International Journal of Information Technologies and Systems Approach,
Volume 10, Issue 1, January-June 2017

Pre-publication Version

The Evolution of the ISO/IEC 29110 Set of Standards and Guides

Rory V. O'Connor
Lero - the Irish Software Research Centre, Dublin City University, Ireland
Rory.OConnor@dcu.ie

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

ABSTRACT

While the quality of products is a competitive advantage for very small software development organizations, the usage of Software and Systems Engineering standards amongst such very small organizations is extremely low. A key factor in the literature explaining this lack of quality standards adoption is the perception by small and very small organizations that such standards have been developed for large multi-national companies and not with small and very small organizations in mind. The ISO/IEC 29110 standard is unique amongst software and systems engineering standards, in that the working group (ISO/IEC JTC1/SC7 WG 24) mandated to develop a new standard approached industry to conduct a needs assessment and gather actual requirements for a new standard as part of the standards development process. This paper presents a historical perspective behind the development of the ISO/IEC 29110 systems and software engineering standard and its constituent components, including the rationale behind its development and the innovative design of implementation guides to assist very small companies in adopting the standards. Further this paper will present an overview of the various parts of the ISO/IEC 29110 family and briefly present the plans for the future evolution of this series of standards.

Keywords: Software Engineering, Systems Engineering, Standards, Software Process, ISO/IEC 29110, very small entities, profiles

1. INTRODUCTION

The purpose of this paper is to explain the rationale and justification for the development of a set of systems and software engineering process standards and guides developed specifically for very small entities and to chart the design and development of the ISO/IEC 29110 series of standards from the perspective of two of the standards authors. Very Small Entities (VSEs) are enterprises, organizations (e.g. public or non-profit), departments or projects having up to 25 people. In addition this paper will present the outline of the standard and its associated deployment supports. Finally this paper will present an overview of the global pilot implementation of ISO/IEC 29110.

The structure of this paper is as follows: Section 2 introduces background concepts and definitions such as the concept of Very Small Entities, Standards and their usage in small companies. Section 2 provides a high level historical summary of the evolution of the ISO/IEC 29110 standards from its initial inception to its current status. Section 4 will present the overall structure of the standard and the Management and Implementation Guides in particular. Section 5 will present a short discussion on the evolution of the standard to include Systems Engineering and section 6 presents a high-level summary the global efforts to implement the standard. Section 7 discussion standards and education, while section 8 discusses the future possible evolution of the standard.

2. BACKGROUND CONTEXT

Software development is a highly complex endeavor (Clarke et al, 2016) and for many small and very small software companies, implementing controls and structures to properly manage their software development activity is a major challenge (Larrucea et al, 2016). Administering software development in this way is usually achieved through the introduction of a software process. All software companies are not the same and vary according to factors including size, market sector, time in business, management style, product range and geographical location. For example, a software company operating in India may have a completely different set of operational problems when compared to a software company in Canada, Mexico or Ireland. Even within a single geographical area such as Ireland, the range of operational issues faced by a small local Irish-owned firm can be radically different to those affecting a multinational subsidiary. The fact that all companies are not the same raises important questions for those who develop software process and process improvement models. To be widely adopted by the software industry, any process or process improvement model should be capable of handling the differences in the operational contexts of the companies making up that industry. But process improvement models, though highly publicized and marketed, are far from being extensively deployed and their influence in the software industry therefore remains more at a theoretical than practical level (Coleman & O'Connor, 2008a).

In a time when software quality is a key to competitive advantage, the use of ISO/IEC systems and software engineering standards remains limited to a few of the most popular ones. Research shows that small and very small companies can find it difficult to relate ISO/IEC standards to their business needs and to justify the application of the standards to their business practices (Laporte et al, 2008) (O'Connor & Coleman, 2009). Most of these companies don't have the expertise or can't afford the resources - in number of employees, cost, and time - or see a net benefit in establishing software life-cycle processes. There is sometimes a disconnect between the short-term vision of the company, looking at what will keep it in business for another six months or so, and the long-term or mid-term benefits of gradually improving the ways the company can manage its software development and maintenance. A primary reason cited by many small software companies for this lack of adoption of software engineering standards, is the perception that they have been developed for large software companies and not with the small organization in mind (Coleman & O'Connor 2008b). To date the industrial reality is that Very Small Entities (VSEs) have limited ways to be recognized, by large organizations, as enterprises that produce quality software systems within budget and calendar in their domain and may therefore be cut off from some economic activities.

Accordingly there is a need to help such organizations understand and use the concepts, processes and practices proposed in the ISO/IEC JTC1/SC7's international software engineering standards. The recently published ISO/IEC 29110 standard "Lifecycle profiles for Very Small Entities" (ISO, 2011a) is aimed at addressing the issues identified above and addresses the specific needs of VSEs.

2.1 Very Small Entities

The definition of "Small" and "Very Small" Entities is challengingly ambiguous, as there is no commonly accepted definition of the terms. For example, the participants of the 1995 Capability Maturity Model (CMM) tailoring workshop (Ginsberg & Quinn, 1995) could not even agree on what "small" really meant. Subsequently in 1998 SEPG conference panel on the CMM and small projects (Hadden, 1998), small was defined as "*3-4 months in duration with 5 or fewer staff*". Johnson & Brodman (1998) define a small organization as "fewer than 50 software developers and a small project as fewer than 20 software developers". Another definition for VSE introduced by Laporte et al. (2006) as "*any IT services, organizations and projects with between 1 and 25 employees*".

To take a legalistic perspective the European Commission (2005) defines three levels of Small to Medium-sized Enterprise (SME) as being: Small to medium - "*employ fewer than 250 persons and which have an annual turnover not exceeding 50 million Euro, and/or an annual balance sheet total not exceeding 43 million Euro*"; Small - "*which employ fewer than 50 persons, and whose annual turnover or annual balance sheet total does not exceed 10 million Euro" and Micro - "which employ fewer than 10 persons and whose annual turnover and/or annual balance sheet total does not exceed EUR 2 million*".

To better understand the dichotomy between the definitions above it is necessary to examine the size of software companies operating in the market today. According to the Organization for Economic Co-operation and Development (OECD) SME and Entrepreneurship Outlook report (2015), "SMEs constitute the dominant form of business organization in all countries world-wide, accounting for over 95% and up to 99% of the business population depending on country"'. In Europe, for instance, 85% of the Information Technology (IT) sector's companies have 1 to 10 employees. In the context of indigenous Irish software firms 1.9% (10 companies), out of a total of 630 employed more than 100 people whilst 61% of the total employed 10 or fewer, with the average size of indigenous Irish software firms being about 16 employees (Coleman & O'Connor, 2008a). In Canada, the Montreal area was surveyed. It was found that 78% of software development enterprises have less than 25 employees and 50% have fewer than 10 employees (Laporte et al., 2006). In Brazil, small IT companies (companies with less than 50 employees) represent about 70% of the total number of companies (Anacleto et al., 2004).

Therefore based on the above discussions and the debate within the ISO community, for the purposes of this paper we are adopting the definition for VSE introduced in Laporte et al. (2006) as "*any enterprise, organization, department and project having up to 25 people*". Furthermore, this is the definition agreed to by the various national representatives of the working group 24 of ISO/IEC JTC1 SC7.

The unique characteristics of small enterprises as well as the uniqueness of their needs make their style of business different (Mtigwe, 2005). Some of the unique differences between small and large enterprises behavior are given in Table 1. Software VSEs are subject to a number of distinctive and intrinsic characteristics that make them different from their larger counterparts, therefore affecting the content, the nature and the extent of the activities. We partition our

discussion of VSE characteristics below based on four main categories: financial constraints, typical customer profile, the focus of internal business processes and the constraints on learning and growth (Basri & O'Connor, 2011).

