Chapter 16

Designing a Mix of Synchronous and Asynchronous Media for VMT

Gerry Stahl

Gerry@GerryStahl.net

Abstract: The challenges of designing computer support for education have shifted considerably in recent years, with, e.g., the rapid growth of the Web, online learning and social networking. New human-computer interaction (HCI) design approaches, methods, tools and theories are now required to analyze and understand interactions and learning of online groups. This chapter first reviews a number of issues related to the new software and pedagogy challenges. It then presents the approach of the VMT research project to address these issues by combining support for integrated synchronous and asynchronous collaboration media. The VMT system integrates a lobby, small-group chat rooms, multiple shared work spaces and community wiki pages to foster learning at the individual, small-group and community levels. The use of this system for a college HCI course is reported. The VMT Project illustrates the application of design-based research to system development, the theory of group cognition as a conceptual framework and an adaptation of chat interaction analysis for HCI design.

Keywords: Educational software, human-computer interaction, user-centered design, design-based research

The Potential of Computer Support for Education

Shifting the Design Perspective on Educational Software

This chapter tries to shift the terms of debate within software design from "human-computer interaction" (HCI) to the more specific topic of *human-human* interaction and *group* learning. This is not to imply that other aspects of the broader theme of HCI are unimportant, just that the focus on group learning is one that has been largely overlooked, much to the disadvantage of the whole field of software design.

Human-computer interaction as a field has historically been oriented predominantly toward the relationship between the individual computer user and the interface of computer software. Classic HCI studies investigated the effects of different designs of desktop software upon individuals using the software. The theory of HCI was, accordingly, closely aligned with the science of individual psychology. For reasons to be discussed in this chapter, we will instead look at human-human (rather than human-computer) interaction that is mediated by computer software and by the networking of computers. The software is here seen largely as a technological communication medium, which both supports and constrains interaction among small groups of users. More precisely, the concern here is with the small-group interaction itself, that is, the group processes, rather than the interaction of one individual as such with other individuals in the group. Conceptually and methodologically, this involves a shift from the psychology of mental processes of individuals to the largely linguistic interactions of small groups.

The proposed shift is from the education of individual minds to learning within groups. The issue changes from tracing effects on students of the transfer of factual knowledge from authorized sources (teachers, textbooks, drill software) to understanding how groups learn. This new focus is sometimes termed *collaborative learning*, which includes both how groups increase knowledge and how the individuals within the groups learn concomitantly. The term *knowledge building* is perhaps preferable to either "education" or "learning." This is partially because the terms "education" and "learning" tend to be closely associated with traditional institutions of schooling and with psychological theories of individual minds. It is also due to the fact that one can observe the building of knowledge in products of group work, such as theories and documents; knowledge building can more easily be operationalized and studied. This perspective also opens new opportunities for teaching HCI, as should become evident latter in this chapter.

The History of Computer Support for Learning

Starting even before personal computers were developed and long before they were networked across the Web, a variety of educational applications of computers were proposed and to a lesser extent disseminated. In a review of instructional technology, Koschmann (1996) identified four broad approaches for incorporating computers in educational practices, namely Computer Aided Instruction (CAI) starting in the 1960s, Intelligent Tutoring Systems (ITS) in the 1970s, Constructive Learning Environments (Logo-as-Latin) in the 1980s and Computer-Supported Collaborative Learning (CSCL) from 1995 on.

These four design paradigms were largely inspired by technological possibilities. Even in the 1960s, mainframe timesharing computers with many terminals were able to present texts to people sitting at the terminals, pose multiple-choice questions and respond based on the choice entered at their terminal. CAI applications were designed to take advantage of this automation mode. Later, tutoring systems took this a step further with a more sophisticated back-end using an AI approach to model both the domain structure (e.g., typical solution paths for a well-defined math problem) and a mental model of the student's domain knowledge (i.e., how the student was approaching the problem solution). More exploratory learning environments took advantage of subsequent 2-D graphics support and personal computer facilities for end-user programming. Finally, CSCL responded to the networking of personal computers and the spread of the Web. Each approach raised new HCI issues—or suffered from a lack of HCI analysis. The four approaches have all had limited successes and are still active in the instructional technology marketplace.

Each of the four approaches has simultaneously offered tools for classroom education and threatened the institutions of schooling. They all allow people to learn outside of school. Some have been particularly popular for home schooling and for after-school programs, as well as for industrial training workshops.

In terms of the focus of this chapter, it is important to distinguish CAI, tutoring and constructivist environments as software for individual usage *versus* CSCL as inherently for small-group usage. While the first CSCL system—CSILE (Scardamalia & Bereiter, 1996), or Knowledge Forum—has been used in classrooms around the world for a decade, most other CSCL systems are still in the research prototype stage.

The New Perspective of the Learning Sciences

Leading, or at least paralleling, the changing paradigms for learning technologies was the evolution of theories in the learning sciences (Sawyer, 2006). Moving away from the traditional educational theories of Thorndike (1914), they recreated many of the ideas of Dewey (1938/1991), supporting them with the developmental and social psychology theories of Piaget (1990) and Vygotsky (1930/1978). In particular, they increasingly recognized the socio-cultural situatedness of learning in communities-of-practice (Lave & Wenger, 1991). In this, they followed much the same path as the situated-cognition critique within AI and computer science (Winograd & Flores, 1986).

