

How applicable is ISO/IEC 29110 in Game Software Development?

Jussi Kasurinen, Risto Laine and Kari Smolander

Department of Software Engineering and Information Management, Lappeenranta University of Technology, P.O. Box 20, FI-53851 Lappeenranta, Finland

{jussi.kasurinen, risto.laine, kari.smolander}@lut.fi

Abstract. Software development in a small development team is a challenge, as people have to fulfill several roles, which in larger groups would have separate, dedicated people for the work. To help small development teams to organize their activities, ISO/IEC standardization body has developed the standard ISO/IEC 29110, Lifecycle profiles for Very Small Entities. Our study focuses on the application of this model in the game industry, an industry that develops software. However, the game industry has its own set of unusual features in software development, such as the demand for artistic vision, need for novelty and demand for creative designs. In this study we analyze how the standard would work in seven different game industry companies and identify a set of additions that would help these organizations to apply the standard in practice. Based on our results, the ISO/IEC 29110 should incorporate more support for iterative development to allow easier adaptation to real-life organizations.

Keywords: game development, game industry, ISO/IEC 29110, software process, empirical study.

1 Introduction

The standardization work and standardized systems are important for many business areas, and the video game industry is not an exception [1]. The video game industry has a number of large international operators such as Sony and Nintendo, which form strong de facto standards for the entire industry [2]. The game industry has also been a constantly growing business for the last decade, having grown in the United States from the annual sales of 6.9 billion in 1999 [1] to 25.1 billion USD in 2010 [3]. Based on this background, it is hardly surprising that the video game industry has more and more new startup companies entering the market.

Fundamentally, video game development is a form of software development, although with several added requirements like visual presentation, artistic aspects and creative design [4, 5]. Game developers may have problems with their technical frameworks, but more often their project management practices need improvement [4]. It seems that game developers should and could learn more from the software engineering discipline in general [5]. In this sense the new ISO/IEC 29110 standard,

Lifecycle profiles for Very Small Entities [6], could benefit the game industry. The standard is aimed towards small and very small software-developing organizations, which, according to the standard, are enterprises, organizations, departments or projects having up to 25 people. The standard introduces a general development process model for development activities, which could be utilized also to organize team activities towards a more systematic and efficient approach to game development.

In this paper we study game industry organizations, compare them against the ISO/IEC 29110 standard, and evaluate how the standard model could be applied in the case organizations. Our research question can be presented as in the title “How applicable is ISO/IEC 29110 in Game Software Development?” Although game development is also software development, it has several unusual features such as the drive for novelty factors, creativity and artistic impression [2], which complicates the software design and development work. Taking this into an account, we wish to understand if the ISO/IEC 29110 standard can cater to the needs of game developing organizations. Based on our results we also develop a revised lifecycle model which should further assist organizations to utilize the current model. This study is also a continuation of our earlier empirical studies in the software industry, in which we, for example, have assessed the problems of software testing [7] and assessed the technical infrastructures of software development companies [8].

The rest of the paper is constructed as follows; Section 2 provides an overview on the related research in game development from the viewpoint of software engineering and introduces the ISO/IEC 29110 model, Section 3 presents the results from the data analysis and observations, while Section 4 discusses these results and their implications. Finally, Section 5 closes the paper with conclusions.

2 Related research

The research of game development from the viewpoint of software engineering is currently quite active. There are several studies discussing software development processes of game developing organizations. For example, game development lifecycles has been studied by Peltoniemi [2] from the viewpoint of creativity and other intrinsic features. This analysis compared game software development to other domains and concluded that in general game software development is not very mature. Peltoniemi also argues that game development differs from the traditional software industry because of the relatively stable hardware platform and the extensive demand for creativity. Because of these differences, the established life-cycle models are not directly applicable to game development. However, Kultima & Alha [9] observed that the game industry has been moving from the “creative chaos” towards process thinking, and that the larger industry operators have reduced innovation and creativity to a publicity act. Successful game products generate game brands, where innovation and creativity are limited, although innovation in game industry is still necessary in the development of new products.

The challenges of game development have been studied by Kanode and Haddad [10]. According to them, the game industry needs to adopt more advanced software

engineering strategies in areas such as management of assets, definition of project requirements and team management. The game industry could benefit from adopting process improvement, project management and development process designs from software engineering. Blow [4] also reports similar findings, emphasizing the need for the game industry to follow software engineering specific project management.