VSEs are economically vulnerable as they are driven by cash flow and depend on project profits, so they need to perform the projects within budget. They tend to have low budgets which have many impacts, such as: Lack of funds to perform corrective post delivery maintenance; Few resources allocated for training; Little or no budget to perform quality assurance activities; No budget for software reuse processes; Low budget to identify, plan and mitigate risks; and Limited budget to perform Process Improvement and /or obtain a certification/assessment of their processes.

Table 1. Characteristic differences between small and large enterprises (from Mtigwe, 2005)

Characteristic	Small enterprise	Large enterprise
Planning orientation	Unstructured/operational	Structured/strategic
Flexibility	High	Structured/strategic
Risk orientation	High	Medium
Managerial process	Informal	Low
Learning and knowledge absorption capacity	Limited	High
Impact of negative market effects	More profound	More manageable
Competitive advantage	Human capital centered	Organizational capital centered

Typically the VSEs product has a single customer at a time, where the customer is in charge of the management of the system; the software integration, installation and operation. It is not a current practice for the customer to define quantitative quality requirements and for customer satisfaction to depend on the fulfillment of specific requirements that may change during the project. A close relationship between all involved project members including the customer shows that software development in small and very small companies is strongly human oriented and communication between them is important. For example, in contrast to small companies, very small companies often do not have regularly formal project meetings (O'Connor et al, 2010).

The internal business process of VSEs is usually focused on developing custom software systems, where the software product is elaborated progressively and incrementally, and typically software projects are independent of one another. Usually most management processes (such as human resource and infrastructure management) are performed through informal mechanisms, with the majority of communication, decision making and problem resolution being performed face to face.

The learning and growth characteristics of VSE are characterized by a lack of knowledge (or acceptance) of software process assessment and improvement and a lack of human resources to engage in standardization.

2.2 Standards: Benefits and Drawbacks

There are multiple approaches to organizing the software development process (Jeners et al, 2013) and multiple factors influencing the software development process (Clarke & O'Connor, 2012). Quality orientated process approaches and standards are maturing and gaining acceptance in many organizations. Standards emphasize communication and shared understanding more than anything. Examples are: any documentation is consistent and what is needed to meet the

needs of the organization; all users understand the same meaning of words used - if one person says, 'Testing is completed 'all affected bodies understand what those words mean. This kind of understanding is not only important in a global development environment; even a small group working in the same office might have difficulties in communication and understanding of issues shared by all. Standards can help in these and other areas to make the business more profitable because less time is spent on non-productive work (Yilmaz et al, 2016).

There are many potential benefits of using standards. From a VSE perspective, the benefits that certification can provide include: increased competitiveness, greater customer confidence and satisfaction, greater software product quality, increased sponsorship for process improvement, decreased development risk, facilitation of marketing and higher potential to export. While good internal software management might help meet the first five claims; the last two can only be the benefits of using widely recognized standards.

Although commercial Software Process Improvement (SPI) models (such as Capability Maturity Model Integration for Development, CMMI-DEV) (SEI, 2010) have been highly publicized and marketed, they are not being widely adopted and their influence in the software industry therefore remains more at a theoretical than practical level (Coleman and O'Connor, 2006) (O'Connor and Coleman, 2009). In the case of CMMI, evidence for this lack of adoption can be seen by examining the SEI (Software Engineering Institute) CMMI data for the two-year period 2014 to 2015 (CMMI, 2016), which shows that worldwide during that period less than 3,500 individual appraisals were reported, which includes many divisions of the same company. It is clear that this represents a very small proportion of the world's software companies and company in-house developers. In addition, there is evidence that the majority of small and very small software organizations are not adopting standards such as CMMI. For example, an Australian study (Staples et al, 2007) found that small organizations considered that adopting CMMI "*would be infeasible*".

Further investigation of the SEI CMMI appraisal data reveals that in the case of Ireland – a country whose indigenous software industry is primarily made of small to medium sized organizations (SME) - fewer than 10 CMMI appraisals were conducted during the ten-year period 2001 - 2011, from a population of more than 900 software companies. Therefore it is also clear that the Irish software industry is largely ignoring the most highly publicized SPI models. In the case of CMMI (and its predecessor Software CMM), Staples and Niazi (2006) discovered, after systematically reviewing 600 papers, that there has been little published evidence about those organizations who have decided not to adopt CMMI.

Though it is not new to claim that SPI has an associated cost, many companies are deterred from investigating SPI models because of a perceived cost. Managers' perceptions are that SPI means increased documentation and bureaucracy (O'Connor et al, 2010). Such a perception is widespread and is seen as a 'feature' of standards such as CMMI. Whether or not this is true is a debatable point. The fact that managers associate CMMI with increased overhead means that most small companies do not see the model as being a viable solution or even worthy of investigation.

There is evidence (Laporte et al, 2008) (Coleman & O'Connor 2008a) (O'Connor & Coleman, 2009) that the majority of small and very small software organizations are not adopting existing standards / proven best practice models because they perceive the standards as being developed by large organizations and orientated towards large organizations, thus provoking the debate in terms of number of employees, size does actually matter. Studies have shown that small firms' negative perceptions of process model standards are primarily driven by

negative views of cost, documentation and bureaucracy. In addition, it has been reported that SMEs find it difficult to relate standards to their business needs and to justify the application of the international standards in their operations. Most SMEs cannot afford the resources for, or see a net benefit in, establishing software processes as defined by current standards (e.g. ISO/IEC/IEEE 12207) and maturity models (e.g. CMMI for Development).

2.3 Recognition of Needs and Problems

Commercial SPI models have not been widely adopted by small and very small companies and their influence in the software industry therefore remains more at a theoretical than practical level. There is now a substantial body of research evidence (Laporte et al, 2008) (Coleman & O'Connor, 2008a) that the majority of small software organizations are not adopting existing standards because they perceive the standards as being orientated towards large organizations. Studies have shown that small firms' negative perceptions of process model standards are primarily driven by negative views of cost, documentation and bureaucracy. In addition, it has been reported that SMEs find it difficult to relate standards to their business needs and to justify the application of the international standards in their operations.

However quality-orientated process approaches and standards are maturing and gaining acceptance in many companies (O'Connor & Laporte, 2011b) and there is a clear benefit even to VSEs in the usage of standards. Amongst other positive effects, standards emphasize communication and shared understanding more than anything. Examples are: any documentation is consistent and what is needed to meet the needs of the organization; all users understand the same meaning of words used - if one person says, 'Testing is completed!', all affected bodies understand what those words mean. This kind of understanding is not only important in a global development environment; even a small group working in the same office might have difficulties in communication and understanding of issues shared by all. Standards can help in these and other areas to make the business more profitable because less time is spent on non-productive work.

2.3 VSE and Standards Usage

In a time when software quality is a key to competitive advantage, the use of ISO/IEC systems and software engineering standards by VSEs remains limited to a few of the most popular ones, such as ISO 9000. Research shows that VSEs can find it difficult to relate ISO/IEC standards to their business needs and to justify the application of the standards to their business practices. Most of these VSEs can't afford the resources - in number of employees, expertise, cost, and time - or see a net benefit in establishing software life-cycle processes. There is sometimes a disconnect between the short-term vision of the organization, looking at what will keep it in business for another six months or so, and the long-term benefits of gradually improving the ways the company can manage its software development and maintenance. A primary reason cited by many small software organizations for this lack of adoption of such ISO standards, is the perception that they have been developed by and for large multi-national software companies and not with the small organization in mind (Ahern et al, 2004). Subsequently, VSEs have no or very limited ways to be recognized as enterprises that produce quality software systems in their domain and may therefore be cut off from some economic activities.

