社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE

クラスルームのためのオープンスペース
ーフェースツウフェース教育を支援するオープンソースソフトウエアー

ブロイヤ ヘニング† バロヤン ネルソン‡ ソウサ クリスティアン† 松本 充司†

† 早稲田大学 〒169-0051 東京都新宿区西早稲田 1-3-10, 29-7 号館

E-mail: † hbreuer@evangelion.ml.giti.waseda.ac.jp, ‡ nbaloiandcc.uchile.cl, † csousa@fuji.waseda.jp, † mmatsumoto@waseda.jp

あらまし 通常，教室における対話型のホワイトボードのためのソフトウェアは，事前に準備した静止画像に加えてリアルタイムのコメントを提示を示すためのツールとで構成されている．このデザインは学生に代わり，教師中心のインタラクションを向上させるであろう.教師とのインタビュー，文献調査，観測を通じて，我々は教育現場での制約を理解し,講義改善のガイドラインを引き出すことができた．我々は対話型のホワイトボードのオープンスペースのインタフェースで，オープンソースソフトを開発した.それはジェスチャーベースの相互作用パラダイムを採用した．教師や学生に学習資料の創作，構造，プレゼンティーションに柔軟性をもたせた．複数の学生との共同作業による学習では,ペンタブレットを通してホワイトボードにアクセスすることもできる．

キーワード クラスルーム技術，双方向電子白板，開放型ラーニング環境，共同作業，学習者中心のデザインパタン，ペンタブレット

An Open Space for the Classroom
－Open Source Software to Support Face-to-Face Education－

Henning BREUER† Nelson BALOIAN‡ Christian SOUSA† and Mitsuji MATSUMOTO†

† Graduate School of Global Information and Telecommunication Studies, Waseda University Waseda Bldg. 29-7, 1-3-10 Nishiwaseda, Shinjuku-ku, Tokyo, 169-0051 Japan

‡ Department of Computer Science, University of Chile, Santiago de Chile

E-mail: † hbreuer@evangelion.ml.giti.waseda.ac.jp, ‡ nbaloiandcc.uchile.cl, † csousa@fuji.waseda.jp, † mmatsumoto@waseda.jp

Abstract Software for interactive whiteboards in the classroom usually combines tools for showing prefabricated presentations with annotation features. Instead of encouraging students to participate by contributing this design promotes a teacher-centred interaction in the classroom. Through literature review, observation, and interviews with teachers we identified pedagogical and situational limitations of this approach and derived design guidelines. We developed open source software with an open space interface for interactive whiteboards. It applies a gesture-based interaction paradigm and allows teachers and students to flexibly create, structure, present and document learning materials. Within collaborative learning activities multiple students may access the whiteboard through pen-tablets.

Keyword Classroom Technologies, Interactive Whiteboards, Pen-Tablets, Open Learning Environment, Collaborative Learning, Learner-Centered Design Pattern

1. Introduction

Educational technologies and the learning environments they compose underwent radical changes within the last years. While students take their PDAs and Laptops into classrooms equipped with wireless connections and teachers use interactive whiteboards and interactive learning material the didactics and practices to support learning in such environments are falling behind the technological advances. Teachers who themselves have studied in completely different media environments need to apply new instructional methods that exploit the potentials these new technologies provide, but they are sometimes hardly coping with the interaction design and the instructional design, implemented in these new technologies. Intuitively usable interfaces, media properties and functionalities flexibly supporting various instructional methods are needed. A reasonably consistent interaction design across programs and devices may ease the interaction and therefore enhance opportunities for learning.

This paper and the prototype we present are part of an ongoing research and development project, in which we try to identify, implement and evaluate interaction design patterns for formal and informal learning environments using computer technology in the classroom and especially, using an electronic whiteboard. Design patterns are re-usable standard solutions to common problems in software design. They describe structural and behavioral features that improve the "habitability" of something – a user interface, an interactive installation, an object-oriented program, or even a physical architecture. In the field of Human-Computer Interaction they are usually being applied to support task-oriented activities. In this work, we adapt them to support learning activities. The main scenario will be interaction design patterns for classroom situations that combine the use of interactive whiteboards with several students' access through pen tablets. Aspects like the access management pose new requirements to this scenario. Moreover when multiple peripheral input and output media face a central means for content creation and presentation (like an interactive whiteboard) there is a proportional increased complexity within the logic of interaction.

