SERIOUS GAME DEVELOPMENT BY DISTRIBUTED TEAMS
A Case Study Based on the EU Project PRIME

Manuel Oliveira
Alfamicro, Alameda da Guia, N.192A, 2750 Cascais, Portugal

Heiko Duin
BIBA-IKAP, Hochschulring 20, D-28359 Bremen, Germany

Keywords: Distributed brainstorming, conceptual design, serious games, distributed teams.

Abstract: This paper describes as a case study the first 12 months of a 2 year project developing a serious game targeted at a number of industrial sectors: aeronautical, automotive, civil construction, software and electronics. The paper presents the devised methodology to address the problems that emerged, mostly associated to the inherent barriers of managing a distributed team with equal participatory roles and responsibilities in the creative process of developing a serious game. Some of the lessons learnt are shared with the reader.

1 INTRODUCTION

The process of creating ideas within a group of people requires good communication to achieve any form of successful outcome. Such a process always benefits from a group setting (Gouran and Hirokawa, 1996).

The initial basic principles proposed by (Osborne, 1957) to structure the process of creativity within a group setting, consisted of deferring judgement whilst instigating quantity, thereby producing quality. However, research has demonstrated the fallacy of this traditional perspective and in fact nominal groups (pooling the ideas of individuals that brainstorm on their own) are more productive than face-to-face groups (Gallupe et al, 1992; Paulus et al, 1995). According to (Diehl and Stroebe, 1987), communication is the culprit behind the ineffectiveness of face-to-face meetings when compared to nominal groups. The communication problems are exacerbated when considering groups of participants that are geographically distributed and rely on technology to create pseudo face-to-face meetings within a virtual space. Independently of the technological advances that have transformed the globe into a digital village, the creation process within a team of geographically distributed individuals continues to present hard challenges to overcome.

This paper presents a case study of a distributed team involved in the highly creative process of developing a serious game targeted at organizations from differing industrial sectors. The focus is on the adopted process and the lessons learnt after 12 months of progress. A review of progress after six months is given in (Oliveira et al 2006b). The development process adopted in PRIME is based on a spiral approach using principles from the *Agile Programming* community, but focused on the end-user from conception to final deployment.

2 THE CASE STUDY

2.1 The Project

The case study presented in this paper is based in the European project PRIME (Providing Real Integration in Multi-disciplinary Environments). The main objective of the PRIME project is to give business professionals a learning environment where they can experiment with new ideas and learn how to handle the entire life cycle of products/processes. PRIME proposes to achieve this by enhancing current work environments with a new paradigm based on *serious gaming* (Annetta et al, 2006).

Oliveira M. and Duin H. (2007).
SERIOUS GAME DEVELOPMENT BY DISTRIBUTED TEAMS - A Case Study Based on the EU Project PRIME.
In *Proceedings of the Third International Conference on Web Information Systems and Technologies - Society, e-Business and e-Government / e-Learning*, pages 296-303
DOI: 10.5220/0001272802960303

Serious games are digital games used as a persuasion or educational technology. They can be similar to educational games, but are primarily intended for an audience outside of primary or secondary education. Serious games might be of any genre and many of them represent a kind of edutainment applications. A serious game could be a simulation which has the look and feel of a game, but corresponds to non-game events or processes, such as business or military operations (e.g. *America's Army* or *Tactical Iraqi* (Squire, 2005)).

In PRIME, the adoption of serious game approach allows the user to learn by experience within a Virtual Business Environment (VBE) that is safe and foments risk taking without detrimental impact on business. The context is based on strategic management, including multi-stakeholder negotiation and business connectivity, with a strong focus on manufacturing.

The VBE is composed of several systems defining its economic fabric: capital markets, population, governments, logistic support, labour markets, business-to-consumer, consumer associations, labour unions, etc. Within this rich environment, the business professional assumes the role of a Business Unit (BU), which may correspond to either a single site or multiple sites. The management breakdown of a BU is based on eight functional units: Production, Product Development, Sales, Human Resources, Strategic Marketing, Distribution, Finance and Information Systems. Associated to each functional unit is a set of operation processes that are based on an input-transformation-output model.

