An Implementation Approach of ISO/IEC 29110 for Government Organizations

Veeraporn Siddoo, Noppachai Wongsai, and Rattana Wetprasit
Department of Information Technology and Communication, Faculty of Technology and Environment,
Prince of Songkla University, Phuket Campus, Thailand

Abstract
Organizations in software industry become aware that adopting comprehensive software process such as Capability Maturity Model Integration (CMMI) and ISO12207 can improve their process and produce high degree of quality software. However, successful implementation of such expensive and complicated Software Process Improvement (SPI) methods in small organizations is generally not possible because these organizations have limited budget and resources. ISO/IEC 29110 is mainly developed for Very Small Entities (VSEs) — up to 25 employees’ organization — which reduces process complexity, cost of implementation and time overhead, but still remains the essential elements of effective processes that finally improve the performance. This paper reviews ISO/IEC 29110, and goes into details of implementing preparation and approach, as well as lesson learnt from adopting this standard into a small IT department in government academic institute based on analysis, recommendations and experiences in the aim of proposing quality process schema that suitable for such kind of organizations.

Keywords: ISO/IEC 29110, Very Small Entities (VSEs), Software Process Improvement (SPI)

1. Introduction
Process (Humphrey, W., 1995), generally, is a set of interconnected tasks that aims to achieve a definite goal when it is adopted and implemented. These sequential tasks may involve all kinds of resources in the organization such as humans, methods and tools required in implementing the procedures specific in the process. The good process is considered as a set of repeatable tasks producing the same result over time.

In software engineering, software process improvement (SPI) (Mathiassen, L.; Ngwenyama, O.K.; Aaen, I, 2005) (Sihvonen, H.-M.; Jantti, M, 2011) refers to activities intending at improving the software development process and is used for reaching a desired improvement objective. SPI methods, associated with standard models, are the instruments used for guiding and managing improvement activities in practice. Many software houses, including in-house development organization, are aware of its benefits and are willing to use it to improve their software process. SPI is generally adopted with the aim of improving software development practices in the organization to meet a certain level of standard. It should be performed gradually by transitions between specific maturity levels. Moreover, there is necessary to collect all information and any kind of knowledge produced during SPIs implementation, and it is essential to continuously evaluate and improve the software process. There is quite a number of currently available maturity process models and standards, such as Capability Maturity Model Integration or CMMI (M. C. Paulk, C. V. Weber, M. B. Chrissis, 1995), ISO12207 (ISO/IEC 12207:2008), and ISO 15504 (ISO/IEC TR 15504, 1998). They are most likely the commonly required and applicable SPI models to the enterprises with all sizes. There is, however, the issues which draw out that successful implementing such expensive and complicated framework in small enterprises is generally not possible when it comes to the fact that most Small & Medium
Enterprises (SME) are reluctant to embrace initiatives. The huge effort to develop, large implementation costs and significant investment are required in SPI implementation. And this matter is also taking place in software industry in Thailand.

Since 1998, Software Park Thailand has initiated the concept of Software Capability Maturity model (SW-CMM) to support the Thai software companies, and then moved to the CMMI. But most of software companies are small. It is generally impossible for those SMEs to implement the CMMI (O’Connor & Gerry Coleman, 2009). Due to this difficulty and drawback, Software Industry Promotion Agency (Public organization) or SIPA, under the administrative supervision of the Ministry of Information and Communication Technology (MICT), has advocated the Thai Quality Software (TQS) that is referred, simplified and strongly related to those defined in the ISO12207. This TQS is mainly developed for Thai software industry with a focus on enhancing quality of software products and improving software development capability in order to reach the certain level of market opportunity in international competition. Subsequently, TQS is promoted to be an international standard called ISO/IEC 29110 Software Engineering — Lifecycle Profiles for Very Small Entities (VSEs) in January 2011. In order to encourage practicing ISO/IEC 29110, SIPA supports the budget to certify this VSEs standard for the Thai software industry, as well as the government agencies.

