Real Time Synchronization for Creativity in Distributed Innovation Teams

Dennis Kjærsgaard Peitersen, Peter Dolog, Esben Staunsbjerg Pedersen,
Kenneth Høst Pedersen, and Yujian Lin

IWIS — Intelligent Web and Information Systems,
Aalborg University, Department of Computer Science,
Selma Lagerloefs Vej 300, DK-9220 Aalborg East, Denmark
dolog@cs.aau.dk

Abstract. In this paper we introduce a synchronization approach for real time collaborative sketching for creativity in distributed innovation teams. We base our approach on reverse AJAX. This way we ensure scalable solution for real time drawing and sketching important in creativity settings.

Keywords: Creativity, Real Time Collaboration, Synchronization, AJAX

1 Introduction

Creativity sessions are usually performed in teams with a lot of interactions between team members. They usually stand at a white board or sit behind a round table discussing new ideas, organizing them, connecting them to each other, validating them and so on.

Several creativity techniques have been introduced to guide and organize such sessions such as 5WH1, SCAMPER, or Six Hats to name just few of them. The main idea behind them is that they describe common practices to elicit new ideas, group members to elicit them, the ways of involving different members and giving them sufficient space in the session and so on. The techniques can be seen as different social conventions depending on what the creativity session goal is (idea generation, exploration, validation and a like). Another strong character of the creativity sessions is the real time collaboration which needs to be supported also in distributed teams as well.

Creativity is a knowledge co-construction process from learning perspective. Participant himself as well as its peers in the creativity session learn by performing brainstorming and by co-constructing knowledge about new ideas when generating them, when validating them and so on.

The creativity techniques inspired us in developing the idSpace platform. As different phases in creativity process are supported by different techniques, we believe that idSpace platform should support a choice from the techniques and an ability to mash them up. Furthermore, it needs to support idea management for further exploration and learning from the work of the others and mash-up such knowledge into the work of the others.

In this paper we discuss a prototype which sets a baseline for supporting such a creativity and knowledge co-construction. We describe a prototype which allow for meshing up different editors for sketching the ideas and subsequent idea management. We describe a solution to synchronize the editors in real time by using so called comet approach which contributes to the fact that the collaboration in the distributed creativity sessions is perceived as real time.

2 Related Work

The use of creativity techniques in industry varies. Some companies just use paper and pencil, some have adopted ICT-based techniques, which are used mostly in isolation though. The frequency and distribution of techniques and tools used varies from country to country. Brainstorming [6] is conducted widely, with mind mapping [1] as the most used supportive technique (see for example [5, 8, 3].

The reasons for using particular techniques differ slightly as well, reflecting culture, industrial domain and working style. Various techniques for the systematic structuring and refinement of ideas have been proposed. Examples are morphological analysis [10] and methods of structured inventive thinking like TRIZ (Teoriya Resheniya Izobretatelskikh Zadatch i.e. theory of inventive problem solving) and SIT (systematic inventive thinking) [4, 7].

In this paper we propose a solution which enables a real time combination of sketches performed under different editors when participants are distributed and are using Internet to collaborate. We do not limit ourselves yet to any of the mentioned creativity techniques as all of them might benefit from the solution we propose in this paper. We base our solution on the Liferay portal. This way we enable the use of any technique as a plugable toolset to the Liferay.

There are different possibilities for realizing real time collaborative editing. As we focus on Internet browser based collaboration, we look here only at those which are commonly used in such situation. One approach is to query server which keeps the share copy for editing in regular time interval. This creates performance problems. Another approach is to used recently introduced AJAX (Asynchronous Javacript) for subscribing and receiving back the request. Our approach is based on AJAX and we will discuss several approaches how to implement real time collaboration in the next section.

There are several real time collaborative editors available such as Adobe Buzzword[1], ZOHO[2] and the others. Most of them however support only text editing. We base our solution on MxGraph[3], graph editing software, to provide scalable solution for creativity sessions. We cannot use supported shared editing functions as they are based on one single model. We need to extend the synchronization to support synchronization of related models which are stored

[1] http://www.adobe.com/acom/buzzword/

[2] http://www.zoho.com/

[3] http://www.mxgraph.com/

separately. Furthermore, comet based approach we have implemented seems to provide more scalable solution.