Perhaps the most important influence on the learning sciences for the focus of this chapter was the reception of Vygotsky's theory of social mediation, in particular his principle of internalization. This says that most higher functions of human thought are first learned socially, as part of interactions among people; they can later be internalized and transformed into individual mental skills (Vygotsky, 1930/1978, pp.

52-57). This principle is associated with his concept of the zone of proximal development, in which a learner can engage socially in collaborative work on a task that they would not yet have been able to accomplish on their own internally.

Vygotsky's theories—although not fully worked out in his brief lifetime—emphasize the importance of small-group interaction to the construction of meaning, representations, tools, symbolic artifacts and knowledge resources—both for the culture and for the individual. The implications of this theory have yet to be taken into account by the aims, procedures and institutions of contemporary schooling.

The Trouble with Computers in the Schools

The primary problem with how schools have adopted computers is their technology-driven view of the social role of computers. Under pressure to do something to improve schooling and to make it seem more up-to-date, politicians, administrators and parents have pushed to equip schools with computer hardware and Internet access. Of course, these are necessary, but not at all sufficient. A major problem is the lack of adequate educational software. In addition, there are needs for providing teacher training and on-site technical support. The hardware is often set up with little provision for meaningful computer-based curriculum and associated infrastructure. HCI was born to address the trouble with computers in industry. Now schools face an analogous—and overlapping—problem.

HCI was able to improve the lot of industrial software by increasing its reliability, usefulness and usability by insisting on a human-centered approach to design (Landauer, 1996). As we shall see, the problem is more complex for educational applications, involving the adoption in practice of the new learning sciences theories.

A Typical Government Study

A recent Congressional study (EETI, 2007) looked at software for reading education and for math education. Let us focus on the math software because that is a main example in this chapter. Three unnamed math applications were tested. According to the standards of the testing, the classroom use of these three applications had no significant effects on learning outcomes. From the characteristics given of the applications, it sounds like they were all examples of the CAI paradigm of drill-and-practice by isolated individual students.

There are many legitimate educational goals for which one might enlist instructional technology. As already pointed out, there are completely different approaches taken by educational software, with many different exemplars in each category. That certain software based on a 50-year-old approach may not inspire millennial children under certain conditions does not mean that software cannot be developed to be effective for educational purposes. Even CAI has its benefits for certain people trying to achieve specific goals.

The first problem in designing and assessing educational software is that one really needs to invent new and innovative approaches, based on current theories of the learning sciences. These are hard to test because one needs to develop prototypes that are robust enough to use in real classrooms over long enough periods that teachers and students can become familiar with them. Furthermore, they may require new kinds of assessments, different from those appropriate for CAI applications.

The Dim Future of the Physical University?

A critical essay in *Science* (Noam, 1995) argued against the use of computers in college education, saying that online education destroys much of the value of traditional university life. Primarily, however, the author conceived of educational software as something to be used by isolated individuals. He then saw that an education based on interaction with a computer would be missing the socializing aspects of, e.g., an undergraduate on-campus experience. However, he never considered that software can promote social contact, as can be seen not only in educational applications that incorporate discussion forums, chat, IM, wikis, websites, etc., but even more in the recent phenomena of social-networking software. Social networking, interestingly enough, is particularly popular among college undergrads. In response to Noam, one might inquire how social networking could be integrated into educational technology so that online learning would be a positive social experience, rather than an isolating, alienating one.

Computer-Supported Collaborative Learning

This is, of course, where CSCL steps in. The research field of CSCL—with its conferences, journal, book series, workshops, projects and labs—is devoted to developing ways to harness computer technology to support the rich social dimension of learning through collaboration.

The computational power of computers has the potential to provide many kinds of tools to extend human capabilities and to transform routine or complex intellectual tasks into tasks that are more interesting or feasible. With its graphic capabilities, the computer can run simulations of scientific or mathematical models and allow groups of students to explore them. With global networking, computers can put students in touch with their peers around the world, to learn each other's language and culture or to work and socialize together. The ability of computers to interact based on programmed instructions allows them to guide students through arbitrarily intricate and adaptable sequences (or scripts) of group and individual activities.

CSCL takes many approaches to mixing these potential benefits of computerization. The CSILE software was designed to allow a classroom full of students to collaboratively build scientific knowledge and theories asynchronously over periods of several weeks (Scardamalia & Bereiter, 1991). Argumentation software typically helps dyads of students to reflect on the structure of their debates and organize the logic of their thinking and persuasion (Andriessen, Baker & Suthers, 2003). The VMT software is designed for groups of 2 to 10 to discuss mathematics in real time.

Whatever the techniques, media and domain, CSCL software is intended to foster collaborative learning and knowledge building by a group. Individuals may learn by participating, and perhaps by internalizing the experience, as Vygotsky described.

The Problem of User-Centered Groupware Design

Software for collaborative learning—like that for workplace learning and community learning—is associated with significant HCI issues, that exceed the difficulties of single-user desktop-interface and web-page design. They call for new theories, assessment tools and principles. They must centrally take into account the interactions among group participants as mediated by the software medium, and not just the interaction of an individual user to an interface. The number of possible combinations of views of the software by different participants at any given time and the variety of interactions possible explodes, making HCI analysis techniques from the 1980s inadequate. Many technical problems and many potential uses of the software are unpredictable and have to emerge from actual usage by groups of people under naturalistic conditions. This limits the utility of scenarios, mockups, walkthroughs, prototypes and lab studies as assessment tools—as essential as they may still be to specific phases of the design process.

