SArt Project: Research in the Intersection between Software and Art

Anna Trifonova
Norwegian University of Science and Technology
Sem Saelands vei 7-9, Trondheim, Norway
trifonova@idi.ntnu.no

Letizia Jaccheri
Norwegian University of Science and Technology
Sem Saelands vei 7-9, Trondheim, Norway
letizia@idi.ntnu.no

ABSTRACT

In this article we present the research of SArt project at the Norwegian University of Science and Technology. In SArt we have the vision that software engineering can benefit from multidisciplinary research at the intersection with art for the purpose of increasing innovation and creativity. The group participates in the development of several interactive art installations and studies the issues and problems in this particular domain. Our findings show several peculiarities of the software development projects which require further study from software engineering perspective. Furthermore, we observe that some important software engineering concepts, like testing and maintenance, which are recognized as important in contemporary software development, but are often underestimated in interactive installation art.

Categories and Subject Descriptors

D.2.0 [**Software Engineering**]: General; J.5. [**Arts and Humanities**]: Performing Arts.

General Terms

Documentation, Management, Design.

Keywords

Interactive Installation Art, Software Engineering Research.

1. INTRODUCTION

Art finds expression in numerous products in society, where development of products is complex, competitive, global and intercultural in scope. The advent of multimedia technology has changed art production processes and the way both music, video, and figurative is fruited by consumers. Contemporary art is often heavily software dependent and the production of the software often needs the involvement of programmers and software engineers. Our assumption is that the interaction between software technology and art is beneficial for both fields.

Within the Software Engineering group at the Norwegian University of Science and Technology (NTNU) a project called SArt (http://prosjekt.idi.ntnu.no/sart) has been created. The focus of the project is the exploration of interesting research issues in the intersection of art and software. This project confronts the interdisciplinary problem of integrating software designs originating from different academic genre into better products for society. Our activities are mainly oriented at academic research, but we also take part in different artistic projects. We collaborate with many academic and industry partners.

In this article we present SArt project. We list the project's goals and objectives (section 2) and discuss the research methodologies utilized (section 3). We persent three interactive installation development projects in which we have participated and which we use as explorative case studies (section 4). We describe the software engineering issues which we have encountered or observed. Conclusions (section 5) are followed by references (section 6).

2. SART OBJECTIVES

The principal objective of the SArt project is to assess, propose and improve methods, models, and tools for software development in art context. We aim at addressing software engineering processes which are of interdisciplinary nature when meeting with arts. Particular attention is paid to art influenced software development. Sub-goals are:

G1. Develop knowledge on the interdisciplinary nature of software production in which the software engineering process interacts with the artistic process (i.e. the creation of heavily software-dependent artworks).

G2. Support artists who utilize or create software for and within their artworks by directing them and providing software engineering knowledge and tools.

G3. Disseminate the findings by publishing in the leading scientific journals and conferences.

G4. Educate competent practitioners and researchers for working in multidisciplinary teams in software industry and artist communities.

Throughout the project we would like to gain better understanding of how software and art are connected in order to facilitate a better collaboration between the artists and software developers. The project aims to carry out research on the following areas:

- Bridging the need of artists with artistic software
- Facilitate collaboration between software developer and artists
- Strengthen communication between new media art technologies and learning
- Establish a relationship with art and software to cultivate creativity and innovation in the software development process
- Improve social learning and increase interests for information technology through games and art.

3. SART RESEARCH METHODOLOGY

In SArt we use research methods based on empirical software engineering and benefit from the interaction with experts from diverse disciplines, e.g. artists.

At the beginning of the project we have performed a *literature review* in order to establish a knowledge base of the studied field, as suggested by [1]. We have started a *systematic literature review*, adopting the guidelines provided in [2]. The systematic review was with a brad focus (details about the process and the first outcomes are available in [3]) and covers scientific articles published in computer science conferences and journals. The systematic review considering interactive installation art has revealed five software engineering concepts discussed in the literature: 1) software requirements; 2) software architecture and design; 3) testing and evaluation; 4) process models and project management; 5) development environments and tools [4]. It also reveals that certain software engineering concepts which have been recognized as important in contemporary software development have not been discussed in relation to interactive installation art.

