's primary focus is establishing the relationship between QMS requirements and software development requirements. Her PhD research addresses the development of a framework for the security assurance of medical devices using cybersecurity assurance cases.

Alec Dorling

Alec Dorling is a member of the research team at the Regulated Software Research Centre in Dundalk Institute of Technology and a member of the Irish Software Engineering Research Centre (Lero). He is the ISO convener of the ISO/IEC 15504 series and ISO/IEC 330xx family of standards on Process Assessment, and the international project leader of SPICE (Software Process Improvement and Capability dEtermination). He has held key positions in software engineering at both national and European levels at IVF's Centre for Software Engineering in Sweden, IT Research and Technology Transfer Centre at the University of Boras in Sweden,

European Software Institute in Spain and the National Computing Centre in the UK. He is on the editorial board/programme committee for a number of leading software engineering conferences and journals.

Sherman Eagles

Sherman Eagles is a convener of IEC/ISO joint working group that developed IEC 62304 Medical device software life cycle processes and IEC 80002-1 Guidance on the application of ISO 14971 to software. He has 45 years of experience in the software field, including 18 years at Medtronic. He is also a Co-chair of the Association for the Advancement of Medical Instrumentation (AAMI) software working group and of the ANSI/AAMI SW87:2012 – Application of quality management system concepts to medical device data systems. He is the author of numerous journal papers on software development, software risk management, and software standards.

Standardization of Software Development Process Model in CIS region: Research Results & Resume on the Company's Level

Denis Pashchenko[1], Andrey Blinov[1]

[1] *Financial University, Moscow, Russia*
denpas@rambler.ru; aoblinov@mail.ru

Abstract

This article is dedicated to the problem of improvement of the model of processes in software production. All resumes & conclusions are based on the 3 rounds Delphi Study in CIS region, organized by the authors in period from 9-th September till 18-th of December of 2013 with 21 experts from Russian language environment in CIS region from software companies. Delphi study is focused on problems of standardization and certification of software production, organizational resistance and key features of changes implementation. Current article is an overview of remained issues on the level of the whole software company.

Keywords

Changes implementation management, software production improvement, organizational resistance in software company, Delphi study in CIS region IT branch.

1. Introduction

The complexity of standardization in the software development is a well-known problem of the IT-branch in all over the world. In CIS region part of this evolution process was missed in the end of 90-th, when new and progressive ISV and out-source companies implemented modern process models, based on CMM and RUP. Also there are a lot of IT-companies from CIS-countries, who have «self-build process models» of software production, based on habits of management despite of end-customer expectations. From the other hand in the last 10 year new coming software companies have tried to use agile and hybrid methodologies. In author's science research all experience and opinions of 21 experts from different kinds of software companies was grouped and identified:

- Effective approaches in process improvement and change management;
- Key factors of resistance and cooperation of participants of processes improvement;
- Possible scenarios of software development improvement in future.

Meanwhile, IT-companies from CIS (and first of all from Russia) are playing an important role in the world's market software development and have a rapid growing share [1]. It means that success in production and business improvement in those companies has a strong influence on local economy. IT branch is changing very rapidly, including technologies, automation tools, modern methodologies, educational standards and end customer expectations. It means that production processes should be flexible and have ability for its rapid changing [2]. Proven approaches and practices in changes implementation give additional chances for successful production and business improvement and meeting customer requirements.

2. Research method and process

Research was conducted from 9-th September to 18-th December of 2013 by the method of Delphi study in 3 rounds. 21 experts from Russian-speaking CIS-countries took part in this research. All experts are leading managers in their companies: from project managers with team 15+ people to quality services directors and the heads of software companies with hundreds software engineers.
On first round panelists were sending their opinion and answers on the list of questions, included 3 kinds of sections:

- Common questions about influence of production processes standardization and company certification on quality of software products;
- Special questions about experience and best practices on the level of production of the whole company;
- Prognoses and opinions about 10 year's perspective of software development process models and instruments in CIS countries.

On second round panelists received a modal opinion of expert's panel for each questions. If expert's answer is not equal with modal opinion, then expert may correct his answer or just give a comment.
On third round panelist gave additional information and comments, which helped to improve Delphi study results and objectivity.
Process of gathering opinions of experts should be described in more details in this article, as well as generalization of the results in the form of ranked lists and bar/pie charts.
When collected responses were analyzed in the first round, for each closed question was selected the dominant (modal) opinion (general consensus of panel). In round №2 responses of each expert were compared with the modal opinion of the panel and could be changed or commented by expert.
As a result, for every multiple-choice question was obtained a ranked list with the dominant response in the beginning. If answer of expert wasn't in the top of list then in round 2 was requested a comment from his side.
For questions with one possible embodiment of the response were created chart's, defined popularity of answers in percepts. It helps to receive a whole panel opinion and develop recommendations.
In next table is situated number of active experts on each round:

	Round 1	Round 2	Round 3
Active experts	21	16	21
Percent of active experts	100%	76%	100%

Table 1. Activity of experts in rounds of Delphi study.

In round 2 we found a traditional decrease of expert's activity.

On next diagrams is displayed different information about experts, their experience and geographical location.

Presented experience almost always is relevant in same type of IT-companies [3]. Types of IT-companies are presented in Delphi Panel in next proportion:
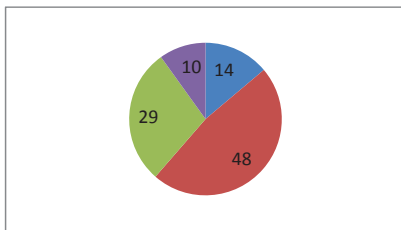
- 10% experts presented their experience from non IT-companies with in-house development;
- 14% experts presented their experience from software vendors (ISV);
- 29% experts presented their experience from system integrators;
- 48% experts presented their experience from custom development software companies (include out-source).

Figure 1. Expert's experience by the types of IT companies in research.

CIS-region geography of research is presented on next diagram:
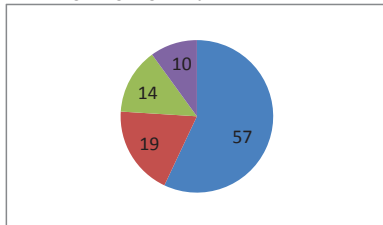
- 57% experts presented their experience from Moscow and Sankt-Petersburg (Russia);
- 19% experts presented their experience from other cities from Russia;
- 14% experts presented their experience from Ukraine;
- 10% experts presented their experience from other countries of CIS-region.

Figure 2. Expert's experience by the regions in CIS.

Experts are presenting middle age group, associated in IT branch with the apex of creativity and professional activity.
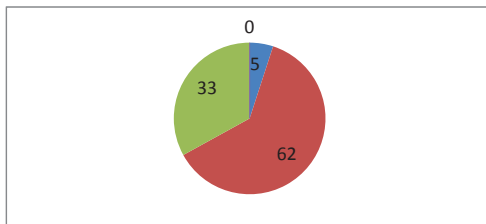
- 5% experts have age in range 20-29;
- 62% experts have age in range 30-39;
- 33% experts have age in range 40-49;
- 0% experts have age in range 50+.

Figure 3. Expert's age groups in research.

Meanwhile, the vast number of experts working in software development considerable number of years, so there are no experts in panel working in IT sector for less than 5 years.
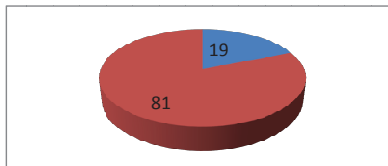
- 19% experts are working in software development from 5 to 10 years;
- 81% experts are working in software development more than 10 years.

Figure 4. Expert's professional experience in research.

3. *Results*

3.1 Section 1. Overview of standardization of software production

In this section experts answered questions, related with importance of software production process standardization and official company's certification. Also in this section are detailed perspectives of process model's improvement.

According modal opinion of panel there is a strong and visible connection between software development process standardization and final quality of software products. Main part of experts is thinking that standardization is a key factor in development high quality software. On next diagram and all others numbers means percent of experts, chosen relevant answer.
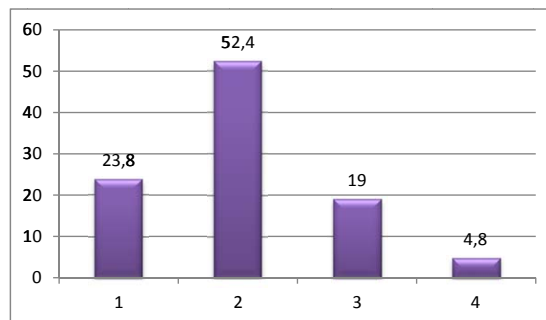
Figure 5. How often software development standardization influences on final quality of software product?
1- Always influences; 2 – Often influences; 3 – Sometimes influences; 4 – Almost do not influences.

Practically it means that one of the approaches of company's management influence on quality of software production is establishing a quality control department that would work over standardization of process model and improve quality of software products. CIS experts believe in standardization like a key method for product quality improvement with some additional remark: time schedule and even opportunity of process standardization depends on many factors like:

      a.   alification of project office and software production management;
      b.   mmon character of all produced software products.

Modal opinion of experts about obligatory certification for the company is not so clear. Around 25% of experts do not see any depends between well-known certifications or appraisal (like ISO, ICAgile or CMMI) and high quality of software product. Big share (around 43%) of experts is thinking that only sometimes official certification could confirm high quality of software. Meanwhile, some experts remarked that even preparation for certification and first audits may temporary increase software quality. But typical opinions tell another:

«… often company do not support requirements between certifications and do not manage the quality of software…» or «there are a lot of cases then CMMI Appraised software companies do not support their own standards in more than one reviewed project».

Expert's opinions about predominance of any kind of process models in software development in perspective of 10 years were different. But in CIS-countries appears a trend of popularity decreasing for classic iterative software development models (like RUF or MSF) and much more attention is paying to hybrid and agile methodologies. However, significant share of market would be placed by companies with own vision of software production models, including picking elements from iterative, flexible and hybrid models. Certainly, last variant means significant predominance own distinctive expertise of project teams and specialists, rather than simply adapting existing standards to specific software product or region.
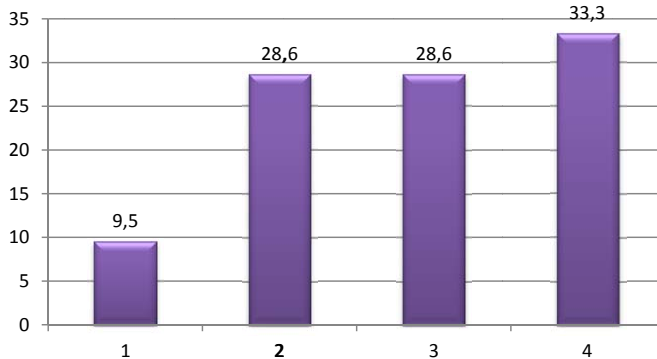
Figure 6. Which methodologies and standards from your point of view are perspectives in next 10 years for practical usage in software commercial development?
1 – Well-known iterative methods (RUP, MSF 3.0, etc); 2 – Well-known agile practices (Agile, Scrum, etc); 3 – Hybrids methods (OpenUP, MSF for Agile, etc); 4 – In-house distinctive models;

Experts didn't agree a common opinion about needs of regular reengineering of current production process model. But majority of experts found that improvement of software production should have a regular character even if it's done not in a way of drastically reengineering.
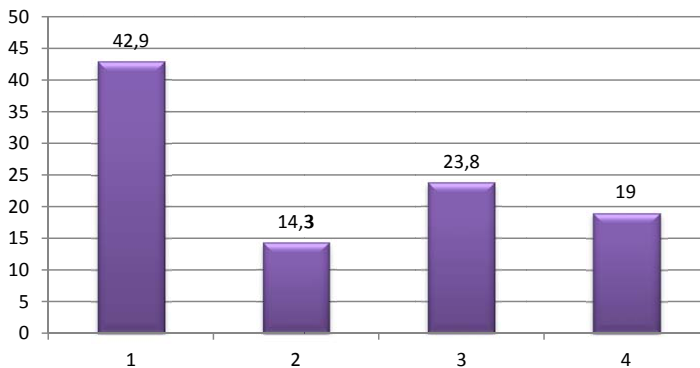
Figure 7. Is it necessary to do regular process reengineering in software production?
1 – Every 2-4 years; 2 – Every 4-7 years; 3 – Only in case of deterioration in production; 4 – No, never;

Additionally panel found, that for CIS companies there are a lot of well-known cases, when after central standardization companies got back to decentralization in quality management (for example, in program of projects or business division). After this kind of decentralization improvement of processes is executing on level of projects or directions.

It means that only dedicated organization unit like SEPG or quality management direction may focus

all efforts on continues process improvement or even just on following agreed production standards. Also such kind of unit may proactive audit the needs of partly of full reengineering of process model in software production.

3.2   Section 2. Changes in processes of software production on the level of the whole company

In this section experts shared their opinions and experience about practices of changes implementation in software development model of processes on the level of the whole company or separated division (subsidiary), focused on software development in company. Of course, this experience is related with significant changes, what touched all chains of process and all project team members. For example, this kind of change may be implementation of CMMI principals in production or new approach in usage of agile practices.

Of course, implementation of any organizational improvements is starting from the stage of planning and includes estimation of its effects [4]. Experts agreed: for internal project of software production process improvement is strong obligatory to have the full and actual set of project documentation (like project manifest, plan, risk table, etc).
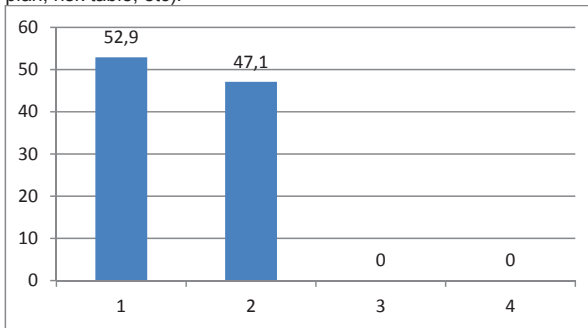
Figure 8. How important in internal process improvement in software production to have a full and actual set of project documentation (project plan, risk table, resource map, etc)?
1 – Very important; 2 –Some importance; 3 – Not important; 4 – Changes do not have a common plan and project;

Also it's important to have a formal phase of planning in this internal project:
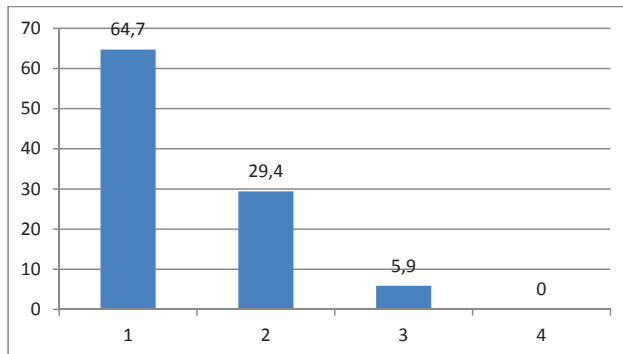
Figure 9. How important the formal phase of planning in project schedule of changes implementation in software production on the level of the whole company?
1 –Very important; 2 – Important; 3 – Some importance; 4 – Not important;

Experts found next list of arrangements, actual for getting prepare company's staff for future organizational and process changes in software development (given in order of popularity, but all presented in Delphi study actions are relevant):

- Kick-off meetings and detailed explaining for all staff;
- Personal meetings with line and project managers;
- Internal marketing support of future changes;
- Announcing of changes by top-management.

So, kick-off meeting is a most common and popular practice and meet in practice of more than 90% of experts. Surprise that «internal marketing support» met only in experience of 47% experts. Despite of well-known factor of success in change management in IT-companies – «support of ordinary engineers» [5].

Also panel is sure about need of creating separated team for internal project of process improvement. This team should be formed from managers and lead specialists, for whom work in this team is not a core job in company, but rather complements their basic functions.
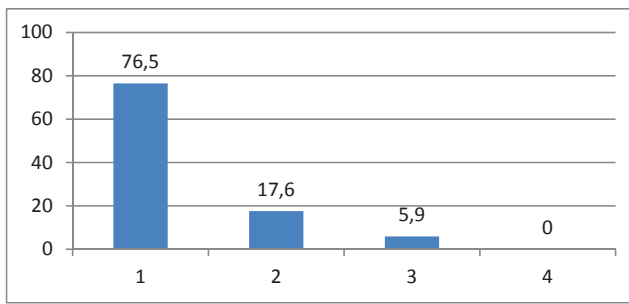
Figure 10. How important is forming a separated team for internal project of production processes improvement (with participation of different company's managers)?
1 – Very important; 2 – Some importance; 3 – Not important; 4 – Damage change management.

Also experts have identified the dominant role of the company's first person (CEO) for initiating and implementing changes in production processes at the level of the whole software company: in practice of more than 90% experts top-manager significantly helps to overcome internal project's crises, such as personal conflicts or lack of resources. While only 6% of experts met a CEO, who directly and immediate manage this kind of project in software companies.

Involvement of top-management on early stages of internal process improvement gives strong benefits and helps overcome a lot of intermediate problems. Innovators in company shouldn't afraid of high expectations or super extra pushing from top-management side, mostly they are prefer to watch process of improvement from the roadside and correct it only in special cases.

Panel agreed that the major and most frequently recurring problem was the problem of a formal attitude from the side of the participants of the process, this formal attitude means change implementation without significant results and deep understanding of the main goals. More than 80% experts met such kind of problem in their practice. Meanwhile more than 50% of experts met a huge resistance of IT-company staff, involved in changed business processes.

It means that explaining and wide debates about goals and process of changes implementation should be started at early stages and continues through the whole project. There are a lot of practices and approaches, supported involvement company's staff in Total Quality Management [6] or following standardized processes [7]. Relevant of them should become regular activities in appropriate internal project plan.

The experts also identified a problem of serious contradictions between the current practices in projects in various stages and new approaches that leads to the simultaneous maintenance of several different methodologies in the company; it takes more efforts and frustrates employers. At the same time causes significant difficulties "imaginary" interest of top management, who declares priority of changes, but at crucial moments do not support corresponding project.

In common panel do not define vector of current customer's influence on internal projects of changes implementation. Significant part of experts (around 30%) is thinking that current stakeholders, interested in final software product, do not influence on such kind of internal activities.

Panel agreed list of major methods of overcoming staff resistance in software companies (in order of method's popularity):

- Involving resisting persons in changes implementation;
- Positive motivation to adopt changes;
- Advocacy with elements of suppression.

Meanwhile, only 10% of experts admitted direct suppression as a using method.

It's important to compare planned and real actual results of changes implementation. Experience of panel was positive.
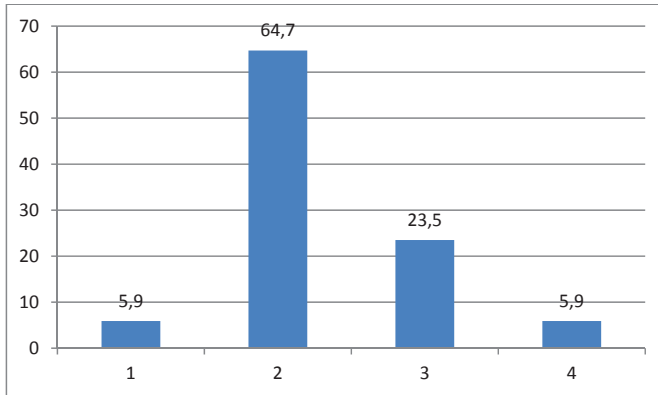
Figure 11. How strong planned goals of changes implementation usually modified in the end of internal project?
1 –Goals are almost lost; 2 – Part of goals are achieved; 3 – Goals are achieved, details became better; 4 – Nobody cares about comparing planned goals and actual results.

Practically it means, that planned goals may be divined on the parts and for each part of goals may be identified a separate project stage in process improvement.

Also time schedule of internal project in dynamic IT branch is an important project parameter. Experience of panel shows, that in most cases originally planned dates were substantially exceeded.
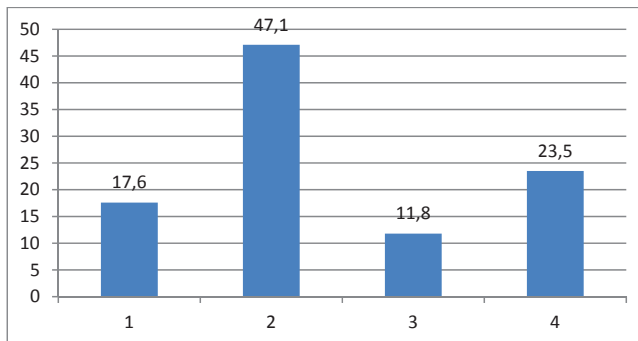
Figure 12. How final time of changes implementation in the production processes at the level of the whole company is comparable with planned schedule of internal project?
1 – Final schedule exceed planned on 50% and more; 2 - Final schedule exceed planned on 20-50%; 3 – Final and initial planned schedule are equal; 4 – Nobody estimates time schedule.

So, additional risk of lack of time is actual for such kind of projects and monitoring of project schedule should be a regular and proactive. On the stage of planning management team should consider possible temporary reserves.

Panel agreed: it's important to have a formal summarizing the implementation of changes in software production processes at the company level:
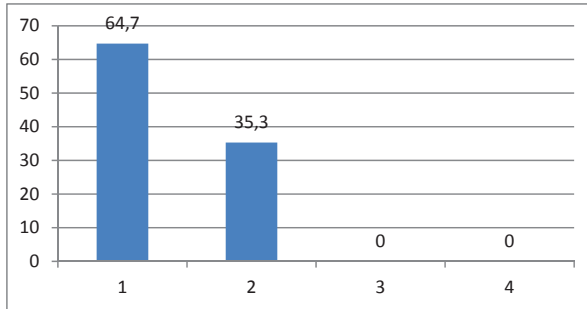
Figure 13. How important is a formal summarizing of the results of change implementation in software production processes at the company's level?
1 – Very important; 2 – Some importance; 3 – Only in case of successful implementation; 4 – Not important.

It means that summarizing in the end of internal project should be formalized and scheduled in project plan. Also such kind of reports may be reviewed from time to time, especially in the beginning of next stage of internal project of process improvement.

Experts have defined set of effective steps for consolidation of implemented changes in the production practice of company (in order of answer's popularity):

- Additional audit of process execution;
- Process documentation in corporate standards and instructions;
- Attention to process or practice in case of recurring defects and problems;
- Internal marketing support of implemented changes.

Also experts add some actions:

- Internal trainings;
- Automation tools configured according new processes.

Panel had identified the role of external consultants in internal projects of production process improvement on the level of the whole company like «important on some stages»:
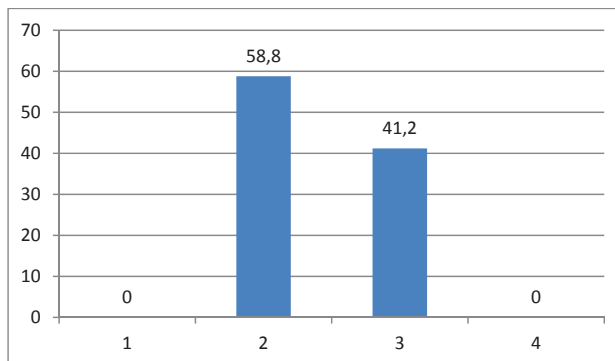
Figure 14. How important is an involving of external consultants in internal project of change implementation in production on the level of the whole software company?
1 – Important for the whole project; 2 – Important on some stages (trainings, audits, etc); 3 – Little of importance; 4 – Damage the project.

4 Conclusions & Recommendations

Expert's panel recommends for process improvement approach of internal project with structured set of documentation not less than for external software or consulting projects. For such kind of project should be formed a team from lead specialists and managers, who execute those roles in company in addition to their core functions.

From the research results authors may recommend to pay a strong attention to the formal stage of planning, when manager of this kind of internal project may manage risks and plan important issues:

- Additional time reserves;
- Involving external consultants in some stages and activities (like trainings or audits);
- All arrangements and actions, aimed on the overcoming typical implementation problems;
- Working with support and loyalty of top-managers, who may help in critical points of project.