Small software organizations, in the first instance, focus mostly on survival. This, in part, explains the success of agile methodologies whose 'light', non-bureaucratic techniques support companies in survival mode attempting to establish good, fundamental software development

practices. Though CMMI is firmly anchored in the belief that better processes means better products, many small Irish software product companies are merely concerned about getting a product released to the market as quickly as possible. Development models, such as those within the agile approach, rather than CMMI or ISO 9000, are perceived as supporting this objective. This clearly poses questions for CMMI and ISO 9000 researchers. However, if SPI models are to be more widely deployed by early stage (start-ups) companies, existing models may have to be broadened to take account of the necessity for these companies to meet their development targets and 'walk before they can run' (Basri & O'Connor, 2010),

2.4 International Organization for Standardization

International Organization for Standardization (ISO) is an independent, non-governmental international organization with a membership of 163 national standards bodies. Through its members, it brings together experts to share knowledge and develop voluntary, consensus-based, market relevant International Standards that support innovation and provide solutions to global challenges. The mandate of the ISO Sub-Committee 7 (SC7) is to develop, maintain, promote and facilitate IT standards required by global markets to meet business and user requirements concerning Software and Systems Engineering. A description of SC7 and of the development of ISO/IEC JTC1/SC7 standards is presented in (Coallier, 2003).

It should be noted that the authors of this paper are all key members of the ISO/IEC JTC1/SC7 WG24 standards development group and are also editors of various parts of the ISO/IEC 29110 family. As such they have a unique insight into the development of a new standard and a direct influence on its development. In addition the papers authors are author/editors of the Deployment Packages and Implementation Guides, which will be described later in this paper. Accordingly the authors are in a position to provide a unique insight into the design, development and initial deployment of this innovative standard.

3. HISTORICAL PERSPECTIVE ON THE DEVELOPMENT OF ISO/IEC 29110

3.1 The Origin of ISO/IEC JTC 1/SC7 WG24

In May 2004 at the Plenary Meeting of ISO/IEC JTC 1/SC7 (held in Australia), the Canadian national body raised the issue of small enterprises requiring standards adapted to their size and maturity level. Following from this a consensus was reached with delegates from five national bodies (Australia, Canada, the Czech Republic, South Africa and Thailand) and a special interest group (SIG) was created to explore the following general objectives:

- To make the current software engineering standards more accessible to VSEs;
- To provide documentation requiring minimal tailoring and adaptation effort;
- To provide harmonized documentation integrating available standards:
- To align profiles, if desirable, with the notions of maturity levels presented in ISO/IEC 33000.

In March 2005, the Thailand Industrial Standards Institute (TISI) and the Thai Software promotion agency (SIPA) invited a Special Working Group (SWG) to advance the work items defined by the SIG at the previous SC7 plenary meeting. A consensus was achieved by the members of the SWG on this study and a target defining for VSE was defined as IT services, organizations and projects with between 1 and 25 employees. The major output of this meeting

was a draft New Work Item Proposal (NWIP) and a. work schedule has also been developed for the new Working Group (WG)

In 2005 at the Plenary Meeting of ISO/IEC JTC 1/SC7 (held in Finland) a new ISO/IEC JCT1/SC7 Working Group (WG 24) was established with a mandate to investigate the need for and propose software life cycle profiles and guidelines for use in very small entities. WG24 initially received a commitment to participate in the new working group from the national bodies of: Belgium, Canada, the Czech Republic, Ireland, Italy, Japan, Korea, Luxemburg, South Africa, Thailand, the UK and the USA.

In October 2005 the first WG24 Meeting was held, hosted by the Italian national body and co-located with the ISO/IEC JTC1 SC7 Interim Meeting. Of particular significance at this meeting was the agreement of the need to elicit requirements from VSE around the world and to question them about their utilization of ISO/SC7 standards and to collect data to identify problems and potential solutions that would help them apply standards and become more competitive.

3.2 Requirements Led Approach to Development

Based on a resolution at the Italian meting in 2005 and in order to ascertain an enhanced understanding of the utilization of ISO/SC7 standards and to collect data to identify problems and potential solutions specific to VSEs, a survey of VSEs was designed to validate some of the groups initial working goals and better understand VSE attitudes to and requirements of standards. A survey questionnaire was developed and translated into 9 languages: English, French, German, Korean, Portuguese, Thai, Turkish, Russian and Spanish. The survey is made up of 20 questions structured in 5 parts: General information, Information about standards utilization in VSEs, Information about implementation and assessment problems in VSEs, Information about VSE needs and Information about justification for compliance to standard(s).

Over 400 responses were collected from 29 countries. The detailed major findings are documented in (Laporte et al, 2008), however some salient points are discussed here. An interesting finding of the survey is the difference in the percentage of certified companies with regard to company size: less than 18% of VSEs are certified, while 53% of larger companies (more than 25 employees) claim to be certified. Furthermore, among those 18% who are certified, 75% of them do not use standards. In larger companies using standards, two families of standards and models emerge from the list: ISO standards (55%) and models from the Software Engineering Institute (SEI) (47%).

The survey anticipated the weak use of standards by VSEs by asking questions designed to provide a better understanding of the reasons for this. The three main ones are: lack of resources; standards are not required; and the natures of the standards themselves, with 15% of the respondents consider that the standards are difficult and bureaucratic, and do not provide adequate guidance for use in a small business environment.

For a large majority (74%) of VSEs, it is very important to be evaluated or certified against a standard. ISO certification is requested by 40% of them. Of those requesting official market recognition, only 4% are interested in a national certification. From the VSE perspective, some benefits provided by certification are:

- Increased competitiveness
- Greater customer confidence and satisfaction
- Greater software product quality
- Increased sponsorship for process improvement

- Decreased development risk
- Facilitation of marketing (e.g. better image)
- Higher potential to export

However, VSEs are expressing the need for assistance in order to adopt and implement standards. Over 62% would like more guidance with examples, and 55% are asking for lightweight and easy-to-understand standards complete with templates. Finally, the respondents indicated that it has to be possible to implement standards with minimum cost, time and resources. All data about VSEs and standards clearly confirm WG24's fundamental assumption and the requirements. Therefore, WG24 uses this information to help define its approach for the development of profiles, guides and templates to meet VSE needs.

3.3 Development of ISO/IEC 29110

During 2006 (at meetings held in Thailand and Luxembourg) WG24 adopted the concept of ISO standardized profiles (SP) to develop the new standard for VSEs. A profile is defined as "A set of one or more base standards and/or SPs, and, where applicable, the identification of chosen classes, conforming subsets, options and parameters of those base standards, or SPs necessary to accomplish a particular function." From a practical point of view, a profile is a kind of bill of material composed of parts of standards such as ISO/IEC/IEEE 12207 or ISO/IEC /IEEE 15288 (ISO, 2015).

The approach (Laporte et al 2008) used to develop ISO/IEC 29110 started with the pre-existing international standard ISO/IEC/IEEE 12207 (ISO, 2008) dedicated to software process lifecycles. The overall approach consisted of three steps: (1) Selecting ISO/IEC/IEEE 12207 process subset applicable to VSEs (2) Tailor the subset to fit VSE needs; and (3) Develop guidelines for VSEs.

WG24 also sought existing standards or models that could be tailored to suit VMEs. In particular a Mexican standard 'MoProsoft' (NMX-059-NYCE, 2005) developed to assist Mexican small and medium enterprises (SMEs) was identified as a suitable base document for examination. As it was felt that MoProsoft addressed the needs of organizations larger than targeted VSEs. Therefore, as a second step, WG24 decided to tailor MoProsoft to address key characteristics of low-capability VSEs. As a starting point, the tailoring approach led to the development of incremental profile targeting of low-capability VSEs of fewer than 10 employees and, in a second phase, those with 10 to 25 employees.

3.4 Preparation and publication of ISO/IEC 29110

During the period 2007 – 2011 WG24 worked on the preparation and publication of the first set of documents. At the core of ISO 29110 is a Management and Engineering Guide (ISO/IEC 29110-5) (ISO, 2011a) focusing on Project Management and Software Implementation and an Assessment Guide (ISO/IEC 29110-3) (ISO, 2011b). It is worth noting that as with all proposed ISO standards, ISO/IEC 29110 is subject to the normal ISO review process. During the development of the standard in excess of 1250 comments have been processed between 2008 and 2010.