In order to gain insight of the issues and problems in the software production for artistic purposes we have followed the development of three installation art projects (description given in section 4). These projects serve as *explorative case studies* [1]. Several data collection methods were used. The historical data was collected through analysis of previous documentation and reports. Analysis was done of the software code, often published as open source. In addition, members of SArt have participated in the organizational meetings between the developers and the artist in the Sonic Onyx. A questionnaire was compiled by the artists (S. M'kadmi and Ø. Brandtsegg, who is also the main developer in Flyndre) and follow-up discussions were made regularly.

Some of the software interventions have been done with *action research* [5] where software engineering students from NTNU have been taking the role of the action researcher, while the artist have been acting as a client. The cyclical process model allows reflection to be done on each step (i.e. each iteration cycle).

4. OUR INSTALLATIONS

Open Digital Canvas project, shown on Figure 1 (http://prosjekt.idi.ntnu.no/sart/wiki), aims to embellish a white wall at NTNU with a number of main boards with LEDs on them, creating a big matrix of light pixels. The installation is created with the goal to inspire reflection about Information and Communication Technology with focus on openness, copyrights, and authorship.

A prototype of the canvas was created during a master thesis [6]. The current complete version of the installation has been developed during the 2007-2008 Experts in Team (EiT) course at NTNU [7]. EiT gives to the students a possibility to experience true multidisciplinary work and intense cooperation. Through the project students meet real-live problems which have to be solved as a group and by applying creativity [8-10]. A software engineering professor and an art consultant were following and helping the students throughout the whole development process.

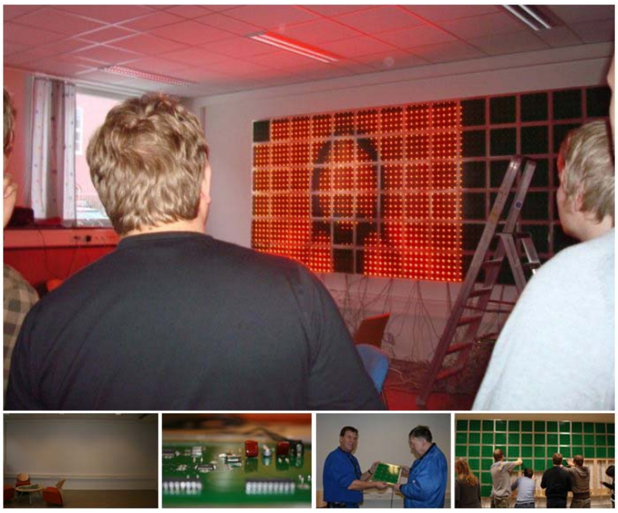

Figure 1: Open Digital Canvas, NTNU, Trondheim, Norway

When choosing the programming languages and the software tools to be used in Open Digital Canvas the developers gave priority to those that support multiple platforms. The software framework was created around the idea of openness, so open source software was chosen whenever needed. User Datagram Protocol (UDP) is used for communication with the boards. An open source Flash player, called Gnash [11] has been chosen for converting Flash animations and real-time movies into the raw data passed to the wall, scaling the image down and translating color into grayscale. The new software for controlling the boards was designed from scratch and was programmed in C. It will be published as open source. The chosen license is BSD.

Several challenges have been met during the installation development. The mounting of the boards have created difficulties, as the framing had to be strong to hold the boards' weight, to allow easy way to change single board in case of malfunction and at the same time aesthetical. The dimension of the installation had to be carefully planned. An aesthetical choice has been taken, i.e. to preserve the size ratio of the wall on which it has been mounted. This, however, influence on what might be presented and how the data should be converted by the software. From software engineering perspective an important observation was that the previously developed tools (i.e. software from similar projects) were not suitable to reuse. The reason was the poor quality of the code. Although it was open source it had little or no commenting, it was not developed in a flexible manner, it did not allow extending and adding new functionalities.

