



**HAL**  
open science

## Software Engineering Support Activities for Very Small Entities

Vincent Ribaud, Philippe Saliou, Rory O'Connor, Claude Laporte

► **To cite this version:**

Vincent Ribaud, Philippe Saliou, Rory O'Connor, Claude Laporte. Software Engineering Support Activities for Very Small Entities. EuroSPI 2010, Sep 2010, Grenoble, France. pp.165-176. hal-00504444

**HAL Id: hal-00504444**

**<https://hal.univ-brest.fr/hal-00504444>**

Submitted on 20 Jul 2010

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Post-print de 17th European Conference, EuroSPI 2010, Grenoble (France)  
Communication in Computer and Information Science,  
Springer-Verlag, Berlin-Heidelberg

## Software Engineering Support Activities for Very Small Entities

Vincent Ribaud<sup>1</sup>, Philippe Saliou<sup>1</sup>, Rory V. O'Connor<sup>2</sup>, Claude Y. Laporte<sup>3</sup>

<sup>1</sup> University of Brest, CS 93837, 29238 Brest Cedex, France  
{Vincent.Ribaud@univ-brest.fr, Philippe.Saliou@univ-brest.fr}

<sup>2</sup> Lero, The Irish Software Engineering Research Centre, Dublin City University, Ireland  
roconnor@computing.dcu.ie

<sup>3</sup> École de technologie supérieure, 1100 rue Notre-Dame Ouest  
Montréal (Québec), Canada, H3C 1K3  
Claude.Y.Laporte@etsmtl.ca

**Abstract.** The emerging ISO/IEC 29110 standard Lifecycle profiles for Very Small Entities has at its core a Management and Engineering Guides which is targeted at very small entity (enterprise, organization, department or project) having up to 25 people, to assist them unlock the potential benefits of using standards which are specifically designed to address there needs. The developers of the standard, ISO/IEC JCT1/SC7 Working Group 24 (WG24), recommend the use of pilot projects as a mean to trial the adoption of the new International standard in small organisations. Accordingly an ISO/IEC 29110 pilot project has been established between the Software Engineering group of Brest University and a 14 person company with the aim of establishing an engineering discipline for a new web-based project. This paper details the lessons learned from the pilot project and based on our experiences with using ISO/IEC 29110 we identify a potential deficiency and accordingly propose new process area, “Infrastructure and Support” for include in the future evolution of ISO/IEC 29110 Process Profiles.

**Keywords:** Very small entities, software engineering processes, ISO/IEC 29110, support activities.

### 1 Introduction

Given the current uncertain economic environment a major key to competitive advantage for software organizations is the quality of the software products they produce. It is there natural to expect that most organization in attempting to ensure high software quality would embrace the usage of software engineering standards, however research [1, 2] tells us that the use of ISO/IEC systems and software engineering standards remains limited to a few of the most popular ones. This research further shows that very small organizations 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. Furthermore they cannot afford the resources - in

number of employees, cost, and time - or see a net benefit in establishing software life-cycle processes. There are also 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 benefits of gradually improving the ways the company can manage its software development and maintenance. 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.

According to [3] "Small software organizations need to manage and improve their software processes, deal with rapid technology advances, maintain their products, operate in a global software environment, and sustain their organizations through growth". But very small organisations are not miniaturization of large entities and often do not have resources to perform complex tasks secondary to their products. Indeed software life cycle processes may well appear to such very small organisations as secondary management and excluded from human and financial investments. The majority of software engineering literature about small and very small software organizations focuses on software process assessment and improvement, such as the Mexican initiative MoProSoft [4] and Adept [5], but with little published work on support processes for very small software organizations.

The Rapid-Q framework, defined as a customizable set of processes that can be directly deployed into organizations with minimal effort, organizes processes in three levels: Project Procedures, Engineering Procedures, and Support Procedures [6]. Rapid-Q support processes are related to Configuration Management, Change Management, Review, Vendor Management, Process Management, Audit, Formal Decision, and Training. However this paper is more focused on support for Software Development and is slightly simpler than the RAPID-Q approach. The International Process Research Consortium reported in [7] a core research constituted of a hierarchy of research themes, research nodes, research questions. This framework defines a research node E.3: Providing Process Engineering Infrastructure, with research questions on organization and training, and technology.