The current WG 24 Document Production schedule is as follows:

- Revision and publication of core documents:
    - ISO 29110-1 Ed2 (2016) - Overview
    - ISO 29110-2-1 Ed2 (2015) - Framework and Taxonomy
    - ISO 29110-3-1 (2015) - Process Assessment Guide

- Preparation and publication of additional guides
  - ISO 29110-5-6-1 (2015) - Systems Engineering Entry Profile
  - ISO 29110-5-6-2 (2014) - Systems Engineering Basic Profile
  - ISO 29110-5-2-1 (2016) - Organizational Management Profile Guide
  - ISO 29110-5-1-3 (2016) – Software Engineering Intermediate Profile
- Addition of conformity assessment documents
  - ISO 29110-3-3 (2015) - Process Assessment Conformity
- Addition of support documents
  - ISO 29110-2-2 (2016) - Domain Specific Guide
  - ISO 29110-3-4 (2015) - Autonomy-based Improvement Method

The ISO/IEC 29110 management and engineering guides became so popular that they had been translated into multiple languages. The current status of these is:

- ISO French Translations
  - Software Entry and Basic profiles translated by Canada
  - Systems engineering Basic profile translated by France (AFIS)
- National Translations
  - Czech
  - German
  - Japanese
  - Portuguese
- Spanish Translations
  - Translation of Part 1, 2, 3, 4, 5 by Peru/Uruguay
  - A Spanish Translation Task Force (STTF) has been reviewing the Spanish translations and will submit them to ISO for an official publication.

3.5 Further Developments

2010 saw the start of the first revision of the Profile Specification and in June 2011, following the recommendation of a Study Group, Canada provided a NWIP (which was later approved) to add a new domain, Systems Engineering, to the scope of the project, and produce new profiles specification and new guides. The objective of this additional NWIP was to establish a common framework for describing assessable system engineering life cycle profiles for VSEs that do not develop critical systems, and associated guidance.

June 2012 - Addition of Organizational Management was added to the scope of WG24 project and produce a new profile specification and a new guide. Organizational Management is defined as: "*The organizational standard processes (Basic profile) the VSE needs to define, deploy and improve to achieve similar good results in all projects*".

In February 2011 following the recommendation of a Study Group, Canada provided a NWIP (which was later approved) to add a new domain to VSEs, i.e. Service Delivery, to the scope of WG24. WG24 will produce a new profile specification (29110-4-3) and a new guide (29110-5-3). Service Delivery is defined as: "*A set of services provided to customers (internal or external) after the system or software development phase. These life cycle processes are generally identified as transition, Operation, Support, and Maintenance*".

4. STRUCTURE OF ISO/IEC 29110

The basic requirements of a software development process are that it should fit the needs of the project and aid project success. And this need should be informed by the situational context

where in the project must operate and therefore, the most suitable software development process is contingent on the context. The core situational characteristic (Clarke and O'Connor, 2012) of the entities targeted by ISO/IEC 29110 is size, however there are other aspects and characteristics of VSEs that may affect profile preparation or selection. Creating one profile for each possible combination of values of the various dimensions introduced above would result in an unmanageable set of profiles. Accordingly VSE's profiles are grouped in such a way as to be applicable to more than one category. Table 2 illustrates a Profile Group, which contains three profiles (labeled A, B and C) that are mapped to nine combinations of business models and situational factors.

Table 2. Allocating VSE characteristics to profile groups.

	Profile Situational Factors		
Business Models	**Critical**	**User Uncertainty**	**Environment Change**
Contract	*Profile A*	*Profile A*	*Profile A*
In-House	*Profile C*	*Profile B*	*Profile A*
Commercial	*Profile B*	*Profile A*	*Profile A*

Profile Groups are a collection of profiles, which are related either by composition of processes (i.e. activities, tasks), or by capability level, or both. The "Generic" profile group is applicable to a vast majority of VSEs that do not develop critical software and have typical situational factors. This profile group does not imply any specific application domain, however, it is envisaged that in the future new domain-specific sub-profiles may be developed in the future. Table 3 illustrates this profile group as a collection of four profiles, providing a progressive approach to satisfying the requirements of profile group. The Generic profile group provides a four-stage roadmap for VSEs that do not develop critical systems or critical software: Entry, Basic, Intermediate and Advanced profiles. VSEs targeted by the Entry profile are those working on small projects (e.g., at most six person-months of effort) and for start-ups. The Basic profile describes the development practices of a single application by a single project team. The Intermediate profile is targeted at VSEs developing multiple projects with more than one team. The Advanced profile is targeted at VSEs wishing to sustain and grow as independent competitive businesses.

Table 3. Graduated profile of the Generic profile group.

	Generic Profile Group		
Entry	Basic	Intermediate	Advanced

4.1 Engineering and Management Guide

At the core of this standard is a Management and Engineering Guide, officially know as ISO/IEC TR 29110-5-1-2 (2011a), which focuses on *Project Management* and *Software Implementation*

as illustrated in figure 1. The purpose of the Basic Profile is to define Software Implementation (SI) and Project Management (PM) processes from a subset of ISO/IEC/IEEE 12207 and ISO/IEC/IEEE 15289 appropriate for VSEs, as illustrated in figure 1.

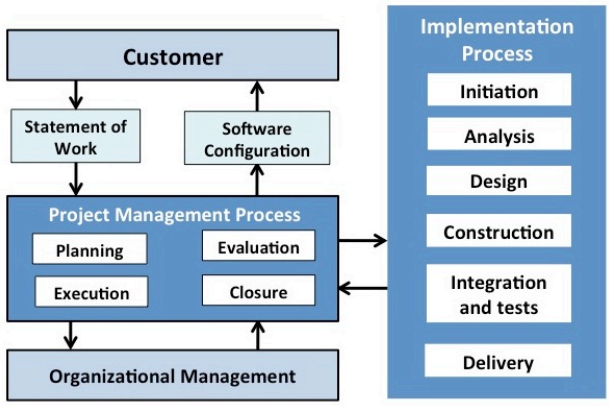

Figure 1. ISO/IEC 29110 Project Management and Software Implementation Relationship.

4.2 Project Management Process

The purpose of the Project Management (PM) process is to establish and carry out the tasks of the software implementation project in a systematic way, which allows compliance with the project's objectives in terms of expected quality, time, and costs (O'Connor and Laporte, 2012). The seven objectives of the PM process are listed in table 4.

Table 4. Objectives of the Project Management process of the Basic Profile (ISO/IEC TR 29110-5-1-2 (2011b).

Objective	Description
PM.O1	The Project Plan for the execution of the project is developed according to the Statement of Work and reviewed and accepted by the Customer. The tasks and resources necessary to complete the work are sized and estimated.
PM.O2	Progress of the project is monitored against the Project Plan and recorded in the Progress Status Record.
PM.O3	The Change Requests are addressed through their reception and analysis. Changes to software requirements are evaluated for cost, schedule and technical impact.
PM.O4	Review meetings with the Work Team and the Customer are held. Agreements are registered and tracked.
PM.O5	Risks are identified as they develop and during the conduct of the project.
PM.O6	A software Version Control Strategy is developed. Items of Software Configuration are identified, defined and baselined. Modifications and releases of the items are controlled and made available to the Customer and Work Team including the storage, handling and delivery of the items.
PM.O7	Software Quality Assurance is performed to provide assurance that work products and processes comply with the Project Plan and Requirements Specification.

Figure 2 illustrates the 4 activities of the project management process as well as their input and output product. The four activities of the Project Management Process are:

- **Project Planning** - The primary objective of this process is to produce and communicate effective and workable project plans. This process determines the scope of the project management and technical activities, identifies process outputs, project tasks and deliverables, establishes schedules for project task conduct, including achievement criteria, and required resources to accomplish project tasks.
- **Project Plan Execution** - To implement the actual work tasks of the project in accordance with the project plan. Ideally when the project plan has been agreed and communicated to all teams members, work of the development of the product, which is the subject of the project, should commence.
- **Project Assessment and Control** - purpose is to determine the status of the project and ensure that the project performs according to plans and schedules, within projected budgets and it satisfies technical objectives.
- **Project Closure** - typically involves releasing the final deliverables to the customer, handing over project documentation to the business, terminating supplier contracts, releasing project resources and communicating project closure to all stakeholders.