There are two well-known problems in such kind projects that may be envisaged on planning stage: lack of time and lack of resources. Additional time reserves could solve first risk and involving top-managers could solve second one. Support of top-managers (like CEO, CTO or COO) could be a strong factor, giving additional chance for success to project of software production process improvement. Involving top-managers in changes management on high level may be a most valuable resource on this stage [8].

Divining internal project on phases while it's planning could prioritize goals and gives additional chances for successful achieving even part of them.

On the next formal stage of internal project – preparation company's staff for future changes [9] – panel recommended to start a set of actions and first of all: Kick-off meetings and detailed explaining for all staff. Well-done preparation of employers for future changes may save a lot of time and efforts for innovators on the next stages of internal project – detailed study of changes and changes implementation.

Changes implementation meets a lot of risks and problems in IT-companies [10]. This Delphi study shows some problems like formal implementation without results and its understanding from employers or organizational resistance. It requires from internal project team a lot of efforts and attention during all implementation stages. Experts recommended:

- Involving resisting persons in change implementation;
- Positive motivation to adopt changes;
- Advocacy with elements of suppression.

From the research's results authors may recommend formalizing and documenting results of internal project despite of its result. Such kind of report may be used in planning future process improvement or during correction actions in next changes implementation.

Experts recommended set of effective steps for consolidation of implemented changes in the production practice of company, such as:

- Additional audit of process execution;
- Process documentation in corporate standards and instructions;
- Attention to process or practice in case of recurring defects and problems;
- Internal marketing support of implemented changes.

Research shows importance of process improvement and standardization that needs a planned and balanced approach for change implementation on the level of the whole company. Panel responses, especially in consensus opinions, are demonstrating needs of considering all factors of organizational resistance and analysis of each stage of changes implementation project.

Contact Us

For any details, related with research process, please contact Pashchenko Denis, e-mail: denpas@rambler.ru

5 Literature

1. Russian Software Industry Overview, http://www.software-russia.com/why_russia/industry_overview
2. Lipaev V.V.: Economical problems of production of huge software products (Проблемы экономики производства крупных программных продуктов), vol.1, pp 3-22. Radio Industry, Moscow (2009)
3. Parabelum A., Zapirkin D: Business Development (Развитие бизнеса), Moscow (2011)
4. Pomeroy-Huff M., Mullaney J., Cannon R., Sebern M., «The Personal Software Process (PSP) Body of Knowledge», ver. 1.0, Special report CMU/SEI (2005)
5. Tsipes G., Kuzmishchev A.: Organizational change projects in large companies: assessment and decision-making methods (Проекты организационных изменений в крупных компаний: методы оценки и принятия решений), vol.37 pp.6-21. Project and Program Management, Moscow (2014)
6. Fowler P., Rifkin S.: Software Engineering Process Group Guide CMU/SEI-90-TR-024. Carnegie Mellon University, Pittsburg (1990)
7. Erik Van Veenendaal: Test Maturity Model Integration, http://www.tmmi.org/pdf/TMMi.Framework.pdf
8. Pashchenko D.S.: The principal role of CEO of IT-company in the implementation of changes in the production processes model (Роль руководителя IT-компании в процессе внедрения изменений в производственные процессы). The News of Tula State University, Tula, 2013, №4, part 1, 164-179 pp.
9. Pashchenko D.S.: Resistance to organizational changes in software development companies (Проектирование организационных изменений в IT-компаниях с учетом факторов противодействия), vol.4, pp 170-179. Management and Business Administration, Moscow (2012)
10. Mosyagin M.V.: Analysis and technologies of organizational changes in IT-companies (Анализ необходимости и технология организационных изменений в ИТ-компании), http://journal.itmane.ru/node/638 Theses of program MBA.IT Management School. Moscow (2012)

[29 January 2014]

6 Author CVs

Denis Pashchenko

Engineer in Software Development (RGTU, Moscow, 2006);
PhD in System Analysis (Moscow, 2010);
General MBA (Moscow Business School, 2014);
Doctoral student in Financial University (www.fa.ru , Moscow, now)

Country Manager in Russia & CIS in IND Group (www.indgroup.eu)

Andrey Blinov

Specialist in Management (Moscow, 1983);
PhD in Economy (Moscow, 1989);
Doctor in Economy (Moscow, 1994);
Professor (Moscow, 1997)
Corresponding Member of the Russian Academy of Natural Science (Moscow, 2010)

Professor in Financial University (www.fa.ru)

Agile Strategy Management Lessons Learned

Soren LYNGSO (soren@lyngso.lu), 24 av. Fr. Clement, L-5612 Mondorf-les-bains, Luxembourg

Abstract

The paper presents practical experience with agile strategy management methods and techniques gathered from multiple missions in private and public enterprises. People and communication are in focus. These elements present at the same time the foundation and primary opportunity for success; and the biggest risk of failure.The experience shows how weakly established organization structures and objectives make communication and people enter into conflict instead of working together to reach feasible and beneficial business results. Without going into detail with method and HOW descriptions the paper outlines through case stories a set of tools that have been used for stakeholder management, for teambuilding, and to bring organizations in crisis back on track. These same framework of tools used early in and throughout important strategic initiatives have been used to achieve at beneficial business results fast end without any conflict using the „no excuse for failure" principle. The conclusion is that all stakeholders must be known and treated in such a way that they are happy from initiation to close out of a strategic initiative. The involved stakeholders need visibly planned for and agreed to organization structures, methods framework, object definitions, and communication to be successful and agile.

Keywords

People, Agile, Strategy, Teambuilding, Management, Quality, Stakeholder, Development, Implementation, Project, Program, Normalization, Accept-test, Satisfaction, Ownership, Agreement.

1 Why Agile Methods

Agility is about people. People formulate wishes and requirements. People form teams and organizations for communication. People implement and develop solutions. People ensure the quality of delivered solutions.

People change their minds and adapt the wishes and requirements accordingly.

But people also enter into conflict and disagreement that can be counter productive in critically important strategic initiatives.

The agile methods presented here were used to deliver strategically aligned Information Systems to our clients. Based on the methods we can promise delivery of fully accepted Information System solutions based on safe estimates. The estimates emanate from standard dialogues with all pertinent stakeholders in the context of fully standardized processes:

- Stakeholder identification and involvement
- Business analysis and scope definition
- Object oriented solution design.

The dialogues take place in scenarios that ensure motivation and best possible contribution from all involved stakeholders to such a degree that these stakeholders want to take ownership of the result – collectively and without conflict.

It is a basic agile principle of the methods that we strive to make the client take ownership of whatever is delivered.

We contribute to the client solution by delivering tools, techniques, and solution components that we happen to have the competence to deliver. The final solution inclusive the knowledge transferred or shared belongs to the client.

Our methods and techniques cater for Strategic Initiative establishment and governance that align Information System development, implementation, support, and governance in any industry with corporate strategy.

The methods and techniques are used successfully in industries such as:

- Finance (Bank and Insurance)
- Pharmaceutical Production
- Health Care
- Oil and Gas Production
- Logistics
- District heating production and distribution
- Electricity distribution
- Public Sector.

Wherever we have contributed to strategically aligned corporate information system solutions we have left our documented standards used for this work with the clients for them to use it without any restrictions.

Today more than 25 years after our first mission we are not first anymore, and probably not even unique, but we do have some stories to tell.

2 The method framework used
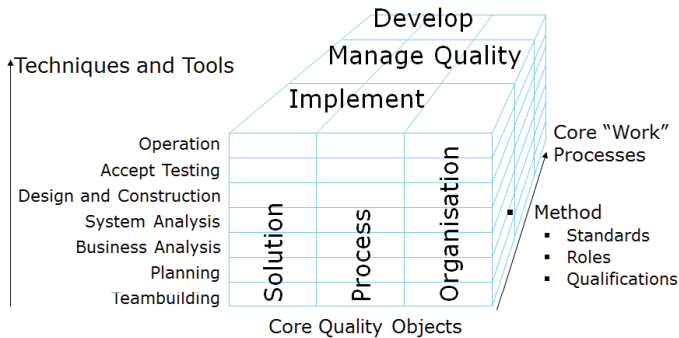
Figure 1 The Agile Strategy Management Method Framework

All method components are explained with respect to the core quality objects:

- Organizational requirements
- Process requirements
- Solution requirements.

The Core Work processes of Implementation, Development, and Quality Management are all established in support of agile behavior, where the concurrent involvement of competent resources ensure fast adaptation to unexpected and risk managed situations and events.
The standard processes and documentation used in the core processes contribute to efficient progress tracking and quality management from the start of a strategic initiative to the delivery of the expected result.
The concrete Techniques and Tools that are used in all development, implementation, and quality management processes ensure the full traceability of all results from idea to solutions in operation. Traceability is especially important while working agile, where results are adapted ongoing to changes in stakeholder demand.
The methods have been developed to be used during the different phases of strategic initiatives, where the strategic initiative can be an information system engineering project or a complete business process engineering program such as the establishment of a new factory.

2.1 Agile Strategy Quality Management

The Agile Quality Management standards used in our strategic initiatives comprise:

- Identification and activation of stakeholders to be involved
- Communication
- Team-building
- Decision making
- Documentation.

The Quality Assurance and the Quality Control methods of Agile Strategy Quality Management ensure that the key-stakeholders are satisfied with the deliverables.

The standards are established to be complementary to and sometimes replace components in industry specific or national method standards for delivery of solutions with the quite ambitious objective that:

> The standards can be understood concurrently by the most hard-core technicians, by the top-level visionary leaders, and by all other strategic initiative stakeholders.

Each strategic initiative stakeholder expects different types of benefits from the initiative, and each one reviews and tests the solution to be delivered for their own reasons, i.e. their proper WHY you need to understand.

The agile ongoing quality assurance of the solution, the organization, and the processes of a strategic initiative has contributed to the success of the methods used.

A fundamental capability of the methods is that they allow rapid solution development in order to be able to capture the solution benefits while they are relevant. This capability is ensured by early visualization of the complete solution structure and by ensuring that solution elements can be delivered and made productive early during a strategic initiative.

In the context of methodology no method is perfect and a great number of different industry, enterprise or national specific standards exist. While establishing our specific methods we have tried to provide the following advantages compared with such other methods and standards:

- People and communication come before processes, and processes come before documentation standards for the very simple reason that no process can run without resources, and even well-chosen people cannot perform well without good processes to govern their work
- Documentation standards are simple and cover only the necessary elements. The documentation standards can be seen as a content suggestion or as checklists. Other methods that promote documentation standards differentiated between complex and simple projects are doomed to fail because they never fit all anyway
- By providing only basic standards and norms, you give the teams and the competent people the ability to expand to standards of their own that are required for the production of best quality or at least feasible results on their specific tasks. Again people and teams freedom to act is in focus
- The methods have one basic requirement to all standard documents, which is that they must answer all WHY questions when used, e.g.:
    - Why has the team been established the way it is?
    - Why is this objective important for the enterprise?
    - Why is this activity conducted the way it is?
    - Why does this solution component functions the way it does?
    - Why is this test done?
    - Why is this suggested improvement an improvement?

2.2   Quality Management Objects

Quality Management comprises three process management classes with explicitly defined organization requirements, procedures, and result standards:

- Quality Assurance ensures that a solution will satisfy the stakeholder needs and requirements. This is handled by establishing agreed standards for all resources, procedures, and solution elements to be involved and/or delivered
- Quality Control is the ongoing effort to maintain the integrity of a method to be able to achieve the required solution quality. We use communication, interview and review techniques combined with advanced testing and verification procedures supported by standard Quality Management Information Systems to perform and document quality control

- Quality Improvement is the purposeful change of a method to improve the reliability of achieving a required solution. We have improved our standards and techniques periodically based on Lessons Learned.

For each quality management process class we use three quality objects as a foundation for evaluation of the performance of this process class:

- The Solution object defines the properties that are required from the solution to be delivered
- The Process object defines the properties that are required for high performance delivery of the required solution
- The Organization object defines the properties that are required for efficient communication, competence establishment, and decision-making.

3 Strategy and Strategic Initiatives

3.1 The Strategy

Corporate leadership establishes the strategy of a corporation. The strategy tells you why the organization has been established the way it is:

- Organization structure and geographical locations
- Products
- Business operations.

The strategy comprises:

- The corporate vision statement that "paints" a picture of how the corporation would like to be observed and how it observes itself in the future
- The corporate mission that tells a story about how the corporation intends to contribute to the happiness of its stakeholders. This is the strategy quality objective
- The confidential corporate objectives known by and sometimes contractually committed to by management tells you the direction followed by the corporation, e.g.:
  - o Internationalization
  - o Growth by acquisition
  - o Profitability (Return on Investment, Return on Equity)
  - o Sustainability
  - o Technological superiority.

All organizations whether public or private have a strategy and perform business activity governed by this strategy more or less successfully.
Corporate management translates the strategy into detailed organizational constructions, business procedures, and strategic initiatives that can ensure and improve the strategy quality.

3.2 The Strategic Initiative

The Strategic Initiatives establish the WHY, the WHAT, the WHEN, the HOW, and the WHO concerned with sustaining, changing, and improving business procedures and infrastructure in support of the corporate strategy.
Strategic Initiatives comprise usually Information System establishment or improvement in support of business operations. When my companies have been involved with corporate strategic initiatives this has always been the case.

Strategic Initiatives are programs and/or projects.

4 Agile Principles

The agile principles of our methods framework ensure that the strategy stakeholders are happy all the time.
The agility is there to overcome the constraints of mistrust and suspicion among strategic initiative stakeholders. The core agility principle is:

> Each process from the definition of the initial need for change to the final delivery of the agreed solution contributes to stakeholder trust, mutual respect, motivation, and willingness to take ownership of the solution components delivered.

This is the philosophy behind our processes, tools, and techniques. They involve the stakeholders in such a way that they feel that the results obtained belong to them. In this way, each process contributes to the motivation for the next one.

5 Stakeholder and Scope Establishment

The initial set of objectives provided by corporate leaders are an indication of what kind of business and stakeholders that must be involved in the establishment of the strategic initiative. In most cases that we have been involved with these objectives were not precisely defined to establish the full scope of the strategic initiative on their own.
Therefore, the first action in the establishment of a strategic initiative scope is to identify and to have open dialogues with potential key-stakeholders to be involved with the initiative, while you respect that you do not know what the real scope is - yet.
In order to identify all stakeholders to get involved or to be communicated with you need to look at the complete value chain, e.g.:

Figure 2 Value Chain Example (http://bettyfeng.us)

In several cases, leaving out potential key-stakeholders has led to the complete failure of the strategic initiative. Some real and recent examples of less efficient stakeholder management are addressed in 5.1 and 5.2.
A complete quality assured scope and stakeholder definition will follow later through a standardized brainstorm based process such as Process Quality Assurance (PQA).

5.1   The Private Bank Case

I was involved with a large Program to automate a private bank giving all clients access to full web banking. Focus was on technology and functionality and all of this was successfully implemented and even Accept-Tested and approved before the disaster was discovered.

> The bank clients only got involved to enter transactions after the fully integrated solution was technically functional and had been approved from "Friends and Family Testing".
> System usage was expected to reach 3000 transactions per day 3 month after the release date.
> The solution never had more than 300 transactions performed by the users in one day and in 90% of the cases, these transactions were performed by known Friends and Family testers.
> Millions of $ were wasted to such an extent that the stock price fell considerably.
> The future users were recognized as stakeholders, but an appropriate dialogue was not established with these stakeholders before it was too late.

5.2   The DANCOIN Cash Card Case

Another Project my organization was involved with was the development and implementation of a Cash Card in Denmark – the DANCOIN case.
Several worldwide recognized patents came out of this exciting project.

> The technical development was a great success. Partners such as banks, credit card facilities, and local transportation were directly involved with development and implementation.
> A whole city was set up for Accept-Testing end to end of the integrated solution with service providers and central bank cash and transaction cost clearing. This acceptance test was a great success also seen from a publication and advertising point of view with press and television coverage.
> So why did the Danish cash card not have success contrary to basically the same card implemented in e.g. Rotterdam, Holland?
> The failure to succeed occurred because of lag of communication with the corporate management people from the core stakeholder, the local transportation organization, HT:
> In parallel with the DANCOIN development HT developed their proper card and card reader devise for their buses and train stations without coordinating this effort with the DANCOIN project.
> On the eve of going live HT refused to implement a DANCOIN reader.
> Only inferior usage such as a few parking terminals, a few laundries and some unmanned newspaper kiosks went into production.
>
> This was not enough to pay off the DANCOIN investment and HT had enough resources to write-off their part of the investment.

6   PQA Agile Team-building

The method to get the teams build and to establish agreement about what the solution will be is the PQA process.
PQA is Risk Management based, but the focus is Opportunity rather than Threat.

PQA does not only document what the scope is, it also documents why the scope is defined the way it is. This ensures the agility of the PQA defined scope because if any why-case changes then the scope must change. The organization to change the scope is explicitly defined in the PQA result.

6.1   No excuse for failure principle

Much too often we have seen projects and major programs moving along with weakly defined organization of responsibility and activity defined on a level, where management is impossible. Such situations lead to crucial lack of commitment from all involved stakeholders because:

- Results don't show up
- Results show up too late to be useful
- Results show up without the quality that was never agreed on or documented.

The "no excuse for failure" principle implies:

- Involved human and technological resources are fully qualified to deliver the required and documented solution
- The involved resources are allocated and committed in such a way that their work is done and that the results are delivered without costly interruption, delay, and cost overrun.

The "no excuse for failure" principle ensures that all involved stakeholders are visibly committed and motivated from start to close out of the strategic initiative.

6.2   Simulated Accept-Testing

Communication of real measurable results is performed on a regular basis during Simulated Accept-Testing (SAT) that allows the not directly involved business and development stakeholders to evaluate intermediate results.
SAT performance allows timely and pertinent decisions about required changes to take place without disturbing the progress of complete solution delivery.
The SAT communication ensures that final Accept-Testing becomes a mere formality because all involved stakeholders already know exactly what they can expect from the delivered solution components during final Accept-Testing.

7   Recovery of Projects in Trouble CASE

Whenever my organization is called upon by major international organizations to deliver our core services of coaching and facilitation they have strategic initiatives in trouble. Most often, they have tried to implement a solution for month just to discover that no progress (except for spending time and money) has been achieved.

7.1   Medical Factory Implementation

We got involved in a major Program to implement a big factory to produce medical equipment using cheap raw material to deliver an end product of high quality to be used worldwide at a price to be acceptable even to poor people.
The Program was run like a Project with one Project manager facing several internal key-stakeholders with considerable internal power and many external stakeholders with legal and political power. The external stakeholders were delivering:

- Buildings
- Production Machinery
- QA Equipment
- Logistics Equipment
- Internal Machine Control Information Systems
- Administrative SAP based Information Systems.

The internal key-stakeholders were:

- The future factory manager
- The CEO.

The work to be done was defined at very low level (work packages by contractor) and a lot of work overlapped between contractors.
Arbitration between contractors and Project manager was handled at weekly Project meetings.
More and more conflicts between all parties surfaced very early. An important reason was that more than one contractor made key decisions and that these decisions were contradictory or at best not visibly aligned with the overall Project objectives. This again was caused by that the overall objectives were very weakly defined.
The Project was finally declared in trouble because major deliveries were slipping without clear responsibility for the delay.
It was obvious that a Program organization was needed to govern all stakeholders. Furthermore, a clearly defined unambiguous requirements specification for each contractor was needed and had to be agreed on by all parties.
Our contributions to this Program that has since delivered one of the most successful solutions in the history of the pharmaceutical industry were:

- Establishment of a Program organization based on visible high-level objectives (Critical Success Factors) and clearly defined high-level activities that each one was a major Project on its own. The organization, the objectives, and the activities were approved by top corporate management that became visibly involved in the Program management. The Project manager of the building implementation said: "This process should have been used from the beginning to avoid the troubles that started more than a year ago …"
- Coaching of the contractor responsible for delivery of the complete factory control system (a network of control computers connected to the numerical control on all production and Quality Control equipment)
- Our coaching comprised the establishment of the detailed Project plan. We also delivered the method and coaching to develop the normalized data structure and content, and the normalized process structure common to all control computers
- The normalized data and process structure allowed fast corrections of failures and resolution of problems and ensured an efficient interface to the SAP based order and production planning and control environment
- The normalized data and process structure was used and verified early in Simulated Accept-Testing to prove the efficiency of the solution to be developed and delivered
- We coached the setup and execution of Simulated Accept-Testing supervised by corporate management (The Program Management Team) that signed the solution development and implementation off.

Our methods used were:

- Process Quality Assurance to establish the objectives and a complete Project plan that allowed reliable estimation, forecasting, and tracking
- The Information Requirements Study to identify all core objects with their purpose and usage and to ensure that they were complete with respect to the overall success factors for the Program and the detailed success factors for the control system production and implementation
- The Object Lifecycle Analysis to detailed define and normalize all data and process objects in such a way that all control systems could be developed and implemented where needed; ensuring full integration among control computers and with external systems (numerical control and SAP)
- Simulated Accept-Testing to prove the feasibility of the data and process structure.

Both the pharmaceutical enterprise and the control system contractor coached by us implemented major method and organization improvements in order to benefit from the methods used also in the future.

8 Conclusion

Qualified and competent people selected to deliver a pertinent solution based on common goals need and deserve the best possible framework of tools and techniques for communication in order to establish a „no excuse for failure" situation without conflicts.

The „no excuse for failure" situation allows the involved teams and people to establish complete and pertinent objectives that can be understood and agreed to by all stakeholders. The involved stakeholders want to take ownership of their part of delivered results.

Agile adaptation to changed conditions is ensured by tracking delivered results early by all involved parties that together possess the competences to:

- Evaluate results
- Make decisions
- Initiate work
- Perform work.

Successful delivery of solutions to complex needs and requirements of strategic initiatives depend on:

- Your knowledge about all pertinent stakeholders
- Your ability to get the stakeholders involved with solution delivery from initiation to close out
- You capability to keep the stakeholders motivated and as happy as possible throughout the strategic initiative.

9 Literature

Shaku Atre: "**Data base**: Structured techniques for design, performance, and management with case studies" (John Wiley & Sons Incorporated, 1988)

E.F. Codd: "The Relational Model for Database Management

Chris Date: "Database in Depth: Relational Theory for Practitioners" (ISBN 0-596-10012-4)

James Martin:
Information Engineering, Savant Institute, Carnforth, U.K., 1987
Strategic Information Planning Methodologies, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1989

Soren Lyngso: "Agile Strategy Management, Techniques for Continuous Alignment and Improvement (ISBN-10: 1466596074, ISBN-13: 978-1466596078), 2014

Soren Lyngso

Soren Lyngso, MA (Econ) from Copenhagen University, PMP, General Manager, Author, Facilitator, and Coach has more than 30 years of experience from general, project, and program management and from implementation of complex business and IT solutions in different industries and countries. Experience has been gathered from Danish and international projects spanning maintenance of oil production platforms, factory implementation in the pharmaceutical industry, container line system implementation, distribution of toys, cash card system implementation, defense facility management, and in private banking and asset management solution implementation.

Currently his main occupation is teaching and coaching project and program managers on all levels in Europe and the Middle East – the favorite subject being "Rapid Assessment and Recovery of Projects in Trouble". He has published his quality model for strategically aligned solution implementation on his web site www.lyngso.lu.

A Methodology for Designing and Evaluating Web Platforms

Martínez Alcalá, Claudia[1,2] ; Calvo-Manzano, Jose A. [1] ; San Feliu, Tomás [1]

[1] Departamento Lenguajes y Sistemas Informáticos e Ingeniería de Software Universidad Politécnica de Madrid, ETS de Ingenieros Informáticos, 28660 Boadilla del Monte, Madrid, España.
[2] Instituto Tecnológico Superior de las Choapas. Carretera Las Choapas Cerro de Nanchital Kilometro 6, J. Mario Rosado, 96980 Las Choapas, Veracruz, México

email: isabel.martinez.alcala@estudiant.upc.edu, {joseantonio.calvomanzano, tomas.sanfeliu}@upm.es

Abstract

This article presents and details a methodological proposal for the developing of accessible web environments. This methodology is based by setting guidelines allowing visualization and correct knowledge management. The results show the users' willingness to adopt the DeCU model proposed in this study. Also, users' satisfaction when validating the proposed web platform is manifested.

