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

Methodology for ISO/IEC 29110 Profile Implementation in EPF Composer

Alena Buchalcevova, Department of Information Technologies, Prague University of Economics, Czech Republic

ABSTRACT

The article presents the ISO/IEC 29110 Profile Implementation Methodology that was developed to manage consistent implementation of individual ISO/IEC 29110 Profiles in the open-source content management tool Eclipse Process Framework Composer. Such an implementation enables effective managing of the standard and its publishing in the form of a web application that can be easily and efficiently used. This methodology represents an example of the usable outputs of the ISO/IEC 29110 standard being utilized in education and research in the Czech Republic. Its main elements described in this article can be also used for implementation purposes in other countries. First, the methodology structure is presented, followed by its individual elements, i.e. General Principles, Profile Structure, Profile Element Mapping, Implementation Conventions, EPF Composer Usage Guidelines, and Implementation Process. The evaluation of this methodology was performed during the implementation of the Entry Profile.

KEYWORDS

Czech Republic, Diffusion, Eclipse Process Framework Composer, Methodology, Small Companies, Software Process Improvement, Standard

INTRODUCTION

Although software systems play a key role in today's society, software projects do not achieve that kind of success their role predicts. According to several surveys (Johnson, 2006; Ambler, 2013), the ratio of successful software projects ranges to 60%, while the rest is categorized as challenged or failed. Software Process Improvement (SPI) represents a way of improving a status of software development. International standards like ISO/IEC 12207 (ISO/IEC 12207, 2008) ISO/IEC 15289 (ISO/IEC 15289, 2006), ISO/IEC 15504 (ISO/IEC 15504. 2004), and ISO 9001 (ISO 9001, 2008) play an important role in SPI initiatives as companies are willing to show compliance with common business rules. However, according to several surveys (Analecto et al, 2004; Laporte et al, 2008), small companies consider implementation of international standards quite difficult as they lack sufficient resources in terms of number of employees, budget and time (Coleman & O'Connor, 2008). This fact is also confirmed in research studies (O'Connor & Coleman, 2009; Sánchez-Gordón & O'Connor, 2015). To overcome difficulties small companies face when implementing heavyweight standards and to enable these companies to be recognized as entities that produce quality software, the ISO/IEC 29110 Systems and Software Engineering - Lifecycle Profiles for Very Small Entities (VSEs) standard is being developed by the ISO community. According to the recent systematic literature review focused on the ISO/IEC 29110 standard (Moreno-Campos et al., 2014), an increasing interest in the standard was confirmed as well as a need of further research. To support standard usage in the Czech Republic, such issues like standard localisation, accessibility, and implementation support are of high importance.

DOI: 10.4018/IJITSA.2017010104

That is why a number of initiatives promoting the use of the ISO/IEC 29110 standard in the Czech Republic have been carried out by the author of this article. Besides localizing the standard as well as all Deployment Packages, the standard was also implemented in an open source content management tool, the Eclipse Process Framework (EPF) Composer. This allows to effectively manage the standard even when changed and above all to publish it online as a web application that can be easily and efficiently used. Implementation of individual Profiles of the ISO/IEC 29110 standard in the EPF Composer poses a challenge itself due to the complexity of this particular tool. Also, it is necessary to assure a consistent implementation of all Profiles to reach a valid overall usage of the standard. So far, two Profiles in the area of software engineering have been published, i.e. Entry and Basic Profile, and the remaining are under development. To manage further profile implementation and assure its high quality and coherence, the ISO/IEC 29110 Profile Implementation Methodology presented in this article was developed. This methodology represents an example of the usable outputs of the ISO/IEC 29110 standard being utilized in education and research in the Czech Republic. Moreover, it does contribute to a further diffusion of the standard. This article is structured as follows. Following the Introduction, the ISO/IEC 29110 standard is introduced and its application in the Czech Republic is discussed. Further, the EPF Composer is described as the tool utilized for the implementation of the standard. The ISO/IEC 29110 Profile Implementation Methodology is presented including its basic principles, conventions, guidelines and implementation process and then evaluated. Lastly, the concluding remarks are outlined.

ISO/IEC 29110 Standard

In order to help small companies to improve their software processes and be recognized as entities that produce quality software, the ISO/IEC 29110 standard is being developed. The term "very small entity" (VSE) was defined by the ISO/IEC JTC1/SC7 Working Group 24 and consequently adopted in the emerging ISO/IEC 29110 standard meaning "an entity (enterprise, organization, department or project) that has up to 25 people". At first, the Working Group 24 focused on developing a standard in the field of software engineering, while it successively expanded its focus also on the area of systems engineering and services. The standard being developed in the area of systems engineering is further described in (Laporte & O'Connor, 2014). The entire history of the ISO/IEC 29110 standard development is then outlined in (O'Connor & Laporte, 2014).