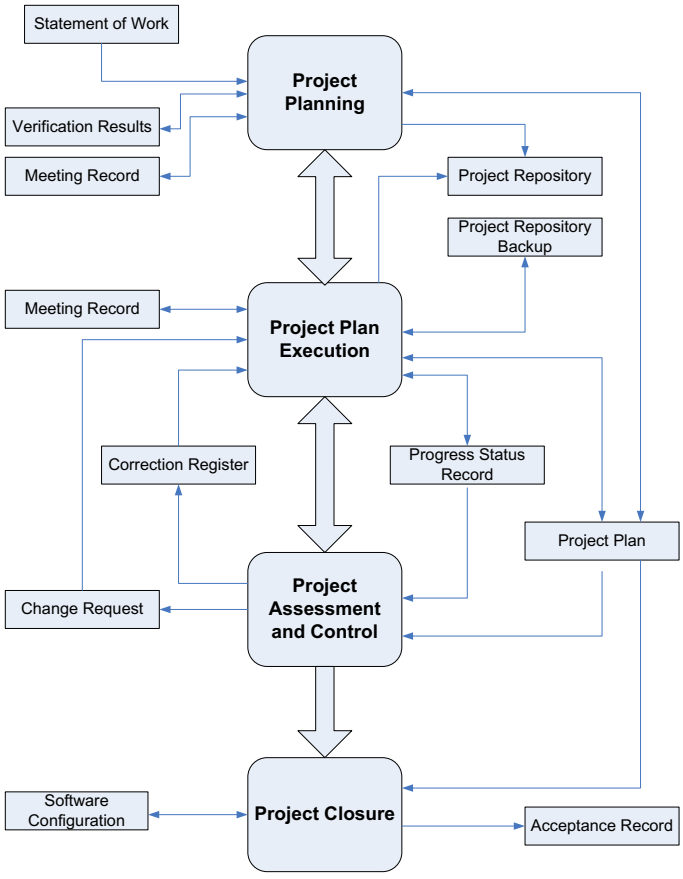

Figure 2. ISO/IEC 29110 Project Management Process (2011b).

For illustration purposes, two tasks of the Project Planning activity are listed in Table 5. The project manager (PM) and the customer (CUS) are involved in these 2 tasks. The customer is involved, during the execution of the project, when he submits change requests, during project review meetings, for the validation and approval of the requirements specifications and for the acceptance of the deliverables.

Table 5. Example of 2 tasks of the Project Planning Activity (ISO/IEC TR 29110-5-1-2 (2011b).

Role	Task	Input	Output
PM CUS	PM.1.2 Define with the Customer the Delivery Instructions of each one of the Deliverables specified in the Statement of Work.	Statement of Work [reviewed	Project Plan Delivery Instructions
PM CUS	PM.1.14 Review and accept the Project Plan. Customer reviews and accepts the Project Plan, making sure that the Project Plan elements match with the Statement of Work.	Project Plan [verified]	Meeting Record Project Plan [accepted]

4.3 Software Implementation Process

The purpose of the Software Implementation (SI) process, illustrated in figure 3, is to achieve systematic performance of the analysis, design, construction, integration, and test activities for new or modified software products according to the specified requirements. The seven objectives of the SI process are listed in table 6.

Table 6. Objectives of the Software Implementation process of the Basic Profile (2011b).

Objective	Description
SI.O21	Tasks of the activities are performed through the accomplishment of the current Project Plan.
SI.O2.	Software requirements are defined, analyzed for correctness and testability, approved by the Customer, baselined and communicated.
SI.O3.	Software architectural and detailed design is developed and baselined. It describes the Software Components and internal and external interfaces of them.
SI.O4.	Software Components defined by the design are produced. Unit test are defined and performed to verify the consistency with requirements and the design. T
SI.O5.	Software is produced performing integration of Software Components and verified using Test Cases and Test Procedures. Results are recorded at the Test Report.
SI.O6.	A Software Configuration, that meets the Requirements Specification as agreed to with the Customer, which includes user, operation and maintenance documentations, is integrated, baselined and stored at the

	Project Repository.
SI.O8.	Verification and Validation Tasks of all required work products are performed using the defined criteria to achieve consistency among output and input products in each activity.

The activities of the Software Implementation Process are:

- **Software Implementation Initiation** - ensures that the Project Plan established in Project Planning activity is committed to by the Work Team
- **Software Requirements Analysis** - analyzes the agreed Customer's requirements and establishes the validated project requirements. The activity provides:
- **Software Architectural and Detailed Design** - transforms the software requirements to the system software architecture and software detailed design
- **Software Construction** - develops the software code and data from the Software Design.
- **Software Integration and Tests** - ensures that the integrated Software Components satisfy the software requirements.
- **Product Delivery** - provides the integrated software product to the Customer.

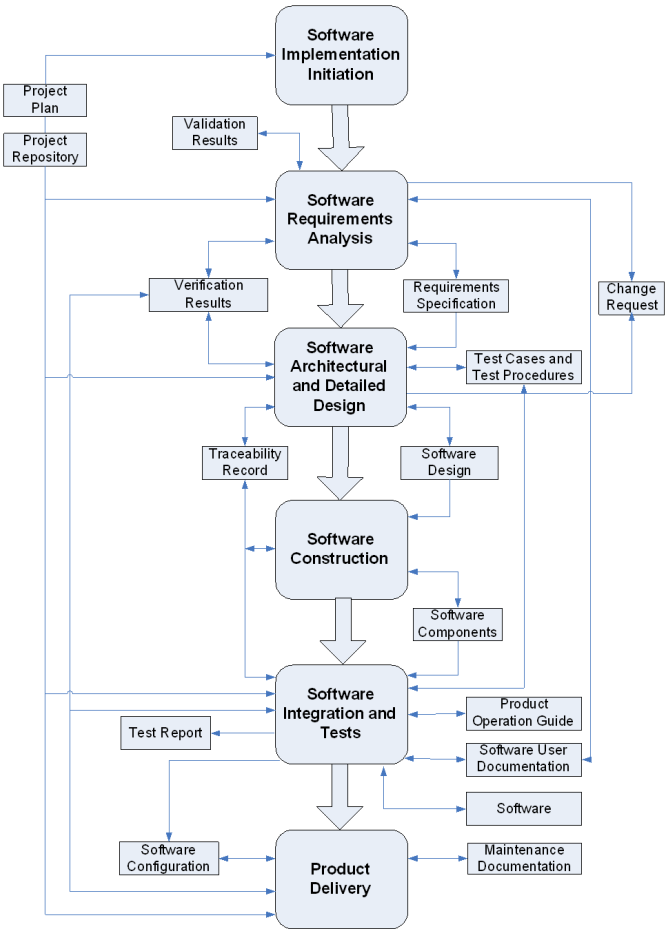

Figure 3. ISO/IEC 29110 Software Implementation Process (2011b).

4.4 Deployment Assistance

A novel approach taken to assist VSEs in the deployment of ISO/IEC 29110 was the development of a series of deployment packages (DPs), to define guidelines explaining in more details the processes defined in the ISO/IEC 29110 profiles (Laporte 2009). These guidelines will be freely accessible on the Internet to VSEs. A DP is a set of artifacts developed to facilitate the implementation of a set of practices, of the selected framework, in a VSE. A DP is not a process reference model (i.e. it is not prescriptive). The elements of a typical DP are: description of processes, activities, tasks, roles and products, template, checklist, example, reference and mapping to standards and models, and a list of tools. The mapping is only given as information to show that a deployment package has explicit links to standards, such as ISO/IEC/IEEE 12207, or models, such as the CMMI for Development, hence by deploying and implementing the package, a VSE can see its concrete step to achieve or demonstrate coverage. Packages are designed such that a VSE can implement its content, without having to implement the complete framework at the same time. A set of nine DPs has been developed to date and are freely available from (DP, 2011). In addition a series of "Implementation Guides" have been developed to help implement a specific process supported by a tool and are freely available from (DP, 2011). To date five such guides have been developed.