Outside the scope of the game industry, there is a study by Kautz, Hansen and Thaysen [11], which discusses the application of process models in software organizations. Their main finding was that organizations can adjust to given models, provided that the model itself is sound and allows adjustment to fit for the organization. Similarly, Dybå [12] conducted a study on software process improvement in different types of organizations. He found that the company size does not hinder or restrict the process improvement activities. Small organizations are at least as effective as large ones in applying process models and doing process improvement. The rationale for this observation was that small organizations tend to be less formal and loosely defined and can adapt to change needs more quickly in turbulent business environments.

This study discusses the ISO/IEC 29110 lifecycle profiles and the general process model defined for very small entities (VSE) [6]. The standard is created to cater to the needs of the VSE software developers, which, according to the OECD study [13], accounts for over 95% of the worldwide business population. The standard defines life cycle processes, a framework, taxonomy and an assessment guide for small organizations to understand and develop their practices. The standard is meant to allow organizations to develop towards more mature software process models such as ISO/IEC 12207 [14] regardless of the current approach in the organization. The activities defined by the standard are divided to the management and software implementation processes. These processes are interconnected, the management steering the implementation and the implementation providing feedback to the management, both with their own separate phases. The model is illustrated in Figure 1.

The management process has four main activities, project planning, project plan execution, project assessment and control, and project closure. These activities run roughly in this order as the project proceeds, with execution and assessment and control as parallel activities. These management activities steer the implementation process, which has six activities. These activities are software implementation initiation, where the project plan is reviewed, necessary creative tasks for designs are done and the implementation is started, software requirements analysis, where the work

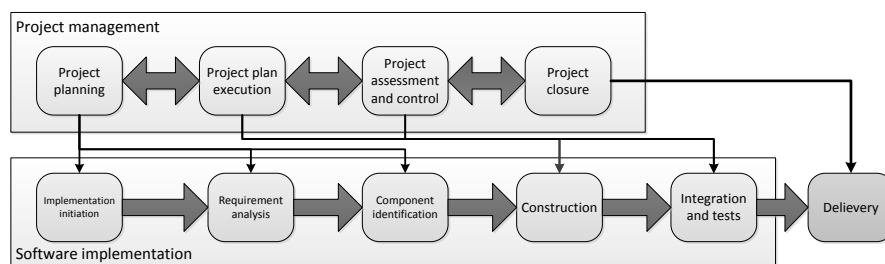


Fig. 1. ISO/IEC 29110 processes. The most important inter-process influences are marked with arrows.

team determines the task assignments and analyses the requirements, software component identification, where architecture is designed, software construction, where the software is developed, software integration and tests, where the construct is tested against the requirements, and product delivery, where the product is finalized and delivered to the customer. Unlike the management process, these activities are not meant to be parallel; the phases advance as product matures.

It is apparent that the game industry could benefit from the project management practices developed for the general software industry, as identified by Petrillo et al. [5], Peltoniemi [2] and Blow [4]. The ISO/IEC 29110 standard offers a general model for software development, which promotes the best practices of the other existing process standards, and is lightweight enough for application use in very small development organizations. In this sense, ISO/IEC 29110 could be an ideal model for the game industry to allow more mature approaches to software development.

3 Research process

Software development, including the design, development and testing of a commercial product is a complex phenomenon with varying solutions even in similar organizations [8]. Acknowledging this we decided to apply empirical qualitative analysis and apply the grounded theory method to understand the practices of game software development and make conclusions on the applicability of the standard in real-life game development [15-17]. We applied the grounded theory research method as introduced by Glaser and Strauss [15] and later by Strauss and Corbin [17] to systematically codify and analyze the interview data regarding the ISO/IEC 29110 model and related concepts such as game development or management activities in practice.

3.1 Data collection

The initial population and population criteria were decided based on the prior research made by our research group [for example 7, 8]. In this study, the population ranges from very small (less than 10 people [18]), startup companies making their first products to large and medium-sized companies with extensive product libraries.

Table 1. Interview rounds and themes

Interviews	Interviewees	Description	Main themes
Qualitative interview with 7 organizations	Team leader or project manager	The interviewee manages development of one product, or one phase for all products.	Development process, test process, quality, outsourcing, tools, organizational aspects.
Qualitative interview with 7 organizations	Developer or tester	The interviewee was responsible for development tasks, preferably also with responsibilities in testing.	Development process, test process, development tools, development methods, quality.
Qualitative interview with 7 organizations	Upper management or owner	The interviewee is upper management, or owner with an active role in the organization.	Organization, quality, marketing, innovation and design process, development process.
Qualitative interview with 7 organizations	Lead designer or Art designer	The interviewee is a game designer, or manager with the ability to affect the product design.	Development process, design and innovation, testing, quality

We carried out four interview rounds in our study (Table 1). The sample of the interview rounds consisted of seven organizations collected from our research partners and supplemented with volunteering organizations to achieve a heterogeneous group of different target audiences, development platforms and organizational backgrounds. The interviews were collected from spring to fall of 2012 by seven different researchers from two different research laboratories.

