MuSwikis: A Graphical Collaboration System

Lex Spoon and Mark Guzdial
College of Computing
Georgia Institute of Technology

Abstract

Many collaborative systems use text as the primary modality and only use multimedia via attachments. MuSwikis are a collaboration system where graphics are the primary modality, and text is treated just like any other graphical element. This paper describes the MuSwiki system for the first time, describes a pilot study that was run on MuSwikis, and discusses ideas generated from that study.

1 Introduction

To date, most collaborative learning systems have been oriented primarily towards text. Where graphics have appeared, they have served as an extension to text (as in most Web-based tools, such as CoNote [7] and CaMILE [6]) or as an anchor for mostly text notes (e.g., as in CSILE [8]).

The focus on text is not a bad design decision. Text really does work. Text can express a wide array of powerful ideas (just consider the content of many libraries– mostly text!), people are capable and comfortable using text to express themselves, and furthermore, text systems are relatively simple to implement. Text works well with the kinds of technologies that have been utilized in collaborative learning systems, mostly Telnet and Web browsers. Finally, if the goal is to capture discussion as it would be face-to-face, the words of the discussion may be enough to capture.

However, for some domains and tasks, text fundamentally cannot capture the kinds of conversations that participants would like to have. For example, design discussions in any domain frequently rely on diagrams and simplified prototypes. Participants create and modify the diagrams and prototypes, and then refer to them in discussion. Trying to converse on a design using only words would be much more difficult, and text is even less rich than spoken word.

Furthermore, even when a textual representation is adequate, a graphical representation can be more expressive. The plain text `(-b+-sqrt(b^2-4ac))/(2a)` is more verbose and less clear than the slightly graphical $\frac{-b \pm \sqrt{b^2-4ac}}{2a}$. The plain text version contains the same information, but the second version uses layout cues and is much easier to read.

Finally, in some domains, text is not the modality of choice for many discussions. Architecture discussions involve diagrams and gestures. Electrical engineering discus-

sions involve equations, graphs (such as waveforms), and even active computational elements such as simulations.

This paper describes MuSwiki, a collaboration system where the primary mode of communication is graphical and computational instead of textual. On a MuSwiki, text is still supported, but it is supported as just another graphical element. Users can drag-and-drop graphical elements into a collaborative space, or even live computational elements such as sound elements and simulation front ends.

The MuSwiki is based on the CoWeb, a collaborative space being used successfully in learning contexts [2] [3]. We see the MuSwiki as a CoWeb-like tool that is more appropriate for some domains (such as architecture and electrical engineering) and some tasks such as design. In this paper, we describe MuSwiki and the results of a small pilot study where computer science students engaged in a collaborative program design activity using the MuSwiki.

2 MuSwikis

A MuSwiki is a CSCL tool which is page-oriented, hyperlinked, peer-to-peer collaborative, and graphical.

First, MuSwikis are page-oriented. A single MuSwiki consists of a number of pages, each of which has a title. The central activity on a MuSwiki is reading a page that exists on the server; all other features are intended to make this activity more effective.

Second, MuSwiki pages are hyperlinked together. Pages and hyperlinking are the foundations of structure on a MuSwiki. Hyperlinks permit any structure to be built, but they do not constrain users to any predetermined structure. In particular, a user can begin a new page by adding a hyperlink from an existing page which has a similar topic; it isn't necessary to find the location the new page should fit into some larger structure before work on the new page even begins.

Third, MuSwikis are collaborative. Furthermore, all collaborators on a MuSwiki are peers–the teacher has no special status. All users are encouraged to modify pages, and the user interface is intended to make page modification very easy. This is a different approach from many existing learning systems, where extra privileges are embedded in the system for the teacher. On a MuSwiki, all relationships must be determined by social mechanisms and not by software.

Finally, as mentioned earlier, MuSwikis are primarily graphical. A MuSwiki page is built of "morphs", which are the fundamental graphical element in the Squeak programming environment (*http://www.squeak.org*).

Morphs are powerful tools. While there are morphs for static elements such as rectangles, circles, bitmaps, and text, there also morphs with continual computation such as clocks, molecule simulations, and keyboard synthesizers. The MuSwiki client and server are themselves morphs. Furthermore, programmers are able to add new morphs as each new class might find useful. For example, a CRC card morph was created just for the pilot study described in this paper.