Social Networking and Web 2.0

Despite the difficulties facing the development of effective collaborative learning technology, the potential benefits loom larger than ever. The recent increase in Internet usage, particularly by high school and college students, bodes well for the adoption of new educational technologies. In particular, the popularity of a range of social networking sites and of so-called Web 2.0 interactive technologies has already instilled a familiarity with computer-supported collaboration, its handiness and its benefits.

Designing Support for a Virtual Learning Community

Use-Centered Research

The VMT Project began by building on a successful service at the Math Forum called Problem-of-the-Week. In the original service, an interesting challenge problem in pre-algebra, algebra or geometry was posted on the Web weekly and students worked on it at home, in school or during math club. Students could submit their solutions and their analyses to get feedback. The best solution statements were posted in the Web archives.

This service had evolved over a dozen years, guided by staff and teachers who had been involved with it from the beginning. The Math Forum itself emerged out of the experience of supporting this service, by adding related services for students, teachers and mathematicians, eventually serving millions of online users. As a digital library with over a million web pages, the Math Forum site grew by archiving user problem solutions, answers to user inquiries and discussions of user groups, such as teachers—anticipating the Web 2.0 philosophy of users as contributors by more than a decade.

Presumably, most of the math problem solving in the Problem-of-the-Week was done individually. The VMT Project set out to make that a collaborative process. We took advantage of the huge popularity of text chat. We initially adopted AOL's instant messaging tool, which was already quite familiar and accessible to students. Students who came to our site were placed into small groups in an AIM chat room and given a math problem to explore. If they wanted to exchange a drawing, they could email it to us and we would post it where the group could view it.

By starting with software and procedures that were already proven in use and were familiar to the students, we finessed the design start-up issues that can bog down groupware development efforts. We were able to quickly observe students "in the wild" doing math collaboratively. By starting simply, we could allow our development process to be driven by observation of actual usage.

We had previously tried to do a face-to-face trial in a Philadelphia public school to get a feel for how collaborative math works in that kind of setting. Although informative, that effort showed how unusual collaboration in school math is and how complex it is to analyze. By contrast, our chat logs immediately revealed that students could quickly adapt to online collaborative math problem solving and that we could observe much of interest about how they accomplished that (see Chapter 9).

A Design-Based Research Process

We adapted the kind of design-based research process (Design-Based Research Collective, 2003) which has been broadly adopted in the learning sciences. This is an iterative inquiry process in which we modify the software environment, the kinds of math problems and the pedagogical script a couple of times a year. We invite students to participate in online groups in the new environments, and then we analyze the logs of their interactions to determine what was good in the service design and where improvement was needed. We thereby gradually build an understanding of chat-mediated interaction and online collaborative math problem solving.

In terms of the technology, we tried a number of commercial and open source environments, combining chat with a shared whiteboard drawing space for geometric figures. Eventually, we contracted with a research lab in Germany (Fraunhofer-IPSI) to modify their ConcertChat software for our needs. We also began to develop a portal front-end to support social networking.

The kinds of math problems evolved considerably. From well-defined challenge problems, we moved toward mathematical mini-worlds for exploration and encouraged groups to define their own math questions to investigate. Over the years, we have gathered a corpus of 1,000 student-hours of interaction logs. We developed a Replayer tool that allows us to recreate the full interaction and review it in detail.

Perhaps the most important development was at the theoretical and methodological level. We gradually developed a theory of group cognition and a methodology of chat interaction analysis, as discussed toward the end of this chapter. This resulted in about a hundred publications reporting findings of the VMT Project and analyzing it, many of them incorporated in this volume.

Supporting Joint Problem Spaces

It became increasingly clear from our analyses and from the related CSCL literature that for our students "collaboration is a coordinated, synchronous activity that is the result of a continued attempt to construct and maintain a shared conception of a problem" (Roschelle & Teasley, 1995, p. 70).

Different technologies can provide different kinds of support for the construction and maintenance of shared conceptions. For instance, chat, whiteboard and wiki have different forms of persistence for inscriptions. Student groups are very sensitive to these differences and exploit them in subtle and inventive ways. Designers cannot predict many ways that these spaces will be used without observing actual groups of interacting students trying to work out their tasks situated within specific environments.

A major issue for groups working in environments with multiple workspaces (e.g., lobby, chat stream, shared whiteboard and wiki) is how to coordinate communications in the spaces and how to shift group attention from one space to the other. Special tools can help with this. When we adopted ConcertChat, it included a referencing tool that could point from the chat to the whiteboard. We observed the power of this tool for supporting the equivalent of pointing gestures and deictic references in the disembodied online context (Chapter 7). We subsequently added wiki spaces and multiple tabs to the whiteboard, facilitating collaborative Web browsing, wiki editing, help access and viewing of the math task. Combining these spaces with the social-navigation portal and its various tools, the VMT environment has come a long way from its AOL IM starting point.

In order to orient students to the current, complex environment, we have had to develop training and help facilities as well as sometimes involving the students' teachers in providing basic training. We have also found that it is effective to engage the same groups of students across multiple sessions, making planning more complicated and fragile. Having sequences of multiple sessions brings enormous learning benefits. Not only do the students become more familiar with the affordances of the environment, but they are able to explore the mathematics more deeply and reflectively. We are able to script the sessions to gradually build understanding. We can also take advantage of the intervals between sessions to provide feedback and suggestions without interfering with the delicate group interactions.

In order to deepen and broaden our user-centered research experience, we tried out the expanded VMT environment outside of the realm of K-12 math. The next section discusses our use of the technology in a graduate-level university course.