The business professional, as a human player, makes strategic decisions based on information available through the BU and the VBE. However, the decisions themselves are not implemented by the PRIME Serious Game, although the interface and game mechanisms may facilitate the decision processes (Kracke et al, 2006). The same is not true of the artificial stakeholders, who share similar decision processes, but are required to have them implemented as a decision engine supported by a knowledge rule based database. A more detailed description of the PRIME project is given in (Oliveira et al 2006a).

2.2 The Stakeholders

The project has a developer team of seven partners (Alfamicro, BIBA, CIST-Sofia, EPFL, Intrapoint, MIP and Sintef), who are geographically distributed across Europe accompanied by six end-user partners from different business sectors (CRF, IAI, Intracom, KESZ, LEGO and Siemens). The diversity of the end-user organizations from different industrial sectors raises additional challenges to the development process.

The need for a geographically distributed team exists from the onset of the project, as evidenced by map illustrated in Table 1. Achieving a cohesive multidisciplinary team with the ability of working together irrespective of the cultural differences, work practices, and geographic distribution across 3 different time zones, proved to be a challenge in process engineering and not coupled to technology.

Table 1: Stakeholders in PRIME.

Short-name	**Longname**	**Country**	**Role**
Alfamicro	Alfamicro	Portugal	Developer
BIBA	Bremen Institute of Industrial Engineering and Applied Work Science	Germany	Developer
CIST	Center of Information Society Technologies, University of Sofia	Bulgaria	Developer
CRF	Fiat Research Centre	Italy	End-User
EPFL	Swiss Federal Institute of Technology	Switzerland	Developer
IAI	Isreal Aircraft Industries	Israel	End-User
Intracom	Intracom	Greece	End-User
Intrapoint	Intrapoint	Norway	Developer
KESZ	Central European Building and Construction ltd.	Hungary	End-User
LEGO	Lego Company	Denmark	End-User
MIP	Business School of Politecnico di Milano	Italy	Developer
Siemens	Siemens Austria	Austria	End-User
Sintef	Sintef Research Institute	Norway	Developer

3 GAME DEVELOPMENT

The process of software development is permeated with creativity (Glass, 1995), and in the case of game development a clear example is embodied in the dynamic evolution of a game design document, which final version corresponds to the synergy of all

the different stakeholders involved in the development process.

The production of games has evolved significantly since the "garage" period when two or three friends would develop a game by doing all the implementation, game design and content with no schedule and minimal budget. Nowadays, a game is usually produced by a multi-disciplinary team of about twenty people or more, with the technical developers being a small core of a few people and the remainder of the team being responsible for content creation and game design. The production cycle usually involves 2 to 2.5 years of desperate development effort chasing ever-shifting deadlines and overcoming problems whilst simultaneously trying to keep to the target release date. The cost involved in maintaining such a large team for an extended amount of time requires an investment budget of a few million Euro. As a result, one verifies in the evidence in many postmortems (Gamasutra, 2006) that it is quite common for projects to exceed the initial budget and overextend the initial time allotted for completion. In many cases, critical problems may be traced to the challenges of managing the creative process of all facets of game development.

Although similar to one another, the development of a serious game diverges in many ways from that of a traditional game. In the case of a traditional game, the development process is driven by the vision of a group of people that expect and plan the result to be successful, with a sufficiently high level of market success to justify the development costs. On the other hand, with a serious game, the development process is driven by the user requirements of a well-defined set of users that will and intend to use the resulting serious game. The addition of requirements engineering requires more creativity (Robertson, 2002), which according to some (Nguyen et al, 2000), adds additional complexity that ultimately may compromise the success of the project.

The gaming industry is fully aware of the high risk associated to game development, both traditional genre and serious game. Therefore attempts have been made to improve the development methodology by addressing the challenges raised. A first approach has been to adopt well structured frameworks and methodologies from software engineering community and streamline the development process. There are numerous different development methodologies (Wiering, 1998) to choose from, but none is able to guarantee the success of a project. This supports the claim "silver bullets" do not exist (Brooks, 1987) to solve the nightmares associated to the software process of complex systems. It is widely accepted that software engineering implies more than sophisticated techniques (e.g. (McConnell, 1998)), which is the only way to tackle the complexity of large scale systems that are inherently multi-disciplinary (Grimson and Kugler, 2000).