The ISO/IEC 29110 standard's structure is as follows. Part 1 Overview (ISO/IEC 29110-1, 2010) explains main concepts, terms and structure of the standard. Part 2 Framework and Taxonomy (ISO/IEC 29110-2, 2010) then presents the principles and mechanism of building the VSE Profiles that represent a key concept of the ISO/IEC 29110 standard. As a starting point, the "Generic" Profile Group was defined being applicable to a vast majority of VSEs that do not develop critical software. Within the Generic Profile Group, four VSE Profiles were proposed, i.e. Entry, Basic, Intermediate, and Advanced. By using these Profiles, very small companies have the chance to improve their processes in a clear and stepwise manner. Part 3 Assessment Guide (ISO/IEC 29110-3, 2010) then defines the process assessment guidelines and compliance requirements needed to meet the objectives of defined VSE Profiles. This part of the standard is used by certified assessors to perform a VSE assessment. Part 4 Specifications of VSE Profiles provides a mapping to the source standards, e.g. ISO/IEC 12207, and is very useful for method developers and assessors (ISO/IEC 29110-4, 2010). On the contrary, Part 5 Management and Engineering Guide is intended for VSEs themselves and comprises technical reports for each Profile, e.g. Entry Profile (ISO/IEC 29110-5-1-1, 2012), Basic Profile (ISO/IEC 29110-5-1-2, 2011). Regarding the VSE Profiles, the Basic Profile intended for a single project with no special risks or situational factors (Clarke & O'Connor, 2012) (Jeners et al, 2013) was developed first and then published. As particular pilot projects of the Basic Profile implementation in VSEs showed, this Profile was still quite difficult for some companies to implement. For this reason, the Entry Profile was developed as a simplification (ISO/IEC 29110-5-1-1, 2012) that applies to a six person-months effort or start-up VSE. The Entry and Basic Profiles are published by the ISO and can be utilized

in practice. The other two Profiles are still under development. The Intermediate Profile is intended for a VSE which handles more than one project at a time, and therefore is aware of assigning project resources and monitoring projects to accomplish business objectives and customer satisfaction. As the last step, the Advanced Profile is proposed to supply business management practices.

To help a VSE with an implementation of the Entry and Basic Profiles, a series of Deployment Packages were developed and offered free of charge (Deployment Packages repository, 2015). A Deployment Package acts as a detailed methodology that guides a company through the process of Profile implementation (O'Connor, 2015). A typical Deployment Package includes process descriptions, activities, tasks, roles and products, templates, checklists, examples, references and mapping to the standards and models, and a list of supporting tools (O'Connor & Laporte, 2010).

STANDARD DIFFUSION AND USAGE IN THE CZECH REPUBLIC

As education of future developers is an important prerequisite for increasing process quality in software development in practice, an informal interest group about education was established within the Working Group 24. This group developed a set of ISO/IEC 29110 courses intended for undergraduate and graduate students in various countries (Laporte, 2015; Laporte et al., 2015; Laporte & O'Connor, 2015). Such courses also formed the basis for university courses at the Prague University of Economics on the undergraduate as well as graduate level. The localisation of the standard poses a great challenge for non-English speaking countries. To tackle this challenge in the Czech Republic, the students of the graduate course Software Process Improvement held by the author of this article translated freely available parts of the standard as well as all Deployment Packages into the Czech language. Similar process of the Deployment Package translation was also conducted for example in Peru (Garcia et al., 2015). Moreover, the Entry and Basic Profiles were implemented in the Eclipse Process Framework Composer within two diploma theses at the Prague University of Economics (Dlugosova, 2014; Simko, 2015). Also, a central website containing all standard related resources was developed and is accessible at http://spicenter.vse.cz/. In addition, the Wikipedia page about the ISO/IEC 29110 standard was translated into Czech as well, accessible at http://cs.wikipedia.org/wiki/ISO_29110.

For the standard to be widely applied, it is its availability and form that play a key role. Instead of a text document, it is more suitable to have the standard and its Profiles accessible as a web application like e.g. the Rational Unified Process methodology. To develop such an application, a specific tool is favourable to be used. The Eclipse Process Framework Composer is an open-source tool accessible at (Eclipse Process Framework Downloads, 2016) that enables to implement individual processes of a standard and then deploy it as a website. This tool is sophisticated and complex providing a number of options for the implementation. It is necessary to select an appropriate representation of individual VSE Profile elements within the Eclipse Process Framework Composer to assure the implemented VSE Profile can be correctly and effectively utilized. Given the fact that the ISO/IEC 29110 standard is based on several VSE Profiles, a consistent implementation of all Profiles is crucial to reach a valid overall usage. So far, two Profiles have been published, i.e. Entry and Basic Profile, and the remaining are under development. To manage further profile implementation and assure its high quality and coherence, the ISO/IEC 29110 Profile Implementation Methodology was developed. The following section briefly introduces the Eclipse Process Framework Composer.

ECLIPSE PROCESS FRAMEWORK COMPOSER

The Eclipse Process Framework (EPF) Composer is a free, open-source tool for enterprise architects, programme managers, process engineers, project leads and project managers to implement, deploy, and maintain processes of organisations or individual projects (Tuft, 2010).

The EPF Composer provides process-engineering capabilities for selecting, tailoring, and rapidly assembling processes for concrete projects. Besides the catalogues of pre-defined processes, the EPF

Composer allows process engineers to set up their own organisation specific process libraries. This is an opportunity for the implementation of the ISO/IEC 29110 standard. The processes created with the EPF Composer can be then published and deployed as a website.