The seven organizations in the study group are all full-time professional software producers, with game development as their main activity and source of income. The smallest organization in the study was a startup with three persons; in the largest organization several hundred people contributed to the product development by directly working in the organization or by producing bought content. The organization sizes in Table 2 vary from small to large according to the EU SME definition [18], where a company is small when it has less than 50 employees, but in fact all small organizations in the study had less than 25 employees and so they fit to the ISO/IEC 29110 VSE definition. The group of organizations included different target platforms and different sizes of development teams. The organizations varied (Table 2) from mobile game developers to developers of browser-based, PC and console games. Their distribution channels included internet stores, general digital distribution, and boxed copies sold at game stores. The objective of this diverse approach was to gain a broader understanding of the practices in the organization to identify the most important factors affecting how the organization felt about the ISO/IEC 29110 standard [6]. To succeed in this objective, the research questionnaires included themes such as development methods, quality requirements, test phases, applied tools and design principles, which helped us to understand the case organizations. The questionnaires were also refined between the interview rounds to gain more role-oriented information and test our observations from the organizations made on the earlier rounds. Before interviews, the questionnaire questions were reviewed by a number of peers from the research group to assess the first round interview questions. The complete questionnaires are available at <http://www2.it.lut.fi/project/SOCES/material>.

The interviews contained semi-structured questions, and the whole sessions were tape-recorded for qualitative analysis. We also allowed the interviewees to get access to the data collection instruments before the interview, ask questions on the topics and gave a brief introduction to the interview themes prior to the actual interview. All

Table 2. - Description of the interviewed organizations

Organization	Release platforms	Team size ¹	Intended distribution channels for products	Number of released games in history
Case A	PC, game consoles	Large	Digital distribution, boxed copies	More than 10 released products
Case B	Mobile platforms	Small	Platform store	Less than 5 released products
Case C	Game consoles, PC	Large	Boxed copies, digital distribution	More than 10 released products
Case D	Mobile platforms, PC	Medium	Platform store, digital distribution	Less than 5 released products
Case E	Mobile platforms	Small	Platform store	Less than 5 released products
Case F	PC	Medium	Digital distribution	Developing first product
Case G	Browser games	Small	Separate service site	Developing first product

¹Number of people contributing to the released product, size by SME definitions [18]

interviews lasted in average about one hour. The interviews were arranged face-to-face with one or two organization participants and one or two researchers present.

The decision to interview project managers during the first round was based on the objective to understand the organizational processes and gain understanding of the operational level of software development in the case organizations. We also wanted to test whether our earlier observations from general software industry [7,8] would be applicable in the game industry context, and gain some insight into the organization.

The interviewees in the second round were selected from a group of developers, who directly contributed to the game product and had experience with the technical details of the developed product. The second round interview topics were heavily focused towards programming techniques, process activities and applied tools.

In the third round, the focus of the interviews was to collect general data of the company and focus on the managerial and business aspects. During this round additional themes such as marketing, innovation and financing were collected to better understand the context in which the game industry operates.

In the fourth round, the focus was on the creative aspects of development, in the game design work. During this round the interviewed employees were game designers or management-level personnel with the ability to affect the designs. Additional topics were change management, testing and external influences on the products.

3.2 Data Analysis

The grounded theory method contains three data analysis steps: open coding, where categories and their related codes are extracted from the data; axial coding, where connections between the categories and codes are identified; and selective coding, where the core category is identified and described [17].

The objective of the open coding is to identify leads in the data and classify the observations. This process was started with “seed categories” [19], which in our case, were derived from our prior studies on general software industry [7,8]. These seed categories were also used to define the main themes for the questionnaires. We used Atlas TI as the tool to manage codings in the interview transcripts. In open coding, the classified observations are organized into larger categories. New categories appear and are merged because of new information that surfaces during the coding and analysis work. For example, we revised our initial choice of collecting data on applied standards as a separate category as the interviews proved that the organizations were not as process-oriented or highly organized as originally thought, not even the larger or older ones. At the end of the open coding, the number of codes was 172. They came from 1572 observations from over 1400 minutes of recorded interview data.