3 Ensuring Real-Time Synchronization

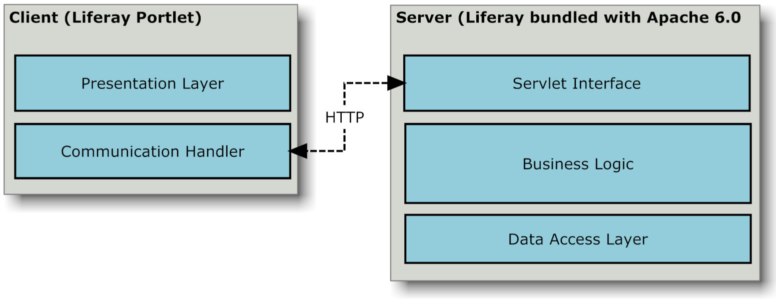

Fig. 1. Outline of the centralized client/server architecture.

The general architecture is designed as a centralized client/server architecture. Meaning that the model layer is handled by the server and clients can show a representation of the model. The centralized server pattern is used to ensure consistency in the graph models and collaborative editors. The architecture is outlined in Figure 1.

The real time collaboration on sketches in creativity sessions is based on synchronization of work performed under different instances of editors in distributed fashion. Traditional way of performing real-time synchronization between web clients is limited by the HTTP protocol. The HTTP protocol is build on a client creating a request for a web page and the server responding with the source of the page. In order to create a traditional real-time synchronization in collaborative editors would involve every client constantly asking the server if there is new updates. This creates a performance and scalability problems. An example of using this technique could be old chat rooms refreshing every second or so, and updating the screen accordingly.

Another approach lies in so called reverse AJAX. As AJAX it is not a technology by itself, but a technique using existing technology. Of course it is not possible to actually push data from the server directly to the clients, without them asking for it, but it can be emulated at a performance level, which is close to real-time.

There is a lot of different implementations of this technique, but the most widely used is the DWR framework. The DWR framework offers three different strategies for reverse AJAX: “Polling”, “Comet” and “Piggybagging”.

Polling: The browser makes a request to the server in regular intervals and receives updates when the server responds.

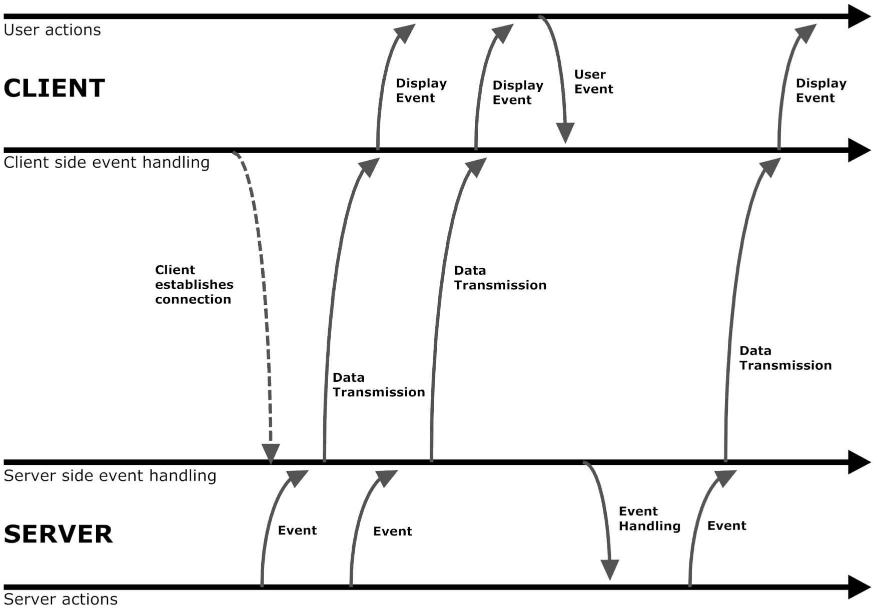

Fig. 2. Sequence diagram showing the comet technique.

Comet: The clients sends a request to the server, which then starts to answer very slowly. It answers so slow, that the connection is kept open at almost all time, which in fact eliminates the limitations of the stateless HTTP protocol. Data can flow in each direction at all times with this technique, so there is no blocking communication either.

Piggybagging: The client does not get updated until it sends a request to the server, the server then sends all updates back.

The solution chosen for the reverse AJAX technique between the client and server is comet, because it gives the best results in terms feeling the application runs in real-time. Figure 2 illustrates a sequence of events happening on both client and server, and outlines how they transmit data between each other. Holding the connection open at all times leads to the question of whether or not it will create thread starvation. Normally it will, but the DWR framework comes with implemented strategies to resolve the issue and ensure that the solution is scalable [9].

DWR is implemented client side by adding auto generated javascript libraries. The libraries is auto generated by a DWR servlet running on the tomcat server. Thereby eliminating synchronization issues, because the client will automatically get the last compiled server interface.