The most fundamental principle in the EPF Composer is a separation of reusable core method content from its application in processes. Almost all of the EPF Composer's concepts are categorised along this separation. Method content describes what is to be produced; the necessary skills required and step-by-step explanations describing how specific development goals are achieved. These method content descriptions are independent from a development lifecycle as processes describe the development lifecycle. Processes take the method content elements and relate them into semi-ordered sequences that are customised to specific types of projects (Tuft, 2010).

Method content is primarily expressed using Work Products, Roles, Tasks, and Guidance. Guidance, such as Checklists, Examples, or Roadmaps, can be also defined to provide exemplary walkthroughs of a process. Among the elements representing processes in the EPF Composer are Activities, Delivery Processes, and Capability Patterns. Main process element is an Activity that can be nested to define Breakdown Structures as well as related to each other to define a Flow of Work. Activities also contain descriptors that reference method content, e.g. Roles, Work Products, etc. Activities are used to define processes. Two main kinds of processes are supported within the EPF Composer, i.e. Delivery Processes and Capability Patterns. Delivery Processes represent a complete and integrated process template for carrying out a specific type of project. They describe a complete end-to-end project lifecycle and are used as a reference for running projects with similar characteristics. Capability Patterns are then processes that express and communicate process knowledge of a key area of interest such as a discipline or best practice. They are also used as building blocks to assemble Delivery Processes or larger Capability Patterns (Tuft, 2010).

ISO/IEC 29110 PROFILE IMPLEMENTATION METHODOLOGY

To manage further profile implementation and publication of the ISO/IEC 29110 standard and assure its uniformity, high quality and coherence, the ISO/IEC 29110 Profile Implementation Methodology was developed and in detail described in (Simko, 2015).

The methodology structure is depicted in Figure 1 consisting of 6 elements: General Principles, Profile Structure, Profile Element Mapping, Implementation Conventions, EPF Composer Usage Guidelines, and Implementation Process. Each element is described in the following sections.

General Principles

The methodology defines 10 principles to follow while implementing the VSE Profiles in the EPF Composer. Each principle is described in a predefined structure: name, explanation, rationale, a way of implementation in EPF Composer, example, and related principles. The principles are complementary and do not exclude other principles when applied. This set of principles is characterised below.

1. **Bottom-up Principle:** When implementing a Profile in the EPF Composer, it is necessary to proceed from bottom to up. That means that basic elements are to be created first, i.e. pages, their parent category and finally main category which covers the entire content. All elements should be supplemented with metadata and their own content (text, images, tables). Once all the elements are created, their individual relations are established and they are assigned to a particular process. The Bottom-up Principle is shown in Figure 2.
2. **Process Assignment Principle:** According to this principle, each content category needs to be assigned to the appropriate process defined within the Profile, i.e. for example to the Project Management or Software Implementation process in the case of the Entry or Basic Profile.

Figure 1. ISO/IEC 29110 Profile Implementation Methodology Structure, source: author

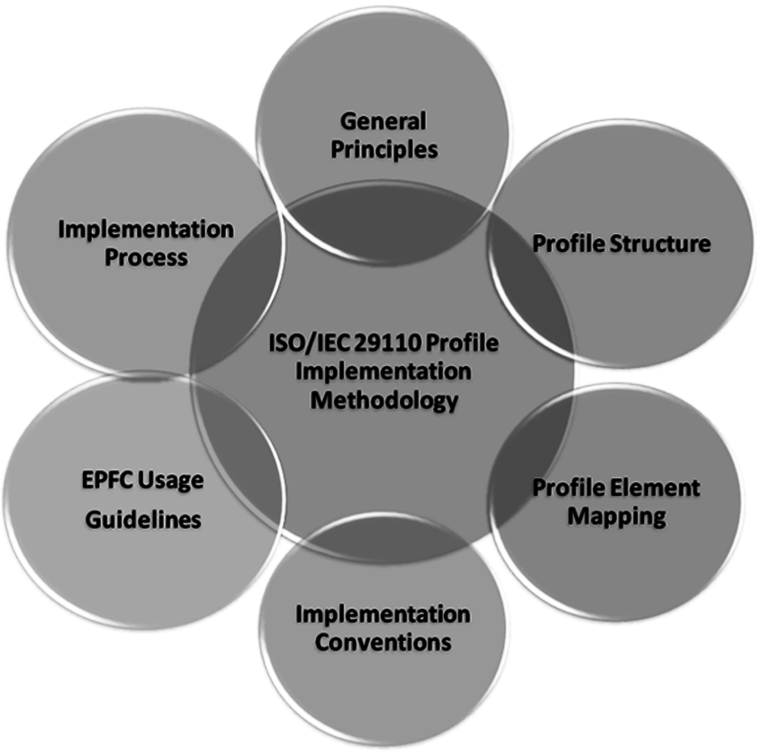
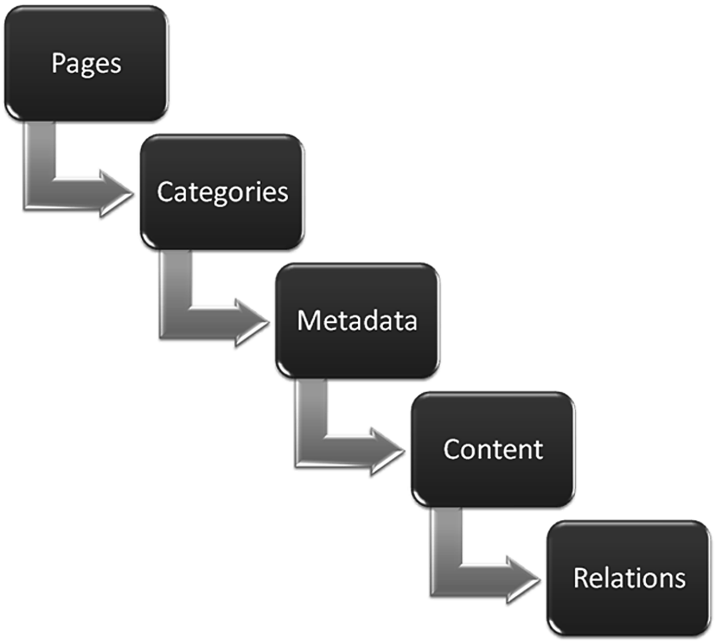