Keywords

Design, Knowledge Management, Usability, Prototypes, DeCU model.

1 Introduction

The integration of new Information and Communication Technologies (ICT) into diverse research groups and communities has enabled the dissemination of new knowledge among different knowledge networks [1], [2], [3]. According to the literature [3], [4], the use of the ICT offers a wide range of possibilities and advantages for research communities due to the variety of the available technological tools. Based on this, web platforms have become an instrument of great importance for research communities, because they have significantly facilitated the sharing and distribution of knowledge. This has allowed that users with little knowledge can use them as supporting tools in order to create and manage contents relatively easily.

However, web platforms are redesigned or modified based on developers' own criteria, and they sometimes do not have the necessary strategies to create products capable of interacting with the user in a more sophisticated manner [5]. Massanari [6] mentions that, functionally, users are quite a different group. That is why, firstly it is essential the user intervention during the whole development process of a web platform to assess the real needs of users. Secondly, it is necessary developers become conscious of the need to adopt methodologies that propose more effective solutions to user experience, making emphasis on a user-centered perspective.

The User-Centered Design approach (UCD) is a methodology that considers all the development process must be led by users, their needs, goals and characteristics. Although this methodology assumes the mandatory necessity of the user during the design process, it does not represent a framework that meets the particular needs of the organization and at the same time considers the technological characteristics, human aspects, social and the context of use of a web environment [7].

The objective of this article is to present the User-Centered Design and Approach Model (DeCU, by its acronym in Spanish) [7]. This model implements methods and techniques that allow the creation of accessible and useful web platforms. Besides, it proposes design solutions based on prototypes evaluated by users, and allows assessing the functionality and usability of the platform. The case study presented in this article evaluates a web platform oriented towards knowledge management of groups dedicated to Software Process Improvement. The obtained results demonstrate the importance of improving the design process of the web platforms, involving all users in a more effective way [8], [9].

The article is organized as follows: Section 2 describes the state of the art of this study. Section 3 details the structure of the DeCU model. Section 4 shows the case study focused on the adoption of the DeCU model. Section 5 presents the results obtained from the practical case, as well as the description of lessons learned during this process. Finally, Section 6 includes the conclusion of the study.

2 State of the Art

The right knowledge management, as intellectual resource, has allowed increasing the competitiveness and productivity of people and organizations. Nowadays, research studies require the integration of new Information and Communication Technologies (ICT), which enable the creation of new communicative and expressive environments for the practice of different activities [10].

Research communities deal with a diversity of data (intangible assets), and in order to know which data are useful and relevant for their members, it is necessary to carry out a study and an exploration of their objectives, activities and tasks. That way, it will be possible to obtain the appropriate information from the database. Also, these communities need to have a space for training and sharing practices, knowledge, interests and scientific knowledge, as well as appropriate knowledge management techniques and an effective control of their relationships with the environment [11]. Moreover, Davenport and Prusak [12] affirm that scientific communities, research institutes, universities and other organizations cannot survive and contribute the socio-economic development, without the possibility of exchanging information.

Because of this, it is not strange that research communities have become interested in adopting technological tools that allow them to define, control and manage innovation and intellectual factors. Gressgård [4] affirms that the introduction of web platforms into such communities has significantly facilitated the exchange and distribution of knowledge, and they have become supporting tools for users with little knowledge to create and manage contents easily.

In the last decade, due to the diversity of users and contexts of use, it have been launched research aimed at developing new design options that create products capable to interacting with the user in a more sophisticated way. And this has caused the adoption of methodologies supporting the development of more useful, acceptable and intuitive technological tools [13], [14], [15].

The User-Centered Design approach (UCD) is born as a methodological framework that consists in identifying some of the user's particularities with the aim of creating more familiar and effective environments for the final user. Also, it offers a set of techniques to apply during the decision-making process of design and organization of information that may be more appropriate for the user's needs and characteristics. The main objective of UCD is to support the whole development process of interfaces or products with user-centered activities in order to create products that are useful and easy to use [16], [17].

Consequently, UCD means that: a) there is an active user participation; b) technological characteristics, human, social and usage context are considered; c) there is a clear knowledge of users' requirements to perform their activities; d) users' physical diversity has been considered, allowing that each of them performs their tasks effectively and efficiently, e) there is an appropriate assignation of functions between the user and the object; and f) an iterative development of design solutions is considered.

However, the UCD approach is not properly applied when developing an environment, even when the participative necessity of the user in the development process is well assumed. This is because often it is not known that techniques, procedures and methods must be chosen to carry out each step of the development process and sometimes the teamwork gets confused when applying them. That is why, in order to create an environment capable of satisfying the final user, it is necessary to set up a methodological guide that allows empirically appropriate solutions based on the real needs of users.

3   Methods and Materials

In this section, a brief description of the DeCU model is included. In section 3.1, the research method used to develop the DeCU model is described. Section 3.2 contains the definition of the characteristics and the phases of the DeCU model.

3.1   DeCU Model: Research Method

The proposal presented in this research was developed based on the User-Centered Design Methodology (UCD). This methodology is defined as a philosophy based on the needs and interest of the user, with an emphasis on making products usable and understandable [18]. The UCD process is an iterative process that consists of 5 steps: 1) Plan the human-centered process; 2) Understand and specify the context of use; 3) Specify the user requeriments; 4) Produce design solutions; and 5) Evaluate design against requeriments.

Another important aspect for developing the DeCU model was to choose an method for analysing and collecting data. The Delphi method was chosen essentially because the results of adopting the model require a collective effort of all participants [19]. In the DeCU model it is contemplated performing three surveys for collecting the data, as established by the DELPHI method. These surveys allow getting the opinions of users with respect to the proposed platform.

3.2   DeCU Model: Definition of the Model

The originality of our model lies in the fact that it links three important aspects: a) the focus on user-centered design and collaborative design, b) the process of schematic representation of interfaces and, c) the evaluation of functionality and usefulness, which together allow to consider a web platform with functional features for the users [7].

The DeCU model is a reference framework that is characterized by setting guidelines allowing visualization and correct knowledge management, ensuring that the modeled environment provides its users with helpful and relevant information of their activities and tasks. Likewise, this model complies with specific goals such as: a) identifying the most relevant characteristics of the users and the organization; b) analyzing and designing prototypes to help the developer in the creation of more acceptable environments for each type of user; c) providing generic solutions to each specific and general needs of the user and his organization, and d) establishing the necessary instruments which allow evaluation and validation of the developed environment. [7]. Figure 1 illustrates the operation of the model and the iteration among its components.
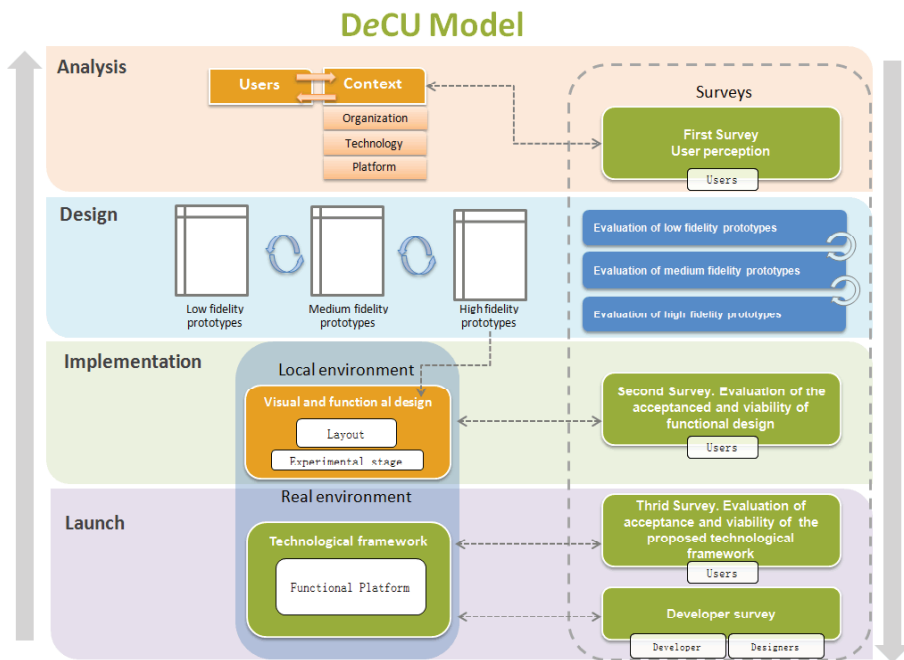
Figure 1. DeCU Model based on the User-Centered Design. Model adaptation [7].

As shown in Figure 1, the DeCU model consists of 4 phases that relate iteratively and create a dialogue among users, designers and developers. In the analysis phase, a study on the user's social context within the organization is performed. Simultaneously, the final user's needs, experiences and preferences are analysed through the application of a survey.

The design phase is responsible of the generation of prototypes that support the design of more acceptable and usable environments for each type of user. To do this, visual guides are used to represent the skeleton or visual structure of the application intended for development, which help determining the functionality and relationships among the different interfaces. Each of the prototypes is evalu-

ated by users through tests. In the implementation phase, the last process of design was performed, the layout, in a local setting in which the functionality of each component integrating the user interfaces was simulated. Within this phase a second survey applies to users. Finally in the launch phase, the technological platform was presented in a real setting and was made available for real users to make use of it. At the close of this phase the latest surveys to users, programmers and designers are applied.

4 Case Study

For this research, the method of case study and exploratory research for a rapprochement between empirical work and the reality of organizations was selected. The study was applied to a research group called Research Chair in Software Processes Improvement for Spain and Latin American Area (Cátedra de Mejora de Procesos Software en el Espacio Iberoamericano, MPSEI by its Spanish acronym), whose main goal is the research, adaptation and diffusion of the software process improvement techniques, the promotion of activities and interdisciplinary research, technological results and knowledge transfer, especially emphasizing their most innovative aspects in the area of Software Process Improvement applied to the Information Systems in Spain and the Latin American Area. Below, the design and evaluation process of the MPSEI Platform applied to the DeCU Model is shown.

4.1 Analysis

During this phase a study on the user's social context within the organization, as well as his goals, tasks and activities through an initial interview with the MPSEI group members is performed. The objectives of these interviews were to identify the problems that currently occur within the research group on the management of knowledge. Table 1 shows the questions asked in the first interview.

Table 1. Questions asked in the first interview.

Question 1	According to your perception, do you consider that the former platform of the MPSEI Research Group satisfies the particular needs of users and of your department?
Question 2	Do you consider it is important to have a visualization of the publications and projects produced in your department?
Question 3	Which do you think are the main problems in the platform you are currently using?
Question 4	Do you consider the functionality of any of the platform features can be upgraded? Which?
Question 5	Are you willing to use a web platform reflecting relevant content and managing your information adequately?

The needs and demands posed by the study group are translated to counting on a system capable of: a) Managing the contents adequately; b) Allowing interaction among users; c) Allowing the visualization of the contents generated by the research group; d) Disseminating Software Processes Improvement products and techniques (articles, projects and conferences); e) Enhancing the research group productivity; and f) Offering the researchers personal space. The user profiles were defined as follows: administrator, researcher, sponsor / committee and anonymous user.

Moreover, according to the analysis of the first interview, it is evident that the best technological solution for the web platform is a web technology environment fitted with Content Management System (CMS). CMS is an option that adapts to the research group needs. The selected environment was WordPress, as it is an open-source software focused on the creation of periodically updated web sites.

4.2　Design

In Figure 2, the evolution of the prototypes created for this study is shown. It is important to mention that for each prototyping category, evaluations were conducted by the user in order to approve the information architecture, usefulness and functionality of each interface.

| Prototipos de baja fidelidad | Prototipos de fidelidad intermedia | Prototipos de alta fidelidad |

Figure 2. Prototyping Evolution.

According to the Figure 2, Wireframes were used as instruments for the conceptual representation of the interfaces. A total of 8 low-fidelity prototypes were designed. Once the initial layouts were approved, we carried on with the elaboration of 12 medium fidelity prototypes. Finally, 6 high-fidelity prototypes were designed where we represented the most precise features of the web platform. It is important to mention that for each prototyping category, evaluations were conducted by the user in order to approve the information architecture, usefulness and functionality of each interface.

4.3　Implementation

In order to produce a more detailed and comprehensive interactive process of the MPSEI web page, a functional prototype in HTML was developed where the functionalities of the web platform were simulated. This prototype was implemented in a testing environment where users could evaluate it. This phase of the DeCU model was evaluated and approved by users through a survey. The objective pursued by this survey was to evaluate and collect information on the acceptance and viability of the functional design implemented in the web platform.

4.4　Launch

In this phase, the technological platform was presented in a real environment (http://mpsei.fi.upm.es/) and was made available for real users. Once the launch phase was carried out, a last survey was conducted among users with the aim of knowing the acceptance and satisfaction levels towards the proposed web platform, as well as evaluating the incorporation and acceptance of the DeCU model.

Likewise, at the end of the web platform launch, an evaluation was administered to the platform developers so as to know the additional effort they put into applying the DeCU model to the web platform creation. It is also important to mention that the participation of MPSEI research group members was required for the evaluation of the web platform design and development. A total of 12 people participated in this study. This group of people has experience in managing and administering the contents of the Software Processes Improvement group.

5 Results

This section contains the results obtained from the analysis of the evaluations performed by the DeCU Model. The survey results provide interesting guidelines and advice for the web platform design with the application of techniques and methods focused on the user. The discussion of the results is organized according to the following three subjects: (1) User perception regarding the need to develop a web platform for their knowledge management, (2) Validation of the MPSEI web platform, and (3) barriers for setting up the DeCU model.

5.1 User perception

This section shows the results of the first survey applied to users, with the aim of identifying their needs and preferences regarding the development of the MPSEI web platform. Additionally, it was necessary to assess the pertinence of incorporating the DeCU into the development of such platform.

The results indicate that all of the users were willing to adopt the DeCU model to shape and personalize the web platform. Also, they indicated that they required a platform capable of managing their knowledge and displaying relevant content. Based on the same analysis, it is possible to observe that 6 out of 12 users indicated that it is important to disseminate the software processes improvement products and/or projects generated by the group, while 4 out of 12 users mentioned that it is relevant for them that the web platform be totally functional. Lastly, 2 out of 12 users considered a good idea to have a personal space within the web platform for their information management.

5.2 Validation of the MPSEI Platform

This second section shows the results obtained from the evaluations applied to each of the prototypes and to the implementation and launch phases. For doing so, it is important to highlight that the prototypes generated for the platform were very close to the final product, so that users could better evaluate each criterion and component displayed. Due to this, few changes were made to achieve a complete functional version of the modelled web platform.

5.2.1 Prototypes evaluation

As for the evaluation applied to low fidelity prototypes, we received the following feedback on the platform's architecture and information design. The results indicate that 9 out of 12 users were satisfied with the home page structure. Likewise, in the evaluation of the researchers' page, 8 out of 12 users showed that the information design was appropriate. From the evaluation of the publication page, it was obtained that 10 out of 12 users reported to be totally satisfied with organization presented in the page, as it is clearer and easier to understand, while only 2 users maintained a neutral opinion. In the evaluation of the page related to the research lines page it was obtained that 9 of 12 users indicated to be totally satisfied with the organization of this section, 2 out of 12 of the users maintained neutral opinion, and only 1 user indicated to disagree with it. Finally, 9 out of 12 users were satisfied with the distribution and design of the projects page.

After the low fidelity prototypes evaluation analysis, readjustments were performed and medium fidelity prototypes were presented. The results indicate that 9 out of 12 users reported to agree with the contents and elements displayed in the home page. Likewise, in the researchers' page 8 out of 12 users indicated that they were completely satisfied with the content of this section. Moreover, 4 out of 12 users said to agree with what is displayed in the researchers' page as long as more elements are added to the page (e.g., researchers' Curriculum Vitae).

After performing readjustments to the prototypes, high fidelity prototypes were presented and the results indicate that 11 out of 12 users reported to be satisfied with the design and contents displayed in each of the MPSEI web platform pages. Only 1 user indicated a neutral opinion.

5.2.2 Evaluation of the technological Platform

The MPSEI web platform was tested in the field during a month. During that period, MPSEI activities were performed. Figure 3 shows the bar graphs with the results obtained in each of the evaluated constructs. As to the ease-of-use, perceived usefulness, attitude towards use and intensity of use constructs, more than 7 of the users were satisfied with the MPSEI web platform. Besides, in the Intension to use construct, 11 users showed a positive discernment when using MPSEI web platform. As the web platform had been implemented in a real environment only 1 month before, users indicated that they needed more time to become familiar with the tool.
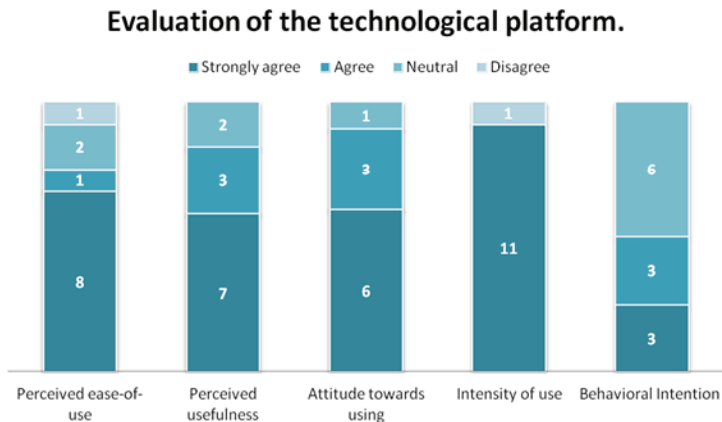

Figure 3. MPSEI Technological Platform Evaluation.

5.3   Barriers for setting up the DeCU model

The DeCU model entails the use of specific instruments to analyse and propose design and development ideas that meet final users' needs and expectations. That is why this model contains a series of iterative steps facilitating the development of intuitive web platforms that completely satisfy the user. However, it is important to consider certain barriers, outside the model, that may appear during its adoption. This section includes the barriers encountered by the work team when applying the DeCU model to the development of the MPSEI web platform, and how they were able to overcome them in order to attain a successful experience in the application of this model.

5.3.1 Barrier 1: Difficulty in conceiving ideas

It is possible to assume that involving users in the whole development process of a web platform is a significant fact, because it is thanks to the user that it is possible to detect improvement opportunities and to create easy-access and easy-to-use environments. However, this task often becomes difficult because users give diverse views and ideas. That is why the needs expressed by users are widely heterogeneous, which hinders the conception of ideas and solutions. As the designers and developers' vision is to design according to a certain type of user, considering specific criteria, they perceive the user as "problematic" or "indecisive" instead of visualizing him as a co-creator, capable of providing solutions and new ideas.

In order to attenuate this barrier, the teamwork needed to establish direct contact with the user, to observe him, understand him and analysed him, so as to identify his needs and specific characteristics, and to give a solution to his ideas. Likewise, it is essential to have a solid teamwork with the ability to adjust to the user's cognitive models, and capable of guiding the user about what he needs in

order to provide a right solution. It is also important for the teamwork to take user feedback into account so that the product can be re-designed if something does not work the way it is expected to.

5.3.2 Barrier 2: Communication Error

Lack of communication usually happens when the user is misinterpreted or his participation in proposing ideas for the design is simply ignored. In fact, there is usually no one in charge of managing the changes proposed by the user. That is why ambiguous decisions are sometimes made, implying that solutions that do not meet all the users' expectations are presented and the user is unsatisfied because he was not taken into account. It is true that it is impossible to satisfy every user, but it is possible to make the product more accessible and suitable for long-term use, so that in the end, users will agree with the solution provided.

To attenuate this barrier, the work team had to assign a person responsible for managing proposed changes and applying them when they needed to be performed, and even to communicate the user which changes were not performed and the reason why they were not. It is also important to make sure that the user understands the impact of his participation on the whole environment creation process.

5.3.3 Barrier 3: Lack of commitment

This lack of commitment in users derives from the fact that they are not widely informed about the importance of their contribution in the whole product development process. Due to this, users start to miss the evaluations and work sessions causing unfinished or low value products. Furthermore, the work team's lack of commitment is often derived from refusing to change the way they work, because these changes generate doubt and disagreement.

To diminish this barrier, it is necessary to establish communication among all the participants of the study, as well as to inform them about each phase implemented in a product development. Communication must be frequent, as communication gaps can motivate doubt and mistakes in the changes. That is why it is ideal that, from the beginning of the process, there is commitment in all the participants (users as well as developers), so as to develop valuable web platforms, based on final user satisfaction.

It should be noted that most of these barriers could be removed with adequate planning. Also, these are not exclusive barriers of the DeCU model described in this research work, so the organization should take these suggestions into account when adopting a model or methodology focused on the user.

6 Conclusions

Recently, the evolution of web platforms has experienced a paradigm shift because the user has become the focus of the design and development process. The adoption of the DeCU model for the creation of web platforms represents in itself a framework through which the needs of a whole ensemble "user-organization-technology" can be met. This research work shows a case study within the MPSEI research group, for which we proposed the development and evaluation of a web platform oriented towards the knowledge management of Software Processes Improvement. Results demonstrate users' willingness to adopt the DeCU model. Furthermore, users' satisfaction was manifested when validating the proposed web platform.

The most significant contributions of this research work are: (a) users' perception regarding the need to develop a web platform for their knowledge management, (b) the MPSEI web platform validation, and (c) the description of the barriers we faced when setting the DeCU model and the strategies implemented in order to surpass such barriers.

In summary, the adoption of the DeCU model has allowed designing an acceptable and useful web platform for the user and the study group, that offers successful use experiences. Nevertheless, this model requires the commitment of users and work team in order to develop valuable web platforms.

7 References

[1]   A. Forkosh-Baruch and A. Hershkovitz, "A case study of Israeli higher-education institutes sharing scholarly information with the community via social networks," *Internet High. Educ.*, vol. 15, no. 1, pp. 58–68, 2012.

[2]   D. Zhang, B. Guo, and Z. Yu, "The emergence of social and community intelligence," *IEEE Comput. Soc.*, vol. 44, no. 7, pp. 21 – 28, 2011.

[3]   H. Waheed, A. B. Salami, D. O. Ali, and A. Rahman, "Collaborative Web-based peacher professional Development system : A new direction for teacher professional development in Malaysia," *Int. J. Humanit. Soc. Sci.*, vol. 1, no. 7, pp. 208–216, 2011.

[4]   L. J. Gressgård, "Virtual team collaboration and innovation in organizations," *Team Perform. Manag.*, vol. 17, no. 1/2, pp. 102 – 119, 2011.

[5]   P. Bharatia and A. Chaudhury, "An empirical investigation of decision-making satisfaction in web-based decision support systems," *Decis. Support Syst.*, vol. 37, no. 2, pp. 187–197, 2004.

[6]   A. L. Massanari, "Designing for Imaginary Friends: Information Architecture, Personas and the Politics of User-Centered Design," *New Media Soc.*, vol. 12, no. 2, pp. 401–416, 2010.

[7]   C. I. Martinez Alcala, "Model customized views to content management in communities I+D+i (Doctoral thesis) - Modelo de vistas personalizadas para la gestión de contenido en comunidades I+D+i," 2012.

[8]   M. Maguire, "Methods to support human-centred design," *Int. J. Hum. Comput. Stud.*, vol. 55, no. 4, pp. 587–634, 2001.

[9]   T. Miaskiewicz and K. A. Kozar, "Personas and user-centered design: How can personas benefit product design processes?," *Des. Stud.*, vol. 32, no. 5, pp. 417–430, 2011.