This paper presents the model of implementing the ISO/IEC 29110 in the Thai government agencies. There are main 4 sections in this paper to represent what the work can be contributed. The first section is the current introduction. The review in complete detail of ISO/IEC 29110 can be found in the second section to illustrate the key elements of this standard. In Section 3, the characteristics of the organization, that is chosen to be the case study, restrictions and the reasons for choosing this standard were discussed.

The highlight of presentation is the implementation approach presented in Section 4. The final section of this paper introduced further research concepts that lead to the concerned issues of ISO/IEC 29110 in the interesting field.

2. ISO/IEC 29110 Software Development Standard

ISO/IEC 29110 is a Software Engineering Standard for very small enterprises (VSE) targeted particularly for software organizations with 25 people and less. Inclusive processes introduced in this standard are sufficient and suitable for small organizations to be able to gradually evolve their quality development. ISO/IEC 29110 Basic Profile (29110-5-1-2:2011, 2011) consists of two main processes: Project Management (PM) and Software Implementation (SI) as shown in Figure 1 (29110-5-1-2:2011, 2011).

![](image)

Figure 1 PM and SI relationship.

The main objective of the Project Management process is to define the start point of the project, establish the systematic tasks of software implementation and carry out project management approach in order to achieve the satisfactory outcome.

The PM process comprises with four core activities as seen in Figure 2 (29110-5-1-2:2011, 2011). (PM.1) Project Planning: the list of project’s execution is developed regarding the Statement of Work and is validated with the Customer. Staff and other IT assets (e.g., hardware and software) are allocated. Tasks cost, and time are estimated for completing the work. (PM.2) Project Plan Execution: the plan...
is carried out. Meetings between staffs and customer are regularly conducted and recorded in the Meeting Record. Any new tasks, commitments, agreements and changes of software requirements raised in the meeting are registered and be able to track when required. (PM.3) Project Assessment and Control: progress status of the project is monitored against the project plan and is recorded to ensure that all tasks assigned to staff are finished by the schedule. Risk Assessment and Mitigation, and Software Quality Assurance are conducted in order to certify that work products and processes complying with the Project Plan and Requirements Specification and project scope are still maintained. (PM.4) Project Closure: the project is closed. Acceptance Record is issued to customer for official project closure sign off. All work products are finalized, collected and sent to maintenance team for software supporting if necessary.

Figure 2 Project Management process.

In Software Implementation process, the sequence of analysis, design, construction, integration and tests activities stated in project plan are performed. The SI process comprises with 6 core activities as seen in Figure 3 (29110-5-1-2:2011, 2011). (SI.1) Software Implementation Initiation: project plan is reviewed for scheduling software development. (SI.2) Software Requirements Analysis: software requirements are defined, analyzed for correctness and testability regarding the initial requirement stated in the Validation Result approved by the Customer. (SI.3) Software Architectural and Detailed Design: software architecture and software design is developed and base-lined. Test Cases and Test Procedures primarily developed. Software components designed in this activity are matched with the Requirements Specification and the Test Cases. This will provide consistency and traceability of constructing and testing software that most fulfills customer requirements. (SI.4) Software Construction: designed components are implemented. Software Version Control may be taking into account to ensure software modifications and releases are controlled. Project Repository and Project Repository Backup are required for handling and storing developing software. (SI.5) Software Integration and Tests: integrating of software components is taking place and Test Cases and Test Procedures are used to verified component integration, as well as system integration. The result of testing is recorded in Test Report. (SI.6) Product Delivery: complete products are delivered to customer. This does not only mean software product, but also all relevant documents required by customer, user and maintenance team. The list of Work Products generated in PM and SI process can be summarized in the Table 1 (29110-5-1-2:2011, 2011).

3. Organization Characteristics

IT infrastructures and equipments in small Thai government agencies are typically provided by their headquarters. On another hand, large departments usually have their own internal IT unit that usually develops their own IT innovation. However, in-house software development in Thai government agencies often faces with many difficulties. For example, the problem about getting themselves recruits, high turnover because of unattractive
wage in Thai bureaucracy and indiscipline working atmosphere. Furthermore, most understand the development process and do not apply Software Engineering discipline in their software development government agencies still do not entirely. Consequently, the finished system often unfulfilled the user requirements. Moreover, general management problems could normally be found in Thai government agencies (Kanchit Malaiyongse, 2003). For example, lack of assessment on organization’s preparedness and required IT assets for the project, lack of support from executives and lack of discipline and awareness of quality software development.