Two example MuSwiki pages are shown in Figures 1 and 2.

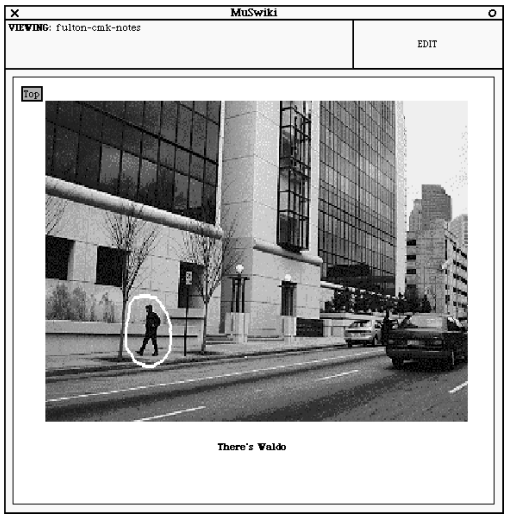
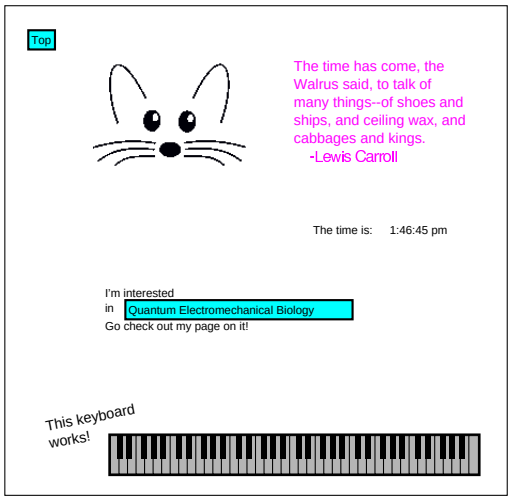


Figure 1: A sample MuSwiki page. Notice the hand-drawn circle used to annotate the page



2.1 Kansas

Kansas is similar enough to MuSwiki to warrant a comparison. Kansas was originally developed as a collaborative programming system, but it has also been used effectively as an educational tool. [4]

In Kansas, there is a persistent two-dimensional world that users view and interact with. That world is built up from the same kinds of morphs that MuSwiki pages have. Kansas is similar to MuSwiki, but there are significant differences.

First, users use a different mechanism to select a portion of the world to focus attention on. In MuSwikis, users choose to look at specific named pages. In Kansas, users position a user-specific viewport somewhere in the world, and their screen only shows things that appear inside that viewport. A "radar view" of the world is provided to make this more convenient and intuitive. Put another way, Kansas viewports are adjusted by scrolling them on continuous axes, while the MuSwiki analogue is selected via discrete hyperlinks and menus.

Second, Kansas is completely synchronous. Actions by one user are shown immediately on the screen of all others users. MuSwikis are fairly asynchronous: changes are broadcast only when a page author explicitly asks for them to be.

Third, Kansas includes a facility for audio and video communication between users whose viewports overlap. MuSwikis contain no such facility.

Because of these attributes, Kansas is better than MuSwikis for live communication and for demonstrations. However, the page metaphor and hyperlinks on a MuSwiki add structure which should make MuSwikis more appropriate for shared knowledge building and document construction. Furthermore, the video links in Kansas are probably only feasible for smaller groups of people; for larger groups on the order of 20-50 people, the number of video links would be impractical to display simultaneously.

3 Pilot Trial Design

MuSwikis were tested in a sophomore Object Oriented Modeling and Design class at Georgia Tech. The focus of the class is on learning to analyze problems in an object-oriented fashion, design a solution using a couple of graphical notations, and implement a solution in Squeak. The class has 81 students this quarter.

The test activity was to design object-oriented systems using CRC cards, a system developed by Kent Beck and Ward Cunningham. [5] CRC cards describe an object system at a high level of abstraction, and they are already taught to students in the class.

A morph to display CRC cards was created to support this activity. Figure 3) shows a typical such morph.