[10]   C. Ferro Soto, A. I. Martínez Senra, and M. C. Otero Neira, "UNIVERSITARIOS ESPAÑOLES," *Rev. Electrónica Tecnol. Educ. Fac. Ciencias Económicas y Empres.*, vol. No.29, no. 1135–9250, pp. 1–12, 2009.

[11]   M. Izquierdo, L. M. Moreno, and J. M. Izquierdo A., "Grupos de investigación en conextos organizacionales acádemicos: una reflexión sobre los procesos de cambio y los retos futuros," vol. 22, pp. 103–141, 2007.

[12]   T. Davenport and L. Prusak, "Working Knowledge: How Organizations Manage What They Know.," *Harvard Bus. Sch. Press*, p. 224, 1998.

[13]   J. Kim, S. Lee, and Y. Park, "User-Centric Service Map for Identifying New Service Opportunities from Potential Needs: A Case of App Store Applications," *User-Centric Serv. Map Identifying New Serv. Oppor. from Potential Needs A Case App Store Appl.*, vol. 22, no. 3, 2013.

[14]   A. Pommeranz, J. Broekens, P. Wiggers, W.-P. Brinkman, and C. M. Jonker, "Designing interfaces for explicit preference elicitation: a user-centered investigation of preference representation and elicitation process," *User Model. User-adapt. Interact.*, vol. 22, no. 4–5, pp. 357–397, Mar. 2012.

[15]   L. C. Koh, A. Slingsby, J. Dykes, and T. S. Kam, "Developing and applying a User-Centered Model for the design and implementation of information visualization tools," in *International Conference on Information Visualisation*, 2011, pp. 90–95.

[16]   M. D. Corry, T. W. Frick, and L. Hansen, "User-centered design and usability testing of a web site: An illustrative case study," *Educ. Technol. Res. Dev.*, vol. 45, no. 4, pp. 65–76, 1997.

[17]   D. Petrelli, M. Beaulieu, M. Sanderson, G. Demetriou, P. Herring, and P. Hansen, "Observing users, designing clarity: A case study on the user-centered design of a cross-language information retrieval system," *J. Am. Soc. Inf. Sci. Technol.*, vol. 55, no. 10, pp. 923–934, 2004.

[18]   D. Norman, *The design of everyday things. New York: Basic Books*. 2002, p. 208.

[19]   J. Cabero, J. Barroso, R. Romero, P. Román, C. Ballestero, M. Carmen, and J. Antonio, "La aplicación de la técnica DELPHI, para la construcción de un instrumento de análisis categorial de investigaciones e-Learning," *Rev. Electron. Tecnol. Educ.*, vol. No.28, pp. 1–35, 2009.

8 Author CVs

Claudia I. Martínez Alcalá

Dr. Martínez-Alcalá earned her Ph.D. in Multimedia Engineering from the Department of Graphic Expression in Engineering at Universidad Politecnica de Cataluña in 2012, where she developed academic researches in design and customization of Telemedicine Systems. From 2009-2012 she began work in the Applications Multimedia Laboratory at UPC where she developed a number of web applications for patients with disabilities.

Her research interests have focused on designing web applications, personalization and user models, knowledge management, visualization of contents models, human-computer interaction, usability, user experience, conceptualization of multiplatform projects. She has published several articles in national and international conferences and journals on User-Centered Design.

She is currently Post-doctoral Research at Universidad Politecnica de Madrid, where she develops projects in the area of Software Process Improvement and Development of Collaborative Applications. Also, she is collaborating in the research line "Iterative and Distributed Systems" of the Department of Computer Systems Engineering, of the del Instituto Tecnológico Superior de las Choapas. The researches associated with this line are: the development of mobile computing systems and collaborative software.

Jose A. Calvo-Manzano

He is PhD in Computer Science. He is assistant professor in the Computer Science School at the Polytechnic University of 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 SEMEPRO 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 has been one of the members' Spanish translation team of the CMMI-DEV v.1.2.

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. He has been one of the members' Spanish translation team of the CMMI-DEV V1.2.

Application of the test select method using mathematical programming model

Hideto Ogasawara, Manami Sasaki, and Takashi Nakano
Software Engineering Center, Toshiba Corporation
{hideto.ogasawara, manami.sasaki, takashi2.nakano}@toshiba.co.jp

Hirohisa Aman
Center for Information Technology, Ehime University
aman@cs.ehime-u.ac.jp

Abstract

A need of effective software testing has been recognized indeed more to ensure quality of the software being more complex and larger in recent years. While the required number of cases to be tested has been increasing enormously to verify and validate the expanding large-scaled and multi-functional software systems, requested software development duration has been getting shorter year by year. In order to resolve such a contradictory situation, a highly sophisticated testing is expected to detect defects earlier, effectively and certainly within a short period.

Toshiba Corporation and University of Ehime have developed the test case selection method using the 0-1 programming model. This method gives a set of the most effective test cases within the consumption of constrained resources.

This paper shows that we developed Test Management System "TETRAPLUS" which has test selection function using the 0-1 programming model and reports an application of the 0-1 programming model to the regression testing plan for an industrial software. The key idea is to formulate a testing plan as a 0-1 programming problem (Knapsack problem). The empirical study shows that the 0-1 programming method can produce a cost-effective testing plan in which all potential regressions are found at only 22% of the cost of running all test cases. After empirical study, we applied this method at a system testing phase of a medical system development. As a result, we confirmed this method can select test sets for regression testing effectively in the limited time.

Keywords

Testing, regression, 0-1 programming model, cost-effectiveness

1. Introduction

System testing—testing behaviors of software—is a crucial activity to develop high-quality software. Any bugs overlooked during system testing may cause failures in the operational phase. Test engineers usually prepare many test cases to check various functions and conditions.

During system testing, many minor upgrades to fix bugs may appear. Then, test engineers should run test cases again to check whether they have regressions (bugs) or not, i.e., regression testing [1]. While it is ideal to run all test cases again for each minor upgrade, such a thorough regression test would be difficult due to realistic limitations such as lack of time and/or manpower. Thus, test engineers have to plan a cost-effective regression test under realistic limitations.

Test engineers often assign some values (worth to be tested) to their test cases in order to prioritize them. Then, those test cases can be selected in descending order of worth. Such a strategy is called the "greedy method" in the field of algorithm theory. While the greedy method is simple and easy-to-perform, it cannot consider testing costs properly. To take into account both the testing worth and the testing cost, Aman [2] proposed to formulate such a testing plan as a 0-1 programming problem (Knapsack problem [3]). The contribution of this paper is to report the application of Aman's method to industrial software.

2. History-based worth to run regression test

A. Test history data

In a system testing for a software, we suppose that we have N test cases Ti (for i = 1, …., N) and M versions Vj (for j = 1, . . .,M) to be tested. Then, let $r (i, j)$ to be the test result of running Ti for version Vj :

$$r (i, j) = \begin{cases} 0 \text{ (failure)}, \\ 1 \text{ (success)}, \\ NA \text{ (non-running)}. \end{cases}$$

These results can form a $N \times M$ matrix $\boldsymbol{R}$ = ($r (i, j)$), and the matrix is our test history data. It is ideal that all elements of the matrix are either 0 or 1; that is to say, all test cases were executed for all versions. In reality, however, some tests are omitted ($r (i, j)$ = NA) if their tests already passed in the previous version, e.g., $r (i, j – 1)$ = 1.

B. Worth to run Regression Test

The test history of Ti is given as the i-th row of $\boldsymbol{R}$:

$r (i, 1), r (i, 2), . . . , r (i, M) .$

Using the above history data, we will evaluate the worth to run T_i again (perform the regression test). Now we will focus on consecutive failures and NAs in the test history.

(1) How to use the number of failures

We will consider that a test case having a high failure rate as needing to be tested again. Then, we define T_i's worth to run the regression test, w_i, as

w_i = the number of failures / the number of execution .

The test case T_i with larger w_i would have a higher risk of poor quality.

Table1 shows an example of **R** with N = 3 and M = 5. In the table, we obtain w_1 = 0, w_2 = 0.5 and w_3 = 0.25. In this case, T_2 has the highest risk.

(2) How to use the number of NAs

Except for the above (1), we will consider that a test case having a long series of NAs as needing to be tested again. We define T_i's worth to be run again (to perform the regression test) as follows:

Suppose that T_i was run at version V_k, but it had never been executed after that version, i.e., $r(i, j)$ = NA (for $j = k + 1, . . .,M$), so there are "$M − k$" consecutive NAs until the latest version. Then, define T_i's worth to run the regression test, w_i, as

$w_i = M − k$.

The test case T_i with larger w_i would have a higher risk of overlooking regressions because it has not been run for a longer time.

Table1 shows an example of **R** with N = 3 and M = 5. In the table, we obtain w_1 = 2, w_2 = 3 and w_3 = 0. T_1 and T_2 have not been run since V_4 and V_3, respectively. That is to say, there are some possibilities of potential regressions in the latest version. T_3 has little or no worth to run again in the latest version because it was just run in the same version.

Table1. Simple example of R, where N = 3, M = 5.

	V_1	V_2	V_3	V_4	V_5
T_1	NA	0	0	NA	NA
T_2	1	0	NA	NA	NA
T_3	0	1	NA	0	0

3   0-1 Programming model

We have introduced the notion of worth to run regression   tests in the previous section. Using theses worth separately, we can select test cases in descending order; such a scheme is referred to as the "greedy method." While the greedy method is simple and easy-to-perform, it would not produce the optimal solution when the available time or effort is limited. To consider not only the testing worth but also the testing cost, Aman [2] proposed to formulate the testing plan as the 0-1 programming problem as follows:

$$\text{maximize} \quad \sum_{i=1}^{N} w_i \cdot x_i ,$$

$$\text{subject to} \quad \sum_{i=1}^{N} c_i \cdot x, \leq L, \ x_i \in \{0, 1\},$$

$$\text{and} \quad x_i \leq x_j \text{ if } T_i \text{ requires to run } T_j ,$$

where c_i is the cost needed to run T_i, and x_i is the 0-1 variable showing whether we run T_i or not; $x_i = 1$ if we run it, otherwise $x_i = 0$. L denotes our upper limit of total cost available for our regression testing.

The above inequality $x_i \le x_j$ represents a causal connection between T_i and T_j. Whenever T_i is selected to be run, T_j is also selected since $x_i = 1$ and $x_i \le x_j$ imply $x_j = 1$. Test engineers can make any causal connections depending on their situation. The authors have empirically used the following co-occurrence relation to make such a causal connection: whenever T_i is run, T_j is also run, we consider that T_i requires running T_j. For example, T_3 is always run when T_2 is run in Table1, so we have $x_2 \le x_3$ as one of our constraints.

The key contribution of the 0-1 programming model is to take into account "testing costs" reasonably. Such a notion of cost can always be crucial matter in the development of industry software.

4   Empirical study

To evaluate the effectiveness of the above 0-1 model, we conducted an empirical study with a software system which has been developed and maintained by the authors' company. Using 300(= N) test cases, we tested 13(= M) versions of the software during system testing. Now we consider the test cost to be the testing time (the unit is minute), and conducted the following comparison experiment:

Table2. Number of regressions found in empirical work.

L	Method	
	greedy	0-1 model
1h (60min)	8	16
2h (120min)	14	22
3h (180min)	14	22

- Worth to select test cases:

  Long series of NAs.

- Method to select test cases:

  1) greedy method,

  2) 0-1 programming method (0-1 model).

- Cost Limit (L):

  1) 1 hour (60 min),

  2) 2 hours (120 min),

  3) 3 hours (180 min).

Table2 shows the number of regressions found by running the selected test cases. For all L, the 0-1 model could find more regressions than the greedy method. Therefore, the 0-1 model would be more useful in planning cost-effective regression tests.

In order to discuss the efficiency of the above methods, we also performed all 300 test cases in 554 minutes. As a result, 22 regressions were found. Thus, the 0-1 model could find all regressions in only two hours, which corresponds to about 22% of total costs needed to run all test cases. While the greedy method is a simple and effective method, the 0-1 model seems to be much more effective.

5   Test Management System TETRAPLUS

Test items and test results can be inputted via web browser and Excel file, and these data are stored in internal databases. Using these data, the latest test items and test execution status can be checked easily. Currently, this system has been used about 30 development departments in TOSHIBA group. Figure1 shows the overview of TETRAPLUS. This system has the following three views. Users can switch display according to the purpose and confirm the situation.

Tree view

Test items are stored by tree structure. There is no restriction in the depth of a tree structure.

Main view

The test items and test results of selected folder by the tree view are displayed.

Command view

The view contains some commands which perform searching test items and results.

By using TETRAPLUS, the test efficiency can be increased at the following three points.
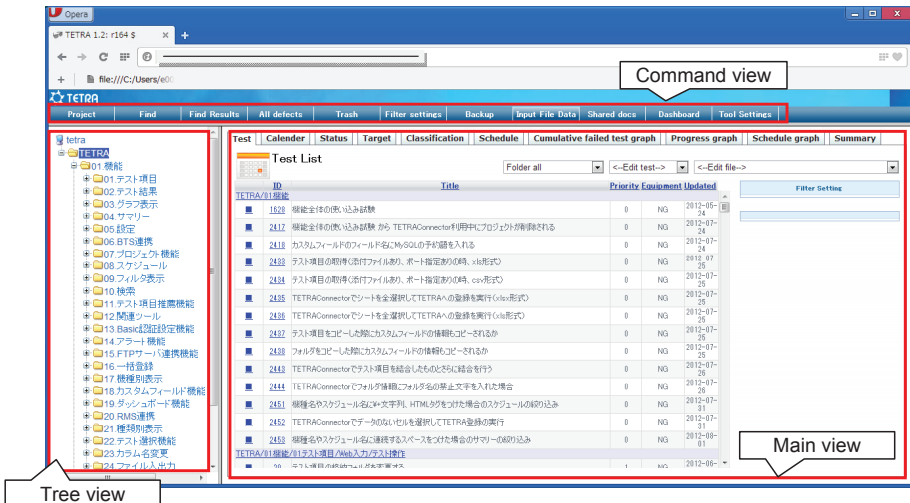

Figure1. The overview of Test Management System TETRAPLUS

(1) Monitoring the progress of test activities

Test monitoring can serve various purposes during the project, including the following:

- Give the test team and the test manager feedback on how the testing work is going, allowing opportunities to guide and improve the testing and the project.

- Provide the project team with visibility about the test results.

- Measure the status of the testing, test coverage and test items against the exit criteria to determine whether the test work is done.

- Gather data for use is estimating future test efforts.

To monitor of the progress, TETRAPLUS has three graphs of Cumulative failed test graph, Progress graph and Schedule graph. Figure2 shows a Progress graph.
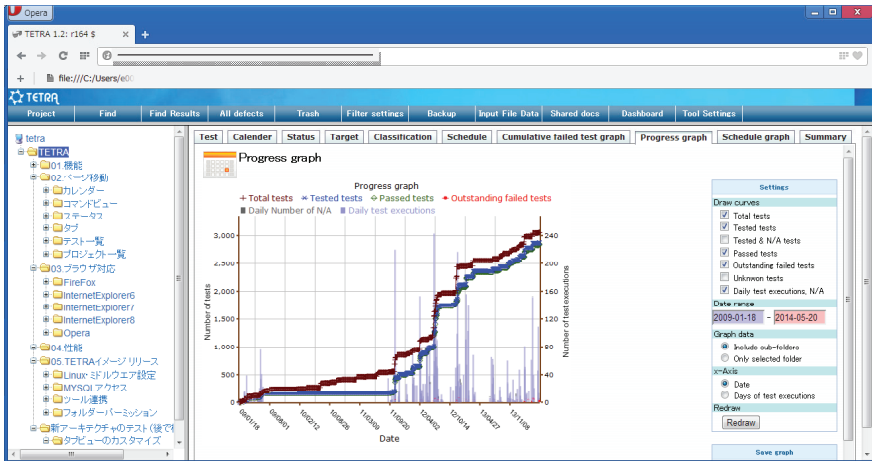
Figure2. Progress graph

(2) Reporting test status

Test summary report is a table summarizing testing activities and results. Figure3 shows a Summary report include the whole progress and each folder progress.
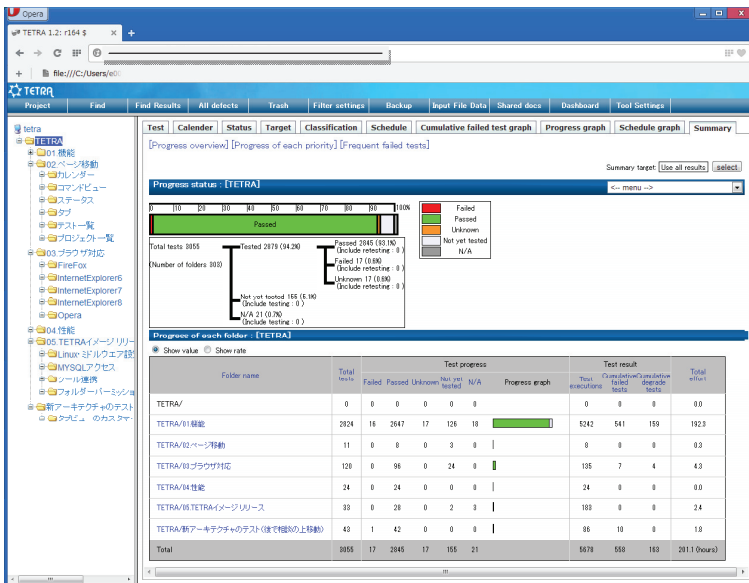
Figure3. Summary progress report

(3) Test control

Projects do not always unfold as planned. Risks become occurrences. When plans and reality diverge, we must act to bring the project back under control. TETRAPLUS has some functions, such as Test schedule graph, Reliability estimation graph and Frequent failed tests table for test control.

6   Practice application

After empirical study, we developed test select function in our developed test management system "TETRAPLUS". We applied the 0-1 programming model to MRI (Magnetic Resonance Imaging) system developed by TOSHIBA MEDICAL SYSTEMS CORPORATION. Usually, this system is released once every year. Test cases were selected using 0-1 programming model before final system functional testing phase carried out in March, 2013. In this phase, test team could use 170 hours for regression testing. Test select function could select 1,087 test cases which can execute within 170 hours from the all 2,997 test cases. The ration of the selected number of test cases was about 35%.

Figure4 show the result of final system functional test phase. Horizontal axis is the date. The left side   vertical axis is the number of accumulation failures, and the right side vertical axis show the number of test executions. The result of regression test using test cases selected by 0-1 programming model is shown in the right end of the graph. As a result, two regressions (bugs) can be detected. It was difficult to detect one of them by the conventional method. Furthermore, regressions have not been detected after the release. From this result, we think that this method can detect regressions more effectively.
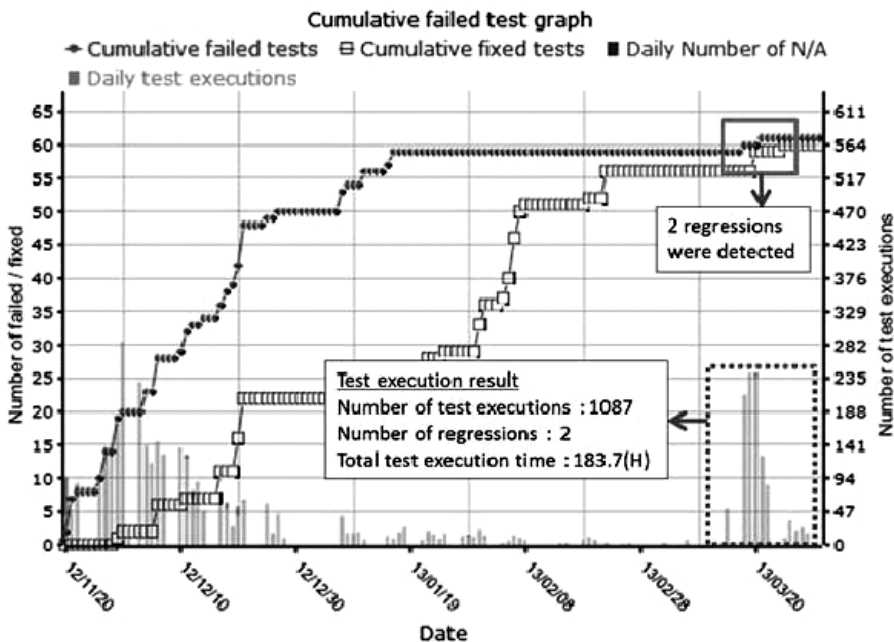
Figure4. Result of final system functional test phase

In order to confirm that there is no degrade, it is difficult to execute all test cases as a regression test in the limited time. This application result shows the effectiveness to select test case using the worth to run regression test.

7   Conclusion

This paper reported an application of 0-1 model to the regression testing plan for an industrial software. The 0-1 model could produce a more cost-effective testing plan than the conventional greedy method, and could find all regressions at only 22% of the costs to run all test cases again. Further, as a result of applying this method at the system testing phase of medical system development, test sets

for regressing testing can be selected effectively in the limited time. Our future work includes further data collection and experiments for comparing the 0-1 model with other intelligent methods (not only greedy one).

Literature

[1] J. Richardson and W. Gwaltney, Jr., Ship It!: A Practical Guide to Successful Software Projects, Pragmatic Bookshelf: O'Reilly Media, CA, 2005.

[2] H.Aman, M.Sasaki, K.Kureishi and H.Ogasawara, Application of the 0-1 Programming Model for Cost-Effective Regression Test, IEEE 37h Annual Computer Software and Applications Conference, 2013.

[3] M.R. Garey and D.S. Johnson, Computers and Intractability:A Guide to the Theory of Np-Completeness, W.H.Freeman and Company, NY, 1979.

Author CVs

Hideto Ogasawara
Dr. Hideto 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).

Manami Sasaki
Manami Sasaki has more than 5 years of experience in testing method and tools promotion at Toshiba Corporation, where she is developing test management system "TETRAPLUSE".

Takashi Nakano
Takashi Nakano has more than 10 years of experience in testing method and tools promotion at Toshiba Corporation, where he is developing test management system "TETRAPLUSE" and test case generating tool using all pair technique.

Hirohisa Aman
Hirohisa Aman is an associate professor at the Center for Information Technology, Ehime University. He is teaching in the area of software engineering, specifically in the domain of software metrics and management. He has been researching topics of software engineering, especially, the empirical data-oriented software development management and quality management. He is author of more than 20 international papers. He is also an Editorial board Member and Associate Editor for several international jornals. He is Publicity Co-Charis of HASE 2014 (15th IEEE International Symposium of High Assurance Systems Engineering).

The Feature Set of TestSPICE 3.0

Tomas Schweigert[1], Andreas Nehfort[2], Mohsen Ekssir-Monfared[3]

[1]*SQS AG, Stollwerck Str. 11, 51149 Köln, Germany*
tomas.schweigert@sqs.com
[2]*NEHFORT IT-Consulting KG*
andreas@nehfort.at
[3]*BDC IT-Engineering, Austria*
mohsen.ekssir@bdc.at

Abstract

The paper describes the feature set of TestSPICE 3.0. It explains the overarching structure of TestSPICE 3.0, its main components which are the "Business Life Cycle Process Category", and the "Technical life Cycle Process Category", the measurement framework, the assessment process and the TestSPICE Assessor Training. The paper also deals with the relationship between TestSPICE 3.0 and ISO/IEC 29119 on the one hand and ISTQB® on the other hand and it explains the support of TestSPICE 3.0 for agile projects.

Keywords

TestSPICE, ISO/IEC 29119, Agile, ISO/IEC 15504, Assessment Process, Measurement Framework, Test Process, ISTQB®, Software Testing, System Testing

European Certification and Qualification Association - Developments in Europe and World Wide

Michael Reiner, University of Applied Sciences Krems, Austria, michael.reiner@fh-krems.ac.at, Gabriele Sauberer, Termnet, Austria, gsauberer@termnet.org, Richard Messnarz, ISCN, Ireland, rmess@iscn.com

Abstract

The European Certification and Qualification Association (ECQA) is the result of a series of EU funded projects from 2005 – 2014. This included European projects such as EQN (European Quality Network, 2005 – 2007), EU Certificates Campus (2008 – 2009) and DEUCERT (Dissemination of EU Certification). Nowadays, ECQA operates as an organization that is independent from funding. The members of ECQA are widely spread all over Europe and vary from universities, companies and NPOs as well as individuals.