As a result, there is a growing trend is search alternate development methodologies with some games adopting principles from the agile software development community (Beck et al, 2006). Such methods promote a light-weight software process based on iterative short cycles where software engineering is deeply ingrained throughout. The lean development philosophy is achieved by reducing the requirement of supporting documentation and relying on best practices to ultimately increase the productivity. However, an analysis (Turk et al, 2002) of the foundational principles, along with the necessary assumptions made about the software process, reveals that there continues to be no such thing as "silver bullets". In addition, the existing agile methods still require some maturity in order to address their limitations (Abrahamson et al, 2003), namely the disregard to generalization to address the issues of reusability and lack of practical guidance with respect to applicability.

One of the most popular and predominant agile methods is eXtreme Programming (XP) (Beck, 2000), which proposes twelve practices to replace the traditional development methodologies of the software process. Although the XP methodology is implementation oriented with much diminished emphasis on analysis and design, a small trial study (Wood and Kleb, 2002) demonstrated the effectiveness of the method in the research arena where the software process is exploratory with an associated high degree of uncertainty to the problem domain.

The High Moon Studios, responsible for games such as Dark Watch, have been using SCRUM (Controlchaos, 2006) for project management with agile principles and XP for the actual development (Keith, 2006). Both agile methodologies have been adapted to fit the nature of the teams involved in the game production. However, none of the methodologies support geographically distributed teams and although the risks of game development are mitigated, the underlying problems persist to plague the projects.

4 DEVELOPMENT METHODOLOGY IN PRIME

The potential pitfalls associated to game development were known from the onset of the project, and it was recognised the need for an iterative development approach that supported both the exploratory nature necessary in research and a high level of task parallelization. In addition, the PRIME project is obligated to develop a serious game that addresses the needs of 6 end-user companies. This has proven to be an additional challenge as the gaming awareness within the corporate knowledge was very low or even non-existent, which led to a high uncertainty concerning the user requirements (Baalsrud Hauge et al, 2006).

4.1 Overview

The uncertainty of the problem domain entailed the adoption of a development approach that was based on an iterative methodology. However, the classic relevant development models, such as exploratory or spiral, were not sufficiently adequate for PRIME, neither did the methods ensure the successful completion of the entire project within 24 months.

A schematic representation of the PRIME development methodology is depicted in Figure 1, illustrating the iterative approach by the spiral process (coloured in light green) with four major milestones coinciding with the major releases of the PRIME software. The first milestone at month 6 corresponds to a cardboard prototype of PRIME; the second milestone corresponds to the release of the alpha version of PRIME at month 12, coupled with the testing and integration methodologies; the third milestone is the beta release of the PRIME software coupled with the PRIME-Time methodologies and evaluation framework; and finally at month 24, the final major version of the PRIME software is released along with the result analysis of the evaluation process.

Although the milestones are pre-determined, the spiral process implies that the development is dynamic, meaning that the nature of PRIME software adapts to the needs of the six end-users and what is identified by the consortium to be effective generalisations to support strategic decisions in the context of global manufacturing. This approach implies that the involvement of the end-users from the start of the project in truly a user-centred approach.

Unlike the classic spiral development methodology, PRIME does not have iterative stages of specifications, design, prototype and evaluation. A more agile approach is taken with the following four deeply ingrained principles:

- People and communication over processes and management tools;