During the first week of the trial, there was an hour and a half activity where students posted a simple design on the MuSwiki which had already been described, and then posted a comment with a text morph on the design they had posted. The goal was to introduce students to MuSwikis and to get an initial impression of MuSwiki usability.

During the second week, groups of students were first given four days to design a

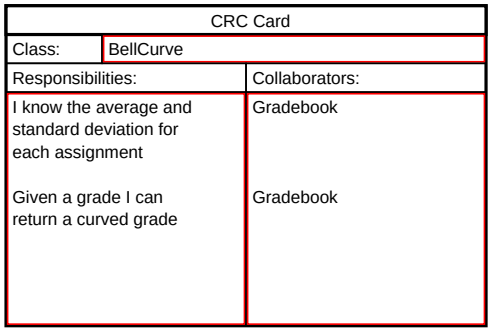


Figure 3: A CRC Card morph, used by students to do a software design

small object-oriented system using CRC card morphs in the MuSwiki. During the next four days, each student was asked to review and annotate at least two other students' designs, by posting text morphs on other students' designs. Students worked in groups of three on the designs, and they were required to post both their designs and their comments on the MuSwiki. Groups were randomly assigned, and thus, had very little time to coordinate activity and complete the task. To ease the collaboration, each student in the group had a pre-specified role (a portion of the design that she was responsible for). The collaborative portion of the design task, then, was to integrate the design pieces and create an effective final design.

Students were asked to complete surveys after the first and second weeks, on the usability system, the effectiveness of their collaboration, and their attitudes toward the system and the activity.

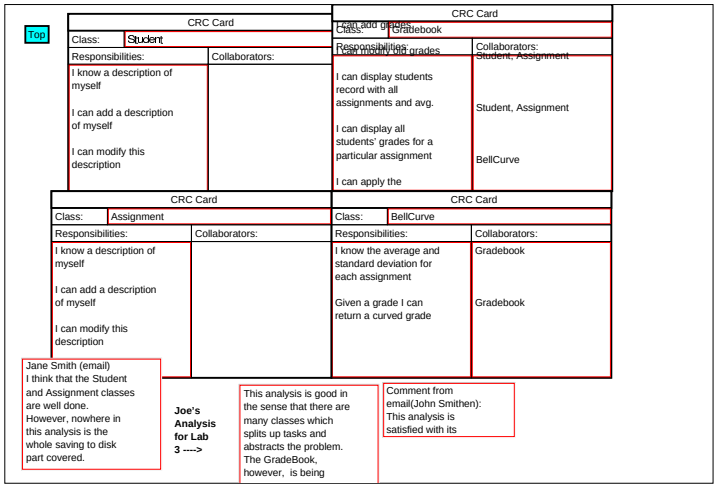
4 Results

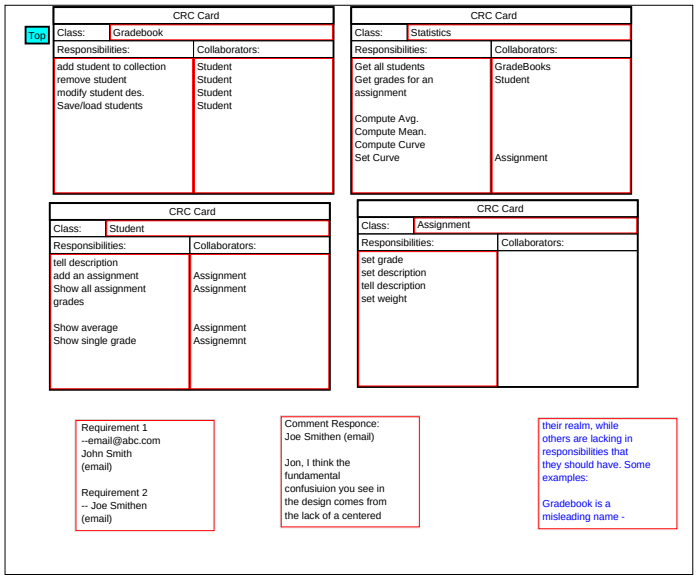
Data come from a number of sources: the final contents of the MuSwiki, the two surveys that were given out, study of the resulting MuSwiki and the logfiles, and review of the class USENET news discussion group.

4.1 Contents of the MuSwiki

A typical page created after the lab is in Figures 4 and 5. There is an analysis based on CRC cards, a listing of the project authors, a critique from a viewer, and a response by the original author.