2. Related Works

2.1. Whiteboards and gesture-based interaction

Primary functions of the traditional blackboard in class are to guide the lesson through different phases, to focus the views and attention of the audience towards a central stage, to document the lesson (enabling the students to rewrite its content for later studies), and to increase learning and understanding by providing structure [12]. The first electronic "LiveBoards" were developed by Xerox PARC. In the last years the SmartBoard™ hardware has been adapted by many schools and universities. These interactive whiteboards usually offer the functionality of taking an instant screenshot of the display which can be freehand annotated and stored as an image file.

Whiteboards have been criticized for promoting a teacher-centered approach giving a passive role to the learners [13]. Constructivist approaches to teaching and learning propose open, problem-oriented classroom activities. Problem-oriented learning environments that are moderated, but not driven by the teacher are considered favorable for relevant learning experiences [14]. They encourage self-directedness and collaboration as they combine explicit instruction (providing for guidance, orientation and help) and interest-driven constructive learner activities. Instead of presenting pre-fabricated learning content and transferring his knowledge to the students the teacher acquires the role of a moderator for his students who co-construct their knowledge with his support, using various media. The presentation-and-annotation philosophy of standard interactive whiteboard software does not support these kind of flexible activities well. Instead, it suggests proceeding operationally slide by slide through a pre-fabricated presentation. Co-creation and reuse of content and visualization is limited. The graphical annotation of a given presentation lacks a semantic dimension. Besides writing, drawing and erasing on the board, users are not supported in subsuming, inducing or deducing concepts or fading in and out aspects on the board – providing depth to the interaction.

In our previous works on software prototypes of the JavaFreestyler environment and the DeepBoard system [3] we tried to overcome the conceptual gap. We analyzed current usage of whiteboards in interactive classroom settings, conducted interviews with teachers and students, and defined guidelines and requirements for enhanced interaction. JavaFreestyler implements an integrated environment for presenting, running and annotating java programs in order to avoid having to switch constantly from one application to another during a lecture, distracting the concentration of the students. DeepBoard implements a hierarchical freehand writing application based on a full gesture approach for implementing interfaces for an electronic whiteboard. The concept of hierarchical writing provides for semantic relations between parts of the contents drawn on the e-board's screen and differentiates our approach from the Flatland interface mentioned in [8]. However, both prototypes we developed did not support simultaneous access by different clients.

Most related research on interaction with large-screen interfaces and interactive whiteboards focused on supporting collaborative workgroups and business meetings. Most "presentation software" that is usually being used in class like Microsoft PowerPoint, or Smart Notebook, has not been specifically designed for educational purposes [11]. Few educational lecture tools are able to mix writing, discussing, and the ability to generate high quality documents or graphics easily, without interrupting the class.

One example is FreeStyler [6]. It supports information retrieval in classroom settings. Different types of knowledge can be represented and structured. However, access is limited to a single user.

Within the business domain, cooperation in multimedia-offices and CSCW-environments has been studied. Productivity tools for primarily informal workgroup meetings [9] or gesture based support for object-oriented modeling [5] have been developed. An early application for the Xerox LiveBoard called Tivoli [16] explored the transition from freeform writing to structured interaction. It allowed groups of users in real time to flexibly organize and arrange materials on the board through direct manipulation of boundaries and recognition of "implicit structures" by the system. Some works examine gesture analysis in pen-based interfaces. "Flatland" uses freeform strokes as basic input and output primitives for office whiteboards, flexible screen segmentation and pluggable applications for different segments [8]. Proximate lines of research deal with display walls for information visualization in control centers, shared displays in meeting rooms, and large screen metaphors.