The research pilot project reported in this paper addresses specifically questions E-44 "What process infrastructures are appropriate to support the new technologies and concurrent engineering?" and E-45 "How do we create easy to use "experience bases" that allow knowledge to be stored, updated, and accessed by developers at varying levels?". The proposal of this paper is focused on the E-44 question and results associated with the E-45 question are presented in another paper [8].

## **1.1 ISO/IEC 29110**

The term "very small entity" (VSE) had been defined by the ISO/IEC JTC1/SC7 Working Group 24 (WG24) as being "an entity (enterprise, organization, department or project) having up to 25 people" and has subsequently been adopted for use in the emerging ISO/IEC 29110 "Lifecycle Profiles for Very Small Entities" [9]. This standard has been aimed at addressing the issues identified above and addresses the specific needs of VSEs. The approach [10] used to develop ISO/IEC 29110 started with the pre-existing international standard ISO/IEC 12207 [11] dedicated to software process lifecycles. The overall approach consisted of three steps: (1) Selecting

ISO/IEC 12207 process subset applicable to VSEs; (2) Tailor the subset to fit VSE needs; and (3) Develop guidelines for VSEs. At the core of this standard is a Management and Engineering Guides (ISO/IEC 29110-5) [12] focusing on Project Management and Software Implementation and an Assessment Guide (ISO/IEC 29110-3) [13]. ISO/IEC 29110 makes use of the concept of a standardized profile, where a profile promotes the integration of base standards by defining how to use a combination of base standards for a given function and environment. These profiles are published as international standards.

In addition to the selection of base standards, a choice is made of permitted options for each base standard and of suitable values for parameters left unspecified in the base standard.

**Profiles:** The core characteristic of the entities targeted by ISO/IEC 29110 is size, however there are other aspects and characteristics of VSEs that may affect profile preparation or selection, such as: Business Models (commercial, contracting, in-house development, etc.); Situational factors (such as criticality, uncertainty environment, etc.); and Risk Levels. Creating one profile for each possible combination of values of the various dimensions introduced above would result in an unmanageable set of profiles. Accordingly VSE's profiles are grouped in such a way as to be applicable to more than one category. Profile Groups are a collection of profiles which are related either by composition of processes (i.e. activities, tasks), or by capability level, or both. The "Generic" profile group has been defined [14] as applicable to a vast majority of VSEs that do not develop critical software and have typical situational factors. The Generic Profile Group is a collection of four profiles (Entry, Basic Intermediate, Advanced), providing a progressive approach to satisfying a vast majority of VSEs that do not develop critical software and have typical situational factors.

To date the Basic Profile has been specified and developed, the purpose of which is to define a software development and project management guide for a subset of processes and outcomes appropriate for characteristics and needs of VSEs. The Basic Profile describes software development of a single application by a single project team [12]. The main reason to include project management is that VSEs core business is software development and their financial success depends on project profits.

**Deployment packages:** However, the issues of assistance to VSEs in understanding and adopting standards, as previously outlined, must be addressed. To this end, some members of the ISO/IEC JTC1/SC7 WG 24 have produced a set of "Deployment Packages" (DP) [15]. 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 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.

### 1.3 Research Pilot Project

As stated in [11], support processes provide a specific focused set of activities for performing a specialized software process. A supporting process assists Project Management and Software Implementation processes as an integral part with a distinct purpose, contributing to the success and quality of the software project. An idea argued in this paper is that even if some support activities are existing and distributed in PM and SI processes of the Basic Profile, the Intermediate or Advanced profiles must include additional support processes.