In general, observation of the MuSwiki shows that students using the system were able to work out a design, to look at each other's designs, and to have discussions over their designs. Thus the system does work. The open question is whether the system is especially effective.


Top
CRC Card
Class: Student
Responsibilities:
I know a description of myself
I can add a description of myself
I can modify this description
Collaborators:
CRC Card
Class: Gradebook
Responsibilities:
I can add grades
I can modify old grades
I can display students record with all assignments and avg.
I can display all students' grades for a particular assignment
I can apply the
Collaborators:
Student, Assignment
Student, Assignment
BellCurve
CRC Card
Class: Assignment
Responsibilities:
I know a description of myself
I can add a description of myself
I can modify this description
Collaborators:
CRC Card
Class: BellCurve
Responsibilities:
I know the average and standard deviation for each assignment
Given a grade I can return a curved grade
Collaborators:
Gradebook
Gradebook
Jane Smith (email)
I think that the Student and Assignment classes are well done. However, nowhere in this analysis is the whole saving to disk part covered.
Joe's Analysis for Lab 3 ---->
This analysis is good in the sense that there are many classes which splits up tasks and abstracts the problem. The GradeBook, however, is being
Comment from email(John Smithen): This analysis is satisfied with its


Top
CRC Card
Class: Gradebook
Responsibilities:
add student to collection
remove student
modify student des.
Save/load students
Collaborators:
Student
Student
Student
Student
CRC Card
Class: Statistics
Responsibilities:
Get all students
Get grades for an assignment
Compute Avg.
Compute Mean.
Compute Curve
Set Curve
Collaborators:
GradeBooks
Student
Assignment
CRC Card
Class: Student
Responsibilities:
tell description
add an assignment
Show all assignment grades
Show average
Show single grade
Collaborators:
Assignment
Assignment
Assignment
Assignemnt
CRC Card
Class: Assignment
Responsibilities:
set grade
set description
tell description
set weight
Collaborators:
Requirement 1
--email@abc.com
John Smith
(email)
Requirement 2
-- Joe Smithen
(email)
Comment Responce:
Joe Smithen (email)
Jon, I think the fundamental confusiuion you see in the design comes from the lack of a centered
their realm, while others are lacking in responsibilities that they should have. Some examples:
Gradebook is a misleading name -

4.2 Initial Survey

After the first lab, students were given a questionnaire which focussed on usability. Some key responses are summarized in Table 1.

Table 1: Usability Survey Responses. The responses are on a scale from 1 to 5, where 5 is the strongest agreement

Proposition	Average	Std. Dev.
It is easy to log into the class MuSwiki.	2.32	1.25
It is easy to identify hyperlinks on a MuSwiki page.	3.24	1.04
It is easy to follow a hyperlink from one MuSwiki page to another.	4.05	0.63
It is easy to find a page on a MuSwiki, if you know the page's name.	3.19	1.13
It is easier to change a MuSwiki page than a CoWeb page.	2.91	1.35
In general, it is easy to modify a MuSwiki page.	3.18	1.28
The process of editing and saving a MuSwiki page is presented in a clear and unconfusing way.	3.05	1.21
The MuSwiki would be better than Web pages for displaying information.	2.23	1.20
The CoWeb is a formal, professional space, while the MuSwiki is an informal, less structured space.	3.09	0.99

4.3 Final Survey

A survey was given to the students after they finished their group designs. Key responses are summarized in Table 2.

4.4 System Logs

The MuSwiki server maintained a log of each login, page access, and page modification. Some interesting things may be learned from perusing that log.

First, the importance of careful instructions is brought out. Several accesses to the system were by users under the alias *lex*, which was the sample alias given in the MuSwiki tutorial. Most likely, the students who used this alias did not realize that it was just an example.

This issue clouds another that the pilot raised: do students view this graphical environment as more informal? Initially, it seemed that they clearly do. Five students modified the front page of the MuSwiki on the first day it was in use; each one changed it an average of three separate times. The front page of a MuSwiki is akin to the home page of a web site: it sets the tone for the site and gives the overall organization. Students probably recognized this, and yet still seemed to feel comfortable changing such an encompassing part of the system. Alone, this would be a clear indication that

Table 2: Responses to Final Survey. The responses are on a scale from 1 to 5, where 5 is the strongest possible agreement.