The objective of the axial coding, which starts when the categories start to emerge and runs somewhat parallel with the open coding [17], is to further develop the categories by looking for causal conditions or any kind of connections between the categories. In this phase, the analysis focuses on identifying the relationships between larger concepts. For example, codes such as “Design process: refining designs”, “Development process: knowledge transfer” and “Problem: Documentation/knowledge transfer related to design” form a chain of evidence for observing organizational doc-

umentation practices and the problems associated to the communication between stakeholders. By following these leads in the data, the relationships were identified.

The third phase of grounded analysis, selective coding, is used to identify the core category [17] and relate it to the surrounding categories. As defined in [17], the core category is sometimes one of the existing categories, and at other times no single category is broad or influential enough to cover the central phenomenon. In this study, the core category “applicability of the ISO/IEC 29110 standard” resulted to the identification of the four different scenarios which could enable better applicability of the standard in game development organizations. This core category was formed by collecting the case organization views on the standard, collecting the existing development process issues and comparing them against the organizational backgrounds. For example, we observed that the most common consideration with the standard was that it was not considered as important in the very small organizations, but more important in the “small-medium” organizations with ten to twenty members. Similarly, the organizations with the background in software industry were the most critical towards the standard itself. Overall, we adjusted the core category to include all issues that explained the opinions on the ISO/IEC 29110 standard and based on this analysis, created an initial model that could combine the simplicity of the ISO/IEC 29110 standard with the concerns the case organizations presented.

4 Results

In this section we first introduce the categories created in the analysis, present the main observations we made on the applicability and usability of ISO/IEC 29110 [6], discuss the implications of the observations and finally present a revised draft for a model that combines the observations with the standard.

4.1 Categories

The core category, *Applicability of ISO/IEC29110* (Table 3), describes our interpretation of how the organization saw that ISO/IEC 29110 could be applied in their organization. The core category was identified from the data because it explained well the differences between the organizations and enabled us to create four scenarios of using the ISO/IEC 29110 process model.

Our analysis identified four scenarios. In scenario 1, named “*As is*”, the organization considers the model to be applicable “as is”, with little to no modifications. In scenario 2, “*development iteration*”, the organization considers that the model is applicable, but the development phase activities from component identification to testing should be modeled iteratively or at least allow iterations over the existing plan-driven definition. In scenario 3, “*full iteration*”, the model is seen applicable, but only if it is used only as a definition of one iteration round. In scenario 3, the entire process model is executed several times, for example once for product design to deliver a design prototype, in development to deliver an alpha release candidate and so on. In each iteration there would all activities from design to delivery. Finally the scenario 4,

“reference tool”, basically considers the model to be unsuitable for the organization, but applicable as a glossary, reference or teaching tool.

The category *Opinion on ISO/IEC 29110 applicability in own organization* describes the overall attitude of the organization towards the ISO/IEC 29110 software process model, should it be applied in the current organization. The category *Most important development need* for the standard describes the most important area or shortcoming, which should be revised to enable better applicability.

The category *Applied development method* defines what kind of development process the organization applies. The applied scale is undefined agile, where the organization uses basically an open-form approach, with little documentation and planning, defined agile, where the organization follows the principles of an established agile practice – also named if followed officially – and plan-driven, if the organization considers itself to follow a more traditional design-first approach.

The category *Typical design deliverables* summarizes the deliverables the organization usually produces before it commits to the construction of a new game product. The term functional prototype denotes to a proof-of-concept prototype that features the main gameplay elements, core features and concept art means that the organization has decided on the main features and the artistic direction. Basic gameplay elements simply means that the organization has decided on the game type and some of the most important features for the product.

The category *External or publisher influence on the development* categorizes the extent and type of influence the external organizations or partners have on the developed product. Low influence means that an external partner, usually a publisher, recommends only few additions or changes, medium that some wanted features are de-