This paper reports some conclusions of a pilot project that authors conducted with a 14-people VSE that builds and sells counting systems about the frequenting of natural spaces and public sites. Only 3 employees are software developers and the VSE asked assistance for the management of software projects – which the emerging ISO/IEC 29110 standard is intended for – and also for its infrastructure management. Infrastructure management, configuration management or documentation management, are examples of support processes which are neither management processes nor implementation processes. Another idea defended in this paper is that implementing support activities in a VSE needs operational mechanisms. The VSE's employees were provided with transfer cards intended to help the VSE to build its own support activities. A transfer cards aims to help the engineer to do [and learn] the task. But it takes in account the necessity for the VSE to become autonomous in supporting its own development process.

This section overviewed SE initiatives for VSEs and related work. The Basic Profile is discussed in section 2, where the pilot project is also introduced. In section 3, we establish a mapping between the ISO/IEC 29110 Process Reference Model (PRM) and the ISO/IEC 12207 PRM; present some facts of the pilot project, and make the proposal to add a support process to the ISO/IEC 29110 initiative. We conclude with short perspectives.

## 2 A Standardized Process Profile for VSE

### 2.1 Basic Profile

**Basic Profile Processes.** Considering deployment of the Basic Profile, Part 5-1-2 [12] defines software implementation and management practices appropriate for characteristics and needs of VSEs described in section 6 of Part 4-1 [16]. The Basic Profile is made of 2 processes: Project Management (PM) and Software Implementation (SI). A process is defined as “a set of interrelated or interacting activities which transforms inputs into outputs [11]”. Table 1 provides the process / activities breakdown, we added the normative references to the ISO/IEC 12207 standard - as it is defined in Clause 8 of the Basic Profile [16]. In italics, were added missing references to the ISO/IEC12207 standard.

**Basic Profile Products.** The purpose of the Project Management (PM) process is to establish and carry out in a systematic way the tasks of the software implementation project, which allows complying with the project’s objectives in the expected quality, time and cost. The PM process also acts as the interface to the outside world. The initiating document for a software project is a *Project Description*, which is also the basis for project planning. Project Management generates a *Project Plan* to direct the software project and establishes a *Project Repository* to store project work products. During the execution of the project, Project Management process receives *Change Requests*, which might cause revisions to the *Project Plan*. The source of a *Change Request* is either one of the processes or Customer.

*Software Implementation (SI)* produces a specified software system implemented as a software product or service [11]. This process starts with the establishment of *Software Requirements*. *Software components* defined by the *Software Design* are produced. *Software* is produced performing integration of *Software components* and verified using *Test Cases and Test Procedures*. The final outcome is a *Software Configuration*, which includes, in addition to executable software and its source, all associated documentation. *Software* shall satisfy architectural design requirements through verification and stakeholder requirements through validation. Defects shall be identified, recorded at the *Verification/Validation Results* and corrected. The customer acceptance is formalized by the *Acceptance Document*.

**Table 1.** Basic Profile Process Breakdown.

29110 Basic Profile Activities	12207:2008 related processes	12207:2008 related activities
PM.1 Project Planning	6.3.1 Project Planning	6.3.1.3.1, 6.3.1.3.2
PM.2 Project Plan Execution	6.4.8 SW Acceptance Support, 7.2.2 SW Configuration Management, 7.2.3 SW Quality Assurance, 7.2.6 Review, and 7.2.8 Problem Resolution	7.2.6.3.3
PM.3 Project Assessment and Control	6.3.2 <i>Project Assessment and Control</i> , 6.3.7 Measurement	6.3.2.3.1, 6.3.2.3.2, and 6.3.2.3.3
PM.4 Project Closure	6.1.2 Supply, 6.3.2 <i>Project Assessment and Control</i>	6.1.2.3.6, 6.3.2.3.4
SI.1 SW Implementation Initiation	6.3.1 Project Planning	6.3.1.3.3
SI.2 SW Requirements Analysis	6.4.1 Stakeholder Requirements Definition, 7.1.2 SW Requirements Analysis	6.4.1.3.1 to 6.4.1.3.5, 7.1.2.3.1
SI.3 SW Architectural and Detailed Design	7.1.3 SW Architectural Design, 7.1.4 SW Detailed Design, and 7.1.5 SW Construction	7.1.3.3.1, 7.1.4.3.1, and 7.1.5.3.1
SI.4 SW Construction	7.1.5 SW Construction	7.1.5.3.1
SI.5 SW Integration and Tests	7.1.5 SW Construction, 7.1.6 SW Integration, 7.1.7 SW Qualification Testing, 7.2.1 Documentation, 7.2.4 SW Verification, and 7.2.5 Validation,	7.1.5.3.1, 7.1.6.3.1
SI.6 Product Delivery	6.1.2 Supply, 6.4.7 SW Installation, 6.4.8 SW Acceptance Support	6.1.2.3.5, 6.4.7.3.1, 6.4.8.3.1