The hardware and the software of Open Digital Canvas have been tested extensively before deployment.

Sonic Onyx (www.soniconyx.org), Figure 2, is an installation commissioned by the Trondheim municipality in 2007 to the artist Samir M'kadmi. The project is part of the municipality's effort to introduce innovative art to public spaces and institutions in Trondheim. Sonic Onyx is an interactive sculptural space in Blussuvoll skole with which people may interact by mobile phones.

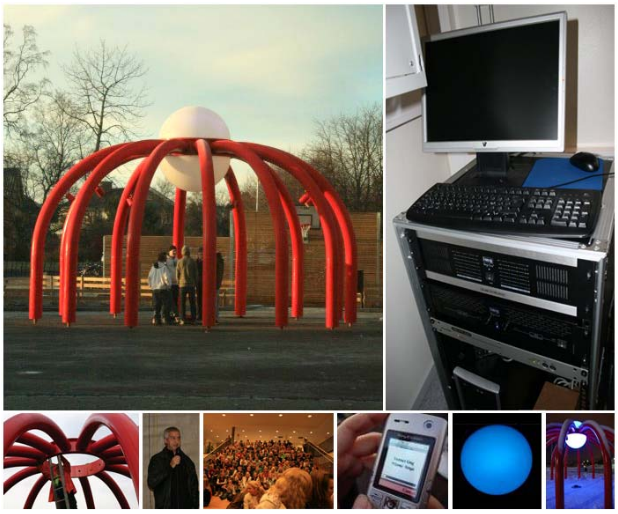

Figure 2: Sonic Onyx, Blussuvoll skole, Trondheim, Norway

The goal of the project is the creation of an interactive 3D sculpture for the court yard of a Secondary School in Trondheim (Norway). The sculpture is about four meters high and the "space" is about seven meters in diameter. The globe on the top is 1.75 m. The 'legs' are metal and static, while the globe is changing its color and intensity. The color depends on the surrounding temperature. The intensity of the light, on the other hand, changes according to the speed of the wind. In this way the sculpture is constantly varying its appearance. Additionally, sound is played all the time, which attracts the pupils (12-14 years old) and other spectators.

Although the appearance itself is very important in its aesthetical manifestation the sculpture provides another important functionality - the possibility to interact with its sound system. The visitors are able to go inside the sculpture to experience the 3-dimensional sound and while near the "space" they are able to interact with the system through their mobile phones. The communication is via Bluetooth and the system accepts messages and files. In real-time the content sent is processed by the software and the interaction influences the played by the sculpture sound. This includes conversion of images and text into sound and combining it with the background music. In cases when sound content is transferred to the system it is transformed into unrecognizable sequence. The music played is archived and played again in a random way when there is no traffic.

Several issues had to be faced for the creation of the sculpture. Apart from the physical creation of the space, the hardware and software of the sound and light systems had to be designed and developed. Part of the software has been created by professional software developers, specialized in the specific hardware (i.e. the one controlling the light ball). The software responsible for the Bluetooth interactions and for the sound transformations have been designed and developed in cooperation with students from HiST school (Trondheim, Norway). The work includes the planning of the communication technology, evaluation of the equipment that is needed and planning and programming of the sound software [12]. The members of the team have chosen to use open source technology (e.g. PureData [13], Apache [14], Python [15]) due to the availability of community support for maintenance and upgrade. The free software also allows development of a low cost system. Both the software developers and the artist have preferred to publish the software as open source.

The development of the software has been completed shortly before the Sonic Onyx opening. Little testing has been done and, later, crashes of the software have been experienced. As the installation is planned to remain in the school yard for a long period (several years) the robustness of the software is crucial. Software maintenance is still in the planning phase.