These Deployment Packages and Implementation Guides mark a significant departure from existing standards development and are specifically designed to ease many of the issues and problems VSE have with implementing standards on a day to day basis, as outlined earlier. In addition a series of Eclipse plug-ins and have been made freely available to the public.

5. ISO/IEC 29110 IN SYSTEMS ENGINEERING

In 2011, the original mandate of WG24 was expanded to produce a standard and a set of management and engineering guides for VSEs involved in systems development (Laporte & O'Connor, 2016a). A system in the context of ISO 29110 is typically composed of hardware and software components. ISO/IEC/IEEE 15288 process elements were selected for the systems engineering standard and guides (Laporte et al, 2014b). A set of Deployment Package has also been developed (Laporte et al, 2015) (Houde et al, 2016). The Entry and Basic systems engineering management and engineering guides have been published; the Intermediate profile is presently being developed and the development of the Advanced profile should start in 2017. We should also be able to complete the development of an autonomous Rover case study, developed under the Eclipse Foundation Polarsys project, using the SE Basic profile and the set of SE Deployment Packages (Laporte & Houde, 2015).

An ISO/IEC 29110 implementation project was created to define and implement project management and systems engineering (SE) processes at CSinTrans Inc. (CSiT), a Canadian company, created in 2011 (Laporte et al, 2016). The company specializes in the integration of interactive systems, communication and security in the field of public transport such as trains, subways and buses and railway stations, and stations bus stops. ISO 29110 standards and guides for SE have been used as the main reference for the development of the processes.

ISO 29110 standard has helped raise the maturity of the young organization by implementing proven practices and developing uniform work products. ISO 29110 was a good starting point to align processes with selected level 2 and 3 practices of the CMMI model.

Recently, the processes of CSiT, based on the Basic Profile of the ISO 29110, have been successfully audited by a third-party audit composed of 2 independent auditors. One member of the audit team was a SE domain expert. In 2016, CSiT had 10 employees.

6. IMPLEMENTING THE ISO/IEC 29110 STANDARD

WG24 members are advocating the use of pilot projects as a means to accelerate the adoption and utilization of ISO/IEC 29110 by VSEs. Pilot projects are an important means of reducing risks and learning more about the organizational and technical issues associated with the deployment of new software engineering practices. A successful pilot project is also an effective means of building adoption of new practices by members of a VSE. Pilot projects are based on the ISO/IEC 29110-5 Management and Engineering Guide (ISO, 2011a) and the deployment package(s). In particular these are aimed to collect, as a minimum, the following data:

- Effort and time to deploy by the VSE
- Usefulness for the VSE
- Verification of the understanding of the VSE
- Self-assessments data - A self-assessment at the beginning of the pilot and at the end of the pilot project DP

To further assist with the roll out of a pilot project and to ensure that all pilot projects are conducted similarly around the world, a set of pilot project guidelines were developed in the form of a Deployment Package (DP, 2011) to describe a process to conduct pilot projects. The primary purpose of this Deployment Package is to provide tailorable and usable guidelines and materials in order to select and conduct pilot projects in VSEs. The high-level tasks of this Deployment Package are:

- Assess the opportunity to conduct a pilot project,
- Plan the pilot project,
- Conduct the pilot project, and
- Evaluate the results of the pilot project.

An additional target audience, and an often forgotten one, in the area of software engineering standards comprise undergraduate and graduate students. In 2009, at the WG24 meeting in India, an informal interest group about education was formed (Laporte 2009). The main objective is to develop a set of courses for software undergraduate and graduate students such that students learn about the ISO standards for VSEs before they graduate. Work is already underway on the development of course modules to support DPs via a VSE Education Special Interest Group. To date four of the six courses have been developed and are freely available (VSE SIG 2011). In addition the WG24's team has created an initial set of Wikipedia information pages in the Czech, English, French, Portuguese and Spanish language versions of Wikipedia and also a set of introductory videos (in both English and French) available on both ISOPlanet on YouTube.

To date a series of pilot projects have been completed in several countries utilizing some of the deployment packages developed. For example in Canada a pilot study has been conducted in an IT department with a staff of 4: 1 analyst and 3 developers, who were involved in the translation and implemented 3 DPs: Software Requirements, Version Control, Project Management (O'Connor & Laporte, 2012) (Laporte et al, 2013). ISO/IEC 29110 has been successfully implemented in VSEs in many countries, including IT start-ups in Canada (Laporte et al, 2014a), Peru (García Paucar et al, 2015) and in a VSE co-located in Tunisia and Canada (Jelljeli & Laporte, 2016) (Laporte & O'Connor, 2016b); VSE located in Ireland (O'Connor and Sanders 2013); in a large engineering firm (Laporte & Chevalier, 2016) in a small medical R&D VSE, in an automotive enterprise, in a large financial institution; and in an electricity provider (Laporte & O'Connor, 2016b). In Belgium a VSE of 25 people started with a process assessment phase

aiming to identify strengths and weaknesses in development related processes (Boucher et al, 2012) (Ribaud et al, 2010). This company is now working on improvement actions mainly based on the following Deployment Packages: Requirement Analysis, Version Control, and Project Management.

In addition some retrospective studies have been completed into early stage adoption of project management practices (O'Connor, 2014) and also exploring VSE management sentiment towards ISO/IEC 29110 (O'Connor, 2012) (Sanchez-Gordon et al 2015).

7. ISO/IEC 29110 in universities and colleges

In the field of undergraduate degree programs, according to the Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering, software process is one of the 10 knowledge areas of the curriculum (Johansen et al, 2016). Although the coverage of software process education is established in curriculum initiatives, increasing its coverage in educational settings is still challenging. The fact that ISO/IEC 29110 guides are easily understandable and freely available has greatly helped their adoption; more than 15 countries are teaching it at the undergraduate and graduate levels. In Thailand, more than 10 universities teach ISO/IEC 29110, and in Canada (Laporte & O'Connor, 2015) (Laporte & O'Connor, 2016c), it's taught in software quality assurance and software process improvement courses. In addition, students are implementing ISO/IEC 29110 in undergraduate and graduate capstone projects.

In Canada, at ÉTS, a 10,000-student engineering school, students of undergraduate and graduate Software quality assurance courses and software process improvement graduate course are using ISO/IEC 29110 as the main framework for their projects (Laporte 2015) (Laporte et al, 2015).

A game called ISOPOLY, based on the well-known board game Monopoly, has been developed at ÉTS. An electronic version has been developed in Peru and should be available in 2017.

A multi-language self-learning site has been published in late 2016. A Moodle web site has been developed in three languages (French, English and Spanish) by a graduate student of ÉTS. The site[1] contains tools, slide presentations, quizzes (i.e. mini test) and videos (e.g. on ISO 29110, implementers) (Laporte & O'Connor, 2015).

8. FUTURE EVOLUTION

As ISO/IEC 29110 is an emerging standard there is much work yet to be completed. Current efforts are concentrated on completing the initial mandates of WG24 are completed (i.e. the publications of the Intermediate and Advanced documents of the systems engineering (SE) and software engineering (SW) Generic Profile Group). After completion of these documents WG24 will be in a position to develop additional material to help VSEs and to accelerate the diffusion and implementation of ISO/IEC 29110. Examples of future projects are:

- Combine Management and Engineering Guides: all SW Generic Profile Guides in one document instead of 4 documents and/or all SE Generic Profile Guides in one document instead of 4 documents
- Get large organizations to do business with VSEs that have implemented ISO/IEC 29110

[1] https://ena.etsmtl.ca/course/index.php?categoryid=35

- Get large organizations to request ISO/IEC 29110 evaluation/audit certification from their suppliers
- Develop a Guide to help VSEs in obtaining/negotiating with large organizations
- Get more Universities and technical colleges to teach ISO/IEC 29110 at the undergraduate and graduate levels
- Develop ISO 29110 Guides/Profiles for specific domains (e.g. medical devices)
- Collaborate with tool developers to enable support for ISO 29110 – e.g. project management, repository management, requirements management.
- Increased VSE support for agile approaches to implementing ISO/IEC 29110 (Galvan et al 2015)

REFERENCES

Ahern, D.M., Clouse, A., Turner, R., (2004). CMMI Distilled: A Practical Introduction to Integrated Process Improvement, 2nd Ed, Addison Wesley, Boston

Anacleto, A.; von Wangenheim, C.G.; Salviano, C.F.; Savi, R. (2004). Experiences gained from applying ISO/IEC 15504 to small software companies in Brazil, 4th International SPICE Conference on Process Assessment and Improvement, Lisbon, Portugal.