## 2.2 Pilot Project

Whilst space limitations do not allow for a detailed discussion on research methodology, it is worth noting that the purpose of a pilot project is as an exploratory study may be used as a prelude to larger research study and for doing causal investigations and is aimed at elucidating and understanding the internal dynamics of program operations. The primary advantage of this type case study approach is "*it often highlights new insights or ideas and when used as a pilot study, it can sometimes generate ideas and focus research*" [17].

A VSE of 13 people (with 2 software engineers) requested help from University of Brest in Spring 2009. The VSE designs, builds, develops and sells counter system intended to collect and analyze the frequentation of public or private sites. Counting systems are based on stand-alone counter boxes (including sensors, power supply, data storage, and data exchange) and a software chain able to collect, to analyze, to present, and to report counting data. In the previous software chain, sets of counting data were downloaded from counters by infrared link or GSM (Global System for Mobile Communications), were stored on personal computers and shall be exchanged via a file transfer utility.

Due to its clients' requirements and products supplied by the concurrence, the VSE started a complete reconstruction of its software chain in order to transform it in a Web-based system called Eco-Visio, intended to host data of fleets of counting systems for each client and able to process statistics and to generate analysis reports on counting. The VSE hired at the end of June 2009 a graduate in Information Technology of Brest university who performed his final internship in the VSE. At the same moment, the professors of University of Brest visited the VSE and initiated a pilot project intended to transfer a part of the ISO/IEC 29110 standard to the VSE and its specific context.

The pilot project started effectively in Fall 2009. The VSE needs fall in two categories, mid-term and urgent. As stated in [10], the pilot should not be on a critical path, such that failure has a significant adverse impact on the VSE. It was decided with the management of the VSE to split the pilot project in two projects: a pilot project to establish standardized processes based on the Basic Profile; a technological transfer project to be assisted on three software engineering activities: establishment of a robust and perennial technical architecture, improvement of the system reliability with a disciplined test process, set-up of processes on change requests management and configuration management. The first project is still running and should complete at the end of August 2010. The second project completed at the end of March 2010 as the first version of the new Eco-Visio Web-based system was released.

### **3 A need for additional supporting activities**

#### **3.1 Mapping of the ISO/IEC 29110 Basic Profile with the ISO/IEC 12207**

ISO 15504:2004 [18] adopts the ISO 12207:2008 [11] terminology for process group, process and activity. These standards group the “things” that may be performed during the life cycle of a software system into seven process groups. Each of the life cycle processes within those groups is described in terms of its purpose and desired outcomes; each process is also divided into a set of activities; each activity is further divided into a set of tasks; activity and tasks are performed to achieve process outcomes [11] [18]. The terminology of ISO/IEC 29110 is the same because the Basic Profile Part 4.1 [16] is an IS, but there is a shift of level:

- Process Group (12207) maps to Process (29110).
- Process or Lower-Level Process (12207) map to Activities (29110).
- Each of the 6 lower-level processes of the 7.1.1 Software Implementation Process has a single activity whose name is the name of the lower-level process; hence there is no need to map the 12207 concept of Activity regarding the 29110.
- Task (12207) or Base Practices (15504) map to Task (29110).