In the educational field, opportunities for free drawing in Kindergarten have been explored through observation [18]. Initiatives for collaborative learning in school [13] and the computer-integrated classroom [1] have been reported. Focusing on reuse of teaching materials, the eClass (or Classroom 2000) project introduced the notion of "teaching and learning as multimedia authoring" [4].

In trying to optimize the use of e-boards in classrooms, different setups of the environment and input devices like tablet-PCs, PDAs and Laptops have been considered. Hardly anyone questioned the quality of the board-show itself, taking the windows oriented desktop projection and annotation philosophy for granted.

2.2. Multiple Clients Access Systems

The individual in front of his screen has been a basic assumption in computing since it was popularized under the paradigm of personal computing. But collaborative learning in computer enhanced environments needs collaborative access to the presentation and content creation media.

Multiple clients are usually connected by a network of PCs that access an application. While in a centralized application a single PC acts as a server, peer-to-peer applications rely on the participants. LiveNotes for example applies tablet PCs, connected by a wireless network [10]. However, a different challenge comes up when multiple input devices access a single PC.

An early attempt achieved simultaneous input to applications with two mice [2]. A more recent approach [24] shows a Multi-cursor Window Manager, based on modifications to the Xserver and the X11 framework. A different solution, the SDG Toolkit [23], based on the Raw Input API of the recent MS Windows XP. The SDG Toolkit was created with the idea to provide an independent layer over which one can build applications with input from multiple keyboards or mice. The toolkit was used [20] to provide multiple inputs in student applications for developing countries where the ratio computer-students are very low. Another similar toolkit is called TIDL [7]. It supports multiple mice and keyboards for legacy and custom applications through an independent layer between the input devices and the applications. This layer also works across several PCs removing the distinction between local and remote input devices, that is, for all the applications all the inputs are local. Pen-tablets (also allowing for gestures) as input devices have not yet been considered, though.

3. An Open Learning Environment

A starting point for our design is the notion of openness as it is being referred to in the definitions of open-source software, open space technologies, and finally our own definition of an open learning environment.

3.1. An Open Projection Space for Learning

Open space is an answer to the question of how to deal with diversity, within an organization, a community, or even a classroom. Open space environments [19] invite participants to take responsibility for their passion. Initially instructed by a facilitator who presents the general theme, the process and its guidelines, the participants move around, interact and contribute freely to designated areas driven by their interests. They engage in self-initiated or ongoing discussions as long as they may contribute or learn something, or disengage and go someplace else. Using various materials they take notes and document their self-organized activities. While the approach has been primarily used for problem-solving activities with small to large groups of people, a primary outcome is the learning and motivation of the participants being achieved. In this respect it complies with constructivist learning theories that also emphasize motivation and active engagement of the learners, and propagate problem-oriented exploration and development of a given field of study.

We took this notion of openness as an inspiration for the design of the whiteboard interface. The interaction begins with an open white space that may be created by the teacher or students making a simple (rectangular) gesture on the whiteboard or pen-tablet. Participants can contribute content, and, moderating the students' discussion, the teacher may add, delete or link pieces of the content on the whiteboard.

3.2. Open Source Implementation

To implement the prototype without needing to start from scratch, we relied on open source libraries already available. We used the open source wrapper of the Wintab driver called WinTabdotNet [25] in order to implement the multiple input based on pen-tablets. This driver provides for the proper function of the pen tablets. Another important component is the RawInputSharp [21], which is a wrapper of the Raw Input. As we already mentioned before, the Raw Input API allows us to identify every input device, in this case every pen-tablet. Finally, to get the basic structure of the application, we used an open source diagramming library called Netron [17]. These three open source applications provide us with the starting framework for developing our application.