Recent key LLP funded projects include

- Automotive Quality Alliance (AQUA) where Automotive Clusters together with ECQA elaborate a new skill set and certificate for Automotive Quality Engineers integrating functional safety, Six Sigma, and Automotive SPICE competencies.

- LEADSUS developing key skills for sustainability management in Europe, including an ECQA certificate, training offer and Europe wide exams.

- I2E Idea 2 Enterprise as a project with schools where ECQA certificates for innovation management will be implemented at high school level (Gymnasium) to empower people for innovation already at an early stage of life.

- LSSH (Lean Six Sigma for Health Care) where the LSSA (European Lean Six Sigma Academy) which was founded by a joined project with ECQA elaborates quality skills in the health care sector.

Recent strategic actions for involvement in university level include:

1. ECQA job role training has been integrated as university lectures and ECTS points are granted for the students. Examples are in functional safety management and Automotive quality management.

Recent globalisation actions include:

2. ECQA was invited as a key note to a Chinese conference organised by Chinese government bodies and a memorandum for future collaboration was signed.

Recent Europe wide recognition:

3. The EU project DEUCERT (dissemination of ECQA certificates) has been published as a good practice by the EACEA (coordinating institute of the LLP program) [12].

1 History

The project EQN [2],[3],[4],[11] was a strategic network project in education in which industry and educational institutions from 13 countries did networking to propose a solution to this demographic problem. Also the project should deliver an initiative of innovation in education as well [6],[8].

This resulted in:

Job Role Based Qualification Strategies [54,[6],[11]. While the universities teach for a domain (e.g. informatics engineer) the job role based qualifications are short courses which in 2-3 weeks re-qualify people in industry (access from the workplace) to sustain their value for the company and remain employed. Job roles are described in form of skills sets (similar to the skill cards in the Department of Trade and Industry in the UK), and skill sets are mapped onto training, exercises and tests.

If someone studies informatics engineer this takes approx. 5 years at a university. Assuming this as a basis a job role qualification is for instance to upgrade to a safety architect, software process improvement manager, etc.

Modular Certification. It was also assumed that people in industry at the work place are under time pressure and cannot do the whole training at once. They might do a part in year 1 and the rest in the next year. They (assuming the European mobility strategy) could do the first part from Spain the rest from a German work place.

Thus, the skills sets were structured into skill elements and each skill element has so called performance criteria. Each certificate lists elements and a full certificate is achieved if all elements are passed.

European Mobility Strategy. If, for instance, in the current economic situation Spanish workers would move to Germany for employment and later would move to UK and then back to Spain, all certifications should be added up and accepted across all countries.

This led to the concept of so called lifelong learning accounts for people. Each person in Europe can register at the system and in one lifelong learning account (like an international bank account but for education) can do self assessment, achieve skills profiles, and receive training. The account sustains and is a central service across EU countries.

This infrastructure was later established in the EU Cert campus project.

Europe Wide Standardized Approach. To be able to roil out this strategy in all countries the design of standard guidelines was important. They were translated in all major European languages.

As a result of this the ECQA [2],[3],[4],[8],[9],[11] has established a set of standard quality guidelines for

- Certification of a new job role

- Certification of training bodies

- Certification of trainers

- Certification of people

See www.ecqa.org and cooperation with ECQA under guidelines.

A job role committee (JRC) is an international working group who signed the standard ECQA JRC agreement and annually maintain the skills card and test questions pool.

ECQA certifies training bodies who train assessors in this new assessment model.

Attendees of courses do an ECQA based exam and receive an ECQA Certificate.

ECQA provides conformity assessment services based on the principles of ISO/IEC 17024:2012. From the very beginning, some core requirements of the international standard ISO/IEC 17024 for certifications of persons have been considered in the ECQA guidelines and have been updated in the current version v05 of the "ECQA Architecture".

For example, one of the core requirements of this standard is that training organisations and examiners are separated and independent from each other. ECQA supports that by

1. Establishing a Europe wide exam system which generates tests randomly per person (each person gets a different test and cannot be controlled by the training body). The exam questions are assigned to skills elements of job roles.

2. Using certified exam bodies who provide examiners to organise the exams.

3. Automatic corrections through the ECQA test system so that none can interfere the tests and the results personally.

4. Job Role Committees who elaborate and annually update a pool of multiple choice test questions.

The work in PAC [1] will be reviewed against the ECQA quality criteria and the coverage of all quality criteria is checked by reviews.

2 EUROPEAN AND GLOBAL INFRASTRUCTURE

To support the concept of lifelong learning accounts (a strategy developed in the EQN project) the EU Cert Campus project (above 20 partners from 18 countries) developed an infrastructure which supports standard procedures for lifelong account management, self ass4essment of skills, exams and administration of certificates [1],[3],[4],[8],[11].

Once this infrastructure was developed and in place (now used by above 11000 professionals and managers in European industry) it was important to find a body maintaining these services. This resulted in the legal foundation of the ECQA (European Certification and Qualification Association) in 2008.

ECQA is an international Non-Profit-Association headquartered in Austria. The income model is based on the certification business.

ECQA follows – as much as possible and useful – the processes of independent exams as outlined in ISO 17024 (ISO/IEC 17024:2012, Conformity assessment — General requirements for bodies operating certification of persons, second edition 2012-07-01).

An external pre-audit in May 2013 revealed that the ECQA self-declaration is justified: „ECQA provides conformity assessment services based on the principles of ISO/IEC 17024".

Since ISO 17024 accreditation is optional and possible only at national level and per job role, ECQA doesn´t envisage to have 30 Job roles certified in more than 20 countries. However, it is foreseen to qualify for the future ISO 9001:2015 certificate for organisations.

ECQA earns a share of each certificate sold. The income is used to further maintain the lifelong learning accounts and exams infrastructure.

The members of ECQA (and owners) are more than 60 universities and training bodies from Europe.

At the end 2013 the EACEA (Institute in the European Union coordinating the LLP program) selected the DEUCERT project (dissemination of ECQA certificates) as a best practice to be publsished in a brochure of good practice.

In 2013, ECQA signed its first memoranda of understanding and cooperation agreements with Asian und North American Partners. e.g. an agreement between the China National Committee on for Terms

in Sciences and Technologies (CNCTST) the International Network for Terminology (TermNet) to introduce ECQA in China and to organise ECQA Certified Terminology Manager certificates and training in China with Chinese and European partners. Additionally a memoranda of understanding was signed with (SOBUS) for future collaboration in Quality management and trainings in China and Europe. A cooperation with Japan already started with Nilsoft becoming the first ECQA member of Japan and ECQA started the talk to other potential partners for trainings and certifications in Japan.

As fast growing network of training and research organisations, examination bodies, industry partners and NPOs, ECQA is on its way to become the world-market leader for industry certificates for persons.

Within the next years, ECQA will implement its expansion and quality strategies, composed of a globalization strategy, a multi-language and terminology strategy and a localisation strategy – localization meaning the adaptation of ECQA products and services to the specific markets and target groups all over the world.

Quality assurance and control will be one of the core activities of ECQA for the years to come: from "mystery shopping" in order to assess the work and conformity of ECQA Certified Training and Exam Organisations and Partners to internal audits on site – ECQA will take all necessary measures to provide standardized quality services all over the world and stay a reliable brand.

The vision of ECQA is that each person in Europe might have an educational skills profile (skill card like a bank card). In ECQA this strategy is adapted in form of an online skills profile which a person can maintain in a private lifelong learning account. Instead of a physical skill card with a chip on the cards, there are now lifelong learning online accounts.

A skills profile is a representation of the coverage of competencies of a person in different skills elements. Each job role contains skills elements, the coverage of skills is shown in form of a percent mark.

Exam results are displayed in the form of a skills profile, demonstrating the coverage per skills element. See Figure 1. To be certified you must reach 66% in each skills area.
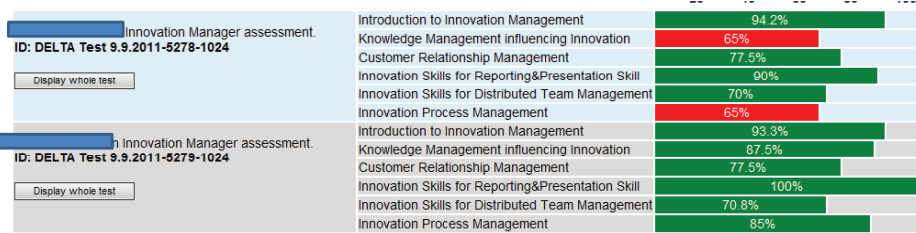

Figure 1: Typical Profile as Result of an Exam

The exam system portals are continuously updated and new functions to guarantee stability of the system during exams is extended since more than 6 years now.

The current number of 31 active professions is reworked due to some new concepts that was generated while taking a closer on the current and future needs of the target group. One idea was to cluster some of the professions, as several of them have similarities that can be used similar. For example to use project management basics in the same way in different job roles and not have differences in different job roles. This would allow participants that have already gained a certificate in a different job role to reuse their knowledge and to concentrate more on the fields being new.

Another approach was to use the current structure of modules (units & elements) to focus more on the performance criteria's. Out of this it is the idea to develop a skill based focus, which allows people to check for their current state of their skills (e.g. self-assessment, online short exam, questionnaire,…) and then be prompted with different additional skills that can be done, to gain a certificate showing the proper Job Role for this skills profile. This will then result in several current job roles, rework their ap-

proach and end up with basic (or foundation) level of certified knowledge/skills. This basic certificate is an ideal start and leads to the expert level that

The basic training will use several online platforms (e.g. MOOCs) or training providers to train the interested in those basic topics. After passing, the basic certificate and proving the knowledge and skills in the needed units and element, participants will need to gain experience in this field by working within companies to apply this knowledge and/or deepen these skills. Later then, the advanced or more experienced knowledge and skills will be then trained in blended courses and onsite courses. In all of these advanced courses, the focus is clearly for the participants to show that they can apply the advanced methods or ideas within their companies or jobs. To verify this, the exams the have to consist of the typical exam parts, where the new gained knowledge and skills are checked but additionally there will be also the need of handed in project papers that are then double blind reviewed by experts from the ECQA network.

While spreading the word in Europe, which was the main and first idea of ECQA, more and more countries all over the world become interested in the European idea of job roles and skills cards for management jobs. Out of this, many possibilities arise, but also many challenges will be faced in the future. The ECQA developed all of their skills cards together with experts from the market and universities from several European countries. This is important to get the idea and viewpoint, how the particular job role is understood in different European countries. As much as we nowadays understand the different aspects of different cultures and approaches in Europe it will be a challenge if we try to apply these skills and ideas to worldwide customers. We have a solid knowledge e.g. how innovation management is to be understood and handled in Europe, as well what a innovation manager should know and how to apply the proper skills to this position. If we now try to export this skill set e.g. to south America, China, Japan this will fail. It will not fail because the idea and the product is not good, it will fail because it has to reflect the different cultures, the different approaches and the different management styles in different regions worldwide. One solution might be the current idea that ECQA is using already: ECQA tries to check for experts in the different fields of ECQA job roles in various regions and countries all over the world and invites them into the discussion to exchange the view of the current European situation with the situation in their region/country. This enables the generated community to enlarge the idea of a European wide certificate into a worldwide view. Now it is hard to tell if there will be a European management certificate that differs from the Asian certificate, which differs from the African certificate… Or if we will agree on a basic worldwide certificate and add regional modules upon this, so people that are working in china and Europe might pick those two regional modules the deal with the differences of cultures and management aspects.

ECQA is eager to enlarge the community and by this the knowledge and understanding of different management roles worldwide by using communication tools like Facebook, LinkedIn, Xing, discussion forum and many other Web 2.0 tools. Web 2.0 tools today provide multiple possibilities for companies to gain new ideas, new customers and provide good customer relationship management [11]. As much as it is appreciated to meet friends and partners in real, time and cost limit this to a certain amount. ECQA strongly supports the idea of the open innovation approach and to exchange ideas with experts from all over the world by inviting them into our community and discuss on several, if not all, of our topics. This helps us to get the best product that helps people to understand the different aspects of their jobs better and additionally shows their skills and knowledge certified to companies all over the world.

3 EUROPEAN LIFELONG LEARNING PLATFORM STRATEGY

Once the ECQA [1] was founded and the exam procedures were in place the next problem to solve was to allow a smooth integration of a critical mass of LLP projects into the platform. The EU LLP program finances a few hundred VET projects per year. The strategy was to empower a selected set of these projects to structure the skills and learning results into ECQA standards and integrate them into the platform.

The partnership of ECQA decided that we cannot accept every LLP project or new job role. Firstly,

they must satisfy the ECQA standards and quality criteria and secondly they should relate to a specific cluster. So it was decided that in the first years of rollout (2008 – 2014) we especially support the sector of IT and services.

To integrate LLP results ECQA demands that skills are described using European standards for skills descriptions. This means that each profession to be accepted needs to create a standard skills set. ECQA reviews the compliance of these quality rules.

With the growing number of collaborating industry training initiatives and LLP projects it was necessary to automate the import of new job roles into the system. Thus a further new functionally added in 2011 was the import and export functions allowing to enter new skills sets and exam questions automatically. This requires that Job Role Committees use a standard Excel template for describing the exam questions.

The Excel includes a macro which allows you to check that syntax and semantics of your test questions are correct.

Job Role Committees reworking the test questions annually can also export the test questions pool into this Excel format, rework the content, check the syntax and semantics with the macro and import again.

QuestionId	Correct?	Question	Possible answers		
		She/he understands that if conflict is not controlled in an innovative environment it can lead to long-term inefficiencies			
592	B,C,D	Which of the following situations would a manager utilise in h	1565	A	Status or power differences
			1566	B	Ensure innovations are within the mas
			1567	C	Do not change team members without
			1568	D	Never place staff on opposity sides of
593	A,B,C	Accepting that conflict can be very distructive what practices w	1569	A	Expand resource base quickly
			1570	B	Set challenging but attainable goals
			1571	C	Never exceed the groups conflict resol
			1572	D	High conflict resolution situations are l
		She/he has gained a thorough understanding of the unterlying factos that help diffuse conflict situations			
QuestionId	Correct?	Question	Possible answers		
594	A,B	In normal circumstances which of the following would be effe	1573	A	Seek out natural leaders
			1574	B	Delegate as much as possible
			1575	C	Become more autocratic
			1576	D	Ignore non-verbal cues
595	B,C	The way forward out of conflict resolution situations would be	1577	A	Set impossible deadlines
			1578	B	Obtain everyones point of view
			1579	C	Situations of serious conflict are best c
			1580	D	Take no action

Figure 2: Import / Export of Test Questions Pool

Some of the universities who are ECQA members decided to not use the multiple choice tests but to extend the system to support the APL (Accreditation of Prior Learning) procedures in the industry-university partnerships for education.

Thus a further new developed software function is the assessment of prior learning function. Here a panel of advisers (assessor of skills) can review existing knowledge of a person and assess a specific skills element to be fulfilled. In this case the exam is only to be done for skills elements where the assessment did not show coverage of the skills.

4 FUNCTIONS OF THE ECQA PLATFORM

The ECQA platform allows people from the work place to attend online skills portals, receive training,

do exams, and receive a certification [2],[3],[4],[8], [11].

With one login people can register for one or more job roles. When they register for the job role they can browse the skills and do a self-assessment.

The learning process (see Fig.3) is based on the following steps:

1. Self study: Attendees have access to skills and leraning portals and receive multimedia lectures, student notes and slides.

2. Exercises: Attendees meet trainers online and receive a "learning by doing" exercise and upload the results for discussion.

3. Group discussion and review: The results are reviewed by trainers and in online group discussions and are refined. Usually the exercises base on examples of the attendees' own organisation so that results of the course can be directly implemented and be useful for the participating organisation.
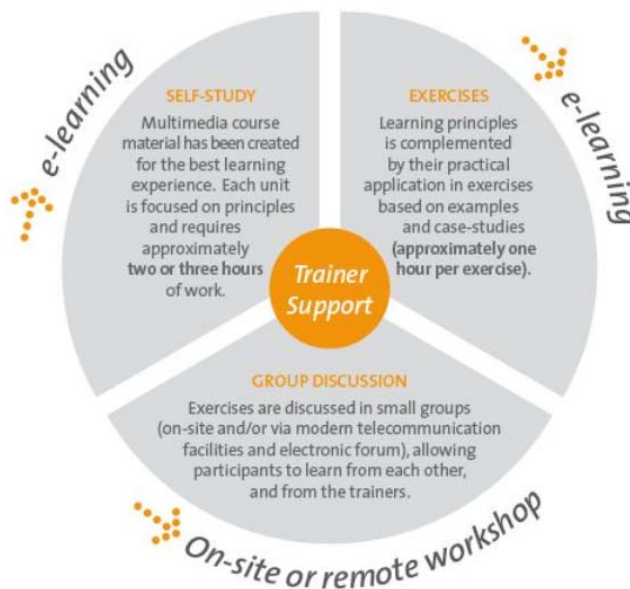

Figure 3: ECQA Learning Process

The experience shows that this type of training is successful because

- Attendees can use their own examples and elaborate them in the course

- Attendees enter a kind of interactive learning environment where the whole team reviews each other's work and the results become reviewed and have a practical value

- In an online course the multimedia lectures can be attended asynchronously (when the attendees have time) and only the group meetings need to be coordinated.

- The uploaded results can be used as learning evidences for the certificates to be issued.

Figure 4: Example Extract from an Integrated E-Learning Course

Figure 5: Example Extract from an exercise Review

5 ECQA and Recent initiatives and Outlook

In the project AQUA (EAC-2012-063) Automotive SPICE (ISO 15504), Functional Safety (ISO 26262) and Lean Six Sigma experts collaborate. The experts identified an architecture of core elements where all three approaches fit together and where a holistic view about improvement is needed. The Automotive Clusters from Austria, Slovenia, and Czech Republic are partners and roll out this knowledge in pilot courses to the industry. ECQA certifies these skills in collaboration with the Automotive clusters.

INTACS (www.intacs.info, International Assessor Certification Schema) is an international association which manages the education and certification of ISO 15504 assessors worldwide. It maintains a pool of many hundred principal, competent and provisional ISO 15504 assessors. From July 2014 onwards ECQA has been appointed the examiner and certifier for ISO 15504 assessors (as certification partner of INTACS).

In the I2E (Idea 2 Enterprise, 2012-1-CZ1-LEO05-09679, EU LLP program) the innovation manager skills will be implemented at school level (Gymnasium) and the first time an ECQA certificate will be jointly implemented between ECQA and schools.

In the LEADSUS project a new skills set for sustainability management is developed and ECQA will be certifier for this new job role (2013-1-RO1-LEO05 – 28771, EU LLP program).

In 2013 ECQA signed a contract with the Irish SW Testing board to act as the eam body and certifier for the software tester exams in Ireland.

The LSSA (www.lssa.eu) Lean Six Sigma Academy is a result of an EU project which uses ECQA as certifier and LSSH (Lean Six Sigma for Health care) is a new project where certificates will be issued for quality management skills in health care.

TRANSCERT (530940-LLP-2012-1-AT-KA3-KA3MP) develops a schema for using ECQA skills definition standards, test procedures and certificates for translators worldwide. Translator associations from Europe and other continents participate.

Recently ECQA started collaboration with universities from 10 countries to discuss and agree an ECQA for ECTS strategy.

The interplay and cross-fertilization of Higher Education (HE) and focused training in the work context has for long been a key objective in the development of a coherent and effective lifelong learning approach at the European level. In the context of HE, the Bologna process has transformed the landscape of trans-national mobility and cooperation. In the context of lifelong learning, different programmes have developed innovation and frameworks as the EQF have fostered trans-national recognition of qualifications. The European Credit system for Vocational Education and Training (ECVET) enables a better compatibility between the different vocational education and training (VET) systems in place across Europe and their qualifications. In HEI, the European Credit Transfer and Accumulation System (ECTS) plays an analogous role, and both ECVET and ECTS are based on the same foundations. These achievements have resulted in a framework enabling the interplay of lifelong learning and HE.

However, the governance and strategic directions in Higher Education Institutions (HEI) institutions are in most cases not effectively partnering with the relevant sectors of industry or doing so only for consulting in the design and evaluation of the curricula. This often results in a mismatch of the needs of enterprises and industry and the programs offered by HEI. Further, the systems for assessment in HEI are often very different from approaches to certification in industry. This points out to a need of devising and experimenting with new models for HEI and enterprise collaboration in aligning and even intertwining their educational and training offerings that consider the standards and practices used in both contexts and allow transferring credits and qualifications between the two realms. This needs to be done also respecting the business models and quality approaches used in training for industry needs together with the attainment of the competencies required in HE degrees.

In the ECQA for ECTS strategy we extended ECQA based industry certificates to become mapped into university lecturing programs and thus offer both to attendees, an industry certificates and ECTS points at the same time.

This will empower also university graduates to have (like with Microsoft diploma) European industry certificates covered in their study as well.

6 Acknowledgment

7 Literature

1. European Union Lifelong Learning website, accessed on November 15th, 2011; http://ec.europa.eu/education/lifelong-learning-programme/

2. European Certification and Qualification Association, www.ecqa.org

3. R. Messnarz, From process improvement to learning organisations (p 287-294), Wiley SPIP Journal (Software Process Improvement in Practice), Volume 11 Issue 3 , Pages 213 - 335 (May/June 2006)

4. Richard Messnarz, Damjan Ekert, Michael Reiner and Gearoid O'Suilleabhain, Human resources based improvement strategies—the learning factor, Wiley SPIP Journal, Volume 13, Issue 3, (pages 355–362), July/August 2008

5. O'Keeffe, T., & D. Harrington, 2001. Learning to Learn: An Examination of Organisational Learning in Selected Irish Multinationals. Journal of European Industrial Training, MCB University Press, Vol. 25: Number 2/3/4

6. DTI - Department of Trade and Industry UK, British Standards for Occupational Qualification, National Vocational Qualification Standards and Levels

7. Gemünden H.G., T. Ritter, Inter-organisational Relationships and Networks, Journal of Business Research, 2001

8. R. Messnarz, et. al, Assessment Based Learning centers, in : Proceedings of the EuroSPI 2006 Conference, Joensuu, Finland, Oct 2006, also published in Wiley SPIP Proceeding in June 2007

9. ISO 17024, International Standardisation Organisation, Standard for the Certification of Persons

10. Knappe & Kracklauer, Verkaufschance Web 2.0, Wiesbaden, 2007, Ed. Dr. Th. Gabler

11. R. Messnarz, M. Reiner, D. Ekert, Europe wide industry certification using standard procedures based on ISO 17024, in: Technologies Applied to Electronics Teaching (TAEE), 2012, IEEE Compurter Society, June 2012

12. Pavol Krempasky, EACEA, European Union LLP Prpogram, Draft Brochure of Good Practice, http://eacea.ec.europa.eu/llp/events/2014/documents/kikoff-lonardo-grundtvig-ka4/ka4-1-pavol-krempasky-&-oscar-casamian-marco.pdf

Standardization of Software Development in Project Practices in CIS-region: Research Results & Resume

Denis Pashchenko[1], Andrey Blinov[1]

[1] *Financial University, Moscow, Russia*
denpas@rambler.ru; aoblinov@mail.ru

Abstract

This article is dedicated to studying the subject and best practices of improving software development at the level of project. All the illations & conclusions are based on the 3 rounds Delphi Study in CIS region, organized by the authors at the period since September 9 till December 18, 2013 with 21 Russian experts from software companies of the CIS region. The Delphi study was focused on problems of standardization of software development in project practices, on engagement of project teams to accept changes and on perspectives of standardization and certification at a company. This article is an overview of mentioned issues on the level of the project.