Figure 3 Software Implementation process.

Faculty of Technology and Environment, Prince of Songkla University, Phuket Campus is one of the government agencies that has its own in-house development entity (Faculty of Technology and Environment, 2012). A group of 4-5 developers in the software development unit under the direction of the Faculty, has developed the Management Information System (MIS) to assist the faculty business. The System Development Life Cycle or SDLC (John W. Satzinger, Robert B. Jackson, Stephen D. Burd, 2006), depicted in Figure 4, combined with Prototyping Software Development processes were used in a number of small projects in several occasions as appropriate. However, the group of developers never completely applied Software Engineering and Project Management disciplines into the software development projects. Hence, the unit came to a decision to adopt the ISO/IEC 29110 in the aim of enhancing the current process. And this unit is chosen to be a case studied in this research.

Figure 4 System Development Life Cycles.

4. Implementation Approach

After studying the ISO/IEC 29110 processes, meeting with all project participants and consulting SPI experts, we came up with the 3-steps approach; (1) Feasibility Study, (2) Risk Management and (3) Execution, for applying the ISO/IEC 29110 in the faculty’s software development unit as seen in Figure 5.

4.1 Feasibility Study (Khoong, C.M.; Ku, Y.W., 1994): we were officially assigned for fully taking responsibility on the project. This step emphasized on finding the possibility of adapting ISO/IEC 29110 standard into existing software processes used by the development unit. Questions were listed, for example, how much did we understand the existing software development process?, how much did we know about the ISO 29110 standard?, what were the organization’s strength that would assist the adoption?, and what were the disadvantages that may impede the standard implementation?. This would allow the working group to evaluate the readiness of the organization concerning whether or not the standard training and expert consulting were needed.
4.2 Risk Management (Beroggi, G.E.G.; Wallace, W.A., 1994): all risks were identified and evaluated based on the results of the feasibility study in order to manage the risk that may occur during the implementation. The risk management plan was created and the risk mitigation was defined.

4.3 Execution (Phillips, D., 2004): after considering the feasibility study result and the risk management report, the implementation of the ISO/IEC 29110 processes (PM and SI) was executed. All constraints defined in the previous steps were reviewed and monitored throughout the implementation plan.

4.4 Knowledge gain in Feasibility Study

Establishing committee: we found that this action should be seriously considered, especially in the bureaucracy working process. Establishing the software project working group with specified roles and responsibilities is important. Moreover, the working group should comprise of the IT development team and all kinds of key stakeholder such as users and project sponsors.

Experiences and skills in the software development process and in IT project management should be taken into account when recruiting the IT development team. In general, the role of Project Sponsor, Project Manager, System Analyst Programmer, Tester, and Technical Leader should be assigned into the working group.

Study and Training: the main point of this activity was to study the objectives, input and outcome of the ISO/IEC 29110 standard processes, and also all work products generated during the process execution for the ISO certificate assessment preparation.

The relationship between activities, work products, roles (activity actors and work product owners) and processes should be illustrated and pointed out their significance. The technique of As-Is analysis (Marc I Kellner, Raymond J Madachy, David M Raffo, 1999) is recommended to perform in order to capture the current working process status, and to gain information on what the problems are, where the root causes of the problems are, what and how things should be done. To-Be analysis (Marc I Kellner, Raymond J Madachy, David M Raffo, 1999) also should be performed to plan the future implementation and get ready for the ISO certificate assessment. We discovered that these two techniques would help the working team to determine the missing process or work product that need to be filled in. Moreover, an activity that found to be valuable for the standard initiative is the assessment simulation with the ISO experts. This pre-assessment imitation gives the working group the self-assessment check list with both close and open questions - should be used to validate the software process’s knowledge of working group members. The questions may lead to the same answers that have been raised by the ISO experts during the pre-assessment imitation.