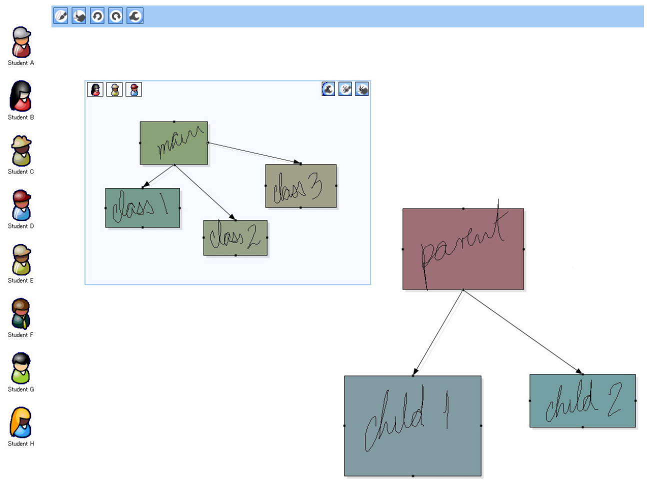
3.3. Gesture-based interaction

Gesture-based input is being supported across various input devices. Gestures are automatically detected as such and interpreted semantically by the system. For easy retrieval and follow-up work reusable materials are stored with their semantic structure. Each element (node) may contain recursively another structure with connected objects. In case of the whiteboards the final goal would be to reduce all necessary interaction to the moderating gestures and documentary writing on the screen. This allows the users to give a semantic, hierarchical structure to content that is elsewise free-hand written and stored as a picture.

There are two modes to operate the whiteboard's interface: command mode and write mode. Users can switch from one mode to another mode by pointing the electronic pen to the whiteboard twice rapidly in an area where no object is located. In order to maximize intuitive usability single-stroke gestures are being applied. For instance we use a simple rectangle to create an open space for someone's contribution that afterwards may be associated via drag and drop to other (groups of) participants, or other pieces of content. Gestures to move, relate, zoom or delete nodes, or import content are documented in [3].

3.4. Collaborative Learning

While experiencing the value of collaboration may itself be an objective of learning activities, the idea of collaborative learning refers to the potentials of motivating and enhancing learning activities through the collaboration of students: it is learning by and learning about collaboration. While systems for computer-supported collaborative learning (CSCL) usually aim at distributed learning environments our open-space environment is intended to support collaborative face-to-face activities.



(The image presents a screenshot of the whiteboard after the teacher and students have created some rectangular node objects for creating or importing content, and connected them by drawing from one node to the other.)

4. Classroom Interaction Design Patterns

The development of an interaction design pattern language to support classroom activities is currently underway. The pattern-approach intents to document, evaluate, and improve good solutions (with argument and context) to reoccurring problems in interaction design, and also to ensure consistency across platforms and applications. They are specific and abstract enough to be adapted and implemented across a range of different devices like interactive whiteboards, pen-tablets or PDAs within a computer-integrated classroom environment. A longer-term perspective of an emerging pattern language for educational technologies is the seamless integration of architectural, interaction design and software patterns for formal and informal learning environments.

Preliminary high level interaction patterns supporting open, problem-oriented learning environments address openness, collaboration, construction, and relation

- Open space for constructive activities: provides a white-space to collaboratively create and edit content. A gesture-based input paradigm for touch sensitive devices defines gestures to perform all necessary interactions directly on the whiteboard or from some tablet.

- Support of collaboration necessitated by tasks. Access orchestration (or "floor-control") allows granting and hindering access to shared screens like the whiteboard. A group presence widget represents the student group and their position in the context of learning activities. A shared screen mode makes individual input accessible for all participants. Access from multiple pen-tablets is being supported. (In our case this access is implemented in MS Windows XP using the WinTab.Net wrapper of the Wintab driver, which allows us to receive independently every input message from the pen tablets input device. The messages sent by the input devices are processed independently by our application.)

The following collection of brief pattern descriptions illustrates our current approach:

4.1. Pattern: Open Space

- Context: Problem-oriented or constructivist in-class learning activities, classroom activities
- Problem: Presentation and annotation of prefabricated materials do not encourage student participation. Problem-oriented learning should start with students' construction and discussion.
- Solution: Provide one or many open spaces for students to create content, and to put pieces into relation.
- Related patterns: gesture-based input, multiple clients' access.