Keywords

Changes implementation management, software development, organizational resistance in project team.

Introduction

Comfortable micro-climate and personal relationships at a software project team are the key factors of successful long-term cooperation, which is a "must" for software product delivery or for long-lasting program of custom development projects at a single company's account. Meanwhile rapid changes in production processes, even if caused by reasonable factors, may damage internal relationships or decrease the authority of project manager [1]. Project teammate's resistance expressed directly or indirectly may become a serious problem for changes implementation and even for regular production of a software product.

In CIS region project teams typically use on one of the following approaches in their production processes [2]:

- Strict following company's standards (in case of if production processes on the level of the whole company are mature enough);
- Mixing of practices and methods with mostly specified by the current project/account manager;
- Following the customer's methodology (in case of custom development / out-source model or playing as a links in a chain of big business process with tough limits for participants);

But there are still a lot of reasons for changes even in stable project practices on different levels of deepness. Sometimes it's a kind of tailoring of the whole company process model; sometimes it is temporary deviation from the rules which aims short-term increase of one of the key project parameters: budgeting, scheduling or product quality. At any case it happens all the time and requires rapid and correct management efforts from the project manager and the supporting project office staff.

This study is observing additional aspects of change management in production at the level of project, including:

- Role of the project manager in changes implementation;
- Comparing the main goals of project with goals of implementing changes;
- Typical approach in overcoming the resistance of the project teams.

1. *Research method and process*

Research was conducted at the period since September 9 till December 18, 2013 by 3 round of Delphi study. Twenty-one experts from Russian-speaking CIS-countries have taken part in this research. All experts have been playing the leading roles in their companies in project management and software development process quality: from project managers with team of 15+ people to software quality directors and the CEO of software companies with hundreds of software engineers.

On the first round, the panelists have sent their opinion and answers on the list of questions with 2 sections:

- special questions about experience and best practices in software production at the level of project;
- prognosis and opinions concerning 10 years perspective of software development process models and tools in CIS countries.

On the second round, the panelists have received a principal opinion of the experts' panel for all the questions, thus having a chance to correct their opinion or just give a comment.

On the third round, the panelists have given additional information and comments, which helped to improve Delphi study results and objectiveness.

Process of gathering experts' opinions is worth describing in details, as well as generalization of the results in the form of ranked lists and bar/pie charts.

When the responses collected have been analyzed during the first round, for each question the dominant (principal) opinion was selected – to become general consensus of the panel.

In the round #2 the responses of each expert have been compared with the principal opinion of the panel to get opportunity to be changed or commented by an expert.

As a result, for every multiple-choice question was obtained a ranked list with the dominant response in the beginning, and for questions with one possible embodiment of the response building an answers' chart has become possible.

Such process helps to receive the overall panel's opinion and to further develop the methods and recommendations.

The following table contains the numbers of active experts for each of the study's round:

	Round 1	Round 2	Round 3
Active experts	21	16	21
Percent of active experts	100%	76%	100%

Table 1. Activity of experts for rounds of the Delphi study.

In round 2 we have faced with obvious decrease of expert's activity.

The following charts show different information about experts, their experience and geographical locations.

Presented experience is usually most relevant for the same type of IT-companies. Types of IT-companies were present in Delphi Panel in the following ratio:
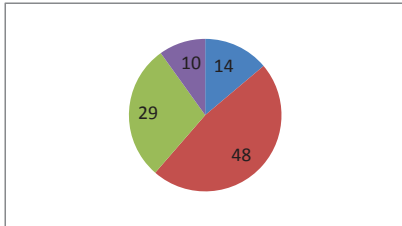
Figure 1. Expert's experience by the types of IT companies

- 10% of the experts with experience at non IT-companies with in-house development;
- 14% of the experts with experience at software vendors (ISV);
- 29% of the experts with experience at software system integrators;
- 48% of the experts with experience from tailor-made software companies (include out-sourcing model).

CIS-region geography of research is presented on the following diagram:
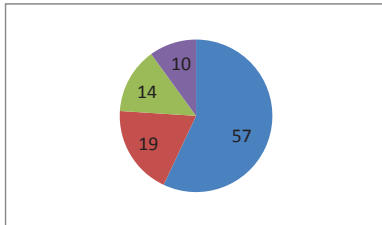
Figure 2. Expert's experience by the regions of CIS.

- 57% of the experts from Moscow and Sankt-Petersburg (Russia);
- 19% of the experts from other cities from Russia;
- 14% of the experts from Ukraine;
- 10% of the experts from other countries of CIS-region.

The experts were of the middle age group, associated in IT branch with the top of creativity and professional activity.
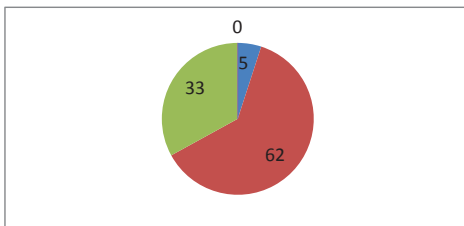
Figure 3. Experts' age groups

- 5% of the experts with age 20-29;
- 62% of the experts with age 30-39;
- 33% of the experts with age 40-49;
- 0% of the experts with age 50+.

Meanwhile, most of the experts have been working in software development area for considerable number of years, so there are no experts in panel working in IT sector for less than 5 years:
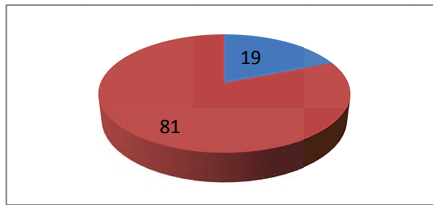
- 19% experts are working in software development area from 5 to 10 years;
- 81% experts are working in software development area for more than 10 years.

Figure 4. Expert's professional experience

2. *Results*

2.1 Change implementation in software development project teams

In this section experts have shared their opinions and experience about practices of changes implementation in software development models at the level of project.

This experience was related with significant changes, influencing all the project team members and core business processes. For example, this kind of change may be implementation of requirements management, or short scheduled releases in agile practice.

First of all, the has panel identified the major role of project manager in implementing changes at the level of project, and has agreed that the main activities of project manager are as follows:

- Set priorities for changes implementation and current goals of production (about 80% of experts agreed);
- Initiate or directly manage changes implementation (more 60% of experts agreed);
- Participate in the evaluation and analysis of changes implementation results (more 60% of experts agreed).

Also the panel has come to agreement that the formal planning of future changes implementation on the project's level is very important activity:
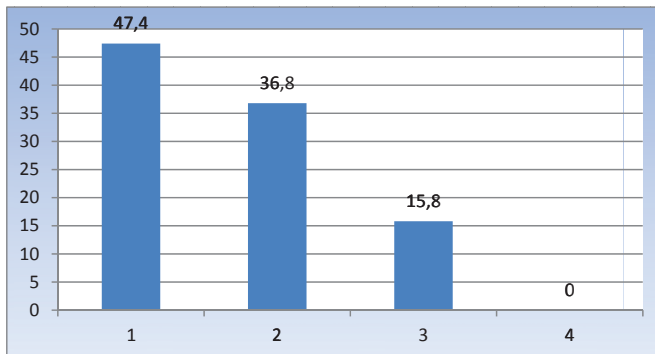
Figure 5. How important is having a formal planning stage in change implementation of software development on the level of project?
1 –Very important; 2 – Some importance; 3 –Not important; 4 – Damage main goals of project.

All types of process changes in software development that have been started with planning stageare receiving more support from the team side, and thus give more chances to collect feedback from the involved engineers on early stages.

Meanwhile formal planning allowed considering more risks of future changes before it goes LIVE and may provide with formal metrics of production process after its improvement.

Also the experts have agreed that such kind of change implementation planning should be document-ed in primary project plan of software delivery. Of course, for some CIS companies it has negative aspects:

- Customer/ project office may notice in project plan, how changes may become reasons for product delivery delays;
- Complexity of project plan is increasing.
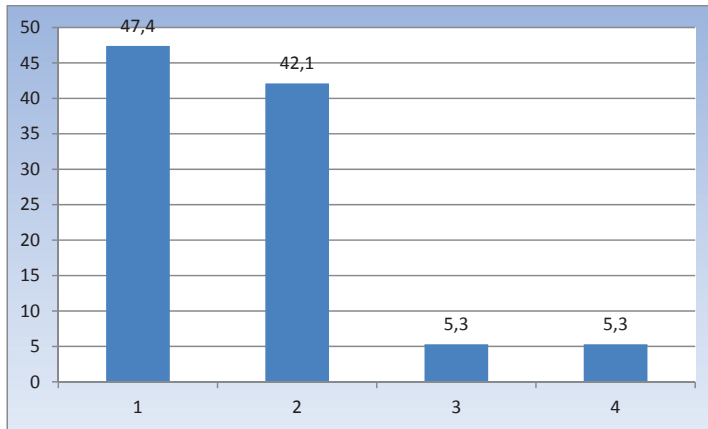
Figure. 6. How important is considering and documenting of change implementation in the primary pro-ject plan of software delivery?
1 –Very important; 2 –Some of importance; 3 –Not important; 4 –Do not put it in a primary plan.

Experts have identified two typical reasons of changes in production process on the project's level:

- Objective reasons for changing (about 90% of panel) related with economic issues;
- Following to external requirements from auditors, customers, market regulations (about 40% of panel).

Meanwhile, also experts added some special cases:

- Not reasonable instructions from management side;
- Following the strict corporate standards;
- Needs of effectiveness in one project/program with lack of processes' maturity at the company level.

Experts have compared the majority of changes implementation goals with the majority of primary project goals, and agreed that primary project goals (like software delivery in schedule or level of soft-ware quality) almost always have high level of priority.
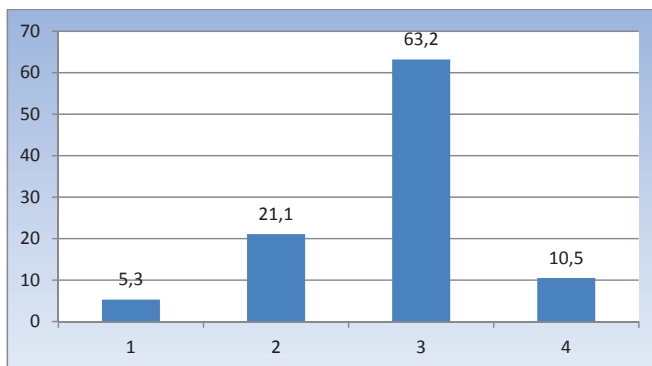
Figure 7. How often goals of changes in production processes could be major then current software de-livery goals?
1 –Very often; 2 – Often; 3 –Seldom; 4 –Never.

The innovators should consider this common opinion of experts at the stage of change implementation planning. Considering this fact one may save time and efforts in overcoming of the internal resistance, related with changes implementation in production processes.

Another important aspect of changes implementation are the recurring problems, related with all the stakeholders (like customers or project sponsor) and project team. Experts have identified two main problems:

- The contradiction (and conflicts) between the goals of the project and the objectives of the im-plementing changes (73% of panel had met this problem in their practice);
- Organizational resistance in project team (more than half of panel's experts had met);

Experts have concluded, that the reasons of such kind of problems are:

- Low involvement of team in changes;
- Lack of advocacy activities;
- Sense and goals of changes are unclear for software engineers.

Of course, it leads to following the changes "pro forma" without understanding of their sense and of modification of production processes coming from these changes.

In software companies of the CIS region changes at the level of project in production processes are often related with internal crisis. The panel's opinion is as wollows in the next Figure:
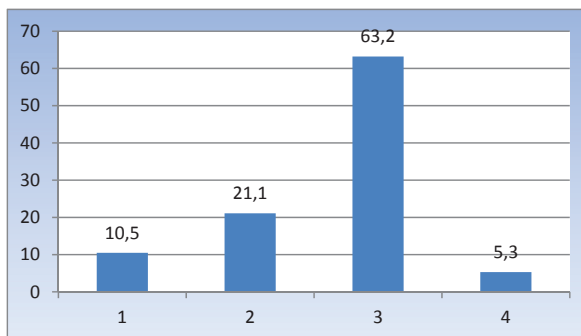
Figure 8. Does the change implementation in the time of project's crisis increases product quality and reduce the timing of the release?
1 – Almost always does; 2 – Often does; 3 – Leads to contradictory situations; 4 – Damage project's pa-rameters.

 Actually the part of expert considers that change implementation in production processes is the key approach in crisis management despite of all fears or risks.

Of course, such kind of changes should be performed under review of the project manager and the project office, and should be flexible at each stage of implementation.

The panel has identified a set of effective arrangements for overcoming of one of the most well-known problems in change implementation – organizational resistance in project team:

1. Involving resisting employees in the process of implementing changes (is met in practice of more than 90% of experts);
2. Positive motivation to work on the new rules (was met by more than 60% of experts);

Of course, both methods require some emotional and time resources from management.

Also the panel has come to a set of effective approaches, which may help in consolidation of new production processes in software delivery team (order by decreasing the popularity):

- Audit and attention from the project manager side;
- Encourage to use the new practices;
- Control from the side of the co-production units;
- Documentation in project standards.

Though documenting the changes in the project standards has not become the most popular method in innovation's consolidation according to the experts, however, this step is one of the important and leads to initiating positive changes in other projects or at the company.

Unfortunately, the panel has declined the important role of external consultants in software project teams while changes are implementing:
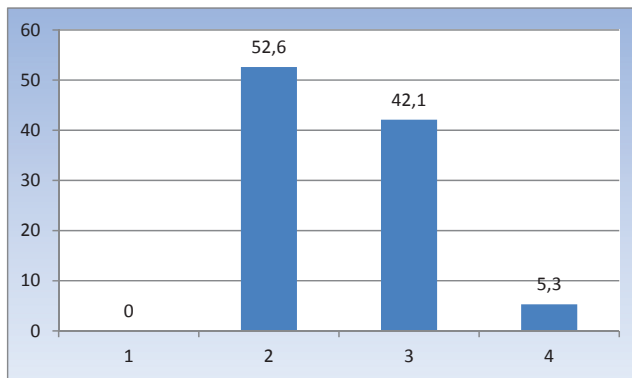
**Figure 9. How is important to involve external consultants for change implementation
in software production on the level of project?
1 – Important; 2 – Important on some stages (trainings, audits, etc); 3- Not important; 4 – Damage the
process.**

2.2 Perspectives of future development for standardization practices

This section presents the consolidated opinions related to the prospects of development of well-known standardized models (CMMI, ISO,etc), certification practices for software companies and the industrial automation tools.

The panel believes that the number of certified IT- companies with standardized production process models in the CIS will increase slightly within next 10 years (both - in absolute and in percentage values). Among the reasons for this trend are not only objective reasons, such as reducing costs or the desire to meet the expectations of final customers, but also the process of absorption of small/medium companies by big players and growth of the medium-sized companies. Experts have agreed that production process standardization in future will still take a lot of time and efforts; they do not see trends of its reducing.

The panel had faced with difficulty with definition of the role of the modern hybrid models in the CIS-region over the next decade. According to some experts, the hybrid model will take its considerable

and even biggest niche, while according to others, it will be replaced by new-fangled approaches, without having a chance to get a considerable affection and popularity among software developers. In general, experts are rather addicted to a more thorough and original adaptation in software companies of the current dominant methodologies - interactive or flexible (depending on the company's history or the type of projects).

Experts have agreed that professional certificates of software institutes and communities do not give a real competitive advantage on CIS market. Real choice of customers depends on other considerations, but certificates may act as formal criteria in tenders or may be used in HR / marketing policy. Rapid growth of internal CIS IT market at last 10 years has decreased the real value of official certificates, which are probably more interesting for EU and USA customers, than for Russian or Ukrainian big customers.

In terms of automating software development processes experts believe that the current set of tools designed for different tasks and methodologies (like IBM, Oracle, Atlassian) is sufficient and plays an essential role. In the CIS countries, automation of software development has a "developing" character, when most companies are still *going* to automate the full cycle of development.

Meanwhile the current software production automation tools can be highly customized by the users themselves [3], they help to achieve the expectations of the developers and managers in software companies.

3 Conclusions & Recommendations

Experts' panel has recommended formal planning for all the significant changes in production practices at the project level with documenting of their results in software product delivery plan. The role of project manager is very important here; it means that even if a PM is not initiator of the changes, he should strongly support them and should be deeply involved in their implementation.

Some basic problems with changes implementation may be considered in main risk management plan. Experts define key issues:

- The contradiction (and conflicts) between the goals of the project and the objectives of the implementing changes;
- Organizational resistance in project team.

Also the panel has recommended managing the latter risk using the following approaches:

1. Involving resisting employees in the process of implementing changes;
2. Positive motivation to work according to the new rules.

Authors would recommend constructing several additional communication channels in project management area: horizontal experience exchange in change implementation between project managers and vertical support line from project management office (PMO) to some of the managers. Project managers usually have brilliant skills in software team/product development, but changes implementation could be not so easy for some of them [4].

PMO and SEPG teammates should understand that primary goals of project (for example, product delivery) always would be at the first priority for project managers. It means that personal motivation in changes Implementation should be strongly related with primary goal of project. And vice versa changes, leading to decrease of success chances of the project, might wake up a serious resistance of both the project manager and the project team. Such kind of organizational resistance would be hard to be overcome without significant losses.

Authors would recommend to innovators and project managers to track relations between changes and primary goals of each project on early stages with full objectivity and transparence.

Also the authors identify an additional factor which may help in change implementation – waiting favorable external conditions for starting changes in a project [5]. It may save a lot of efforts in overcoming organizational resistance in project teams.

Experts and authors cannot recommend implementing changes at a project in the state of crisis management. Although crisis management in common cases means significant changes in production processes this path should be done accurately and consistently [6].

One of the key problems in change implementation is consolidation of successful results [7]. The panel recommends:

- Audit and attention from the project manager side;
- To encourage the usage of new practices.

The research conducted also shows the key problems in process improvement on the level of project. The role of project manager and his personal attitude to change implementation have a big value. Correct impacts of implementing changes on primary goals of project may become a strong factor in its successful implementation. Panel responses, especially in consensus opinions, are demonstrating needs of systematic approach and thorough risk management in changes implementation on the level of project in production processes.

Contact Us
For any details related with this research please contact Mr. Denis Pashchenko via e-mail: denpas@rambler.ru

4 Literature

1   DeCarlo D.: eXtreme Project Management: Using Leadership and Tools to Deliver Value in the Face of Volatility, Jossey-Boss A Whiley Imprint, 2004
2   Gergert D., Artemiev D.: Methodic features in project management in in a volatile external environment (Методические особенности управления проектами в условиях нестабильной внешней среды), vol.3, pp 56-69. Management and Business Administration, Moscow, 2013
3   Base of knowledge – Atlassian Jira, http://www.teamlead.ru/pages/viewpage.action?pageId=3801115
4   Pashchenko D.: Changes in project management process model on example of Russian system integrator (Изменение процессной модели управления проектами на примере российского системного интегратора). Intellect. Innovations. Investments, Ufa, 2013, №4.
5   Pashchenko D.: Features of organizational changes in Russian software companies (Особенности реализации проектов организационных изменений в российской софтверной компании), Program & Project Management №1,Moscow, 2014 pp.22-32
6   Collyer S.: Project Management Approaches for Dynamic Environments, https://espace.library.uq.edu.au/eserv/UQ:175094/Project_Management.pdf
7   Thomsett R., Radical Project Management, Prentice Hall PTR, 2002

[24 April 2014]

5 Author CVs

Denis Pashchenko

Engineer in Software Development (RGTU, Moscow, 2006);
PhD in System Analysis (Moscow, 2010);
General MBA (Moscow Business School, 2014);
Doctoral student in Financial University (www.fa.ru , Moscow, now)
Country Manager in Russia & CIS in IND Group (www.indgroup.eu)

Research interests: software production, project of changes implementation in software companies and organization resistance.

Andrey Blinov

Specialist in Management (Moscow, 1983);
PhD in Economy (Moscow, 1989);
Doctor in Economy (Moscow, 1994);
Professor (Moscow, 1997)
Corresponding Member of the Russian Academy of Natural Science (Moscow, 2010)
Professor in Financial University (www.fa.ru)

Research interests: modern small and medium business conditions in Russia, excellence in production, long-term influence rules on SME economy.

Agile Adoption: Developing projects in parallel with Agile and Traditional life-cycles

Alexssander A Siqueira[1], Sheila Reinehr[2], Andreia Malucelli[3]

[1,2,3]Programa de Pós-Graduação em Informática (PPGIa)
Pontifícia Universidade Católica do Paraná (PUCPR)
Curitiba – PR – Brasil
[1]alexssander@yahoo.com, [2]sheila.reinehr@pucpr.br, [3]malu@ppgia.pucpr.br

Abstract

There are many reports and surveys that present successful cases of agile methods adoption to replace traditional plan-driven software development processes. To understand their success, it is necessary to consider some key factors such as: organization support, people skills, process flexibility, technical definitions and project type. This report describes an agile adoption experience that was developed within a Telecommunication company in Brazil that adopted an agile software development process as an alternative to their traditional plan-driven processes. The traditional ones were presenting several issues regarding productivity and quality. The agile adoption started with changes on the organization culture, knowledge sharing and an agile software development process definition that included the agile methods: Scrum, FDD and Pair-Programming. Then, a set of 10 projects were selected to be implemented in parallel using the company's waterfall based life-cycle and the custom agile software development process. It was possible to compare the processes performance using their results in terms of effort in hours, delivery delays and number of defects found and fixed during the integration test.

Keywords

Agile Development, Agile Adoption, Software Development, Process Improvement.

1 Introduction

In software business the pressure to continuously develop business processes in order to stay competitive is great. The productivity of companies is heavily founded on the effectiveness of their software development processes [1]. Telecommunication companies are operating on increasingly dynamic markets. The new business challenges reflect on their software development, which has to be able to respond faster to changing situations. Several Agile software development methods are proposed as an alternative to the traditional heavy weight approaches [2].

As more and more software projects engage agile methods, there are emerging patterns of success and failure. With growing adoption of agile methods, project managers increasingly need to understand the applicability to their projects and factors that drive key project performance characteristics [3].

2 Context and Life-cycles

The reported experience was developed within a telecommunication company in Brazil that offers services all over the country, such as phone calls, internet access and cable TV. This company has an enterprise of integrated systems that are updated and modified to support change requests from internal customers. Each of these change requests are managed as a single project (new customer services, special offers, regulatory and laws updates, etc) that are critical for sales and marketing initiatives.

The company had adopted in the past years many plan-driven life-cycles as the waterfall, spiral and a RUP based process. The main concerns are the difficulties to accept customer's change requests, a high number of software defects that are found in the integrated test and specially the projects costs that are impacted by rework generated during the development phase. The main company's software development process (that it is the company's traditional process reference) is a waterfall based life-cycle that has 5 simple phases: business analysis, solution design, build, integrated test and user acceptance test that are presented in the Figure 1.
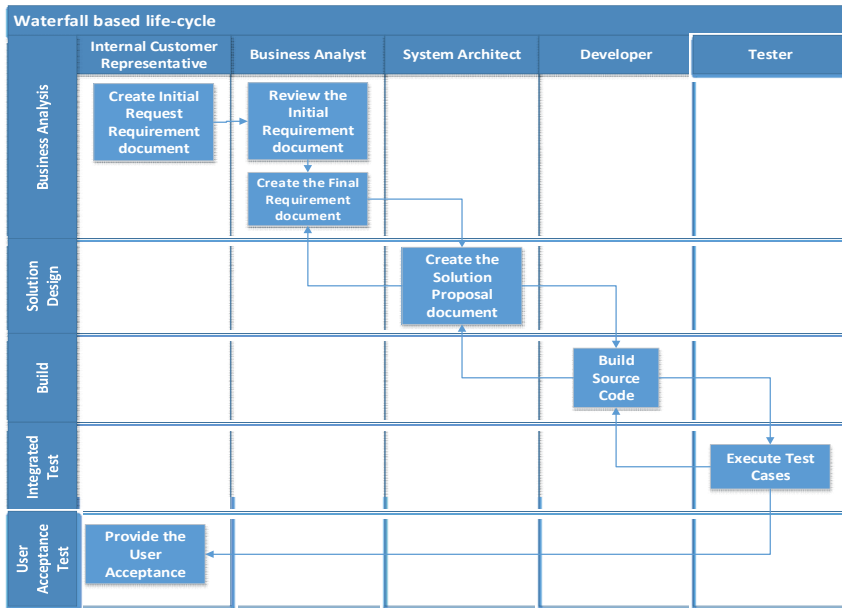
Figure 1: Waterfall based life-cycle

The development process starts with the internal customer representative writing the Initial Business Requirement document that contains the project's general scope that is described in functionalities and business constraints that must be implemented. The life-cycle is composed by the following phases' activities:

- Business Analysis: A business analyst reviews the Initial Business Requirement document to describe in details the new requirements in the Final Business Requirement document and identify any additional business constraint that must respected, other functionalities and projects that can be impacted by the change request. To complete the Final Business Requirement document, the business analyst inform the test cases that will be executed during the Test phase.