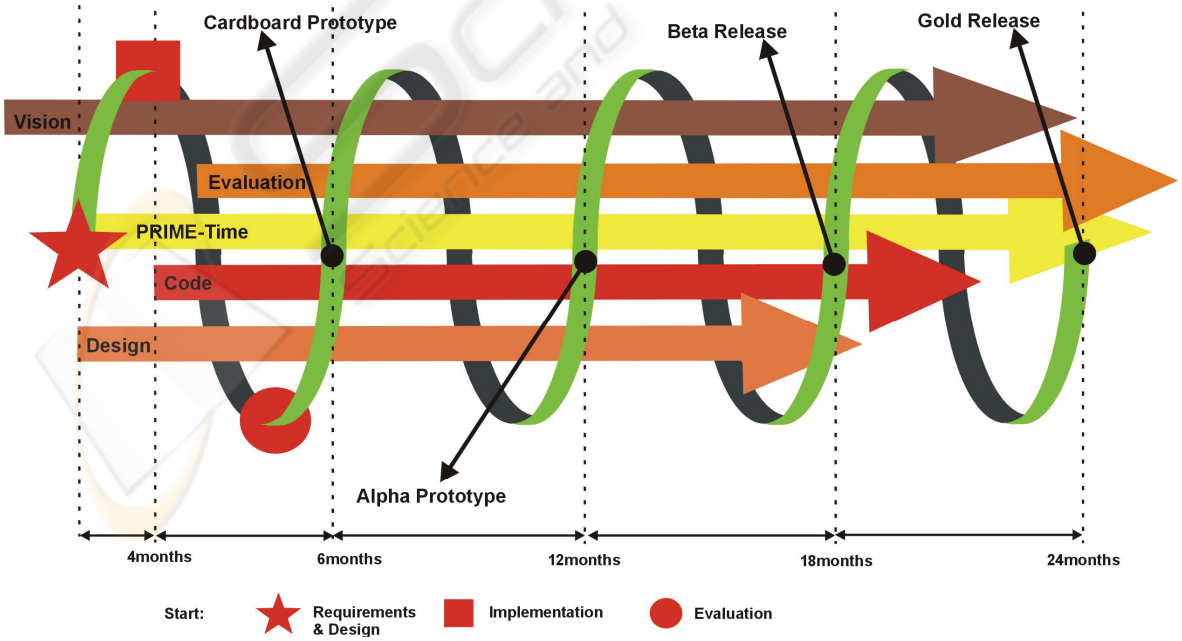

Figure 1: PRIME development methodology (Oliveira, 2006 b).

- Working documents and visual artefacts (prototypes, storyboards, images, animations) over comprehensive documentation;
- End-user collaboration over frozen functionality;
- Response to change over established plan.

In addition, the development activities are aggregated into five well-defined strands that operate in parallel:

- **Vision**. The aim of the Vision strand is to convey a consolidated view of the project outcome within the consortium, which is essential in managing the user expectations and to provide a common understanding amongst the development team of developers, who are geographically distributed and working in parallel. The vision will continue to evolve as the user centred approach integrates the feedback from the end-users. This strand was initiated 4 months in advance from the official start date of the project.
- **PRIME-Time**. The PRIME-Time strand encompasses all the methodologies aimed at the creation of new work environments based on the usage of PRIME integrated into current work environments, in both the Industrial or Academic environments. There are nine key concepts that are part of PRIME-Time: Player (knowledge worker), training, evaluation, time, place, award, motivation, monitoring and management.
- **Design**. The Design strand corresponds to the activities that shape and mould the game design of the PRIME serious game, defining the game play and the game mechanics, along with the underlying simulation model. The strand takes into account the feedback from the end-users and the concerns related to the integration of PRIME-Time into real work environments.
- **Code**. This strand corresponds to many of the traditional implementation activities, namely the user requirements, technical specifications and the actual implementation.
- **Evaluation**. The Evaluation strand covers both the testing of the output of the development and the validation of the initial hypothesis that managers will gain experience within a virtual environment that allows soft-failure.

All the strands operate in parallel, with information flowing between them, thus influencing the final output in terms of the PRIME software and PRIME-Time methodologies.

4.2 Developer-End-User Partnerships

To facilitate the development process, developer-end-user partnerships were established according to Table 2. Each one of the developers acquired detailed knowledge of their end-user during the solicitation and elaboration of the user requirements by means of close collaboration. This deep synergy, which was established in the initial phase of the project, allows a developer to champion the interests of the corresponding end-user whilst absent from the developer meetings and technical brainstorming sessions.

Table 2: Developer-end-user partnerships in PRIME.

End-User	Developer
CRF	MIP
IAI	BIBA and Alfamicro
Intracom	EPFL
KESZ	Alfamicro
LEGO	Sintef
Siemens	BIBA

It was quickly identified that the lack of gaming culture, or awareness of what a serious game was, would add to the development challenges. Therefore, the consortium as a whole and by means of the developer/end-user partnerships worked to create the game culture within the consortium, which included the realisation of internal workshops presenting examples of serious games and holding discussions on broad user scenarios. Although 6 partners of the consortium are characterised as end-users, some of the developers themselves are also end-users. Alfamicro will be using PRIME as a consultancy tool, whilst BIBA, EPFL, MIP, Sintef and Sofia will be using PRIME as an education tool. This promoted ownership and interest in the usage of the PRIME results by all the partners within the consortium.