Figure 2. Bottom-up Principle, source: author



3. **Multiple Views Principle:** The methodology leads us to look at the elements from different views and perspectives. The Profile in a printed form is a comprehensive text, which is quite difficult for the user to navigate through and lacks an emphasis on their clarity and coherence. The Profile Implementation Methodology thus suggests to apply the Multiple Views Principle and look at the Profile from various perspectives, i.e. Goals, Roles, Work Products, etc.
4. **Reference Principle:** The methodology advises using hyperlinks to refer to methodological content or external sites that appropriately extend and/or supplement the content.
5. **Predefined Category Priority Principle:** This principle suggests that if predefined content categories do exist, they should be applied. Only in the case that the EPF Composer does not contain a required category, a new category can be created.
6. **Completeness Principle:** The Completeness principle assures that an electronic version of the Profile includes all the information and relations that are present in the printed version.
7. **Correspondence Principle:** According to the Correspondence principle, the published and printed version of the Profile should correspond with and further supplement each other. The electronic version is divided into two sections. The first one follows the printed version of the Profile, while the second one expands the Profile with other elements, relations and context. The user has the option to compare the published and printed version in the first section. The second section then provides him with the possibility to discover and understand various links that are not apparent from the first section.
8. **Formatting Consistency Principle:** This principle assures that the electronic version follows the format of the printed version. As the formatting in the text within the printed version is used to distinguish various elements of the Profile and point out to specific relations between the elements, it is necessary to manage same text formatting in both versions.
9. **Order Consistency Principle:** According to the Order Consistency principle, the electronic version should follow the exact order of the elements established in the printed version.
10. **Desired Duplicity Principle:** The methodology utilizes duplicity in views to capture all relations between the elements even the hidden ones. Thus, duplicity should be a desired attribute of the implementation.

These principles are the most important aspect of the developed Profile Implementation Methodology. The relation among individual principles is then shown in Table 1. The principles that influence each other are marked with a cross.

Profile Structure

The content structure in the EPF Composer consists of categories and pages. A category is a methodological content that contains other categories and/or pages. Categories are used to divide the elements into logical groups or cover the entire content. On the other hand, pages do not contain other child elements, thus carry only their own content and are not further divided.

Each implementation of VSE Profile has two sections. The first section corresponds with the printed version of the Profile regarding the entire text as well as its order. The aim is to maintain a linkage between the printed and electronic version according to the Correspondence principle. For example, the Entry Profile contains the following elements in the first section:

- Introduction
- Project Management Process
- Software Implementation Process
- Roles
- Work Products
- Software Tools

Table 1. Relation among principles, source: author based on (Simko, 2015)

Principle	Bottom-up	Process Assignment	Multiple Views	Reference	Predefined Category Priority	Completeness	Correspondence	Formatting Consistency	Order Consistency	Desired Duplicity
Bottom-up		x	x		x	x	x			x
Process Assignment	x		x		x	x	x			x
Multiple Views	x	x		x	x					x
Reference			x							x
Predefined Category Priority	x	x	x			x	x		x	x
Completeness	x	x			x		x		x	x
Correspondence	x	x			x	x		x	x	
Formatting Consistency							x			
Order Consistency					x	x	x			
Desired Duplicity	x	x	x	x	x	x				

The second section duplicates some parts of the Profile content in order to make all relations and connections between the content apparent and comprehensible. The second section adheres to the principles of Desired Duplicity, Completeness, and Multiple Views. For example, the Entry Profile contains the following elements in the second section:

- Processes
- Goals
- Activities
- Deployment Packages

Profile Element Mapping to EPF Composer Content Categories

Table 2 shows mapping of individual elements of the VSE Profile to recommended EPF Composer content categories on the concrete example of the elements of the Entry Profile.

According to the Predefined Category Priority principle, existing EPF Composer categories should be applied first. In the case of the Profile elements such as Activity, Product Kind, Process, Product, Role, Software Tool, Role Sets, Status, and Task, their predefined category was assigned. Looking at the Objective element, there is not any methodological element defined within the EPF Composer. That is why the Guidelines subcategory from the Guides category was chosen. Regarding the Deployment Package element, the Template subcategory from the Guides category was selected. In the case of the Views element, we strived to eliminate English terms in the titles of tree structure within the electronic version. The impossibility of changing particular titles from English within this tool is pointed out in (Champagne, 2016). Thus, the Custom Category was chosen for the Views element to address this issue.