- Solution Design: A system architect reviews the Final Business Requirement document to describe the necessary changes in the involved components and databases. The architect output is the Solution Proposal document that will provide guidance to the developers during the Build's phase.

- Build: The developers will execute the necessary changes in the identified components and databases using the Solution Proposal document as reference. The updated source code is uploaded into a specific source code repository to further deployment in the test environment.

- Integrated Test: The project's source code is merged with other projects' source codes and deployed in the test environment. The test cases are executed as was previously informed in the Final Business Requirement document.

- User Acceptance Test: The internal customer representative checks if the delivered solution was developed as requested in the Initial Business Requirement document. After obtaining the user acceptance, the solution is deployed in the production environment for general availability.

The waterfall based life-cycle presents some issues during the project development that impact the development productivity (extra effort to projects completion, delivery delays) and quality (a high number of defects found during the integration test). The main issues are:

- Documentation: The communication channel to share information among the team members are the project's documentation. As the communication is very formal the requirements and the technical solution details are extensible documented. Even with the available documentation architects have problems to understand the project's requirements and developers have problems to understanding and implement the solutions described by the architects. The quality of the software produced is impacted by defects that are generated during the solution coding and implementation.

- Change Requests: Any minimal change or update in the project scope impacts the project development. The previous workflow phases must be re-executed to support the necessary changes in the documents and components source code. In this case, changes in the project scope impact the cost and the project delivery date that is usually postponed.

- User Acceptance: The business representatives are deeply involved in the beginning of the project when the Initial Business Requirement document is received and reviewed by the business analyst. But during the project development they are not. Many scope misunderstood are identified lately in the project life-cycle, avoiding projects to be delivered in the production environment and generating unnecessary rework.

- Concurrent Projects: During merge of the source code in the test environment, it is common to the project team figures out dependencies with other concurrent projects that are delivering or changing the same functionality or component. There is not a clear communication channel among the team members to share information about project's scope, solutions and issues.

Due to listed productivity and quality issues that the company's traditional software development processes have presented in the last years, the company was looking for an alternative software development process to replace the traditional ones. Agile methods are a departure from plan-driven traditional approaches, where the focus is on generating early releases of working software using collaborative techniques, code refactoring, and on-site customer involvement [4]. Research and surveys have shown that agile methodologies are an efficient way of producing software with significant advantages in production costs, time-to-market, complexity, and quality improvement over heavy-weight traditional methodologies [4][5][6].

2.1   Agile Methods

The term 'Agile Methods' has been around for more than a decade, while the underpinning concepts and most of the practices associated with agile software development have been around for much longer. In fact, there is still no complete agreement on what agile software development is, but certainly agile methods aim to answer a need to develop software quickly, in an environment of rapidly changing requirements. The use of iterative development is common to all agile methods and usually there are frequent releases to customers. Close (preferably onsite) collaboration with customers is encouraged and requirements change is an accepted, even welcome, part of the process [7].

As stated in the Agile Manifesto [8], the agile software development is based on a set of principles: Individuals and interactions over processes and tools; Working software over comprehensive documentation; Customer collaboration over contract negotiation; Responding to change over following a plan.

A host of methods, adhering to varying degrees to the tenets of the manifesto, appeared on the landscape. These include eXtreme programming (XP), Scrum, Lean Software Development, Feature-Driven Development (FDD), and Crystal methodologies, to name but a few. Broadly speaking, all these methods endeavored to address the core principles of the manifesto [9].

2.2   Agile Adoption

The transition from a plan-driven to an agile software development process affects not only the development team members, but also other teams, departments, and the management. Any new process will likely attract developers excited to try it while repelling those opposed to change [10].

In practice, few organizations are able, psychologically or technically, to take on agile development

approaches immediately and adopt them successfully over a short period. A full transition often takes a few years [11].

To reduce the transition impact, there are agile development frameworks that support the agile adoption [12]. These frameworks and good practices are based on a set of success factors such as: organization support, people skills, process flexibility, technical definitions and project type [13]. These factors were considered by the company to proceed with the agile adoption.

Firstly, the company's executive board members allowed the team to proceed with the experiment assigning idle resources (business analysts, architects, developers and tester that would be available for 2 months). The team was composed by high skilled professionals (experience with critical projects). Next, several knowledge sharing sessions were conducted to spread the agile principles and culture. Then, the team members worked together to define and document the agile software development process, that they would be adopting further. The process defined was composed by agile techniques as Scrum for activities development planning and control, FDD for extensible code testing and Pair-Programming to support knowledge transfer among developers and architects (applied when changing critical components or the design was not clear enough). Finally, a set of 10 web based projects were selected to be developed twice, using the existing waterfall based life-cycle and the agile process defined by the team.

In addition, some stakeholders from internal departments (marketing, finance and engineering) were involved in the agile transition. They were the projects owners which would assume the customer role.

2.3   The new Agile Software Development Process

The new agile software development process that was defined by the project team has an iterative incremental life-cycle that is presented in the Figure 2. In each iteration, all process phases' activities are executed resulting in a new software deliverable.
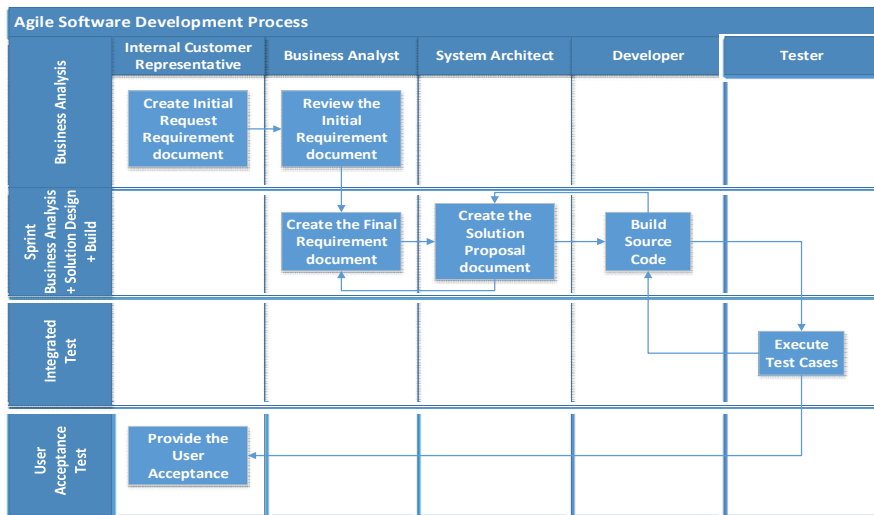
Figure 2 – Agile software development process

The agile software development process definition was based on the core XP practices [14], such as:

- Planning game: the iterations are planned and controlled using Scrum. The stories (activities) progress are visible for the project team and the internal customer representative. The iteration duration is 2 weeks. There is a planning meeting in the iteration beginning, daily meetings and a meeting in the iteration end to verify its completion and discuss about the main goals and issues found;

- Metaphor: the business requirements are described using a simple and direct language (customer friendly language);

- Small Releases: all iterations produce a software artifact that may be part of the architectural solution or a functional feature (working software);

- Simple Design: the solution proposal document has the fewest possible classes and methods. Just the minimal information to clarify the problem solution. The architects when specifying the solution, they define a set of technical stories that will be addressed to the developers in the planning game (Scrum meetings). In this case, these stories will be used as guidance to the developers to assure that important components or databases will be correctly modified;

- Tests: the source code development starts with the elaboration of a test case. The FDD agile method supports the developers activities;

- Pair-programming: this practice was applied just when a story involves a critical component change. Or in case of knowledge and experience sharing between two developers working with a complex implementation.

All team members attended the workshop session to contribute with the process definition that were documented and shared in a knowledge repository.

3 Developing projects

Due to a lack of new projects requests, part of the team's members would be idle for a period of 2 months. Then, developing projects using the traditional waterfall based life-cycle and the agile development process in parallel was an option to keep all team members engaged during the low demand period.

Before starting the projects development, the following assumptions were stated:

- Parallel development: the same project will be developed using the waterfall based life-cycle and the agile development process. The start date will be the same for both initiatives.

- Team member assignment: as the projects are implemented in parallel, the team members will be assigned for one of the projects implementation. It is important to balance the number of traditional and agile projects that each team member is assigned.

- Projects similarity: the selected projects involve technologies and platforms that the project team already have expertise.

3.1 Projects Characteristics

The projects were classified by the business analyst and the system architect regarding the project's characteristics as described below:

- Requirements: the project scope is represented by the number of functional requirements that are part of the project's scope and have to be implemented.

- Technologies: the number of technologies that will be involved in the project's solution can indicate how complex is the project proposed solution.

- Estimative: based on the team's metrics, it is possible to calculate an initial estimative of hours that will be necessary to complete the project development.

The Table 1 presents the list of projects after the business analyst and system architect classification.

Projects	Requirements	Technologies	Estimative (in hours)
Project 01	3	2	48
Project 02	22	3	124
Project 03	35	5	204
Project 04	38	4	184
Project 05	18	3	116
Project 06	29	4	172
Project 07	37	4	184
Project 08	25	4	156
Project 09	26	4	154
Project 10	23	4	150

Table 1: List of selected projects

After the projects selection and classification, the activities were distributed to the development team.

3.2   Experiment Parameters

As reported by Dyba and Dingsoyr [15] the most part of the comparative studies between agile and traditional models employed the number of produced source code lines as the productivity parameter. However, none of these studies had an appropriate recruitment strategy to ensure an unbiased comparison. In this case, the following parameters will be defined and registered for further analysis:

- Effort: the amount of hours that were expended to complete the project development. This value will be compared with the initial hours estimative;

- Delay: when the project's delivery day is not reached as previously planned, it is necessary to register the number of days that the project delivery was postponed;

- Quality: the number of defects that were found and fixed during the integrated test.

This work was motivated by several issues that impact the productivity and quality the company's traditional software development processes. In this case, the further analysis will be using these quantitative and qualitative parameters results to compare the projects development performance and quality.

4 Results Analysis

All 10 selected projects were successfully executed in 2 months. The experiment assumptions were respected and followed. The parameters value were registered as planned and the comparative study started analyzing the effort expended to develop the projects.

In terms of effort, the agile projects consumed more hours that the waterfall based life-cycle model process as shown in the Table 2. The additional effort expended on the agile projects was caused due the use of the pair-programming method. Some complex solutions were developed by 2 programmers at the same time.

Projects	Estimative (in hours)	Effort variance	
		Agile	Traditional
Project 01	48	0%	0%
Project 02	124	16%	0%
Project 03	204	11%	6%
Project 04	184	12%	7%
Project 05	116	18%	5%
Project 06	172	12%	5%
Project 07	184	6%	6%
Project 08	156	17%	5%
Project 09	154	6%	6%
Project 10	150	19%	4%
Effort variance average		12%	4%

Table 2: Projects effort variance

The projects delivery delay was minimal for the developed projects using the agile development process. There were 3 projects using the waterfall based process that had the delivery delayed (Projects 3, 4 and 9). Regarding the projects developed with the agile development process, just 2 projects were delayed (Projects 5 and 10).

The number of defects found and fixed during the integrated test was very different for both development process. There were 131 defects in the agile projects against 186 defects in the traditional development process (42% more defects) as presented in the Table 3.

Projects	Defects - Integration Test	
	Agile	Traditional
Project 01	13	13
Project 02	13	22
Project 03	15	27
Project 04	14	14
Project 05	11	21
Project 06	15	19
Project 07	12	20
Project 08	12	18
Project 09	12	10
Project 10	14	22
Total	131	186
Difference	42%	

Table 3: Number of defects found and fixed in the integration test phase

Analyzing the results, it is possible to affirm that the quality (number of defects) was improved by the use of agile methods FDD and Pair-programming. Applying these methods together, the amount of defects found and fixed during the integration test will be reduced, because the most part of the defects will be detected by the developers earlier (in the build phase).

Regarding the project management, the application of the Scrum method did not present a significant advantage against the traditional plan-drive development process. However, the Scrum technique allowed the team members to discuss the projects issues and progresses.

5 Conclusion

Firstly, the experiment reported in this paper was motivated by a set of issues that often impact the traditional plan-driven software development processes. Many works in the literature presented successful cases of agile methods adoption. In this case, the agile software development process is a good alternative to replace the company's traditional life-cycle models.

Then, as presented in the results analysis section, the quality improvement in terms of software defects reduction it was the main report achieved goal. The productivity improvement, which could be understood as the assertive quality software production, was also achieved on this experiment due to the defects number reduction.

Next, the company will adopt the agile software development process for other similar projects and reduce progressively the use of traditional software development processes. It is possible, to conduct more parallel developments to explore more agile methods, practices and productivity issues.

To conclude, the parallel projects development using two different software development process is an approach that helps company's managers to identified and mitigate the concerns of the agile methods adoption. This strategy's cost may be an issue for many companies, but represents an investment to avoid unsuccessful projects.

6 Literature

[1] Kukko, M., Helander N. and Virtanen, P. **Knowledge management in renewing software development processes**. Proceedings of the 41st Hawaii International Conference on System Sciences, 2008.

[2] Kähkönen, T. **Agile Methods for Large Organizations – Building Communities of Practice**. Nokia Corporation. Agile Conference, 2004.

[3] Coram, M., Bohner, S. **The Impact of Agile Methods on Software Project Management**. Proceeding of the 12th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems, 2005.

[4] Mahanti, A. **Challenges in Enterprise Adoption of Agile Methods - A Survey**. Journal of Computing and Information Technology, 2007.

[5] Mishra, D., Mishra, A. **Complex software project development: agile methods adoption**. Journal of Software Maintenance and Evolution, 2011.

[6] Hajjdiab, H., Shaima. T. A., **Agile adoption experience: A case study in the U.A.E**. IEEE 2nd International Conference on Software Engineering and Service Science, pg 31-34, 2011.

[7] Greer, D., Hamon, Y. **Agile Software Development**. Software Practice and Experience, 2011.

[8] Beck, K., et all. **Agile Manifesto**. Available on www.agilemanifesto.org, 2001.

[9] Dingsøyr, T., Nerur, S., Balijepally, V., Moe, N. B. **A decade of agile methodologies: Towards explaining agile software development**. Journal of Systems and Software 85, 1213-1221, 2012.

[10] Cohn, M. & Ford, D. **Introducing an agile process to an organization [software development]**. *Computer* **36**, 2003.

[11] Qumer, A., Henderson-Sellers, B. **A framework to support the evaluation, adoption and improvement of agile methods in practice**. The Journal of Systems and Software 81, pgs 1899tems , 2008.

[12] Sidky, A. Arthur, J.,Bohner, S. **A disciplined approach to adopting agile practices: the agile adoption framework**. The Journal of Innovations in Systems and Software Engineering, 2007.

[13] Chow, T., Cao, D. **A survey study of critical success factors in agile software projects**. Journal of Systems and Software, 2008.

[14] Beck, K. **Embracing Change with Extreme Programming.** Computer vol. 32, 1999.

[15] Dybå, T., Dingsøyr, T. **What Do We Know about Agile Software Development?** *IEEE Software* 26, 6-9, 2009.

7 Author CVs

Alexssander Siqueira

Has a Bachelor's degree in computer science and is Master's student in Informatics at the Pontifícia Universidade Católica do Paraná. Project manager at HSBC Global Technology Brazil. Has experience in computer science, software engineering, systems development, and information security.

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.

An SME Transition from Plan-Driven to Hybrid Project Management with Agile Software Development Methods

Stefan Biffl[a]   Richard Mordinyi[a]   Günter Raidl[b]
Heinrich Steininger[b]   Dietmar Winkler[a]

[a]Institute of Software Technology and Interactive Systems, CDL-Flex[1],
Vienna University of Technology
Favoritenstrasse 9/188, 1040 Vienna, Austria
<firstname>.<lastname>@tuwien.ac.at

[b] logi.cals GmbH
Europaplatz 7/1, 3100 St. Pölten, Austria
<firstname>.<lastname>@logicals.com

Abstract

In critical software development contexts, such as industrial software product development, plan-driven methods for project management are established to achieve well-defined goals. However, for new product development agile software development and management methods are attractive due to the flexibility they provide as new requirements or findings emerge.

In this paper we report on the process improvement experience of a 25-person small-to-medium enterprise (SME) with new software product development of programming tools for large-scale industrial automation systems. Plan-driven project management methods were well aligned to contracts with customers but not to the challenges of market-driven new product development. Therefore, the SME adopted agile methods and developed a hybrid project management approach with agile software development methods and the tool support to optimize the benefits from these process improvements. We discuss lessons learned from the case study, risks, and success factors regarding the transition from plan-driven to hybrid project management.

Major effects from the process improvement to hybrid project management with agile software development methods were (a) better awareness of development needs and progress on the team and management levels, (b) more efficient controlling of resources and cost, and (c) the innovative integration of research and development partners into agile sprint management.

Keywords

Project Management, Hybrid Project Management, Plan-Driven Development, Agile Software Development, Software Process Improvement, SME.

---

[1] Christian Doppler Laboratory "Software Engineering Integration for Flexible Automation Systems".

1   Introduction

Industrial software product development concerns often high-risk tasks that require high quality from the software development process. Plan-driven methods [6] are well suited to the goals and risks in this context as they provide a clear top-level plan and control. However, detailed plans may lose touch with real-world software development due to challenges from changing priorities based on new information from customers and research, or frequent changes to available resources.

In recent years software development teams have embraced agile methods [17], which promise less overhead than plan-driven methods and the necessary flexibility to react quickly and efficiently to emergent requirements, urgent fixes of quality issues, and changes regarding resources. However, agile methods can become risky if they are not implemented sufficiently, e.g., if project participants are not trained and agile means no real plan and minimal documentation only.

In many software developing small-to-medium-enterprise (SME) companies, project managers, and software developers want to benefit from the strengths of both plan-driven management and agile software development practices, like predictability and flexibility, but find it difficult to combine the approaches. If not done wisely, the combination of plan-driven management and agile software development practices can bring the limitations of both approaches into a project, making it hard to control. A key question is how to combine a plan-driven project management (PM) with agile software development practices to enhance benefits and mitigate risks from limitations.

In this paper we report on the process improvement experience of the 25-person SME, logi.cals GmbH (LCS). LCS conducts new software product development of programming tools in multi-disciplinary engineering environments for safety-critical and large-scale industrial automation systems, including a third of the world's medium and large hydro power plants and 75% of all busses in Europe. A typical project takes 20 to 30 person years with 10 to 15 in-house developers and changing teams at development partners in different locations. Development includes technology exploration, interfaces to research prototyping, early adopting development partners, and to sales/marketing.

Plan-driven software PM methods [1][5] were well aligned to contracts with customers, but not to the challenges of market-driven new product development [6][11] resulting in major challenges from visibility of actual progress and needs. Therefore, LCS adopted agile methods and developed a hybrid project-management approach with agile software development methods and the tool support to optimize the benefits from these process improvements. Major issues were how to organize the interaction between plan-driven and agile methods; and how to efficiently coordinate the needs and results of interacting sprint-driven development projects, research, and marketing.

Success criteria for LCS, similar to many other SMEs, are (a) for *marketing and management* good software delivery effectiveness, effort, cost to fulfill contracts with customers and provide competitive products on the market; (b) for *PM planning and control* effectiveness and efficiency; and (c) for *all project participants* overview on and awareness of needs and status of work. Major effects from the transition to hybrid PM with agile software development methods were: (a) better awareness of development needs and progress on the team and management levels, (b) more efficient controlling of resources and cost, and (c) the innovative integration of research and development partners into agile sprint management. We discuss lessons learned from the case study as well as risks and success factors regarding the transition from a plan-driven to a hybrid project management approach.

2   Related Work – Plan-Driven and Agile Development/Management

In their book "Balancing Agility and Discipline" Boehm and Turner [6] discuss in case studies the risks of plan-driven and agile approaches and propose a risk mitigation approach to choose a fitting approach for a given project context. The authors describe a measurement framework consisting of the following risk factors to indicate whether a plan-driven or an agile approach is more appropriate:

- *Size (number of personnel):* small projects are more amenable to agile approaches, while large projects need stronger structure and coordination.

- *Criticality (loss due to the impact of defects):* Software development projects in critical environments (danger to lose essential funds or lives) typically have to fulfill process and documentation

standards that go well with plan-driven methods, while agile approaches do well in uncritical environments.

- *Personnel (training):* The share of personnel with strong training and experience in relevant process and software development methods is related to the project complexity that the team is likely to conquer successfully.

- *Dynamism (% requirements-changes/month):* A high share of changing or newly emerging requirements needs high flexibility in software development that is more suitable for agile practices.

- *Culture (% Thriving on chaos vs. order):* The company and project culture needs to support the chosen approach, where thriving on order is better aligned to plan-driven methods and thriving on chaos fits better to agile approaches. This means that a change of the development and management paradigm has to be accompanied with measures to also change the company culture to avoid major risks from a mismatch between culture and development paradigm.

Boehm and Turner [6] discuss rather large case studies and divide a project into sub-projects, which are managed with either plan-driven or agile methods. However, it remains unclear how to combine these approaches and get to a stable top-level project process with predictable outcomes.

For new product development, researchers and software product developers have to cooperate for exploring new software functions in research prototypes and bringing selected software functions into industrial product development. Figure 1 illustrates five maturity levels of research concepts and prototypes towards quality-assured prototypes and industry products [23]. The management of the interface between research prototyping and software product development is often challenging with agile approaches, as the user stories for software functions for prototypes do not consider the quality needed in higher maturity levels and the integration effort in the product environment.
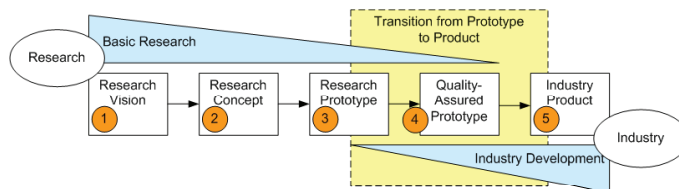
Figure 1: Maturity Levels from Research Prototypes to Industry Products [23].

Several studies investigate the advantages and disadvantages of plan-driven and agile processes, or compare them in general [20]. While plan-driven approaches like waterfall processes are still well recognized in companies [10], requirements cannot be managed well – a main reason for failure [15] – or are hard to learn and require a high level of knowledge [8] as it is the case with Rational Unified Process (RUP). In the context of agile methods, communication and feedback help to transfer knowledge from customer to developer more effectively [2], which has been noticed as valuable by developers [18], but lacks paying enough attention to architectural issues [18]. Studies on large-scale development ([9] and [16]) evaluate the effects of migrating from plan-driven to agile approaches. However, we have not found empirical studies investigating the combination of advantages using both approaches.