The developer/end-user relationships will remain in place throughout the duration of the PRIME project. Each pair developer/end-user holds virtual and face-to-face meetings to carry out the necessary work.

4.3 Working Groups

During the first twelve months of the project, there has been one kick-off meeting (September 2005 on Madeira Island) and three project meetings (Novem-

ber 2005 in Vienna, March 2006 in Budapest and June 2006 in Athens). The project meeting in Budapest coincided with the six month milestone and the "cardboard" demonstrator was presented to the end-users. With just four project meetings, all the development activities have been done together by means of a collaborative platform and a dynamic flexible management methodology to effectively support creativity within a virtualised space.

With seven geographically distributed developer partners, it was necessary to effectively coordinate all the developers involved, which exceeded 25 individuals with differing cultural backgrounds. At the first instance, coordination was based on a classic management hierarchical structure with virtual meetings between workpackage leaders and the corresponding task leaders. The workpackage leaders would in turn have a meeting with the Operation Manager. Only the individuals responsible for active tasks and workpackages needed to participate in the meetings. This structural organization was in operation until the second project meeting (Vienna – November 2005) where it was acknowledged that there were delays in the project. The major culprit was the inherent "waterfall" mentality affecting most of the developers, whom recognised the spiral process of the PRIME development methodology, but remained adamant about sequential task execution. It was evident that an alternative approach was necessary. In addition, the digital platform put in place to support electronic brainstorming was only used by a few individuals.

The adopted solution was to set up Working Groups (WGs), which would have a dynamic life cycle, being created to address a particular need and lasting until their purpose had been achieved. These WGs consisted of small number of developers, where each individual would be representative of their own local team. Consequently, the number of participants would not exceed five at the most, but the development team consisted on average of 25 individuals from seven different organizations. On occasion, some additional individuals would participate to a particular WG meeting, but solely as observers.

The initial two WGs were focused on the game object model and the system object model respectively. Other WGs were created, dissolved and merged, with the current WGs consisting of (in parenthesis are identified the participating partners with the WG leaders in bold):

- Game Design WG (Alfamicro, BIBA, MIP, Intrapoint and Sintef). This working group is the result of the merging of the Game Object Model and Gamer WG. Their responsibility is to maintain and validate the Game Object Model, the game play and the corresponding game design. The active members of the WG are all individuals who are considered gamers, thus with experience in either playing extensively games during their growing up or developing games.
- System Design WG (Alfamicro, BIBA and Intrapoint). This working group evolved from the System Object Model group. The group is responsible for the maintenance of the technical specifications and overseeing the implementation of the PRIME server, client and middleware. However, the WG is driven by the outputs of the Game Design WG.
- Simulation Model WG (Alfamicro, BIBA, EPFL and Intrapoint). This working group is relatively new and emerged form the System Design WG to address all issues concerning the hierarchical simulation model that supports the Virtual Business Environment.
- Toolset and Artificial Stakeholders WG (Alfamicro, BIBA and Sofia). This working group tackles the orthogonal development activities to the PRIME server and client, namely the PRIME Toolset and the Artificial Stakeholders.
- PRIME-Time WG (Alfamicro, EPFL and MIP). This working group focuses in developing the PRIME-Time model, identifying barriers and developing effective adoption methodologies.

Initially there would be on average 4-6 virtual meetings throughout a month. With the WGs, the number of meetings increased to an average of eight meetings a week, with some groups having more than two meetings a week. The aim of a WG meeting would be to brainstorm using supporting working documents and the existence of clear objectives would keep the duration to less than two hours. Once the WG concept was implemented, it fomented the realization of additional bi-lateral meetings, which would be more brainstorming intensive and exceed the duration of two hours threshold.

Although each WG had a group leader, the overall coordination of all WG was done by someone who assumed the role of Producer. This person became responsible for driving all the development activities and being actively involved in all the working groups. The Producer would in turn liaise with the Operation Manager and together, they would identify risks and develop contingency plans that were discussed with the Quality Risk Manager.