Basri, S. and O'Connor, R. (2010). Understanding the Perception of Very Small Software Companies towards the Adoption of Process Standards, in Riel et al (Eds), Systems, Software and Services Process Improvement, CCIS Vol. 99, Springer-Verlag, pp. 153-164.

Basri, S. and O'Connor, R. (2011). Towards an Understanding of Software Development Process Knowledge in Very Small Companies, in Abd Manaf et al (Eds), Informatics Engineering and Information Science, CCIS Vol. 253, Springer-Verlag, pp. 62-71.

Boucher, Q, Perrouin, G., Deprez, J.C. and Heymans, P. (2012). Towards Configurable ISO/IEC 29110-Compliant Software Development Processes for Very Small Entities in Winkler et al (Eds), Systems, Software and Services Process Improvement, CCIS Vol. 330, Springer-Verlag, pp. 169 – 180.

Coleman, G., O'Connor, R. (2006). Software Process in Practice: A Grounded Theory of the Irish Software Industry. In: Richardson, I., Runeson, P., Messnarz, R. (eds.) EuroSPI 2006. LNCS, vol. 4257, pp. 28–39. Springer, Heidelberg.

Coleman, G., O'Connor, R. (2008a). Investigating Software Process in Practice: A Grounded Theory Perspective. Journal of Systems and Software 81 (5), 772–784.

Coleman G. and O'Connor R. (2008b). The Influence of Managerial Experience and Style on Software Development Projects, International Journal of Technology, Policy and Management, 8 (1).

Coallier, F. (2003). International Standardization in Software and Systems Engineering, Crosstalk, February 2003, pp. 18-22.

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

Clarke, P., O'Connor, R. V., & Leavy, B. (2016, May). A complexity theory viewpoint on the software development process and situational context. In Proceedings of the International Workshop on Software and Systems Process (pp. 86-90). ACM.

CMMI Published Appraisal Results List, Software Engineering Institute, https://sas.cmmiinstitute.com/pars/, accessed June 2016.

Deployment Packages repository available from http://profs.logti.etsmtl.ca/claporte/English/VSE/index.html

European Commission (2005). The New SME Definition: User Guide and Model Declaration, available at:
http://europa.eu.int/comm/enterprise/enterprise_policy/sme_definition/sme_user_guide.pdf

García Paucar, L. H., Laporte, C. Y., Arteaga, Y., & Bruggmann, M. (2015). Implementation and Certification of ISO/IEC 29110 in an IT Startup in Peru. Software Quality Professional (SQP).

Galvan, S., Mora, M., O'Connor, R. V., Acosta, F., & Alvarez, F. (2015). A compliance analysis of agile methodologies with the ISO/IEC 29110 project management process. Procedia Computer Science, 64, 188-195.

Ginsberg, M.; Quinn, L. (1995). Process Tailoring and the Software Capability Maturity Model, Software Engineering Institute, CMU/SEI-94-TR-024.

Hadden, R., Key Practices to the CMM: Inappropriate for Small Projects (1998). Proceedings of the Software Engineering Process Group Conference.

Houde, R., Laporte, C.Y., Blondelle, G., ISO/IEC 29110 Deployment Packages and Case Study for Systems Engineering: The "Not-So-Secret" Ingredients That Power the Standard, 26th Annual International Symposium of INCOSE (International Council on Systems Engineering), Edinburgh, UK, July 18-21, 2016.

International Organization for Standardization (ISO): ISO/IEC/IEEE 12207:2008, Systems and software engineering – Software life cycle processes. Geneva (2008).

International Organization for Standardization (ISO): ISO/IEC/IEEE 15288:2015, Systems and software engineering- System life cycle processes. Geneva (2015).

International Organization for Standardization (ISO): ISO/IEC TR 29110-5-1-2 Software Engineering - Lifecycle Profiles for Very Small Entities (VSEs) Part 5-1-2: Management and

engineering guide: Generic profile group: Basic profile, Geneva (2011a). Available at no cost from ISO at: http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html

International Organization for Standardization (ISO): ISO/IEC TR 29110-3, Software Engineering - Lifecycle Profiles for Very Small Entities (VSE) - Part 3: Assessment Guide. Geneva (2011b). Available at no cost from ISO at: http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html

Jelljeli, H., Laporte, C.Y., Mise en oeuvre de processus logiciels à l'aide de la norme ISO/CEI 29110 dans une grande entreprise en démarrage, Revue Génie logiciel, Numéro 117, Juin 2016, pp. 34-41 (in French).

Jeners, S., Clarke, P., O'Connor, R. V., Buglione, L., and Lepmets, M. Harmonizing Software Development Processes with Software Development Settings - A Systematic Approach, In McCafery, F., O'Connor, R.V. and Messnarz R. (Eds), Systems, Software and Services Process Improvement, CCIS 364, Springer-Verlag, 2013

Johansen, J., Colomo-Palacios, R., & O'Connor, R. V. (2016). Towards a Manifesto for Software Process Education, Training and Professionalism. In International Conference on Software Process Improvement and Capability Determination (pp. 98-105). Springer International Publishing.

Johnson, D., Brodman, J. (1998). Applying the CMM to Small Organizations and Small Projects. In: Proceedings of Software Engineering Process Group Conference.

Laporte, C.Y.; April, A. and Renault, A. (2006). Applying ISO/IEC Software Engineering Standards in Small Settings: Historical Perspectives and Initial Achievements, Proceedings of SPICE Conference, Luxembourg.

Laporte, C.Y., Alexandre, S., O'Connor, R. (2008). A Software Engineering Lifecycle Standard for Very Small Enterprises. In: O'Connor, R., et al. (eds.) Proceedings of EuroSPI. CCIS, vol. 16, Springer-Verlag, pp. 129–141.

Laporte, C. Y. (2009). Contributions to software engineering and the development and deployment of international software engineering standards for very small entities (Doctoral dissertation, Université de Bretagne Occidentale, Brest).

Laporte, C. Y., O'Connor, R., & Fanmuy, G. (2013). International Systems and Software Engineering Standards for Very Small Entities. In Proceedings of CrossTalk - The Journal of Defense Software Engineering, 26 (3), pp 28-33.

Laporte, C. Y., Hébert, C., & Mineau, C. (2014b). Development of a social network website using the new ISO/IEC 29110 standard developed specifically for very small entities. Software Quality Professional, 16(4).

Laporte, C., O'Connor, R., Garcia Paucar, L., Gerancon, B.: An innovative approach in developing standard professionals by involving software engineering students in implementing and improving international standards. J. SES (The Society for Standards Professionals) 67(2), 2–9 (2015)

Laporte, C.Y., International Software Engineering Standards Applied in Undergraduate and Graduate Software Quality Assurance Courses, IEEE Standards University, IEEE Standards Education E-Magazine, November 2015. On line at: http://www.standardsuniversity.org/issue/november-2015/

Laporte, C.Y., Houde, R., Open Source Systems Engineering Guides, Deployment Packages and Support Tools for Very Small Enterprises – A Case Study, 25th Annual International Symposium of INCOSE (International Council on Systems Engineering), Seattle, US, July 13-16, 2015.