Proposition	Average	Std. Dev.
My group's MuSwiki page expresses our design thoroughly and accurately.	3.68	0.67
We had a hard time getting our design expressed right in the MuSwiki.	3.05	1.31
In the annotation I made, I was able to fully express my thoughts.	3.11	1.05
The annotations pasted on my group's page were useful to us.	3.11	0.9
The annotations pasted on my group's page were interesting to us.	2.67	1.03
I didn't really look at the annotations pasted on my group's page	2.82	1.38

students treat the MuSwiki fairly informally. However, considering that this happened more frequently the first day than it did later, some unknown number of the front page modifications were probably due to student confusion.

Finally, the logs show that most project pages went through 10 to 20 editions in their development. Thus, students posted several intermediate versions of their pages, instead of posting just their part of the design.

5 Discussion, Conclusions

5.1 Usability

Observation of the final system showed the system to be usable: most groups did successfully finish their designs. However, on the surveys students were widely divided on their usability responses. It seems that MuSwikis are usable, but that prolonged use seems to lower users' opinions for some reason.

5.2 Learning

Students were able to work in small groups to develop and agree on an an object-oriented analysis for the given problem. They did not do this *solely* using the MuSwiki to communicate, but developing the analysis directly on the MuSwiki proved possible. Thus, the use of MuSwikis did not strongly hinder the collaborative analysis process.

Furthermore, this MuSwiki-based project allowed for a mechanism that is normally difficult to achieve in classes: students had reasoned discussions *between* groups instead of just among the members of individual groups. These discussions probably came about because it is easy and fast to do all of the following needed activities: view-

ing projects of other groups, sending critiques to other groups, viewing critiques made by others, and responding to critiques.

5.3 Student Attitudes toward the Space

Data suggests that students feel less constrained on MuSwikis than they do on CoWebs. For some reason, students seem more likely to "scribble" on a MuSwiki page than they are on a CoWeb page. It may be that students already understand some norms of practice for the Web, where reading is much more common than writing, and they don't carry those norms to a MuSwiki. Or perhaps the ease of editing on a MuSwiki as compared to a Web page makes this difference.

In any case, this is an attitude worth investigating. In many attempts at collaboration in an educational setting, the largest problem is getting students to actively take part. A more informal environment should help.

5.4 Iterative Development

It is unclear how much students collaborated on their designs, but it is clear that the tool worked as expected. Most design pages were submitted between 10 and 20 times, indicating that students did indeed make small changes at a time and then save and broadcast them. The fact that incomplete versions of the project existed on the server, means that other members of a group were able to view and comment on the incomplete designs. This is an improvement over the typical case where collaboration tools aren't involved, and students in a group only see partial results at the two or three physical meetings they schedule.

6 Future Work

6.1 Outside Channels of Communication

Most groups reported exchanging at least a few email messages, and most groups met face to face at least once. This was not strictly necessary for the assignment: nothing would prevent groups from collaborating solely through the MuSwiki, posting intermediate results to be worked on by future members.

This suggests that the MuSwiki alone is not a sufficient collaboration mechanism. This really isn't too surprising. Much of collaboration does involve discussion. By emphasizing graphics and de-emphasizing text, we did make discussion more difficult.

We plan to add, at the least, a text-based synchronous communication mechanism to future incarnations of the MuSwiki. People should be able to chat with others visiting the same page, and they should probably be able to chat with arbitrary individuals who are logged into the server. This mechanism might take the form of a simple textual window as is used in IRC, AOL Instant Messenger, and ICQ. On the other hand, it might take advantage of Squeak's digital audio abilities and allow for full verbal communication.

6.2 Space Management

Students' pages became crowded quickly, even though the trial activity was relatively simple. A screen-sized page just isn't always large enough. In the final surveys, students mentioned a desire for scrolling capabilities. Furthermore, several of the comments posted on the MuSwiki made oblique reference to the difficulty of finding a space to actually place the comment.