Table 3. Summary of the essential categories in organizations

Category	Case A	Case B	Case C	Case D	Case E	Case F	Case G
Applicability of ISO/IEC29110	Scenario 3, “full iteration”	scenario 1, “As is”	Scenario 4, “reference tool”	Scenario 2, “development iteration”	Scenario 1, “As is”	Scenario 2, “development iteration”	Scenario 4, “reference tool”
Opinion on ISO/IEC 29110 applicability in own organization	Game development has additional issues	Management-part is close to current state,	Basis for contracts, problematic to implement	More or less defines the basic activities done in each project.	Too close to waterfall model.	Would limit design.	Overtly rigid, too close to waterfall model.
Most important development need for the standard	Too generic model to be directly usable.	Too generic model to be directly usable.	Design cannot be finalized before development	Needs experienced crew to use.	Design cannot be finalized before development	Design cannot be finalized before development	Too close to waterfall model
Applied development method	Defined agile, SCRUM	Plan-driven	Defined agile, SCRUM	Undefined agile	Undefined agile	Defined agile, SCRUM	Defined agile
Typical design deliverables	Functional prototype	Basic gameplay elements	Functional prototype	Core features, concept art	Basic gameplay elements	Core features, concept art	Basic gameplay elements
External or publisher influence in development	Medium/Large: Some features dictated by the publisher	Low/Medium: Some requested features, negotiable	Large: rejected features, cuts project schedules	Medium: Own testing, requested features	Low: Rarely requested features	Medium: Some requested features and changes	-
Project size	Large	Small	Large	Medium	Small	Medium	Small
Organizational experience	More than 10 products	Less than 5 products	More than 10 products	Less than 5 products	Less than 5 products	Developing first products	Developing first products
Key personnel background	Media	Academic	Media	Media	Software Industry	Academic	Software Industry

¹See SME definitions [18]

manded and large that the external influence may reject features, enforce changes, or even dictate themes or genre of the designed product.

The categories *Project size* and *Organizational experience* express the size and the experience of the development team. The project size is small when there are less than ten people, medium when ten to fifty and large when more than fifty developers contribute to the game product. Third-party members such as insourced musicians, artists or asset developers are also included to the size estimate. Organizational experience expresses the experience of the organization in game development. The experience is evaluated with the number of commercially launched products, excluding expansion disks or other additional content developed for earlier product. The applied scale is “developing the first products”, “less than five”, “less than ten” and “ten or more” to give a rough estimate on how experienced the organization is.

The category *Key personnel background* lists the business domain from which the key personnel such as company owners or founding members have emigrated to the company. The backgrounds were classified to three domains: media which includes areas such as advertisement agencies or journalism, academic which includes university-level students or teachers and software industry which means software business domains other than entertainment or video games.

4.2 Observations

ISO/IEC 29110 should offer a more iterative and flexible approach to development.

The ISO/IEC 29110 standard was introduced and discussed with the representatives of the seven study organizations. In most organizations the standard was considered to be at least somewhat useful, although in need of some revisions. Cases A and B considered the model to be too generic to apply as such, while organizations E and G considered it to resemble too much the classical waterfall approach. However, several case organizations (A, B, D, E and F) considered the model to be adjustable with some revisions to cater to their organization.

“Yes, I do recognize stuff from our current processes, but I think that in game development design cannot be finalized beforehand.” ... “We should try to be more Agile than this.” – Case C, Project manager

“The first thing I see here is that requirement analysis is completely done before construction. So design is finished before anything is implemented... that’s just not the way it happens.” – Case G, Designer

Interestingly, the most critical outlook against the ISO/IEC 29110 standard in its current form came from the organizations with background in the software industry. Both this kind of organizations criticized the model for steering the process activities towards waterfall model, which in their opinion is very unsuitable for the game industry. In both organizations, cases E and G, the main concern was that the waterfall-like approach would lock the design too early in the development process.

“I think that the current tools have made that approach redundant”... “Why should we bother with strict decisions when we can make a prototype in two hours to test things out?” – Case E, Developer

“I think that the [waterfall] model is no longer feasible approach for any kind of development project.” – Case G, Designer

Game development has many stakeholders, but they are not interested in certification or process standards.

In all organizations with the sole exception of the Case G the interviewees reported that they get regularly requests from their external stakeholders. The requests varied between negotiable feature additions (for example cases B and E), to large interventions where the requests could determine the features and dictate project schedules. However, these external stakeholders did not require defined development processes. In fact the opposite was observed: the organizations (cases B and E) which had most limited external influence on their products were also the most positive towards the ISO/IEC 29110 model, while case organizations A and C with the most external interference were also very critical towards the standard.

“Well, practically this is how we work. All the stuff we do is in this model somewhere.” – Case E, Developer

“Considering our team I think this would be good to use as a backbone” ... “I think this should be usable in the game industry. At least I do not see why it would not be.” – Case B, Project manager

It is also worthwhile to notice that even if the external organizations would promote standardization for game development, the developers are rather critical towards using it in small organizations. All organizations expressed that their organization either was too small for such a rigid model (cases B, D, E and G) or that very small organizations in general do not require defined process models (cases A, C and F).

“In our team everyone does a bit of everything, there really is no separate project management organization.” – Case B, Project manager

“If our team was 10 people, our approach would be hard, with 20, probably impossible. But since we are still so small, we can go with ad hoc.” – Case D, Upper management

Game-developing organizations have problems in following the current ISO/IEC 29110 process model because of the nature of the industry.