Implementation Conventions

The methodology also defines specific conventions that are to be followed while working with the EPF Composer. These are the following: naming conventions, usage of predefined styles, hypertext

Table 2. Entry Profile Element Mapping to EPF Composer Content Categories, source: author based on (Simko, 2015)

Profile Element	Chosen EPF Composer Content Category	Entry Profile Example Element
Objective	Guides – Guidelines	PM.O1
Activity	Tasks	PM.1 Project Planning
Introduction	Guides – Supporting Material	Introduction
Product Kind	Work Product Kinds	PM Input Products
Deployment Package	Guides – Template	Project Management Deployment Package
View	Custom Category	All categories, e.g. Software Implementation (SI) Process
Process	Disciplines	Project Management (PM) Process
Product	Artifact	Project Plan
Role	Roles	Project Manager
Software Tool	Tools	Software Tools
Role Sets	Role Sets	Roles
Status	Available States	Accepted
Task	Steps	PM.1.1 Review the Statement of Work

content guidelines, and metadata management. Within metadata management, mandatory and optional metadata elements are defined. Their overview is presented in Table 3.

EPF Composer Usage Guidelines

The Profile Implementation Methodology contains a number of practical recommendations and guidelines for the implementation of individual Profiles. These are further shortly described.

Text Editing Guideline

When adding a text in the EPF Composer, it is recommended to insert it in smaller batches as problems with formatting sometimes occur in the text editor. After inserting a short text, using the "Preview" tab to check the formatting is suggested. To indent text, the bullet points and styles intended for this purpose should be applied. It is not advisable to use a tab key or spacebar since spaces and lines are spontaneously added between paragraphs.

Content Check Guideline

While drafting the content, it is necessary to check that each category has an assigned content as there must be at least a basic element, i.e. a page, defined within a category. Otherwise, the editor

Table 3. Mandatory and Optional Content Metadata, source: author based on (Simko, 2015)

Element	Applicable to EPF Composer Content	Mandatory	Optional
Alternatives	Task		X
Assignment Approaches	Role		X
Authors	All	X	
Base	All		X
Brief Description	All	X	
Brief Outline	Work Product		X
Copyright	All		X
Change Date	All	X	
Change Description	All		X
Impact of Not Having	Work Product		X
Key Considerations	All		X
Main Description	All	X	
Name	All	X	
Notation	Work Product		X
Presentation Name	All	X	
Purpose	Task and Work Product	X	
Reason for Not Needing	Work Product		X
Representation Options	Work Product		X
Selected Representation	Work Product		X
Skills	Role	X	
Slots	Work Product		X
Synonyms	Role	X	
Version	All	X	

skips an empty category when in the publishing process. Some categories even require a particular content that has to be specified for such a category, e.g. in the case of Tools.

Icon Usage Guideline

Regarding the usage of icons, the amount of icons in the EPF Composer is extensive. Nevertheless, it is suitable to create custom icons if they capture specific content in a better way. To create an icon for a category or page, it is required to reduce the size of a selected image to 32x32 pixels. To create an icon for a tree structure, the image size needs to be reduced to 16x16 pixels.

Configuration Guideline

While working with the EPF Composer, configuration settings should be conscientiously selected. Some operations are related to configuration and if not selected in the dropdown menu, these operations cannot be performed.

Saving and Backup Guideline

As the tool does not have an autosave function, the project should be saved after each step by pressing CTRL+S. Moreover, it is advisable to save the project under a different name in order to be able to go back to older versions.

Problem Solving Guideline

In case of problems with the EPF Composer, the following resources are recommended to be consulted. Basic information about the EPF Composer can be found at the "Getting Started" page (Getting Started, 2016). There are available courses, expert articles, manuals, tutorials, work examples, presentations, and publications. All these information serve as a necessary basis for understanding the tool itself. The "Practice: Method Development" page then contains concepts, recommendations and conventions for developing a methodological content (EPFC4, 2013). An extensive knowledge base and discussion is to be found at the "EPF Wiki" page (Eclipse Process Framework Project, 2016). It is also desirable to subscribe to RSS feeds that are available at the same page. Practical examples of published websites can be then found at the "EPF Published Websites Downloads" page (Eclipse Process Framework Downloads, 2016) illustrating what can be achieved with this tool. To supplement the sources, video based information is available on the social network YouTube after entering an EPF Composer keyword into the search.

Error Message Guideline

If an error message occurs while working with the editor, consulting the application Help or visiting discussion forums (Eclipse Community Forums, 2016) is recommended. In case the concrete issue was not addressed before, a new discussion thread could be created. The community is quite active and willing to help.

GUI Behaviour Guideline

In case the EPF Composer user interface does not behave in a standard manner, saving the project first is recommended followed by closing the active window and reopening it. If this option does not solve the issue, a restart of the EPF Composer is suggested.