Future versions of MuSwiki will in some way support pages larger than the screen. The simplest mechanism would be to make each page a Kansas-like world, where users have individual rectangular viewports. Also intriguing, however, is the idea of allowing different users to zoom in to varying degrees, as was studied in Pad++. [13] With such a mechanism, it would still be possible to get an overview of the entire page before delving into the details.

6.3 Active Content

MuSwikis form an ideal testbed to experiment with Ted Kaehler's *active essays*. [14] An active essay is a document which contains not only static text and graphics, but which also includes computational elements. Computational elements, used carefully, should be able to make a presentation much more lucid than static elements alone. Making the use of active content practical, especially for people who are not expert programmers, is an interesting area of research.

7 Acknowledgements

Thanks to the students and faculty involved in these studies. Funding for some of this work is from National Science Foundation, grant REC-9814770.

References

[1] Guzdial, M., *Collaborative websites to support an authoring community on the Web*. Journal of the Learning Sciences, 1999. **Submitted**.

[2] Guzdial, M., *Supporting Learners as Users*. The Journal of Computer Documentation, 1999. **23**(2): p. 3-13.

[3] Guzdial, M., *Teacher and Student Authoring on the Web for Shifting Agency*. 1999, Presented at the Annual Meeting of the American Educational Research Association as part of the session "How can CSCL (Computer Supported Collaborative Learning) change classroom culture and patterns of interaction among participants?"

[4] Scanlon, Eileen, Tim O'Shea, Randall B. Smith, and Yibing Li. *Supporting the Distributed Synchronous Learning of Probability: learning from an experiment*, in *Proceedings of Computer-Supported Collaborative Learning197*, R. Hall, N. Miyake, and N. Enyedy, Editors. 1997, Toronto, Ontario, Canada. p. 224-230.

[5] Kent Beck, Ward Cunningham. *A Laboratory for Teaching Object-Oriented Thinking*, in *Proccedings of OOPSLA 1998, ACM*.

[6] Hmelo, C. E., M. Guzdial, and J. Turns, *Computer-support for collaborative learning: Learning to Support Student Engagement*. Journal of Interactive Learning Research, 1998. **9**(2): p. 107-130.

[7] Davis, J. R. and D. P. Huttenlocher, *Shared Annotation for Cooperative Learning*, in *CSCL'95 Proceedings*, J. L. Schnase and E. L. Cunnius, Editors. 1995, Lawrence Erlbaum and Associates: Bloomington, IN. p. 84-88.

[8] Scardamalia, M., C. Bereiter, and M. Lamon, *The CSILE Project: Trying to bring the classroom into World 3*, in *Classroom Lessons: Integrating Cognitive Theory and Classroom Practice*, K. McGilly, Editor. 1994, MIT Press: Cambridge, Mass. p. 201-228.

[9] Guzdial, M., *Information ecology of collaborations in educational settings: Influence of tool*, in *Proceedings of Computer-Supported Collaborative Learning'97*, R. Hall, N. Miyake, and N. Enyedy, Editors. 1997, Toronto, Ontario, Canada. p. 83-90.

http://guzdial.cc.gatech.edu/papers/infoecol/

[10] Guzdial, M., et al., *Integrating and Guiding Collaboration: Lessons learned in computer-supported collaboration learning research at Georgia Tech*, in *Proceedings of Computer-Supported Collaborative Learning'97*, R. Hall, N. Miyake, and N. Enyedy, Editors. 1997, Toronto, Ontario, CANADA. p. 91-100.

[11] Guzdial, M., *Making project-based learning work in undergraduate educational support: Lessons in computer-supported collaborative learning*, in *CALISCE'98: 4th International Conference on Computer Aided Learning and Instruction in Science and Engineering Proceedings*, C. Alvegrd, Editor. 1998, Chalmers University of Technology: Gteborg, Sweden. p. 3-8.

[12] Guzdial, M., C. Kehoe, and J. Turns, *What We Know About Technological Support for Project-Based Learning*, in *Proceedings of the Frontiers in Education Conference*. 1997, IEEE: Pittsburgh, PA. p. 918-922.

[13] Bederson, B.B., Stead, L., Hollan, J.D. *Pad++: Advances in Multiscale Interfaces*. *Proceedings of ACM SIGCHI '94*.

[14] Homepage of Ted Kaehler.
http://www.webpage.com/~kaehler2/