In almost all organizations the design process was considered to be at least somewhat problematic area in the standard. Cases C, E and F expressed their concern over how the standard enables design activities. In many organizations product design is not very specific, and the decisions are made based on the usability, playability and “fun factor” tested with product prototypes.

“If a tester comes to say that this does not work, there is not fun in it, you really cannot leave that in the game, you have to fix it.” – Case E, Designer

“When the production started, the specifications went out of the window” ... “There simply is not enough knowledge to make a full design at the early stages.” – Case G, Designer

“We sometimes have to make drastic changes [to keep the game fun].” – Case F, Designer

Another common concern with the standard was the level of abstraction. Case organizations A and C considered the standard very abstract to the point of being hard to implement as a development guideline. Case D considered the standard to be generally close to what they are already doing, adding that the standard would require an experienced team to work properly. Case B mentioned that the management model is a rather accurate representation of the project management tasks in their organization.

“With these states and arrows... pretty much every step we take with development falls into some of the [ISO/IEC 29110] model categories.” – Case D, Project manager

Besides general comments, case organizations A, B and C considered the standard to be usable in software development, but lacking the features and parts that are specific to the game industry. Features like design with prototypes, demo versions to attract business partners and art design were mentioned to be such features missing.

“The management parts, that’s OK, but the implementation [is not].” – Case B, Developer

“I do identify parts of our current work, but in my opinion game programming with this is difficult.” – Case C, Project manager

4.3 Implications

Based on the observations it is apparent that the ISO/IEC 29110 standard is a reasonable start for creating a model for small and very small software organizations. However, for the game industry in particular there are some troublesome areas which require further attention. For example, the model needs to be able to support iterative development, like in the scenarios 2 and 3. In addition, the game industry has some unusual features such as the requirement for late changes to designs and focus on fine-tuning the user experience. It seems that the most useful standard for the game industry would be a lean iterative process standard for development tasks, targeted at small and small-medium sized organizations. Very small organizations operate with highly informal and flexible methods, and do not benefit from the standard as much as a small to small medium (roughly ten to twenty participants) sized organizations would. ISO/IEC 29110 can be developed to meet the needs of these organizations, but that requires revisions to allow more iterative approaches. The requirements for a VSE model suitable for application should have the following changes:

- Design should not be finalized before the implementation.
- Varying amounts of details should be allowed in the initial design.
- Changes in the product should be allowed even near the end of the project.
- The standard should support iterative approaches.
- The standard should define all of the main activities to allow its use as the basis for business contracts, but leave the definition of details to organizations.
- The standard should support delivery-phase activities such as acceptance testing and planning phase activities such as proof-of-concept prototypes.

A draft model based on these requirements is illustrated in Figure 2. The most important difference between the standard and the revised model is the implementation

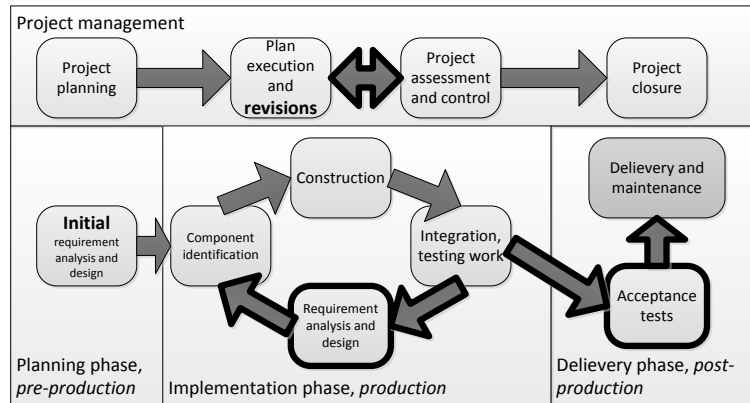


Fig. 2. The revised model based on the analysis, revised elements bolded.

of iterative development and separating the process to three phases. Initial requirements analysis and design, which includes experimenting new ideas with functional prototypes before the implementation is located in the planning phase (also called pre-production by the game industry), while the implementation phase (production in game industry) has its own requirement analysis and design phase to allow revisions to the plans during the development. Finally the finalization activities such as acceptance testing or identification of reusable assets are separated to a new phase called delivery phase (post-production).

The management process is kept relatively intact, as it was considered to be the most accurate part of the model. The project planning is done before the development starts, and in the plan execution process some minor revisions can be done to the project plans to cater to the change needs during development. Project closure happens in the delivery phase, when final adjustments and quality assurance activities to validate the requirements are done. With these additions the model is still conceptually similar to the standard. Iteration is not necessary but it is allowed. Varying degrees of design are allowed and if revisions are not needed, the process is similar to the original.