HTML Knowledge Guideline

To efficiently work with the EPF Composer tool, a basic knowledge of HTML is recommended. Although the tool includes an editor for text editing, some results cannot be achieved within the editor alone. Similarly, when detecting errors, it is often impossible to reveal an error in the environment of the editor as it is necessary to analyse the HTML code. Thus, the option to switch between the modes of "Rich Text" and "HTML" should be used.

Profile Implementation Process

The profile implementation process in the EPF Composer consists of seven steps. The first step is to create a library that contains the source files and is used to store a methodological content. After creating the library, it is necessary to set a configuration that allows publishing the content. Further, a plug-in has to be developed enabling to reuse and create the content. The final stage of the first step consists in a package development. Basic elements (roles, activities, work products and manuals) are created within the package. In the second step, basic elements, categories and pages are created. The third step then builds a relation between all elements created in the previous step. In the fourth step, a reference to the content of individual categories and pages is supplemented. In the fifth step, a view (tree structure) is built to arrange the entire content clearly. A thorough content check is performed in the sixth step before publishing. In the seventh step, the prepared content is published in the form of a website. The last step represents an official publication, i.e. loading the website to the internet/intranet. During the implementation process, the principles of the methodology are followed as well as the proposed structure, element mapping, established conventions and guidelines for the tool usage are respected. The implementation process and its relation to defined rules are depicted in Figure 3.

PROFILE IMPLEMENTATION METHODOLOGY EVALUATION

The ISO/IEC 29110 Profile Implementation Methodology was evaluated during the implementation of the Entry Profile. The entire process composed of the steps outlined in the previous section is described in detail in (Simko, 2015) including screenshots and detailed guides. Thus, it represents a manual for publication of the VSE Profiles in the EPF Composer. The result of this whole process is an electronic Entry Profile as depicted in Figure 4, available at the website spicenter.vse.cz.

The electronic Profile also comprises localized Deployment Packages. In addition to the Entry Profile, the Basic Profile was implemented in the EPF Composer as well. Both Profiles are used in university courses at the Prague University of Economics on the undergraduate as well as graduate level. Students are able to easily access the content of the ISO/IEC 29110 standard as well as all Deployment Packages and use them within their projects. The VSE Profiles implemented in the EPF Composer are used also in a public course focused on the introduction of the ISO/IEC 29110 standard to small businesses. The main benefit of the ISO/IEC 29110 Profile Implementation Methodology is the consistent, usable and effective implementation of all VSE Profiles including those under

Figure 3. Profile Implementation Process, source: author

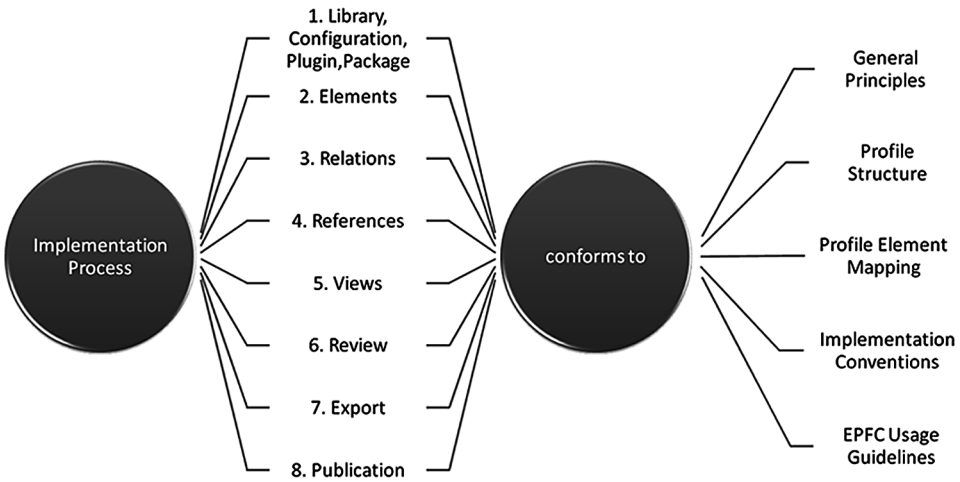
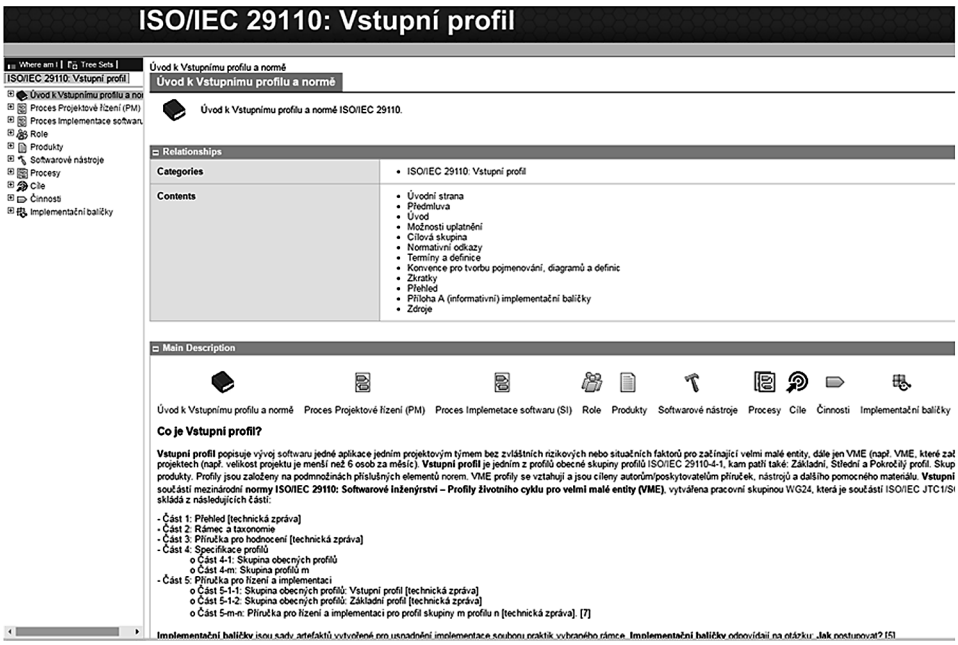