Reflexively, the course was about designing the VMT system, and encouraged the course students to use, analyze and re-design the technology.

Using the VMT Environment to Teach HCI

In Spring Quarter 2007, while we were completing our latest major software upgrade to VMT, we decided to try basing an online HCI masters-level course at Drexel on the VMT system. In our research, we were working on integrating wikis into VMT, so we decided to move the course home and the student-group websites into a wiki—away from Blackboard and HTML websites.

This gave us an opportunity to try out the new VMT lobby/chat/tabs/wiki environment in a context where we could define the course and guide the students first-hand. It turned out to be surprisingly easy to set up the entire course in a wiki, with clear instructions for the students and a clean organization of resources. Each week, the students held several online meetings with their workgroups in VMT chat rooms, where they discussed the readings and their design project assignments. They summarized their discussions in their shared whiteboards and then posted their summaries to the course wiki. We provided general feedback and guidance in the wiki as well.

The students read the whole of a newly-revised and comprehensive HCI textbook (Preece, Rogers & Sharp, 2007) as well as 18 research papers about CSCL and VMT. The textbook provided a thorough overview of the field and related background information. The papers served in place of lectures. Students maintained individual journals on the textbook chapters and reflected collaboratively on the papers. Each group posted its critiques of the papers in the wiki, where the other groups could read them and the instructor could comment on them.

The heart of the course was a group project, spanning most of the quarter, with weekly milestones requiring postings to the wiki by each group. The groups met several times a week at their convenience in VMT chat rooms to work on their group project and to discuss the readings for the week. As they discussed, they summarized their ideas on the whiteboard for posting to the wiki. That way, the whole group could draft the postings and if anyone missed a meeting they could catch up quickly without going through a long chat log.

The group project was to design an extension to the VMT software that they were using. The extension was supposed to support social networking, so that potential users of the VMT system could find others with similar skills, interests and availabilities to form groups.

The ten-week hands-on project was divided into weekly assignments, which paralleled the stages of the textbook's design model and matched the chapters and papers read the previous week:

1. An ice-breaker design project to help the students get used to working together in the environment.
2. Literature search on social networking and Web 2.0.
3. Analysis and statement of problems in social networking.

4. Establish requirements with use cases and scenarios.
5. Conceptual design (this was done individually by the students).
6. Interactive prototype and scenario.
7. Heuristic evaluation of another team's prototype.
8. Cognitive walkthrough of one's own team's prototype following a scenario.
9. Final, revised design for a new social network function in VMT.
10. In the final week, individual students submit their textbook journals and a reflection paper on their experience learning about HCI in the course.

Classroom learning is contextualized within a global horizon by situating the knowledge built by the groups within current HCI research issues. These are explicitly discussed as the student groups design and prototype solutions that apply the HCI concepts in the readings. The issues emerge mainly in the collaborative chat interactions: practice and group discussion inform each other.

The idea of collaborative peer learning through hands-on practice—which is fundamental to the course approach—is presented to the students through the syllabus document and some of the readings. The grading system stated in the syllabus shows that collaborative learning is a combination of efforts at the individual, small-group and classroom level: the grade is based on a combination of these. The assignments mix individual and small-group efforts, and the results are mostly shared at the class level.

By having the students work in the environment that they are designing for, they acquire first-person experience from a user perspective. Comprehensive histories of the interactions within the system are persistently available, so the students as designers can study their own usage of the system reflectively and analytically. It is thereby natural for the students to compare their subjective and objective analyses of the user experience. The collaborative structure of the course stimulates, encourages and supports discussion of issues of HCI and education.

The collaborative learning approach of the course is in many ways at odds with the culture at Drexel, which is traditionally an engineering school. Yet, as evidenced by the reflection papers, the students learned to appreciate the many aspects of collaborative learning in the course. Perhaps because they were mature students who knew that the work-world is increasingly organized into collaborative teams, they could understand the advantages better than undergraduates. Perhaps because they were accustomed to taking online courses in which there is no social contact, these students enjoyed the interaction with their peers.

Similarly to the schools in the EETI study discussed above, Drexel has long been committed to the visible hardware aspects of a twenty-first century education, but has not as thoroughly recognized the shifts in pedagogy that should go along with this. As the learning sciences have concluded, it is important to involve students in active, authentic, hands-on collaborative-learning experiences. Students need to take responsibility for their own learning and that of their peers. Only this way will they be prepared for the life-long learning that they will be involved in after graduation in rapidly changing high-tech fields.

Drexel was originally founded to provide educational opportunities for working-class people. One way this mission is met today is to offer online classes for people who are working fulltime. A majority of the Information School's students are now online graduate students. They typically work during the day and often have substantial family responsibilities. Many of the students in HCI courses work in computer fields and have first-hand experience with HCI issues at their work. This is a great advantage in a course, particularly when it is run collaboratively, so the experienced students can share their expertise and perspectives.

As Noam (1995) argued, colleges must redefine the benefits they offer in the contemporary educational marketplace. To some extent, this will depend upon local specifics. Perhaps a more general way colleges can promote their advantages is to emphasize social experiences through collaborative learning and other human-human interaction—including online. This applies, of course to curricula in HCI as well as to other disciplines.

The challenge is that current software support for online collaborative learning is primitive at best. There is a tremendous need for HCI work to help develop effective collaborative learning software. The help is needed at a deep level, not just superficial changes to the look-and-feel of the interface. The nature of computer-mediated human-human interaction must be understood and new media and functionality must be designed to support it.