4.2. Pattern: Gesture-based input

- Context: Problem-oriented or constructivist in-class learning activities, learning with whiteboards
- Problem: A central stage for projection helps to focus attention in class. Interactive whiteboards allow for flexible creation, editing and documentation of learning materials, but their use is often constrained by rigid software. Spending too much time in activities, which are not directly related to teaching (typing long commands or queries, or searching for files) may interrupt the dynamic flow of the lecture and distract the attention of the audience.
- Solution: Provide for a gesture-based interaction that enables users to perform all activities related to teaching on the interactive whiteboard as the unique input and output device.
- Subordinate patterns include the individual gestures, such as creating and linking nodes [3], related patterns may include multiple clients' presentation, group presence widget, remote access, access orchestration.

4.3. Multiple Client Access

- Context: Collaborative learning, learning with whiteboards
- Problem: The blackboard has been criticized for promoting a teacher centered instruction while students may contribute content only verbally. In remote lectures even this opportunity is lost. Collaborative learning activities may afford synchronous access of various students to the same shared resources.
- Solution: provide for multiple and distributed access to a shared screen (e.g. on a whiteboard) for individual students through individual mice, keyboards, pen-tablets or tablet PCs.

4.4. Pattern: Group Presence Widget

- Context: Collaborative Learning in class and between remote places, multiple client access
- Forces: Learners need to control their speed of progression. Seeing oneself in relation to others may motivate learners.
- Problem statement: Users should see their progression in relation to others but screen real estate is limited.
- Solution: Provide for a widget that indicates the current page number of each user in the current session using a small icon, which is displayed in the user's selected ink color, along the slider at a position relative to other users. The user moves between pages by dragging his icon along the slider. To allow for flexibility, automatic page turns are not supported and the user is responsible for advancing to the next slide.

5. Future Work

In order to support systems like Linux, it is indispensable to find an equivalent of the WinTab.Net and Rawinputsharp, which are dependent of the underlying operative system. We should proceed by the same means as the TIDL Toolkit. In Linux, we can run our code using the open source framework called Mono [15], which is an open implementation of the Microsoft .NET framework.

Currently we are preparing an evaluation of the open classroom system that was designed according to ideas indicated above. Java Programming classes will be conducted applying different media environments and comparing students' preference for either standard software of interactive whiteboards, or the open classroom system prototype with access from tablet PCs. Accompanying research deals with didactic implications and potentials.

6. Acknowledgement

This research project is funded by NICT, the National Institute of Information and Communication Technology of Japan (http://www.nict.go.jp/index.html).

Literature

[1] Baloian, N., Berges, A., Buschmann, S., Gaßner, K., Hardings, J., Hoppe, H.U. & Luther, W. (2002). Document management in a computer integrated classroom. In Joerg M. Haake, & Jose A. Pino, (Eds.). Proceedings of CRIWG 2002, 8th International Workshop on Groupware, LNCS. Springer (pp. 35–46).

[2] Bier, B. & Freeman, S. (1991). MMM: A user interface architecture for shared editors on a single screen. Proceedings of the ACM Conference on User Interface and Software Technologies (UIST '91), Hilton Head, pp. 79-86.

[3] Breuer, H. & Baloian, N. (2005). Augmenting Whiteboard Interaction in the Classroom. In: P. Kommers, G. Richards. (Eds.). Proceedings of ED-Media 2005. World Conference on Educational Multimedia, Hypermedia and Telecommunications. Association for the Advancement of Computing in Education (AACE), pp.1214-1221. Montreal, Canada.

[4] Brotherton, J., & Abowd, G. (2004). Lessons Learned From eClass: Assessing Automated Capture and Access in the Classroom, ACM Transactions on Computer-Human Interaction (TOCHI), Volume 11, Issue 2 (pp. 121-155).

[5] Damm, C.H., Hansen, K.M., Thomsen, M. (2000). Tool Support for Cooperative Object-Oriented Design: Gesture Based modeling on an Electronic Whiteboard. Proceedings of CHI 2000, ACM Press (pp. 518-525).

[6] Hoppe, H. U., & Gassner, K. (2002). Integrating collaborative concept mapping tools with group memory and retrieval functions. Proceedings of CSCL-2002 (International Conference on Computer-Supported Collaborative Learning). Boulder, Colorado (USA).