Figure 4. Published Entry Profile, source: author



development. Implemented VSE Profiles are then available to businesses for their pilot projects and ISO/IEC 29110 implementation. As of now, we are starting a pilot project of the ISO/IEC 29110 Basic Profile implementation that will follow the Profile implemented in the EPF Composer.

CONCLUSION

To help small companies improve their software development processes and to be recognized as entities that produce quality software, the ISO/IEC 29110 Systems and Software Engineering - Lifecycle Profiles for Very Small Entities (VSEs) standard is being developed. The standard itself was shortly presented together with initiatives towards its diffusion in the Czech Republic. To support standard diffusion, such issues like ease of use, and implementation support are of high importance. Therefore, the ISO/IEC 29110 standard was implemented in the Eclipse Process Framework (EPF) Composer, the open-source content management tool. Such an implementation enables effective managing of the standard and its publishing in the form of a web application that can be easily and efficiently used. To manage consistent implementation of individual Profiles and its quality, the ISO/IEC 29110 Profile Implementation Methodology was developed and presented in this article. This methodology represents an example of the usable outputs of the ISO/IEC 29110 standard being utilized in education and research in the Czech Republic. First, the methodology structure was outlined, followed by its individual elements, i.e. General Principles, Profile Structure, Profile Element Mapping, Implementation Conventions, EPF Composer Usage Guidelines, and Implementation Process. The evaluation of the ISO/IEC 29110 Profile Implementation Methodology was conducted during the implementation of the Entry Profile and presented in the following section alongside with the results of this approach and its impacts.

ACKNOWLEDGMENT

The work reported in this article was processed with contribution of long term institutional support of research activities by Faculty of Informatics and Statistics, University of Economics, Prague.

REFERENCES

Ambler, S. W. (2013). IT Project Success Rates Survey Results. Retrieved from http://www.ambysoft.com/surveys/success2013.html

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. *Proceedings of the4th International SPICE Conference on Process Assessment and Improvement*, Lisbon, Portugal.

Champagne, R. (2016). Developing a basic EPF plugin from scratch. École de technologie supérieure. Retrieved from http://etsmtl.ca/Professeurs/rchampagne/documents/epftutorial/index.html

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), 433–447. doi:10.1016/j.infsof.2011.12.003

Coleman, G., & O'Connor, R. (2008). Investigating Software Process in Practice: A Grounded Theory Perspective. *Journal of Systems and Software*, *81*(5), 772–784. doi:10.1016/j.jss.2007.07.027

Deployment Packages repository (2015). Retrieved from http://profs.etsmtl.ca/claporte/English/VSE/index.html

Dlugosova, S. (2014). *Publication of the ISO/IEC 29110 Basic profile in Eclipse Process Framework Composer* (in Czech) [Diploma thesis]. Prague University of Economics. Supervisor Alena Buchalcevova.

Eclipse Community Forums. (2016). Retrieved from http://www.eclipse.org/forums/index.php/f/49/

Eclipse Process Framework Downloads. (2016). Retrieved from http://www.eclipse.org/epf/downloads/downloads.php

Eclipse Process Framework Project. (2016). Retrieved from http://www.eclipse.org/epf/

Garcia, L., Laporte, C. Y., Arteaga, J., & Bruggmann, M. (2015). Implementation and Certification of ISO/IEC 29110 in an IT Startup in Peru, *Software Quality Professional Journal ASQ*, *17*(2), 2015.

Getting Started. (2016) Eclipse Process Framework. Retrieved from http://www.eclipse.org/epf/general/getting_started.php

ISO. IEC 15504 (2004). Information technology – Process Assessment. Geneva: International Organization for Standardization (ISO).

ISO. IEC 15289 (2006). Systems and Software Engineering — Content of systems and software life cycle process information products (Documentation). Geneva: International Organization for Standardization (ISO).

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

ISO. IEC 29110-1 (2010). Software Engineering – Lifecycle Profiles for Very Small Entities (VSE) -- Part 1: VSE profiles Overview. Geneva: International Organization for Standardization (ISO).

ISO. IEC 29110-2 (2010). Software Engineering – Lifecycle Profiles for Very Small Entities (VSE) - Part 2: Framework and Taxonomy. Geneva: International Organization for Standardization (ISO).