5 Discussion

A common opinion was that a life-cycle standard for very small organizations could be beneficial. Out of the seven organizations only two considered ISO/IEC 29110 as unsuitable for their use, but the others considered the standard at least as a useful basis for contracts. The most common problem with the standard was the apparent lack of support for iterative development especially in design. The ISO/IEC 29110 documentation mentions that it can be applied with any methodology, like agile, evolutionary, incremental, test driven or other [6], but this is not very evident from the viewpoint of the organizations. The model design steers the development towards plan-driven principles, or at least does not support agile development well.

The interviewed organizations also considered the model target audience to be problematic; very small organizations can operate with an ad-hoc structure. However,

most organizations agreed that a small-medium organization (roughly ten to twenty developers) needs a more or less defined process model. Overall, the conclusions for the applicability and the suitability of the ISO/IEC 29110 standard can be summarized as follows: *For very-small-sized game development companies the standard should promote more agility. For the current standard the best target audience seems to be small and small-medium-sized organizations.*

Many interviewees were happy with their informal and, to some degree, unstructured approach to software development. The small organization size and unstructured process also affected the assessment of the standard. The smallest organizations represented the key demographic of the standard, while the larger, more experienced organizations provided insight into the general applicability. Interestingly, the more the organization had external stakeholders like financiers and publishers, the more they considered the model to need changes. On the other hand, this could also indicate that the model is suitable for small, autonomous developer teams such as case organizations B and E, should there be any incentive like an external publisher or a financier requiring the implementation of some process model. However, since expertise with process models or process development was not required for participating in this study, these results should be considered more like indications or recommendations.

Based on all of the identified needs and observations, the ISO/IEC 29110 standard can be considered to be somewhat difficult to implement in a game organization as is. However, dividing the process model to three main phases, the design phase, the implementation phase and the delivery phase, with added support for iteration in implementation, the model caters to the most of the identified game industry problems. In addition, should the implementation phase be completed without iteration, the revised model should still be compatible with the original ISO/IEC 29110 standard.

There are threats that should be acknowledged when addressing the validity of this type of qualitative research [20,21]. Whittemore et al. [20] lists integrity, authenticity, credibility and criticality as the most important attributes that are universally required for a qualitative study. In their conclusion, the most important aspect of presenting validity is to present enough evidence for the reader to accept that the researchers truly understand the phenomena they are observing, “know what they claim to know”. The most common threat is personal biasing, disregarding the patterns and observations which do not fit or support their own theories [21]. In this study these risks have been taken into account when planning and implementing the study with several cautionary actions. The study questionnaires were designed by seven researchers to avoid personal bias, the data collection was conducted by six researchers and the analysis process for this paper was conducted by two and validated by three researchers working separately on the same data. The two analyzers did their work independently, and the different interpretations were discussed and resolved in meetings during the study.

Additionally, all interviewed organizations were small and medium size game organizations, located in Southeastern Finland. This can be considered a threat, as some phenomena can be caused by the geographical proximity, such as local policies or support from the local educational institutions. However, the game business is international and all our organizations aimed their products at international markets. The possibility for the bias caused by the cultural similarity was addressed by selecting the

organizations to represent different release platforms, company backgrounds, different maturity levels as businesses and different production sizes.

6 Conclusions

In this paper we have presented our study on seven game software development organizations with the purpose of evaluating the applicability of the ISO/IEC 29110 standard. The results indicate that very small game developing organizations do not tend to follow any systematic development method, working mostly ad hoc. The application of the ISO/IEC 29110 standard could be possible, but the model should support an iterations, as in the game business product design is mostly a guideline, which changes throughout the development process. The current standard version is reasonably accurate for the management tasks, but does not support the continuous design or the approach where late changes to the product features are expected. The applicability of the current version can be summarized as follows: Should the target audience of VSE-sized companies be kept, the model should promote more agile approaches, and should the current model be kept, the best target audience in the game industry would be not a VSE, but small and medium-sized organizations.

The results of this study can be used to help very small new game developing organizations to understand how they could apply the ISO/IEC 29110 standard, or at least the concepts presented in the standard to help in managing development. For the future work, the applicability of the revised model should be assessed with the game developing organizations. Another interesting direction could also be the inclusion of creative elements to the process thinking; how game organizations actually work, and how they could be supported with the existing software engineering knowledge.