All the supporting documentation and working documents generated by the WG would be available in an online document management system for easy access by all.

4.4 Development Status Overview

The PRIME vision has been developed and matured, providing the framework for all development activities. In addition to the development milestones, many more activities produced results, ranging from surveys to the PRIME portal.

At month 6 (end of month February 2006), the release of a cardboard prototype of the PRIME client coupled with a storyboard describing the context of the human player interaction with the VBE. This prototype was limited in functionality, but allowed the end-users to interact with the demonstrator and sign-off their approval.

In addition to the demonstrator, a Game Object Model with 200 game entities was developed, along with the game play encompassing the various strategic decisions that a manager of a Business Unit will take. The game design itself has gone through 3 major iterations, taking into account the expectations and gaming background of the end-users.

The PRIME-Time concept has been developed and matured, thus providing a framework for the integration of PRIME in existing work environments and the evaluation activities.

At month 12 (end of August 2006), the implementation of the alpha version has progressed. The feedback of the end-user organizations on the prototypes has enriched the features and contributed to an increase in the complexity of the problem domain. This has led to rescheduling and reprioritization of tasks to accommodate the minor delay introduced.

5 CONCLUSIONS

In PRIME, it was observed that technology alone is not sufficient to enable a process, namely formulation, discussion and refinement of ideas. However, it is not sufficient either to enforce a singular methodology since the process is dependent on a multidisciplinary team of individuals, each with their own cognitive mental models that most likely differ from one another. Another two important factors are the geographic dispersion of the team and the different contexts associated to a task. The project began with a strict hierarchical managerial approach to the work supported by collaborative tools, namely electronic support for brainstorming. During the third month of the project, it became obvious that delay was incurred and it was necessary to adopt an alternative approach or risking non-completion of the project within the designated time with the allocated resources.

In the case of the PRIME team, it was necessary to adopt a flexible project management process that evolved according to the needs as dictated by the circumstances. At the heart of the process was the producer with a hands-on approach, who supports the operational manager of the project. Although during the first twelve months, the producer centralized the information flow, this has gradually changed as the work shifts more into implementation and the creative entropy wanes.

The producer is responsible for the coordination of dynamic working groups, which would be responsible for a single or set of tasks for a given time period. In addition to the working groups, there were the end-user/developer partnerships to ensure a user centred development approach. This pairing of partners proved particularly successful with the phase of requirement engineering, where it was very important to guide end-users through user requirement analysis and to inform them at an early stage of the advantages and limitation of the simulation underlying the game.

The sharing of information was done via the collaborative platform, which adopted principles of agile programming community, promoting clear communication based on working documents and visual artefacts.

The success of the methodology has allowed the project to recuperate the initial delay and accommodate the emerging problems within the development process and the PRIME functionality. The developer partners have initiated the adoption of the methodology into other of their projects, in particular during the phases where creativity is predominant and the team is geographically distributed.

The PRIME methodology has a wider applicability, as in the case of the designing and creating the manufacturing processes of a new product, namely involving the use of plastic moulds and special tooling. The design and engineering of a new product strongly benefits from a closer contact between product designers, tool makers and product manufacturers. However, further research is necessary to evaluate the effectiveness of the PRIME development methodology in a broader case.

ACKNOWLEDGEMENTS

The authors thank the respective project partners, as well as the European Commission for all scientific, organisational and financial support. The PRIME project is partially funded under contract number FP6-016542 within the Sixth Framework Programme, Priority IST/NMP.

REFERENCES

Abrahamsson, P., Warsta, J., Siponen, M., Ronkainen, J., 2003. New Directions on Agile Methods: A Comparative Analysis. *IEEE Proc. International Conference on Software Engineering.* Portland.

Annetta, L. A. et al, 2006. Serious Gaming: Incorporating Video Games in the Classroom. Educause Quarterly. Number 3, 2006.

Baalsrud Hauge, J., Duin, H., Oliveira, M., Thoben, K.-D., 2006. User Requirements Analysis for Educational Games in Manufacturing. In: *Proceedings of the 12th International Conference on Concurrent Enterprising: Innovative Products and Services through Collaborative Networks (ICE 2006)*. 26-28 June 2006, Milano, Italy.