The Basic Profile addresses quality assurance with tasks, such as review, verification and validation, test and defect correction, and products such as the project plan, test report and verification results. As an example the project plan should include this information: Tasks, including verification, validation and reviews with Customer and Work Team, to assure the quality of work products. Tasks may be represented as a Work Breakdown Structure (WBS). Additional quality assurance tasks and products will be added to the Intermediate and Advance Profiles of the Generic profile group. The Basic Profile does not cover all the activities of the 12207 Software Configuration management process. 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. The storage, handling and delivery of the items are controlled [12].

Hence, except PM.2 Project Plan Execution, SI.5 SW Integration and Tests and Configuration Management tasks, the ISO/IEC 29110 Basic Profile does not refer to and do not have dedicated support processes.

#### **3.2 Back to the Pilot Project**

Ed Yourdon, cited in [19], wrote: “The data processing profession has been slowly learning that technology transfer is one of its biggest problems, if not the biggest problem”. As mentioned in section 3.2, a technology transfer project with the VSE was initiated to resolve urgent problems about architecture, reliability and change management of the software under development. The software release was scheduled for Spring 2010, hence the assistance was on the two former points.

The absence of requirement traceability and systematic testing was rapidly recognized by all stakeholders. A short audit was performed by the second author to identify potential risks of the technical solution chosen by the VSE. Both professors



from University of Brest thought also that project management has to be improved but this point has been deliberately omitted from the scope of this pilot project. A plan of action in two steps was proposed: - 1- implementation of the Deployment Package “Software Requirements Analysis” and - 2 - support to the definition and the establishment of a robust and perennial architecture.

The starting point of the ISO/IEC 29110 use for requirement is a step-by-step approach to perform the required SI.2 tasks is given in the Deployment Package - Software Requirement Analysis [15]. An employee of the VSE integrated a training session on requirements based on the training material associated with this Deployment Package. Lessons learned about this training are reported in [8]. Reliability problems reported by the VSE were attributed to the architecture. The software under development was using the architecture of the Internet provider: Tomcat application server and MySQL database. During the software tests, frequent crashes occur and the VSE was encouraged to invest in its own 3-tier architecture. VSE employees had no skills about databases and application servers’ installation, configuration and administration. Hence, a dedicated assistance approach, presented in the next section, was developed.

### 3.3 Support Processes for VSEs

**Process Reference Model.** The variety of SE standards provides different Process Reference Models that may differ slightly on the composition of Support Processes. Our proposition is to use several infrastructure and support processes, such as Infrastructure management, Methods and tools support, Documentation, Installation and deployment. This decomposition is a pragmatic answer to the lack of support that may be experienced in small-sized project where the main effort is concentrated on project management and software development tasks.

**Transfer cards.** It was decided to use the 15504 standard to provide software engineers with an exemplar model of software engineering support. While documenting a support process, the engineer’s activity was organized in small units of work called a support task. The whole process can be performed by a dedicated support group or distributed on the development team.

The description of the task is designed as a theater scene: the scene is the reference context where action happens; the scene aims at a unity of place, time and action; the scene is together a situation where people do [and learn], a scenario of actions, a role distribution, an area mobilizing resources and means. The different components of a scene are depicted in a standardized transfer card. We will present briefly the main elements of a card.

- Related 29110 Process / Activity (e.g. SI / SI.4 Software Construction) provides a smooth link to the ISO/IEC 29110 and through the Standardized Profiles to the 12207 standard.
- Role (e.g. Designer) is a quick reference to the ISO/IEC 29110 Role.
- Task Title and Objectives is similar to Process Title, Process Purpose, and Process Outcomes as defined in ISO/IEC 12207.
- Step-by-step is a comprehensive description of the work to be done intended to be a practical guide to complete the task. A set of resources required. It may include

the hosting of a technical support (such as Oracle Metalink) that a technology transfer center is able to afford when the cost is out of reach of a VSE.