Integrating Asynchronous and Synchronous Media in VMT

This section will describe the combination of asynchronous and synchronous media in the version of the VMT environment that was used in the HCI course as well as in the VMT Spring Fest 2007. The technological integration of the lobby, chat room and wiki should be understood as a pedagogical integration of learning at the individual, small-group and community levels.

Figure 16-1 shows an image of the VMT social networking portal in its current state. On the left are tools for defining and viewing personal profiles—in general, students in a VMT group have no knowledge about each other except for what is revealed in the chat interaction. With the functionality available in the VMT Lobby, they can define their own profiles and view profiles of each other, as well as send messages to individuals or groups in their communities. Communities are defined for various VMT constituencies, such as participants in a given Spring Fest or in a given course. There is also support for defining buddies, listing favorite chat rooms, etc. On the right is an interface for searching and browsing available chat rooms, usually listed for a given community. For the HCI course, each group met in a different chat room each week, to avoid overcrowding of the chat log and the whiteboard.

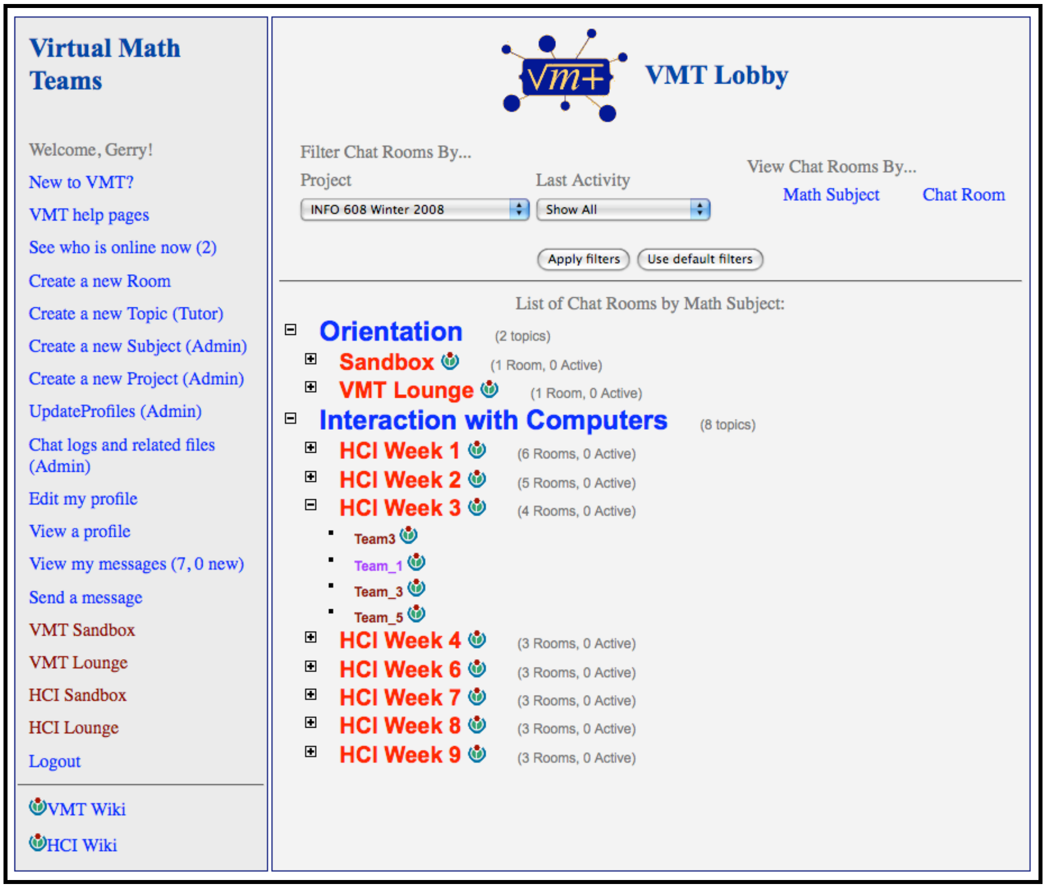


Figure 16-1. The VMT Lobby.

Figure 16-2 shows a typical chat room, consisting of the text chat interface on the right and the shared whiteboard on the left. Note that the user who is typing is currently pointing to a translucent rectangle selecting part of the whiteboard, as is a highlighted previous chat posting. The history of the whiteboard state can be scrolled through, much like that of the chat, but unlike the chat it usually retains inscriptions in the visible board as long as they are relevant. Here, in the HCI course, the whiteboard is being used to collect and organize design issues for subsequent posting to the course wiki as part of the group's weekly report.