ISO. IEC 29110-3 (2010). Software Engineering – Lifecycle Profiles for Very Small Entities (VSE) - Part 3: Assessment Guide. Geneva: International Organization for Standardization (ISO).

ISO. IEC 29110-4-1 (2010). Software Engineering – Lifecycle Profiles for Very Small Entities (VSE) - Part 4-1: Profile specifications: Generic profile group. Geneva: International Organization for Standardization (ISO).

ISO. IEC 29110-5-1-2 (2011). Software Engineering – Lifecycle Profiles for Very Small Entities (VSE) - Management and engineering guide: Generic profile group: Basic profile. Geneva: International Organization for Standardization (ISO).

ISO. IEC 29110-5-1-1 (2012). Software Engineering – Lifecycle Profiles for Very Small Entities (VSE) - Management and engineering guide: Generic profile group: Entry profile. Geneva: International Organization for Standardization (ISO).

ISO 9001 (2008). Quality management – Systems Requirements. Geneva: International Organization for Standardization (ISO).

Jeners, S., O'Connor, R. V., Clarke, P., Lichter, H., Lepmets, M., & Buglione, L. (2013). Harnessing Software Development Contexts to inform Software Process Selection Decisions. Software Quality Professional, 16(1).

Johnson, J. (2006). *My Life is Failure*. The Standish Group International, Inc.

Laporte, C. Y. (2015). International Software Engineering Standards Applied in Undergraduate and Graduate Software Quality Assurance Courses. IEEE Standards University. *IEEE Standards Education E-Magazine.* Retrieved from http://www.standardsuniversity.org/issue/november-2015/

Laporte, C. Y., & O'Connor, Rory V. (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., Alexandre, S., & O'Connor, R. (2008). A Software Engineering Lifecycle Standard for Very Small Enterprises, R.V. O'Connor et al. (Eds.), Proceedings of EuroSPI (pp. 129-141). Springer-Verlag.

Laporte, C. Y., & O'Connor, R. (2014, June 25-27). A Systems Process Lifecycle Standard for Very Small Entities: Development and Pilot Trials. *Proceedings of the21th European Software Process Improvement Conference(Euro SPI 2014), CCIS* (Vol. 425, pp. 13-24). Heidelberg, Luxembourg: Springer-Verlag. doi:10.1007/978-3-662-43896-1_2

Laporte, C. Y., O'Connor, R. V., García, L., & Gerançon, B. (2015). An Innovative Approach in Developing Standard Professionals by Involving Software Engineering Students in Implementing and Improving International Standards. *Journal of the Society for Standards Professionals*, *67*(2), 1–9.

Moreno-Campos, E., Sanchez-Gordón, M. L., Colomo-Palacios, R., & de Amescua Seco, A. (2014, June). Towards measuring the impact of the ISO/IEC 29110 standard: a systematic review. *Proceedings of theEuropean Conference on Software Process Improvement* (pp. 1-12). Springer Berlin Heidelberg. doi:10.1007/978-3-662-43896-1_1

O'Connor, R., & Coleman, G. (2009). Ignoring "Best Practice": Why Irish software SMEs are rejecting CMMI and ISO 9000. *Journal of Information Systems and Small Business*, *16*(1), 780–789.

O'Connor, R. V. (2015). *Deploying a Software Process Lifecycle Standard in Very Small Companies." Encyclopedia of Information Science and Technology* (3rd ed., pp. 762–772). IGI Global; doi:10.4018/978-1-4666-5888-2.ch073

O'Connor, R. V., & Laporte, C. Y. (2010). Towards the provision of assistance for very small entities in deploying software lifecycle standards.*Proceedings of the 11th International Conference on Product Focused Software* (pp. 4-7). ACM. doi:10.1145/1961258.1961259

O'Connor, R. V., & Laporte, C. Y. (2014, January-June). An innovative approach to the development of an international software process lifecycle standard for very small entities. *International Journal of Information Technologies and Systems Approach*, *7*(1), 1–22. doi:10.4018/ijitsa.2014010101

Sánchez-Gordón, M. L., & O'Connor, R. V. (2015). Understanding the gap between software process practices and actual practice in very small companies. *Software Quality Journal*, 2015, 1–22.

Simko, D. (2015). *ISO/IEC 29110 Entry Profile Localization and its Publication in Eclipse Process Framework Composer* (in Czech) [Diploma thesis]. Prague University of Economics.

Tuft, B. (2010). Eclipse Process Framework (EPF) Composer Installation, Introduction, Tutorial and Manual. Retrieved from http://www.eclipse.org/epf/general/EPF_Installation_Tutorial_User_Manual.pdf

Alena Buchalcevova graduated in 1981 at the Prague University of Economics, Faculty of Management, major Information Systems Management. Since then she has been working at the faculty and at present she is an Associate Professor at the Department of Information Technologies. Her research field is mainly object oriented analysis, design and programming, Software Process Improvement, Software Quality Assurance, management of business informatics, and Enterprise Architecture. She is an author or co-author of 8 books, a number of journal articles and conference papers. She is a programme committee member of several international information systems and software engineering conferences and reviewer of several journals. Since 2008 she has been a member of the JTC1/SC7 Working Group 24 and has participated in the ISO/IEC 29110 standard development.