- Output products are generally a usage or an installation guide. It aims to initiate and develop a strategy of capitalizing and transferring knowledge to the VSE employees.

The concept of transfer cards was an answer to the VSE specific needs that wishes to be accompanied on the architectural design and SW construction activities (SI.3 and SI.4 in the 29110 Basic Profile). Transfer cards is intended to help the engineer to do [and learn] the task. However the concept of transfer cards is also hybridized with a wish (and a necessity) for the VSE to become autonomous in supporting its own development process: beside the support itself, one result of a supporting activity is an output product - a guide - that may be used as an input product for the next occurrence of this support activity.

**Empirical evaluation.** Using and continually enhancing support guides is a virtuous cycle that, in our opinion, is a kind of software improvement. Although no measurements are easy to define and perform to confirm this point, empirical evidence is given by the “customer” satisfaction. Three supporting cards were delivered to the VSE: Set-up of deployment architecture, Set-up of the database architecture, Migration MySQL to Oracle XE. The VSE fully employed the three cards and was able to set-up a 3-tier architecture for its software system. The system is deployed since the end of March 2010 and load testing and application optimization should be soon complete. As an empirical measure of its satisfaction, the VSE asked for similar cards in documentation management with a Content Management System (CMS) and in software configuration management (SCM). The former card is under construction, the DP “Version Control” [15] should be useful as a basis for the latter card.

### **3.4 Support Processes for the ISO/IEC 29110 Intermediate and Advanced Profiles**

**Overview of the proposal.** The empirical results of this pilot project in technology transfer lead us to address a proposal to the WG24 in charge of the 29110 development: add the process “Infrastructure & Support” to the 29110 intermediate and advanced profiles. “Infrastructure & Support” was preferred to “Software Support Processes” (12207) or “Supporting Life Cycle Processes” (15504) because it emphasizes VSE concerns about mastering its infrastructure.

**ISO 12207 Support Processes.** The process category “Software Support Processes” contains 8 processes: 7.2.1 SW Documentation Management, 7.2.2 Configuration Management, 7.2.3 SW Quality Assurance, 7.2.4 SW Verification, 7.2.5 SW Validation, 7.2.6 SW Review, 7.2.7 SW Audit, and 7.2.8 SW Problem Resolution.

Processes of the process category “Organizational Project-Enabling Processes” manage the organization’s capability to acquire and supply products or services through the initiation, support and control of projects [11]. Hence, all processes are, more and less, concerned with support but we will concentrate on those related with SW development: 6.2.1 Life Cycle Model Management, and 6.2.2 Infrastructure Management. Some processes of the process category “Technical Processes” provide

support to the SW team, particularly 6.4.7 SW Installation, 6.4.8 SW Acceptance Support, and 6.4.9 SW Operation.

**Review of the 29110 support activities.** Let us examine how support processes above are taken in account in the 29110.

- As mentioned in the DP “Version Control” [15], Configuration Management is distributed in (almost) all activities of the SI process with a “generic” task: Incorporate the deliverable to the Software Configuration as part of the baseline.
- In a similar manner, documentation management is distributed to the PM and SI activities that produce output documents. Documentation is managed with a “generic” task: Incorporate the document to the Software Configuration as part of the baseline.
- All tasks related with SW testing (including SW verification, validation and acceptance) are localized in activities SI.3, SI.4, and SI.5.
- Reviews are performed within the PM2.3 (internal) and PM2.4 (external) tasks. Audit is not in the scope of the Basic Profile.
- Problem resolution is performed in a VSE on a day-to-day basis and no formalized process can address the variety of problems that VSE employees have to solve.
- Life Cycle Model Management is generally absent in a VSE. Implementing the 29110 in a VSE will provide Life Cycle Model Management..
- The 29110 Basic Profile provide, on purpose, few indications about Infrastructure Management, Installation and Operation Processes; the SI.1.2 task is intended to set or update the implementation environment and the SI.6.6 task is devoted to perform delivery according to Delivery Instruction.