Self-Assessment: In order to ensure that the working group sees the same picture of the ISO certificate assessment process, the self-assessment check list - with both close and open questions - should be used to validate the software process’s knowledge of working group members. The questions may lead to the same answers that have been raised by the ISO experts during the pre-assessment imitation. The technique of Gap Analysis (Prakash Pol, Madhup Paturkar, 2012) could be used based on the As-Is and To-Be analysis results to identify what are still misplaced and what should be taken into considerations in the
Focus Group Discussion (Zarinah M. K. and Siti Salwah S., 2009).

**Report:** the report of current status of organization’s software development is strongly encouraged to conduct. The result of self-assessment and the summary of issues in the Focus Group Discussion (Zarinah M. K. and Siti Salwah S., 2009) should be used and logically analyzed in these As-Is and To-Be Gap Analysis reports. With the report outcome, the working group will be able to know the current software process status, to plan what must be done in the future, and to propose the approach to improve current software process.

### 4.5 Lesson learned in Risk Management

**Risk Management** (Beroggi, G.E.G.; Wallace, W.A., 1994) is highly recommended to implement along with the ISO implementation. Difficulties and obstructions regularly occur during the project action time and hard to avoid. Therefore, analysis and assessment of potential risk (Jung-Ho Eom; Young-Hyun Choi; Seon-Ho Park; Tai-Myoung Chung, 2010) can be used as tools to manage such obstacles.

**Risk Analysis** (Jung-Ho Eom; Young-Hyun Choi; Seon-Ho Park; Tai-Myoung Chung, 2010): we realized that the information about resources, technologies and working processes in the Gap Analysis report is useful when it comes to the risk analysis activity. It makes easier to identify risks in relevant aspects.

**Risk Assessment** (Jung-Ho Eom; Young-Hyun Choi; Seon-Ho Park; Tai-Myoung Chung, 2010): all possible risks should cautiously be rated and then their solution or prevention should be proposed in
order to reduce and mitigate their impacts during the execution phase.

We learned that these two activities were the most vital keys to achievement. To keep away from painful implementation failure or project’s time extension, risk analysis and assessment need to be performed and reported frequently at appropriate milestones.

4.6 Management Experience in Execution

Implementation: The initial user requirements and work commitment stated in the Statement of Work were the start point of the ISO implementation. The PM and SI processes should be strictly followed. Once again, the simple validation technique like Check List of work products (documents and activities) is used to control the quality and accuracy in every step of the ISO implementation. The Gap Analysis Report, Focus Group Report and Risk Management Report are maintained. Residual risks should be continually monitoring, and sudden risks that might occur during the execution have to be recorded every single occasion and reported to all stakeholders. Changes in policy, requirement, even implementation plan may have to be made, and resources allocation may have to reconsider in order to response on both kinds of risk.

Review and Monitor: Monitoring and assessment of work products are suggested as the technique for controlling completeness and correctness according to the ISO standard in this activity. Another necessary activity that we strongly recommend to consider is entity knowledge management. Lesson learned document recording problems and solutions, experiences and knowledge gain from the ISO implementation project will value for the next ISO/IEC 29110 software development project.

5. Conclusion and Further research

With the three-step implementation approach, the case study unit gained a better understanding in the ISO/IEC 29110 standard, subsequently was able to effectively handle and prepare documents under the standard. As a result, the unit has a clear, well-defined, step-by-step approach in the software development that leads to a better reputation of the organization. Sustainable uses of the ISO/IEC 29110 standard in the organization depend on the executives’ attitude toward the standard and the working group’s recognition in effectively developing software. Since this standard is fairly new and thus its study is limited. Our findings pointed out the significance of such standard with respect to apply its process in in-house software unit under government agencies.

Further study should consider other aspects. Evaluating the software development unit performance should be done and applying the ISO/IEC 29110 in other projects that used other developing model such as Agile, Incremental and RUP model. And this will lead us to develop the implementing package and tools suitable for different government agencies profile.

6. References