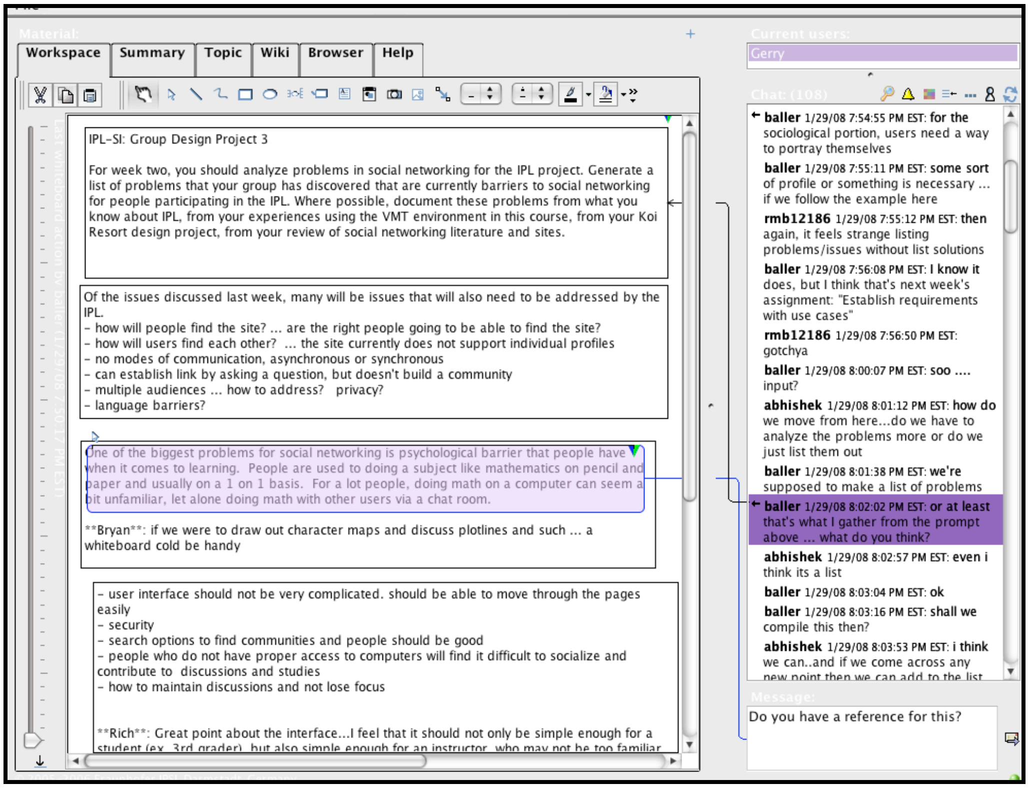


Figure 16-2. The VMT tabbed workspace.

The workspace on the left has a tabbed interface, with six default workspaces—users can add additional spaces. The first is the old shared whiteboard, supporting graphics and text boxes. The second is a similar shared whiteboard, intended for preparing a summary of the week's work for automatic posting to a special wiki page associated with this chat room. The third tab displays the topic for the course that week, stored on a wiki page by the instructor. The "wiki" tab displays a web page, using the user's default browser software. This tab initially points to the group's wiki page for their week's report. The "browser" tab uses a simplified web browser that can support the graphical referencing tool from the chat and a history scrollbar. The final tab displays wiki pages containing the VMT help manual and associated information.

Figure 16-3 shows the wiki home page for the HCI course. It points to pages describing the course and each assignment. Group assignments are all posted to linked wiki pages. The course wiki includes index pages that bring together the student assignments in various combinations and allow the instructor to post feedback that is visible to all. The student groups also rate and provide feedback to each other's previous reports.