**Synthesis of 29110 Basic Profile support activities.** We synthesize how support activities are addressed in the 29110, we classify them in three categories:

- **Difficult to be applicable**, such as Audit, and Life Cycle Management;
- **Missing**, such as Infrastructure Management, Installation, and Operation;
- **Ubiquitous**, as Configuration Management, Documentation Management.

*Difficult to be applicable* is related to activities out of reach of a VSE. *Missing* qualifies, in our opinion, activities that are not covered by ISO/IEC 29110, but should be. *Ubiquitous* means that the support activities do exist, and are distributed and performed elsewhere (even everywhere) in PM and SI activities and tasks; but these support activities are not identified, defined, maintained and improved.

It is a reasonable (and pragmatic) point of view to distribute the tasks of a SE support activity such as Configuration Management and include it into all others SE activities that involve it rather than isolate the distributed tasks in a dedicated activity. It is a general principle in a VSE to have polyvalent employees able to perform a lot of different tasks. However, when the activity bears high risks or becomes too complex, this approach reaches its limits. A careful study is required to establish which SE support activities fall in the “risky” or “complex” categories and require (at least conceptually) a dedicated support activity.

One of the contributions of this paper is to propose a draft of an “Infrastructure & Support” (IS) process for the Intermediate or Advanced profiles. Table 2 presents a first decomposition in activities and examples of tasks. Table 2 is subdivided in three sections: the first section lists the existing PM process, the second section lists the

existing SI process and the last section lists the proposed IM process. Column 1 links to the 12207 processes, column 2 lists the 29110 tasks concerned (for PM and SI), column 3 proposes a name for the potential support process and column 4 gives some examples of related tasks.

**Table 2.** Breakdown of the proposed Infrastructure & Support Processes.

12207 Processes and activities	29110 tasks	Proposed name for the activity	Exemplar tasks
<i>“Project Management” PM Process</i>			
7.2.2	PM.1.10 PM.1.15	Configuration Process Planning	<ul style="list-style-type: none"> <li>• Develop a SW Configuration Management Plan</li> </ul>
<i>“Software Implementation” SI Process</i>			
7.2.2	PM.2.2, PM.2.3 SI.2.3, SI.2.5	Change Request Management	<ul style="list-style-type: none"> <li>• Develop a change management strategy</li> <li>• Establish the dependencies to other change requests</li> <li>• Implement and track the changes</li> </ul>
7.2.2	SI.2.7, SI.3.8, SI.4.7, SI.5.11, SI.6.7	Configuration Management	<ul style="list-style-type: none"> <li>• Establish a SW Configuration Repository</li> <li>• Maintain SW configuration.</li> </ul>
7.2.1	SI.2.2, SI.3.3, SI.3.5, SI.5.7, SI.5.9, SI.6.4	Documentation Management	<ul style="list-style-type: none"> <li>• Establish standards for documents</li> <li>• Establish a Documentation Repository</li> <li>• Maintain documents.</li> </ul>
6.4.7 6.4.9		Installation and Deployment	<ul style="list-style-type: none"> <li>• Develop installation strategy</li> <li>• Specify installation requirements</li> <li>• Install software product</li> </ul>
<i>“Infrastructure &amp; Support” IM Process</i>			
6.2.2		Infrastructure Management	<ul style="list-style-type: none"> <li>• Establish the infrastructure</li> <li>• Maintain the infrastructure.</li> </ul>
6.4.7 6.4.9		Experience Management	<ul style="list-style-type: none"> <li>• Establish an experience repository</li> <li>• Gather knowledge and experience</li> <li>• Provide access to experience</li> </ul>

## 4 Conclusion and future work

We presented a pilot project conducted in a VSE of 13 people who wished to master its infrastructure. Two hypotheses were discussed (1) supporting activities can be transferred to the VSE through the use of transfer cards - a kind of self-training activity; (2) the 29110 Intermediate or Advanced Profiles need additional Support Process drawn from the 12207 and 15504 standards. As a case study, the solutions that we provided to the VSE are reported.