Collaborative editing uses a shared model, which different client represents and is able to manipulate. In order to create the one-to-many dependency between the model and the clients, an observer pattern is applied [2]. Figure 3

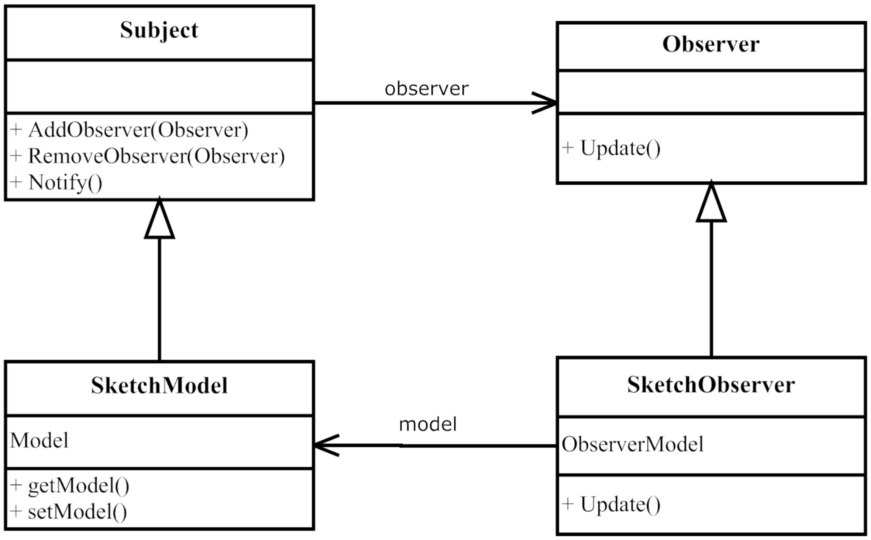

Fig. 3. Observer pattern applied to the sketch editor.

shows the class diagram of the observer pattern applied to the sketch model, which is observed by the clients. This design creates transparency for the developer and allows for further extensions to other editor types, such as idea editors, topic map representations etc.

4 Conclusions

In this paper we have introduced an approach for real time synchronization of different sketching editors for real time collaboration in creativity sessions. The approach uses comet and reverse AJAX as techniques of doing that. The advantage of the proposed architecture is that it keeps the communication channel between connected clients and server almost constantly allowing to send any kind of information between each other. However, with large models, we might still encounter performance issues. We will further study how to exchange only changed part of the model to improve performance and study further technologies to improve collaborative real time sketching supporting also larger teams. In the future work we plan to evaluate this approach with real users in the context of idSpace project.

Acknowledgement

We would like to thank our various project partners for inputs and discussions on earlier versions of the ideas presented in this paper. The present work was carried out as part of the idSpace project on Tooling and Training for collaborative product innovation http://idspace-project.org. This project is partially supported by the European Community under the Information and Communication Technologies (ICT) theme of the 7th Framework Programme for research and development (FP7-IST-2007-1-41, project number 216799). This document does not represent the opinion of the European Community, and the European Community is not responsible for any use that might be made of its content.

References

1. T. Buzan. *The Mind Map Book.* New York: Penguin, 1991.
2. Erich Gamma, Richard Helm, Ralph Johnson, and John Vissides. *Design patterns : elements of reusable object-oriented software.* Addison Wesley, 1995.
3. H. Geschka. Creativity techniques in germany. *Creativity and Innovation Management 5 (2), 87-92*, 1996.
4. N. Leon and H. Aguayo. A new model of the conceptual design process using qfd/fa/triz. In *Paper Presented at the QFD-Symposium, Novi, Michigan*, 2002.
5. J. Nielsen and K. F. M. Lieshout (Eds.). Awareness, use and effectiveness of models and methods for new product development. *European Journal of Marketing, Vol. 29 No. 10*, 1995.
6. A. F. Osborn. *Applied imagination: Principles and procedures of creative problem solving.* New York: Charles Scribners Sons., 1963.
7. J. Terninko, A. Zusman, and B. Zlotin. *Systematic Innovation — An Introduction to TRIZ.* CRC Press, USA, 1998.
8. Y. Xin, K. H. Chai, and J. A. Bauly. The application of new product development tools in industry. In *Proceedings of the IEEE International Engineering Management Conference 2004. Oct. 2004. IEEE Press*, 2004.
9. Frank W. Zammetti. *Practical DWR 2 Projects.* Appress, 2008.
10. F. Zwicky. *Discovery, Invention, Research - Through the Morphological Approach.* Toronto: The Macmillian Company, 1969.