Flyndre (www.flyndresang.no), see Figure 3, is a sculpture located in Inderøy, Norway. It has an interactive sound system that reflects the nature around the sculpture and changes depending on parameters like the local time, light level, temperature, water level, etc. The main creator of the sound installation of Flyndre is Øyvind Brandtsegg - composer, musician and programmer. The original sculpture was created by Nils Aas without interactivity and sound.

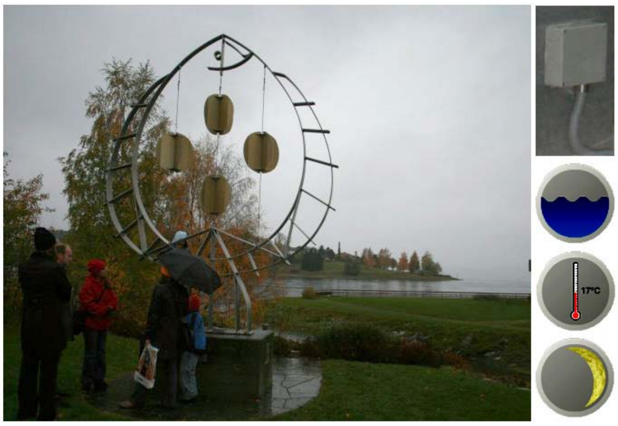

Figure 3: Flyndre, Inderøy, Norway

Initially, the software behind the sculpture, called ImproSculpt, has been created entirely by the artist in CSound [16]. The artist had implemented his artistic ideas without following software engineering principles, as the software was not expected to grow into a very complex system. As the system became more complex it became obvious that a long-term planning is required.

Through a series of discussions, informal meetings and more formal interviews with the artist two NTNU students analyzed the non-functional requirements [17]. They have used a miniature version of Architectural Trade-off Analysis Method process (ATAM, www.sei.cmu.edu/architecture/ata_method.html). The chosen intervention consisted in re-designing the former software to follow the Model-View-Controller pattern.

The proposed modular architecture facilitates easy modification, code reuse and continuous extension of the system. In order to modify the architecture they have prioritized the quality attributes according to their importance: performance, modifiability, availability, usability, testability, security, safety. The students concentrated their efforts on improving modifiability. The tactics which the student used is of polarizing responsibilities into separate packages in the system, thus improving the object orientation and maintaining semantic coherence. Although the software has evolved after the intervention the artist, who is also the main developer, has continued to follow the Model-View-Controller pattern.

Other NTNU students, as part of their pre-thesis study [18], were involved in the publishing of ImproSculpt as OSS. Students' intervention included licensing of the ImproSculpt source code, adding the appropriate terms and conditions documents in the software packages, publish the software on SourceForge, setting up a CVS environment and a Wiki site. Currently, ImproSculpt is licensed under the GNU General Public License (GPL) schema (http://sourceforge.net/projects/improsculpt/). This type of license was chosen mainly due to its popularity. The artist required a license that will provide protection against theft of the code into commercial closed source projects. The developers believe that the GPL license will best guarantee such high protection.

Flyndre installation is planned to exist until 2016, thus the maintenance of the software is essential.

5. CONCLUSIONS

Throughout the SArt project at NTNU we aim at gaining better understanding of how software and art are connected. The goal of our research is to facilitate artists and software engineers by studying the way software is being developed and by providing appropriate literature and guidelines to the best practices. In addition NTNU students profit from the multidisciplinary collaboration which gives them unique experience.

We have discussed the research methods we employ in our work. The literature review approach has allowed us to create a knowledge base – a collection of reported experiences from other computer science researchers and software developers in the domain of interactive installation art. The case studies (i.e. the installation projects described here) give us a first-hand experience of the problems and issues not reported in the literature.

We have observed several software engineering issues that have been underestimated, but need particular attention and further research. For example, robustness proves to be an important quality attribute but is rarely discussed. Testing and maintenance of the software are also essential as interactive installations often have a long life. Project management is rarely done by experienced people. This might lead to oversights, like lack of risk assessment, unrealistic schedule or budget.