Laporte, C.Y., Houde, R., Blondelle, G., *Systems and Software Engineering Standards and Open Source Tools for Very Small Enterprises – A Case Study*, Embedded World Exhibition & Conference, Nuremburg, Germany, February 23-25, 2016

Laporte, C.Y., O'Connor, R, (2015). Software Process Improvement in Graduate Software Engineering Programs, in: O'Connor, R.V., Mitasiunas, A. and Ross, M. (Eds), Proceeding of the 1st International Workshop on Software Process Education, Training and Professionalism (SPETP 2015), CEUR Electronic Workshop Proceedings, Vol. 1368, pp. 18–24

Laporte, C.Y., O'Connor, R.V. (2016a), Systems and Software Engineering Standards for Very Small Entities: Accomplishments and Overview, IEEE Computer, Volume 49, number 8, pp. 84-87.

Laporte, C. Y., & O'Connor, R. V. (2016b) mplementing Process Improvement in Very Small Enterprises with ISO/IEC 29110: A Multiple Case Study Analysis. In Quality of Information and Communications Technology (QUATIC), 2016 10th International Conference on the. IEEE.

Laporte, C. Y., & O'Connor, R. V. (2016c). Software Process Improvement in Industry in a Graduate Software Engineering Curriculum. Software Quality Professional Magazine, 18(3).

Laporte, C. Y., Chevalier, F. (2016). An Innovative Approach to the Development of Project Management Processes for Small-Scale Projects in a Large Engineering Company. In K. Jakobs (Ed.), *Effective Standardization Management in Corporate Settings* (pp. 123-160). Hershey, PA: Business Science Reference. doi:10.4018/978-1-4666-9737-9.ch007

Laporte, C.Y., Tremblay, N., Menaceur, J., Poliquin, D., Houde, R., Implementing the New ISO/IEC 29110 Systems Engineering Process Standard in a Small Public Transportation Company, In Proceedings of 23rd European Software Process Improvement Conference (Euro SPI 2016), Springer-Verlag, 2016.

Laporte, Claude Y., Houde, R., Marvin, J., Systems Engineering International Standards and Support Tools for Very Small Enterprises, 24th Annual International Symposium of INCOSE (International Council on Systems Engineering), Las Vegas, US, June 30th-July 3, (2014a).

Larrucea, X., O'Connor, R.V., Colomo-Palacios, R., Laporte, C.Y.: Software process improvement in very small organizations. IEEE Softw. 33(2), 85–89 (2016)

Mtigwe, B. (2005). The entrepreneurial firm internationalization process in the Southern African context: A comparative approach, International Journal of Entrepreneurial Behavior & Research, Vol. 11 No. 5, pp. 358-377

NMX-059-NYCE (2005), Information Technology – Software Models of Processes and Assessment for Software Development and Maintenance. Part 01: Definition of Concepts and Products; Part 02: Process Requirements (MoProSoft); Part 03: Guidelines for Process
Pre-Publication Version Implementation; Part 04: Guidelines for Process Assessment (EvalProSoft). Ministry of Economy, Mexico, 2005.

OECD (1915). SME and Entrepreneurship Outlook, Organization for Economic Co-operation and Development.

O'Connor, R. (2012). Evaluating Management Sentiment towards ISO/IEC 29110 in Very Small Software Development Companies, in Mas et al (Eds), Software Process Improvement and Capability Determination, CCIS Vol. 290, Springer-Verlag, pp. 277-281.

O'Connor R. and Coleman G., (2009). Ignoring 'Best Practice': Why Irish Software SMEs are rejecting CMMI and ISO 9000, Australasian Journal of Information Systems, Vol. 16, No. 1, pp. 7-29.

O'Connor, R., Basri, S. and Coleman, G. (2010). Exploring Managerial Commitment towards SPI in Small and Very Small Enterprises, in Riel et al (Eds), Systems, Software and Services Process Improvement, CCIS Vol. 99, Springer-Verlag, pp. 268-278.

O'Connor, R. and Laporte, C.Y. (2011a). Using ISO/IEC 29110 to Harness Process Improvement in Very Small Entities, Workshop on SPI in SMEs, 18th European Software Process Improvement Conference (EuroSPI 2011), CCIS Vol. 172, Springer-Verlag, pp. 225-235.

O'Connor, R. and Laporte, C.Y. (2011b). Deploying Lifecycle profiles for Very Small Entities: An Early Stage Industry View, In O'Connor et al (Eds), Software Process Improvement and Capability Determination, CCIS Vol. 155, Springer-Verlag, pp. 227-230.

O'Connor, R. and Laporte (2012). Software Project Management in Very Small Entities with ISO/IEC 29110, in Winkler et al (Eds), Systems, Software and Services Process Improvement, CCIS Vol. 330, Springer-Verlag, pp. 330-341.

O'Connor, R. and Sanders, M., (2013) Lessons from a Pilot Implementation of ISO/IEC 29110 in a Group of Very Small Irish Companies, Proceedings 13th International Conference on

Software Process Improvement and Capability dEtermination (SPICE 2013), CCIS Vol. 349, Springer-Verlag.

O'Connor, R. V. (2014). Early stage adoption of ISO/IEC 29110 software project management practices: a case study. In International Conference on Software Process Improvement and Capability Determination (pp. 226-237). Springer International Publishing.

Ribaud, V., Saliou, P., O'Connor, R. and Laporte C. (2010). Software Engineering Support Activities for Very Small Entities , in Riel et al (Eds), Systems, Software and Services Process Improvement, CCIS Vol. 99, Springer-Verlag, pp. 165-176.

Sanchez-Gordon, M. L., O'Connor, R. V., & Colomo-Palacios, R. (2015). Evaluating VSEs viewpoint and sentiment towards the ISO/IEC 29110 standard: a two country grounded theory study. In International Conference on Software Process Improvement and Capability Determination (pp. 114-127). Springer International Publishing.

Software Engineering Institute. (2010). CMMI for Development (CMMI-DEV), Version 1.3, Pittsburgh, PA. Carnegie Mellon University.

Staples, M, Niazi M, Jeffery, R, Abrahams, A, Byatt, P and Murphy, R. (2007). An exploratory study of why organizations do not adopt CMMI, The Journal of Systems and Software, Vol. 80, No. 6, 883–895

Staples, M., Niazi, M., (2006). Systematic Review of Organizational Motivations for Adopting CMM-based SPI. Technical Report PA005957, National ICT Australia.

VSE Education Special Interest Group, available from http://profs.logti.etsmtl.ca/claporte/English/VSE/VSE-Education.html

Yilmaz, M., O'Connor, R. V., & Clarke, P. (2016). Effective Social Productivity Measurements during Software Development—An Empirical Study. International Journal of Software Engineering and Knowledge Engineering, 26(03), 457-490.

BIOGRAPHY

Rory V. O'Connor is an Associate Professor of Software Engineering at Dublin City University (Ireland) where he is currently serving as the Head of the School of Computing. He is also a Senior Researcher with Lero, the Irish Software Research Centre. Prof. O'Connor is Ireland's Head of Delegation to the International Organization for Standardization (ISO) for Software & Systems Engineering (ISO/IEC JCT1/ SC7). His research interests are cantered on the processes and standards whereby software intensive systems are designed, implemented and managed. His focus is on researching methods, techniques, tools and standards for supporting the work of software project managers and software developers in relation to software process improvement, and the management of software development projects. He can be reached at Rory.OConor@dcu.ie.

Claude Y Laporte is a Professor of software engineering at the École de technologie supérieure. His research interests include software process improvement in small and very entities and software quality assurance. He is the co-author of 2 French textbooks about software quality assurance published by Hermes-Lavoisier in 2011. The IEEE Computer Society and John Wiley and Sons will publish an English version of the French textbooks in 2017. He is the Project Editor of the ISO/IEC JTC1 SC7 Working Group tasked to develop ISO/IEC 29110 systems and software engineering standards and guides for use in very small entities. He may be contacted at: claude.laporte@etsmtl.ca.