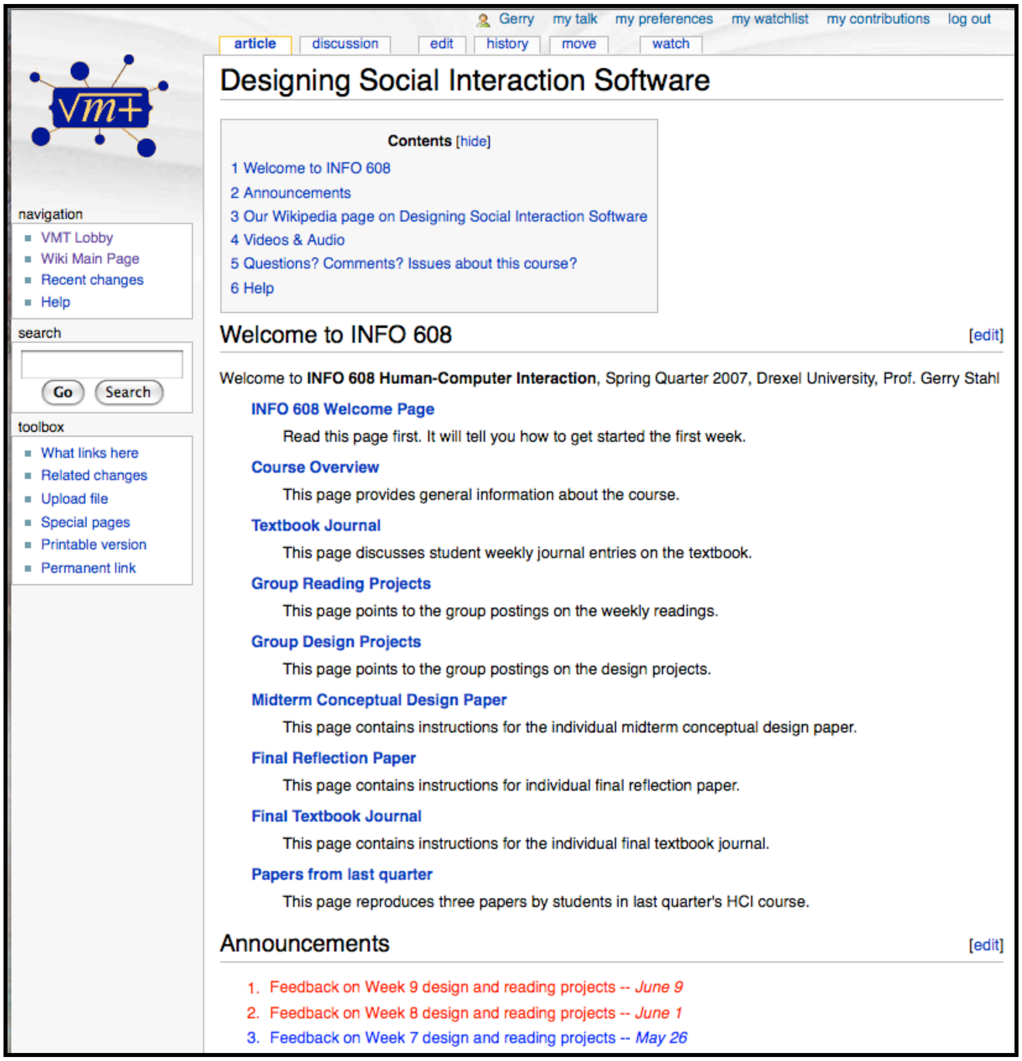
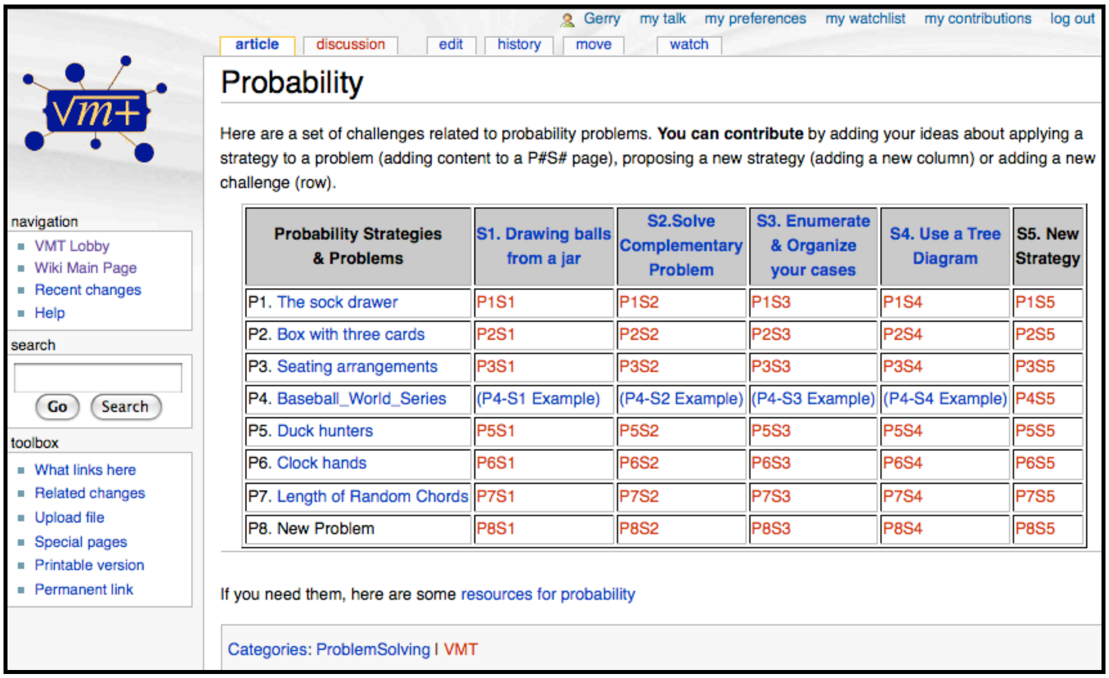


Figure 16-3. The VMT course wiki.

While the chat rooms are open to all users, people rarely visit rooms other than those of their own group. So the chat rooms are basically meeting and work places for the small groups as they engage in collaborative learning. The VMT Lobby provides a portal for the individual user to browse the people and topics of the community and to select a room for group work. The wiki, on the other hand, primarily provides a community space in which the work of all groups is coordinated, commented upon and perhaps summarized.

Figure 16-4 shows a wiki page for Spring Fest 2007, which involved probability problems. This main page for the community participating in this event provided a knowledge-building space, analogous to Wikipedia. That is, anyone in the community could add information to this catalog of knowledge about K-12 probability as well as browsing the space. The space is seeded with a number of different probability problems and several strategies for solving such problems. During the Spring Fest, student groups were to each initially select a problem and try to solve it with one of the strategies. Then they would post a summary of their solution path on the wiki page linked to from the table in Figure 16-4 for that problem and that strategy. Subsequent work would involve trying the same strategy on other problems or other strategies on the same problem, followed by comparing the results posted by other groups. The idea was that this kind of knowledge-building repository could persist and evolve through use in the future.


Gerry my talk my preferences my watchlist my contributions log out

article discussion edit history move watch

Probability

Here are a set of challenges related to probability problems. **You can contribute** by adding your ideas about applying a strategy to a problem (adding content to a P#S# page), proposing a new strategy (adding a new column) or adding a new challenge (row).

Probability Strategies & Problems	S1. Drawing balls from a jar	S2.Solve Complementary Problem	S3. Enumerate & Organize your cases	S4. Use a Tree Diagram	S5. New Strategy
P1. The sock drawer	P1S1	P1S2	P1S3	P1S4	P1S5
P2. Box with three cards	P2S1	P2S2	P2S3	P2S4	P2S5
P3. Seating arrangements	P3S1	P3S2	P3S3	P3S4	P3S5
P4. Baseball_World_Series	(P4-S1 Example)	(P4-S2 Example)	(P4-S3 Example)	(P4-S4 Example)	P4S5
P5. Duck hunters	P5S1	P5S2	P5S3	P5S4	P5S5
P6. Clock hands	P6S1	P6S2	P6S3	P6S4	P6S5
P7. Length of Random Chords	P7S1	P7S2	P7S3	P7S4	P7S5
P8. New Problem	P8S1	P8S2	P8S3	P8S4	P8S5

If you need them, here are some resources for probability

Categories: ProblemSolving | VMT

Figure 16-4. The VMT probability wiki page.