6. REFERENCES

[1] Oates, B. J., *Researching Information Systems and Computing*: SAGE Publications Ltd, 2006.

[2] Kitchenham, B., "Procedures for Performing Systematic Reviews", Keele University Technical Report TR/SE-0401 and NICTA Technical Report 0400011T.1, 2004.

[3] Trifonova, A., S. U. Ahmed, and L. Jaccheri, "SArt: Towards Innovation at the intersection of Software engineering and art", in *Proceedings of The 16th International Conference on Information Systems Development (ISD'07),* Galway, Ireland: Springer, 2008.

[4] Trifonova, A., L. Jaccheri, and K. Bergaust, "Software Engineering Issues in Interactive Installation Art", *Int. J. of Arts and Technology (IJART),* vol. 1 (1), 2008.

[5] Davison, R., M. G. Martinsons, and N. Kock, "Principles of canonical action research", *Information Systems Journal*, vol. 14 (1), pp. 65-86, Jan 2004.

[6] Mendoza, N., "Open Digital Canvas", in *Department of Computer and Information Science,* Master thesis, Trondheim, Norway: Norwegian University of Science and Technology (NTNU), 2007, p. 123.

[7] Jaccheri, M. L. and G. Sindre, "Software Engineering Students meet Interdisciplinary Project work and Art", in *11th International Conference on Information Visualisation (IV),* Zurich, Switzerland, 2007.

[8] Abrahamsen, E., E. T. Baadshaug, K. Høyland, V. M. Iversen, C. R. Jensen, and C.-E. Kim, "Heart and Software", Norwegian University of Science and Technology (NTNU), Trondheim, Norway, Project report from Experts in Team (TBA 4850), 2008, online at http://prosjekt.idi.ntnu.no/sart/publications/product_report-group_green-heart_and_software.pdf.

[9] Pedersen, A. F., A. J. Rasmussen, F. Kristian, J. B. Knyvi, M. H. Omdahl, and P. D. D. Tran, "Lux Vitae", Norwegian University of Science and Technology (NTNU), Trondheim, Norway, Project report from Experts in Team (TBA 4850), 2008, available online at prosjekt.idi.ntnu.no/sart/publications/productOransje.pdf.

[10] Grana, P., B. Raae, L. Skjelbreia, M. Nordahl, S. Pedersen, and C. Carlson, "The IDI-wall Project", Norwegian University of Science and Technology (NTNU), Trondheim, Norway, Project report from Experts in Team prosjekt.idi.ntnu.no/sart/publications/ProductYellow.pdf.

[11] "Gnash (GNU Flash movie player) web site", www.gnu.org/software/gnash/

[12] Asphaug, T., C. Bielsa, E. M. Grytå, E. Thorsrud, and T. Tollefsen, "Sculpture with 3D Interactive Sound", in *Department of Electronical and Computer Engineering, Faculty of Technology*. Bachelor HiST, Norway, 2007.

[13] "PureData web site", http://puredata.org/

[14] "Apache web site", http://www.apache.org/

[15] "Python web site", http://www.python.org

[16] "CSound web site (sound design, music synthesis, and signal processing system)", www.csounds.com

[17] Semb, T. A. G. and A. Småge, "Software Architecture of the Algorithmic Music System ImproSculpt", in *Dep. of Computer and Information Science,* Master thesis, Trondheim, Norway: Norwegian University of Science and Technology (NTNU), 2006, p. 70.

[18] Collet, T. and M. Ramirez, "IMPRO SCULPT: Open Source - Artistic Software ", Norwegian University of Science and Technology (NTNU), Trondheim, Norway, Report for TDT4705 - Software Engineering, Depth Study, 2006, www.idi.ntnu.no/grupper/su/fordypningsprosjekt-2006/When_OS_meets_AS.pdf