[7] Hutterer, P., Close, B., Thomas, B. (2006). TIDL: mixed presence groupware support for legacy and custom applications. ACM International Conference Proceeding Series, Vol. 169. Proceedings of the 7th Australasian User interface conference - Volume 50 (http://wearables.unisa.edu.au/mpx)

[8] Igarashi, T. et al. (2000). An Architecture for Pen-based Interaction on Electronic Whiteboards, Proc. working conference on advanced visual interfaces. Palermo, Italy (pp. 68-75).

[9] Ishii, H., Kobayashi, M., & Grudin, J. (1993). Integration of interpersonal space and shared work-space: ClearBoard design and experiments. ACM Transactions on Information Systems (pp. 349-375).

[10] Kam, M., Wang, J., Iles, A., Tse, E., Chiu, J., Glaser, D., Tarshish, O., and Canny, J. (2005). Livenotes: a system for cooperative and augmented note-taking in lectures. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (Portland, Oregon, USA, April 02 - 07, 2005). CHI '05. ACM Press, New York, NY, 531-540.

[11] Llanza, L.. (2004). Human-Computer Interfaces for Electronic Board Supporting face-to-face Knowledge Exchange, GITS, Tokyo, Japan.

[12] Leisen, J., & Mentges, H. (2000). Handwerkskasten für Lehrer. Staatliches Studienseminar für das Lehramt an Gymnasien in Koblenz Koblenz (http://www.uni-koblenz.de/~odsleis/Handwerkskasten)

[13] Levy, P. (2002). Interactive Whiteboards in Teaching and Learning. Department of Information Studies, University of Sheffield, published by the Sheffield Excellence in Cities Partnership. (http://www.shefield.gov.uk/education/schools-area/services-to-schools/ excellence-in-cities/interactive-whiteboards-report)

[14] Mandl, H., & Reinmann-Rothmeier, G. (1998). Auf dem Weg zu einer neuen Kultur des Lehrens und Lernens. In Dörr G. and Jüngst, K. L. (Eds.). Lernen mit Medien. Juventa: Weinheim (pp.193-205).

[15] Mono Project (http://www.mono-project.com)

[16] Moran, T. P., Chiu, P., & van Melle, W. (1997). Pen-based interaction techniques for organizing material on an electronic whiteboard. Proceedings of UIST'97 (pp. 45-54).

[17] Netron Reloaded Project (http://sourceforge.net/projects/netron-reloaded)

[18] Ovaska, S., Hietala, P. & Kangassalo, M. Electronic whiteboard in kindergarten: opportunities and require-ments. Proc. of the 2003 conference on interaction design and children. ACM Press: New York (pp. 15-22).

[19] Owen, H. (1997). Open Space Technology: A User's Guide. (2nd Edition). Berrett Koehler Publishers: San Francisco.

[20] Pawar, U., Pal, J., Toyama, K. (2006). Multiple Mice for Computers in Education in Developing Countries. 1st International Conference on ICT and Development, UC Berkeley, May 2006.

[21] RawInputSharp Source Code (http://jstookey.com/ arcade/rawmouse/rawinputsharp.zip)

[22] Sousa, C. & Matsumoto, M. (2006). Application for Integrating Simultaneous Input Devices into Electronic Whiteboards. IEICE General Conference, Technologies and Architectures for Ubiquitous Network, Kokushikan University, Setagaya Campus, Japan, March 2006.

[23] Tse, E. & Greenberg, S. (2004). Rapidly Prototyping Single Display Groupware through the SDGToolkit. Proc Fifth Australasian User Interface Conference, Volume 28 in the CRPIT Conferences in Research and Practice in Information Technology Series, Australian Computer Society Inc., pp. 101ff

[24] Wallace, G., Bi, P., Li, K., Anshus, O. (2004). A MultiCursor X Window Manager Supporting Control Room Collaboration. Princeton University, Computer Science, Technical Report TR-707-04, July 2004.

[25] WinTab.NET Project (http://sourceforge.jp/projects/wintabdotnet)