The VMT environment has come a long way from the simple AOL Instant Messaging system to the current lobby/chat/tabbed-spaces/wiki multiple interaction space. In part, this increased complexity parallels the shift from simple math exercises to open-ended explorations of math worlds, from one-shot meetings to multiple-session Fests, from problem-solving tasks to knowledge-building efforts. Along with the considerable gain in functionality come substantial increase in complexity and the potential for confusion. This has been countered by trying to extend and supplement the integration approaches of ConcertChat (see Chapter 15). The graphical referencing and the history scrollbars have been extended to the multiple tabs. New social awareness notices have been added to track which tab each group member is viewing or referencing.

Integration across modules has been important. Logins and passwords have been unified across the Lobby, chat rooms and wiki, so that logging into one automatically logs into the others. People registered in one show up in the profiles and messaging system, by their selected community. When a new chat room is created, it is categorized by a community (e.g., HCI), subject (e.g., Interaction with Computers), a topic (e.g., Week 3's assignment) and a group (e.g., Group 3). A new wiki page is generated for posting the summary from this room. The media-wiki functionality of categories automatically associates this new page with aggregation pages for the community, subject, topic and group. The version of MathML that was developed for chat postings in ConcertChat has been implemented for the textboxes in the shared whiteboards as well as in the VMT wiki, so that math expressions copied from one of these media to another retains its formatting.

While the VMT environment has been tuned to the needs of high-school math students, it has proven effective for other collaborative activities as well. The specifically math-oriented functions—like our implementation of MathML for displaying equations and the whiteboard's stock of Euclidean shapes—play a relatively small role. The tools for integrating the multiple work spaces—like the graphical referencing from chat, the creation of wiki pages corresponding to each chat room and the automatic posting of summary text to the proper wiki page—are more important and are applicable to all knowledge domains.

Our collaborators at other locations (Singapore, Montreal, Pittsburgh, Wisconsin, Romania, Hawaii, Brazil, New Jersey) and we use the VMT environment for coordination of our design work on VMT, for collaboratively critiquing each other's research papers, for holding virtual committee meetings, for pre-teacher training and for student collaborations in other domains like physics or argumentation. Each of these different uses can work effectively in our current environment, but each also suggests new features tuned to the new application. A characteristic of design-based research as used in the VMT Project is that it makes no pretense to ever produce a final version of the software. We continue to use and evolve the VMT environment.

References

Andriessen, J., Baker, M., & Suthers, D. (Eds.). (2003). *Arguing to learn: Confronting cognitions in computer-supported collaborative learning environments*. Dordrecht, Netherlands: Kluwer Academic Publishers. Computer-supported collaborative learning book series, vol 1.

Design-Based Research Collective. (2003). Design-based research: An emerging paradigm for educational inquiry. *Educational Researcher, 32*(1), 5-8.

Dewey, J. (1938/1991). Logic: The theory of inquiry. In J. A. Boydston (Ed.), *John dewey: The later works, 1925-1953* (Vol. 12, pp. 1-5). Carbondale, IL: Southern Illinois University Press.

EETI. (2007). *Effectiveness of reading and mathematics software products: Findings from the first student cohort*: US Congress.

Koschmann, T. (1996). Paradigm shifts and instructional technology. In T. Koschmann (Ed.), *CSCL: Theory and practice of an emerging paradigm* (pp. 1-23). Mahwah, NJ: Lawrence Erlbaum.

Landauer, T. K. (1996). *The trouble with computers: Usefulness, usability, and productivity*. Cambridge, MA: MIT Press.

Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge, UK: Cambridge University Press.

Noam, E. (1995). Electronics and the dim future of the university. *Science, 270*, 247-249.

Piaget, J. (1990). *The child's conception of the world*. New York, NY: Littlefield Adams.

Preece, J., Rogers, Y., & Sharp, H. (2007). *Interaction design: Beyond human-computer interaction* (2nd ed.). New York, NY: John Wiley & Sons.

Roschelle, J., & Teasley, S. (1995). The construction of shared knowledge in collaborative problem solving. In C. O'Malley (Ed.), *Computer-supported collaborative learning* (pp. 69-197). Berlin, Germany: Springer Verlag.

Sawyer, R. K. (Ed.). (2006). *Cambridge handbook of the learning sciences*. Cambridge, UK: Cambridge University Press.

Scardamalia, M., & Bereiter, C. (1991). Higher levels of agency in knowledge building: A challenge for the design of new knowledge media. *Journal of the Learning Sciences, 1*, 37-68.

Scardamalia, M., & Bereiter, C. (1996). Computer support for knowledge-building communities. In T. Koschmann (Ed.), *CSCL: Theory and practice of an emerging paradigm* (pp. 249-268). Hillsdale, NJ: Lawrence Erlbaum Associates.

Thorndike, E. L. (1914). *Educational psychology* (Vol. I-III). New York, NY: Teachers College.

Vygotsky, L. (1930/1978). *Mind in society*. Cambridge, MA: Harvard University Press.

Winograd, T., & Flores, F. (1986). *Understanding computers and cognition: A new foundation of design*. Reading, MA: Addison-Wesley.