The development context of the case-study company LCS focuses on multi-disciplinary engineering projects [2][13], where heterogeneous groups of project participants have to collaborate. Different tool sets of project participants make collaborative engineering difficult because of strong limitations on tool interoperability and data exchange capabilities. Typically, applied tools provide less well-integrated and open interfaces for other tools to access their data than typical in business software engineering. Resulting challenges for software process improvement based on the industry standard VDI 3695 [21] from project experience in such an environment are: (a) how to organize PM in each project and how to standardize PM across engineering projects; (b) effective and efficient tool and data integration to collect experience for process improvement; and (c) ways to identify and elicit the relevant engineering knowledge for reuse across engineering teams in a company. The following case study reports experiences at LCS on balancing agile and plan-driven methods for software developments that may include research exploration and multi-disciplinary engineering environments.

3 Case Study Context and Research Questions

The case study summarizes how the SME Company LCS was motivated by a crisis project to reconsider their traditional use of plan-driven methods, to take up agile methods, and to develop a hybrid PM approach.

3.1   Project A – Crisis with a Plan-Driven Approach

Project A focused on the development of new product – designed for safety certification – involving 9 persons with a consolidated overall effort of 30 person years. Project A was set up strictly with plan-driven methods as these were well aligned with regulations of and contracts with the certification authority, development partners, and customers, who defined product features, time-to-market, and budget. In a CMMI assessment LCS would be likely to be rated at CMMI-levels 2 to 3. The plan-driven approach was found appropriate on the top level to ensure traceability of the process.

Unfortunately, a review after 60% of the project duration found the project targets actually moving away from completion. The analysis showed that the applied plan-driven management led to developers to not communicate development problems because of time pressure and rapidly increasing feature requests. As a consequence, the project culture penalized information sharing and learning, leading to information hiding even between developers. Also, risks from unrealistic expectations in a new product development project [11] were not addressed well. As a reaction, effective project tracking was suspended. Table 1 shows ex-post ratings of selected risks based on [6] for project A.

3.2   Introduction of Agile Practices to Project A

As a reaction to the software development crisis an external agile methods consultant helped to introduce basic agile practices [12][17] to improve visibility of the ongoing work and consolidated planning of emerging requirements. The agile practices included: agile roles, the Kanban process with user stories derived from overall project plan, agile rituals, such as daily standup meetings, for better transparency and communication in the development team. All relevant project participants were trained in agile practices, 15% of the staff gained over time sufficient expertise to fill leading roles.

These changes had to be agreed with the critical project customer, who feared not to know what he really would get with an agile setup. LCS argued that – following a plan-driven approach – the deliverables were agreed with the customer and had to be delivered. Therefore, the risk of development (how the results are achieved) remains at the software development organization. Finally, project A ended successfully although with a reduced set of results.

Lessons learned from project A regarding risks from plan-driven methods were:

- A plan-driven approach may lead to a disconnect between plan and real life, without contingency for this case.
- Plan-driven management can degrade to local best-effort management without good coordination.

Lessons learned from project A regarding risks from agile methods were:

- Kanban works well with tasks that can be defined as stories with a testable (delta) outcome.
- Some tasks are difficult to manage with Kanban, e.g., software documentation as these sprint deltas do not provide clearly specified customer benefits.
- With Kanban there is the risk to lose the view on the big picture and traction for important tasks. In the scope of Kanban, there is no systems-of-systems view, which is essential for parallel multi-disciplinary engineering projects.
- The product owner has to provide the market view, which may introduce new risks.

Based on the mainly good experiences with agile practices, the project team opted for continuing the use of these practices in the next project, project B.

3.3   Project B – Transition with External Management Consulting

Project B is a software research and development project for an engineering system in a systems-of-systems multi-disciplinary engineering environment to develop industrial production plants. Project B was the next large project at LCS after completing project A, planned for 3 years with yearly major deliverables to customers.

Based on the lessons learned in project A the project team set up a hybrid PM approach for project B. Supported by an IPMA[2]-certified consultant, a plan-driven top-level framework and agile sprints for software development, research, and marketing was implemented. According to the measurement framework of the risk factors in [6], (a) the size of the project, i.e., 20 persons, is suitable both for agile and plan-driven approaches; (b) the criticality of the resulting product (serious funds but not lives) is in favor of a plan-driven approach; (c) the personnel training has been improved to allow plan-driven and agile approaches; (d) dynamism, with an expected share of changing or newly emerging requirements of 5% to 20% per month is in favor of an agile approach; and (e) the culture, thriving on order or chaos, is balanced to allow both plan-driven and agile approaches.

Risk Items	Risk Ratings	
	Project A	**Project B**
Environmental Risks		
E-Tech. Technology and certification process uncertainties.	3-4	3
E-Coord. Stakeholder diversity leading to conflict and misunderstandings.	2	2
E-SoS. Systems-of-systems environment making control more difficult.	1	2
Risks of using agile methods		
A-Scale. Scalability and criticality of the product.	3	2
A-YAGNI. Use of simple design that does not scale up.	1	1
A-Churn. Personnel turnover with loss of expert knowledge.	3	2
A-Skill. Not enough people skilled in agile methods.	3	1
Risks of using plan-driven methods		
P-Change. Rapid change	1	2
P-Speed. Need for rapid results	0	1
P-Emerge. Emergent requirements	1	3
P-Plan. Unrealistic planning, high planning uncertainty.	3-4	2-3
P-Skill. Not enough people skilled in plan-driven methods.	2	1
Risk rating scale: 0: Minimal risk; 1: Moderate risk; 2: Serious but manageable risk; 3: Very serious but manageable risk; 4: Show stopper risk.		

Table 1: List of Risks and Risk Exposure according to Boehm and Turner [6].

Table 1 presents ex-post ratings of selected risks based on [6] for project B. Risk ratings are overall more favorable than for project A, also due to the work with external expert consultants. Therefore, the goal was to provide a strong plan-driven framework [5][22] to define the big picture of the project phases, while using agile methods for software development [12][17].

Major research questions in a hybrid setting were (a) how to organize the interaction between plan-driven and agile methods; and (b) how to efficiently coordinate the needs and results of interacting sprint-driven development projects, research, and marketing. Figure 2 illustrates the challenges in a hybrid plan-driven approach with agile methods: (1) in the plan-driven project structure plan (PSP) the agile sprints have to be represented for planning, coordination, controlling, and measurement of progress; (2) the process interface between PSP and sprints has to be defined; and (3) in the sprint backlog the needs coming from other work packages in the PSP have to be represented for effective coordination. The LCS team managed to address these issues in project B.

---

[2] International Project Management Association: http://ipma.ch/
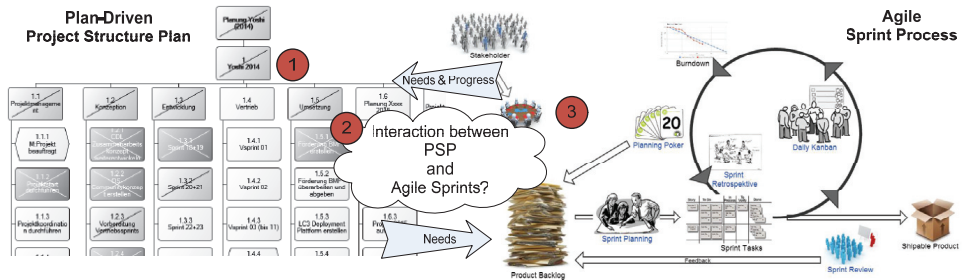
Figure 2: Project Structure Plan [6] and typical Scrum Process [17].

4 Hybrid Project Management Solution and Results in Project B

This section summarizes the "hybrid project management" approach and presents results in project B from more than a year of experience including successful shipping of selected components to clients. Core goal was to ship on time and with high quality. Core needs of project participants were: (a) High chance of sufficiently good product at the end of each time box; (b) Management visibility on progress and issues from developers; and (c) the tool set has to provide developers effectively and efficiently with the necessary information to plan their work, e.g., with a task management system. Based on core goals and needs of project participants the following core roles, processes, and artifacts were defined.

4.1   Core Roles, Processes, and Artifacts

Plan-driven project definition according to the International Project Management Association (IPMA) augmented with agile software development methods.

Core Roles: Project sponsor, project manager, product owner, Scrum master, and development team.

Planning Aspects

- *Definition of goals, milestones* on rough results (i.e., high-level and abstract product features), results, work packages (WPs), resources, analysis of the project environment and external interfaces, risk management, and controlling.

- Define the *big picture goals* in the project team.

- Application of a *project structure plan* (PSP) (see Figure 3) containing Scrum WPs. These WPs can be assigned to sprints or define dependencies between plan-driven WPs and Scrum WPs.

Coordination Aspects

- *Project goals and measurement criteria* get agreed between project sponsor and project team.

- The *product backlog* is the central interface between plan-driven WPs and agile sprints (see also Figure 2), including sprints for software development, research, and marketing (see Figure 3).

- *Sprint Planning.* At the start of a sprint, the most important stories in the backlog are estimated again for planning the current sprint. WP progress measures and actual effort reports provide the input to a traditional "earned value" analysis and to "social project controlling".

- *Visibility of needs:* Ensure that all tasks, which are hard to handle in a sprint, get planned and managed, initially by the project manager or a specified WP.

Controlling Aspects

- Controlling cycles need to be adjusted to milestones and sprints to refresh the big picture of the project in the project team. Measurement of progress: *burn down* of Scrum WPs and progress according to WP specification for plan-driven WPs.

- *Project controlling reporting* was initially planned to be conducted every 8 weeks. However, due to the very efficient automated availability of the controlling information, the project participants used this controlling information on a weekly basis, ensuring that plans fit well to the actual project activities. In critical situation the relevant information is available to react fast based on measured data.

Communication Mechanisms

- *Management* aspects include stand-up meetings twice a week and bi-weekly project team meeting (involving the project manager, project sponsor, and work package responsible persons).

- *Agile* communication mechanisms include daily and weekly meetings, sprint planning (involving the software development team, product owner, and the Scrum master), and Backlog grooming.

- *Sprint reviews* focus on (a) the progress of sprint results and (b) feedback on the needs for further concepts, training, and consulting.

Key artifacts as bridges between plan-driven management and agile practices

- *Feature map* perspectives include marketing experts, the product owner (innovative new product design features), and the development team.

- *Backlogs* collect ideas (idea backlog) and features (product backlog) with epics and stories (more or less specified).

- *Tool support* to automate visibility of coordination artifacts and reporting.

These roles, processes, and artifacts formed the key elements for (a) defining interactions between WPs; (b) balancing software development, and (c) establishing effective tool support.

4.2   Interaction of agile and non-agile work packages

A key question was how to combine plan-driven project management (PM) with agile software development methods to enhance the benefits and mitigate risks from limitations.

Legend: AP … work package; ES … development sprint; FS … research sprint; VS ... marketing sprint

Figure 3: Work Packages and Parallel Research & Development, and Marketing Sprints.

Figure 3 illustrates the solution approach in the hybrid PM approach used in project B at LCS:

(1) *Plan-Driven PM.* The top part of Figure 3 shows WPs according to plan-driven PM, their dependencies, resource usage, and progress. Examples are WPs for technology exploration, training, and concept development that provide input to software development tasks. All activities get represented as WPs/stories, including sprints. WPs can generate requirements for sprints. The product owner puts these requirements as stories into the product backlog.

(2) *Parallel Sprints.* The bottom part of Figure 3 presents individual sprints for software development to design and implement concepts. Software development has also technology spikes as input to plan-driven WPs. As an innovation there are parallel sprints for software development, research prototypes, and marketing in order to simplify the communication of needs and results between the departments in a timely manner. Before the sprints were of different lengths and end dates leading to unnecessary delays in communication and re-planning.

(3) *Synchronization.* Needs coming up from sprint tasks that cannot well be worked on with agile methods get communicated to the PM and get planned in plan-driven WPs.

These interactions between agile and non-agile WPs ensured that all project-relevant needs get planned in the appropriate context.

4.3   Balancing the Software Development Process

A key need is to provide an overview for the management, marketing, product owner, and development team on the roadmap for product development options.

Figure 4: Feature Map to Balance Marketing Needs and Development Capabilities.

Figure 4 illustrates the "*Feature Map*" for negotiation and planning between the product owner and the development team, based on a snapshot of the current process status.

(1) The white and light grey circles represent marketing features that can become epics and stories for development. Circles can be just ideas or concrete needs with customer value priority. In addition, developers can identify dependencies between features and estimates development effort and risk.

(2) Basic features are planned for shipping to a customer.

(3) Dependencies between features are input to selecting a set of features for potential development paths. Colored areas represent different parts of the product.

(4) The backlog holds ideas that are candidates for the roadmap but not yet analyzed sufficiently. The product backlog is the central interface between WPs and sprints. In backlog grooming sessions the development team, the Scrum master, and the product owner refine new stories in the backlog and roughly estimate the complexity in story points.

Thus, the features map enables the project team to take informed decisions in updating the backlog.

4.4 Tool Support for Hybrid Project Management

The software tool set to support the integration of plan-driven and agile methods was essential to enable the overview in the team and efficient reporting without incurring a prohibitive overhead for administrative documentation. The tool infrastructure for coordination and collaboration [4] consists of the following components:

- *Tools for PM*, collaborative process support, and knowledge management via ticketing systems (e.g., Jira[3], Jira Agile[4]), overview on the burn down of WPs and of software development story points.

- *Corporate Wiki* (e.g., Confluence[5]) for documentation linked to tickets.

- *Continuous Integration and Build.* Permanent build, continuous integration and test servers (e.g., Jenkins[6]).

- *Review Tool Support* .Reviewing of change sets coming from continuous integration and test processes (e.g., Google Gerrit[7]). The product owner can see the review results as a progress/risk indicator.
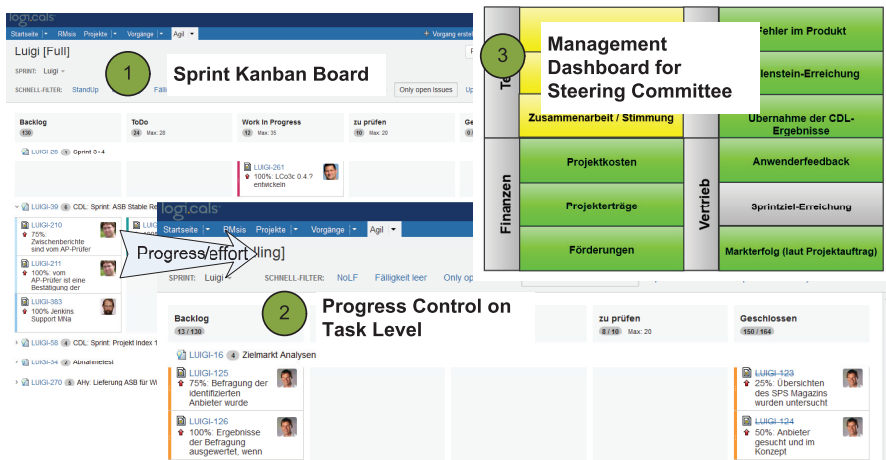

Figure 5: Kanban Boards (Jira Agile) for Sprints and Plan-Driven Work Packages.

Figure 5 shows screenshots from selected core systems for progress tracking:

(1) *Sprint Planning*. Kanban boards are used by the development team to *organize the work tasks in sprints*, showing the work load of resources and progress control for daily stand-up meetings.

---

[3]Jira: https://www.atlassian.com/en/software/jira
[4]Jira Agile: https://www.atlassian.com/en/software/jira/agile
[5]Confluence: https://www.atlassian.com/en/software/confluence
[6]Jenkins: http://jenkins-ci.org
[7]Google Gerrit: https://code.google.com/p/gerrit

(2) *Plan-Driven progress control.* Kanban boards also provide for the project management *progress control on task level* from sprints. As all PSP WPs are represented as work tasks (and sub-tasks), managers and software developers can also see the progress of plan-driven WPs in their system.

(3) *Management dashboard.* The data from the Kanban boards is aggregated in the bi-weekly project team meetings for controlling to allow the effective and efficient update of the management dashboard for reporting. Dashboard topics include the level of defects and achievement of milestones.

5 Discussion and Conclusions

After more than one year of using and fine-tuning the hybrid PM approach in project B, we can report that the method has been successfully applied at LCS regarding the following basic success criteria: (a) software delivery was effective to fulfill contracts with customers and provide competitive products to the market within the planned effort and time plan; (b) PM planning and control was effective and considerably more efficient than planned (compared to experiences from Project A); and (c) the overview on needs and status of work for all project participants enabled a very effective and flexible work culture.

Lessons learned

- Agile approaches need a strong framework for success in practice. The progress of sprint WPs has to be translated to the progress of plan-driven WPs as sprints have time-boxed goals. Well-defined milestones can avoid losing the overall perspective on progress goals.

- For time-boxed approaches the synchronization of time boxes is important to avoid loss of opportunities, e.g., synchronize the sprint time windows of all involved development teams, and also research and sales/marketing, who have strong goal interactions with software development.

- The product owner is a central role at the interface between plan-driven WPs and sprints and must not interfere with concrete sprint-internal tasks.

- Benefits from the integration of agile sprints with the plan-driven PM process: Controlling of cost, effort, and progress benefits from using the same methods in all parts of the project. There is an efficient overview at all times on the status and progress of sprints and (aggregated) WPs enabling more effective and flexible planning

Success factors

- A systematic, goal-oriented approach for priority setting mitigates the risk of jumping between ideas and not achieving overall goals.

- An efficient and tool-supported continuous integration and test process provides visibility of progress and ensures the required software product quality. At the time of transition from project A to project B, the agile PM tool set available on the software market was mature enough (and inexpensive) to be effectively adapted to the needs of the hybrid PM process.

- A feature network that provides planning data enables goal-oriented negotiation of the development strategy: Marketing features and value, dependencies between features, and developer estimates on effort/risk of features.

- Strong roles for sprint planning: product owner, Scrum master.

- Developers build their decisions on experiences from recent sprints.

- No extra work for developers for administration documentation. Tool environment supports developers in their work (and they see that); at the same time the tool environment provides the necessary data to management without extra cost and delay.

- Strong risk management mitigates issues that may jeopardize development effectiveness.

Conclusions

The SME company LCS has systematically developed a hybrid PM approach for software research and development projects. Major innovations in the approach are:

- Parallel coordinated sprints of software development", research, and marketing.

- Integrated and very efficient overview on all WPs in the hybrid PM due to a well-integrated tool set, customized to the hybrid PM needs and methods.

- Tool data integration needs are well addressed in the business software engineering context, but integration remains a major risk in multi-disciplinary engineering projects, which can be addressed with the logi.cals' Automation Service Bus® tool integration approach.

The risk framework by Boehm and Turner [6] was found useful to discuss the decisions taken in the PM transition project. However, it was found challenging to include different stakeholders, e.g., partners, developers, users, and researchers within one project team. In this context LCS found innovative ways to organize their work and manage risks, which are likely to help SMEs in comparable settings.

Future work

Future work will be the evaluation of the hybrid PM approach in research and development groups at a variety of research organizations and SMEs. While continuous integration and testing is well supported within one organization and the basis for progress monitoring, we see a need for better supporting continuous integration and test services across development organizations [4][7]. In addition, we see the need to improve quality assurance approaches of reusable information and the opportunity of using collective intelligence and social computing approaches [14] for eliciting and integrating engineering knowledge.

Acknowledgements

Literature

[1]   Abran A., Moore J.W.: "Guide to the Software Engineering Body of Knowledge", v3, IEEE, 2014.
[2]   Bahli B., Abou-Zeid E.S.: "The role of knowledge creation in adopting XP programming model: An empirical study", Proc. of Int. Conf. on Information and Communication Techn., 2005.
[3]   Biffl S., Moser T., Winkler D.: "Risk Assessment in Multi-Disciplinary (Software+) Engineering Projects", IJSEKE, Special Session on Risk Assessment, 21(2), 2011.
[4]   Biffl S., Schatten A.: "A platform for service-oriented integration of software engineering environments", Proc. of New Trends in Software Methodologies Conference, IOS Press, 2009.
[5]   Biffl S., Winkler D., Höhn R., Wetzel H.: "Software Process Improvement in Europe: Potential of the new V-Model XT and Research Issues", SPIP Journal, 11(3), pp.229-238, Wiley, 2006.
[6]   Boehm B., Turner R.: "Balancing Agility and Discipline – A Guide for the Perplexed", Addison Wesley, 2004.
[7]   Dösinger S., Mordinyi R., Biffl S.: "Communicating Continuous Integration Servers for Increasing Effectiveness of Automated Testing", Proc. of ASE, pp. 374-377, 2012.
[8]   Heijstek W., Chaudron M.R.V.: "Evaluating RUP Software Development Processes through Visualization on Effort Distribution", Proc. of Euromicro SEAA, pp 266-273, 2008.
[9]   Hirsch M.: "Moving from a plan driven culture to agile development", Proc. of ICSE, 2005.
[10] Johnson J.: "Build only the Features You Need", Proc. of XP, Keynote Speech, 2002.
[11] Kontio J.: "The Riskit Method for Software Risk Management", Univ. of Maryland, 1997.
[12] Ladas C.: "Scrumban - Essays on Kanban Systems for Lean Software Development", Modus Cooperandi Press, 2009.
[13] Moser T., Biffl S., Sunindyo W.D., Winkler D.: "Integrating Production Automation Expert Knowledge Across Engineering Domains", IJDST, Special Issue, 2(3), pp.88-103, 2011.
[14] Musil J., Schweda A., Winkler D., Biffl S.: "Improving Video Game Development: Facilitating Heterogeneous Team Collaboration Through Flexible Software Processes", EuroSPI, 2010.

[15] Petersen K., Wohlin C., Baca D.: "The Waterfall Model in large-scale Development", Proc. of PROFES, pp.386-400, 2009.
[16] Petersen K., Wohlin C.: "The effect of moving from a plan-driven to an incremental software development approach with agile practices", Empirical Software Eng. Journal, 15(6), 2010.
[17] Schwaber K.: "Agile Project Management with Scrum", Microsoft Press, 2004.
[18] Stephens M., Roserberg D.: "Extreme Programming refactored: The case against XP", Apress, Berkeley, 2003.
[19] Svensson H., Höst M.: "Introducing an agile process in a software maintenance and evolution organization", Proc. of CSMR, pp.256-264, 2005.
[20] Thomas M.: "It Project Sink or Swim", British Computer Society Review, 2001.
[21] VDI 3695, Part 2: "Engineering of industrial plants – evaluation and optimization; subject processes", Association of German Engineers, 2010.
[22] Winkler D., Biffl S.: "Improving Quality Assurance in Automation Systems Development Projects", Book Chapter, In: Quality Assurance and Management, INTEC Publishing, 2012.
[23] Winkler D., Mordinyi R., Biffl S.: "Research Prototypes versus Products: Lessons Learned from Software Development Processes in Research Projects", Proc. of EuroSPI, 2013.

Author CVs

Stefan Biffl

Stefan Biffl is an associate professor at Vienna University of Technology and head of the research laboratory "Software Engineering Integration for Flexible Automation Systems" (CDL-Flex). His main research interests are software and systems process improvement and value-based software engineering. For his detailed CV see http://qse.ifs.tuwien.ac.at/~biffl/

Richard Mordinyi

Richard Mordinyi is a PostDoc researcher at Vienna University of Technology in the research laboratory "Software Engineering Integration for Flexible Automation Systems" (CDL-Flex).His main research interests are service-oriented tool integration and project management. For his detailed CV see http://www.ifs.tuwien.ac.at/~mordinyi/

Günter Raidl

Günter Raidl is a project manager at logi.cals GmbH with over 15 years of development experience with software for safety- and mission-critical products. His main professional interests are project management practices and agile software development methods.

Heinrich Steininger

Heinrich Steininger is chief executive officer at logi.cals GmbH with over 25 years of development experience with software for safety- and mission-critical products. His main professional interests are customer benefits exploration and risk management.

Dietmar Winkler

Dietmar Winkler is a consultant on software process improvement, a lecturer at Vienna University of Technology, and researcher in the research laboratory "Software Engineering Integration for Flexible Automation Systems" (CDL-Flex). His main research interests are software and systems process improvement and empirical software engineering. For his detailed CV see qse.ifs.tuwien.ac.at/~winkler/