Beck, K. et al, 2006. *Manifesto for Agile Software Development.* www.agilemanifesto.org, last accessed April 2006.

Beck, K., 2000. *Extreme Programming Explained: Embrace Change.* Addison Wesley.

Brooks, F., 1987. No Silver Bullet: Essence and Accidents of Software Engineering. *Computer*, Vol.20, N. 4.

Controlchaos, 2006. www.controlchaos.com. last accessed April 2006

Diehl, M., Stroebe, W., 1987. Productivity loss in brainstorming groups: Toward the solution of a riddle. *Journal of Personality and Social Psychology*, Vol. 53.

Gamasutra, 2006. www.gamasutra.com. Last accessed in April 2006.

Gallupe, R., Dennis, A., Cooper, W., Valacich, J., Bastianutti, L., Nunamaker, J., 1992. Electronic Brainstorming and Group Size. *Academy of Management Journal*, Vol. 35.

Gouran, D., Hirokawa, R., 1996. Functional theory and communication in decision-making and problem-solving groups: An expanded view. In: R. Hirokawa and M. Poole (Eds.): *Communication and group decision making*. Thousand Oaks, CA, 2nd Edition.

Grimson, J., Kugler, H., 2000. Software Needs Engineering – a Position Paper. *Proceeding of ICSE*, Limrick.

Glass, R., 1995. *Software Creativity*. Prentice Hall, Englewood Cliffs, NJ.

Kracke, R., Baalsrud Hauge, J., Duin, H., Thoben, K.-D., 2006. Training of Strategic Decisions in Collaborative Networks through Serious Games. In Camarinha-Matos, L. M., Afsarmanesh, H., Ollus, M. (Eds.): *Network-Centric Collaboration and Supporting Frameworks. IFIP TC5 WG 5.5 Seventh IFIP Working Conference on Virtual Enterprises, 25-27 September 2006, Helsinki, Finland.* Springer.

Keith, C., 2006. Agile Methodology in Game Development: Year 3. *Proc. Computer Games Developer Conference*. San Jose.

McConnell, S., 1998. The Art, Science, and Engineering of Software Development. *IEEE Software*. Jan/Feb.

Nguyen, L., Carroll, J., Swatman, P., 2000. Supporting and monitoring the creativity of IS personnel during the requirements engineering process. In: *Proc. of 33rd Hawaii International Conference on System Sciences (HICSS-33)*. Maui.

Oliveira, M., Andersen, B., Oliveira, A., Rolstadas, A., 2006a. Using Serious Games to Improve European Competitiveness. In: Cunningham, P., Cunningham, M. ,(eds) 2000. *Exploiting the Knowledge Economy: Issues, Applications and Case Studies*. IOS Press.

Oliveira, M., Andersen, B., Oliveira, A., Rolstadas, A., 2006b. A Case Study of Applying Agile Principles to Build a Serious Game in Strategic Manufactuirng. In: Cunningham, P., Cunningham, M. ,(eds) 2000. *Exploiting the Knowledge Economy: Issues, Applications and Case Studies*. IOS Press.

Osborn, A, 1957. *Applied Imagination: Principles and Procedures of Creative Thinking.* Scribner, New York.

Paulus, P., Larey, T., Ortega, A., 1995. Performance and perceptions of brainstormers in an organizational setting. *Basic and Applied Social Psychology*, Vol. 17.

Robertson, J., 2002. Eureka! Why analysts should invent requirements. *IEEE Software*, Vol. 19.

Squire, K., 2005. Game-Based Learninig. An x-Learn Perspective Paper. Masie Center. E-Learning Consortium.

Turk, D., France, R., Rumpe, B., 2002. Limitations of Agile Software Processes. *Proc. of the Third International Conference on eXtreme Programming and Agile Processes in Software Engineering.* Alghero.

Wiering, R., 1998. A Survey of Structured and Object-Oriented Software Specification Methods and Techniques, *ACM Computing Surveys*, Vol. 30, N. 4.

Wood, W., Kleb, W., 2002. Extreme Programming in a Research Environment. *Proc. of the Second XP Universe and First Agile Universe Conference on eXtreme Programming and Agile Methods.* London.