Further work is required to develop and pilot the required supporting activities and to determine how far the scope of transfer cards can be extended to non-supporting processes such as requirements or tests.

## References

1. Laporte, C.Y., Alexandre, S., O'Connor, R.: A Software Engineering Lifecycle Standard for Very Small Enterprises. In: R.V. O'Connor et al (eds) EuroSPI 2008. CCIS, vol. 16, pp. 129-141. Springer-Verlag, Heidelberg (2008)
2. Coleman, G., O'Connor, R.: Investigating Software Process in Practice: A Grounded Theory Perspective. *Journal of Systems and Software* 81(5), pp 772-784 (2008)
3. Richardson, I. , Wangenheim, C. G.: Why are Small Software Organizations Different?. *IEEE Software* 24 (1), pp.18-22 (2007)
4. Oktaba, H.: MoProSoft: A Software Process Model for Small Enterprises. In: 1<sup>st</sup> International Research Workshop for Process Improvement in Small Settings, pp. 93–101. Carnegie Mellon University, Pittsburgh (2006)
5. McCaffery, F., Taylor, P., Coleman, G.: Adept: A Unified Assessment Method for Small Software Companies. *IEEE Software* 24 (1), pp.24-31 (2007)
6. Revankar, A., Mithare, R., Nallagonda, V. M.: Accelerated Process Improvements for Small Settings. In 1<sup>st</sup> Int. Research Workshop for Process Improvement in Small Settings, pp. 135–144. Carnegie Mellon University, Pittsburgh (2006)
7. Forrester, E.: A Process Research Framework - The International Process Research Consortium. Carnegie Mellon University, Pittsburgh (2006).
8. Ribaud, V., Saliou, P., Laporte, C. Y.: Experience Management for Very Small Entities: Improving the Copy-paste Model". In: Fifth International Conference on Software Engineering Advances, IEEE Press, New York (2010)
9. International Organization for Standardization (ISO): ISO/IEC DTR 29110-1 Software Engineering - Lifecycle Profiles for Very Small Entities (VSEs) -- Part 1: Overview. ISO, Geneva (2010)
10. Laporte, C. Y.: Contributions to Software Engineering and the Development and Deployment of International Software Engineering Standards for Very Small Entities. PhD thesis of the Université de Bretagne Occidentale, Brest (2009), <http://tel.archives-ouvertes.fr/tel-00483255/fr/>
11. International Organization for Standardization (ISO): ISO/IEC 12207:2008 Information technology -- Software life cycle processes. ISO, Geneva (2008)
12. International Organization for Standardization (ISO): ISO/IEC DTR 29110-5-1-2 Software Engineering - Lifecycle Profiles for Very Small Entities (VSEs) -- Part 5: Management and Engineering Guide-Basic VSE Profile. ISO, Geneva (2010)
13. International Organization for Standardization (ISO): ISO/IEC DTR 29110-3 Software Engineering - Lifecycle Profiles for Very Small Entities (VSEs) -- Part 3: Assessment Guide. ISO, Geneva (2010)
14. International Organization for Standardization (ISO): ISO/IEC FDIS 29110-2 Software Engineering - Lifecycle Profiles for Very Small Entities (VSEs) -- Part 2: Framework and Taxonomy. ISO, Geneva (2010)
15. Deployment Packages Repository. Available from <http://profs.logti.etsmtl.ca/claporte/English/VSE/index.html>
16. International Organization for Standardization (ISO): ISO/IEC FDIS 29110-4-1 Software Engineering - Lifecycle Profiles for Very Small Entities (VSEs) -- Part 4: Specification-VSE Generic Profile Group. ISO, Geneva (2010)
17. Patton, M., "How to Use Qualitative Methods in Evaluation. Sage, London (1987)
18. International Organization for Standardization (ISO): ISO/IEC 15504 Information technology -- Process assessment. ISO, Geneva (2004)
19. Bouldin, B.: Forward, Agents of change. Yourdon Press, New Jersey (1989)