7 Acknowledgements

This study was supported by the European Union Regional Development Fund projects number A31814, “Kaakon Peliklusteri” administered by the Council of Southern Karelia and number A32139 “Game Cluster” administered by the Council of Päijät-Häme. We would also like to thank all the interviewed organizations and the project partners, namely Cursor and Innovire research team at LUT Kouvola.

8 References

1. Gallagher, S., Park, S.H.: Innovation and competition in standard-based industries: a historical analysis of the US home video game market, *Engineering Management, IEEE Transactions on*, vol.49, no.1, pp.67-82, Feb 2002, doi: 10.1109/17.985749 (2002)
2. Peltoniemi, M.: Life-cycle of the Games Industry The Specificities of Creative Industries, *Proceedings of the Mindtrek'08*, 7.-9.10.2008, Tampere, Finland. (2008)
3. Entertainment Software Association: 2011 Sales, demographic and usage data: Essential facts about computer and video game industry, (2011)

4. Blow, J.: Game Development: Harder Than You Think, Queue, vol. 1, nro. 10, ss. 28–37, February (2004)
5. Petrillo, F., Pimenta, M., Trindade, F., Dietrich, C.: Houston, we have a problem...: A survey of Actual Problems in Computer Games Development, Proceedings of SAC'08, 16.-20.3.2008, Fortaleza, Brazil (2008)
6. ISO/IEC: ISO/IEC 29110 Software Engineering – Lifecycle Profiles for Very Small Entities (VSEs), Part 5-1-2: Management and engineering guide:Generic profile group: Basic profile ISO/IEC JTC1/SC7, Technical Report dated 2011-05-15. (2011)
7. Kasurinen, J., Taipale, O., Smolander, K.: Analysis of Problems in Testing Practices, Proceedings of the 16th Asia-Pacific Software Engineering Conference (APSEC), 1.12.-3.12.2009, Penang, Malaysia. doi: /10.1109/APSEC.2009.17 (2009)
8. Kasurinen, J., Taipale, O., Smolander, K.: Software Test Automation in Practice: Empirical Observations, Advances in Software Engineering, Special Issue on Software Test Automation, Hindawi Publishing Co. doi: 10.1155/2010/620836 (2010)
9. Kultima, A., Alha, K.: Hopefully Everything I'm Doing Has to Do with Innovation: Games industry professionals on innovation in 2009, Proc. 2nd International IEEE Consumer Electronics Society's Games Innovation Conference, Hong Kong, China, (2010)
10. Kanode, C. M., Haddad, H. M.: Software Engineering Challenges in Game Development, in Information Technology: New Generations, ITNG '09. Sixth International Conference on, 2009, ss. 260 –265. (2009)
11. Kautz, K., Hansen, H.W., Thaysen, K.: Applying and adjusting a software process improvement model in practice: the use of the IDEAL model in a small software enterprise, Proceedings of the 22nd international conference on Software engineering, Limerick, Ireland, pages 626-633. doi: 10.1145/337180.337492 (2000)
12. Dybå, T.: Factors of software process improvement success in small and large organizations: an empirical study in the scandinavian context, Proceedings of the 9th European software engineering conference held jointly with 11th ACM SIGSOFT international symposium on Foundations of software engineering, pages 148-157, Helsinki, Finland. doi: 10.1145/940071.940092 (2003)
13. OECD: OECD SME and Entrepreneurship Outlook, 2005 Edition, Organisation for Economic Co-operation and Development, Paris (2005)
14. ISO/IEC: ISO/IEC 12207:2008 Systems and Software engineering - Software life cycle processes, (2008)
15. Glaser, B.G.: Constructivist Grounded Theory?, Forum: Qualitative Social Research (FQS), Vol 3(3). (2002)
16. Glaser, B., Strauss, A.L.: The Discovery of Grounded Theory: Strategies for Qualitative Research. Chicago: Aldine. (1967)
17. Strauss, A., Corbin J.: Basics of Qualitative Research: Grounded Theory Procedures and Techniques. SAGE Publications, Newbury Park, CA, USA. (1990)
18. EU: SME Definition, European Commission. (2003).
19. Seaman, C.B.: Qualitative methods in empirical studies of software engineering, IEEE Transactions on Software Engineering, vol. 25, pp. 557-572. (2002)
20. Whittemore, R., Chase, S.K., Mandle, C.L.: Validity in Qualitative Research”, Qual Health Res, July 2001, 11: 522-537, doi:10.1177/104973201129119299 (2001)
21. Golafshani, N.: Understanding Reliability and Validity in Qualitative Research, The Qualitative Report, Vol 8(4), December 2003, pages 596-607. (2003)