Design, Science, *and* Engineering Topics? Teaching HCI with a Unified Method

D. Scott McCrickard, C. M. Chewar, and Jacob Somervell
Center for Human Computer Interaction and Department of Computer Science
Virginia Polytechnic Institute and State University (Virginia Tech)
Blacksburg, VA 24061-0106
1-540-231-7409
{mccricks, cchewar, jsomerve}@cs.vt.edu

ABSTRACT
Reacting to challenges that have been observed in human-computer interaction (HCI) education, as well as the multidisciplinary design, science, and engineering underpinnings, we investigate a pedagogical approach based on case methods. Our study of various case method techniques in an undergraduate HCI class provides insights into challenges that can be expected in the employment of case methods, student learning outcomes, and considerations for HCI curriculum planning. In general, case methods show great promise with a wide variety of topics, and we present broad recommendations for future work that will improve integration of HCI professional practice, research, and education.

Categories and Subject Descriptors
K.3.2 [**Computer and Information Science Education**]: Computer science education

H.5.2 [**User Interfaces**]: Theory and methods

General Terms
Design, Human Factors

Keywords
Case methods, human-computer interaction

1. INTRODUCTION
The emergence of human-computer interaction (HCI) as a driving force within software development practice and computer science research has sparked wide interest in effective teaching methods for the discipline. As the focus of design has shifted from the command line to graphical interfaces to off-the-desktop ubiquitous computing paradigms, the computer science undergraduate major must have a solid appreciation of HCI topics to succeed. Understanding HCI will allow the interface designer to produce products that are usable by everyone, extending the impact of computing and communication to a diverse set of users within many domains, academic disciplines, and outside demographics. Yet HCI educators continue to struggle with the key issues of their discipline. Reports from recent SIGCSE conferences have noted that colleagues within computer science departments perceive that HCI is (and perhaps even should be) little more than a course on Visual Basic, with a large focus on implementation of interfaces [2]. Others have noted that HCI courses within the computer science curriculum tend to be survey courses on various elements of interface design, or a course that has students building cool interfaces [4]. Yet, even in the early years of HCI, leaders in the field suggested adapting educational approaches from related disciplines [10]. This argument provides inspiration to the approach taken in this paper.

ACM SIGCSE '04, March 3-7, 2004, Norfolk, VA.

1.1 Examining the Roots of HCI Education
HCI has elements of design. Creativity has long been valued in HCI, as researchers start with abstract patterns, then tailor them to the situation at hand. There is truly an art to balancing various constraints—embracing principles of design like contrast, opposition, and repetition, with the concerns of efficient information communication and user interaction. As HCI educators, we must teach students to value approaches that provide novel, satisfying, affective experiences.

HCI can be viewed as a science. Some argue that interface designs are perfected over time, starting with observation that leads to hypotheses and testing, accumulating knowledge that eventually forms theory. Reducing interfaces to basic units that can be observed and tested in a variety of conditions provides laws that describe how these units interact, leading to new hypotheses and more constraints, rules, and exceptions to rules. The resulting network of requirements for observation and data collection is only valid with methods that enforce control and replication of results. As we grow as a scientific discipline, we can analyze new events and make reasonable hypotheses to predict their outcomes. Educators must endow their students with appreciation for this vision, as well as the tools to enact it.

HCI is emerging as an engineering discipline. In reflecting on key objectives of engineering, primary concerns are with efficiency and reliability. Responding to the realistic needs of the interface development community, HCI practitioners with an engineering approach seek to build interfaces quickly and consistently in a way that will serve the desired process. Engineering as a discipline seeks procedures to operationalize best practices, allowing others to create usable interfaces and lending analytical structures to guide analysis within diverse contexts. Training HCI students to solve problems by using procedures and analytical methods supports and extends usability engineering practices.

So how should we teach HCI? Opinions differ on where HCI education should be focused, who should deliver it, and how [9]. It does not fall in any one discipline—design, science, or

engineering—nor is it a clear combination of them. Rather, it appears to be an ordering of them, with boundaries and hand-offs not yet well understood or defined.

1.2 Case Methods: A Promising Pedagogy

In exploring methods for teaching HCI, we seek an approach common in all three disciplines, ensuring continuity in method and content as a student progresses from topic to topic. An HCI learning experience is not a passive experience, but an active exchange that builds on successful endeavors within other computer science disciplines, where interactive lectures have provided an active learning environment for fundamental CS courses [8] and puzzles and games enrich the teaching of operating systems [6]. Playing a game, providing short answers, and engaging with peers and instructors gives students the opportunity to shows mastery and highlights problem areas in classrooms to instructors. Centering the educational experience on *cases*, a common artifact in active learning approaches in design, science, and engineering, provides the opportunity to unify these diverse but important disciplines within HCI education.

We analyzed three distinct types of case methods for teaching identified design, science, and engineering topics in human-computer interaction: case history review, problem-based learning, and decision-making cases. The *case history review* method [1] is best suited for design topics, as it generally relies on information sources that are rich in background detail and that focus on early design work; information that is necessary for successful design analysis. *Problem-based learning* is an approach commonly used in science subjects [5] and can leverage a variety of actual materials (familiar interfaces, ongoing development projects, contemporary articles) [3][5]. For engineering topics, we used *decision-making cases*, which rely on stories of real engineering experience, usually ending at a point where a specific decision must be reached by employing some engineering process [7][6].

We used several problem-based learning techniques, three which employ a discussion format (issue case, single subject focus, and resolution of conflict) and two which provide practical experience (concept application and lab-based experiment simulation). The *issue case* technique generally uses a current news article with controversial content to draw out various perspectives with respect to a specific situation, requiring students to formulate arguments related to a key concept [3][5]. The *single subject focus* technique is used to develop analytical skills about a broad process that can be carried out with a variety of methods by examining a particular method in depth—although the discussion reviews the method in question, students would be able to transfer the critique to alternate methods. A *resolution of conflict* discussion is similar to an issue case, requiring students to analyze a situation and formulate arguments, however, the materials normally provide conflicting information that the students must sort out in order to reach a consensus. To gain practical experience, the *concept application* technique requires students to apply a theory or concept to a specific situation and draw conclusions. A simulated *lab-based experiment* provides practical experience with testing methods and procedures [5]. To ensure that these various methods can indeed provide an effective connected approach to teaching HCI, we investigated implementation in an undergraduate course.

2. APPROACH

As we considered how to integrate these methods with the topics we teach in introductory HCI courses, we followed a few guidelines to ensure a rigorous and measurable experience. First, we were certain to distribute the case method techniques throughout various parts of the course to ensure that students encountered them at different stages in their learning. Each technique was exercised at least three times, often with different types of supporting materials. Although we employed different techniques to investigate a wide variety of HCI topics, we thought it important to gain an element of consistency by building themes that continued between multiple activities, such as development of an immersive virtual environment and critiquing familiar graphical editing interfaces. We also recognized that not all HCI topics could be appropriately addressed with just any of the case methods, so we discarded any but the strongest topic-method pairings to include in our study.

Using these guidelines, we developed a study around an undergraduate HCI class, conducted over a seven-week summer semester with 75-minute daily classroom meetings. The course plan included eleven opportunities for case activities, which we designed to address the topics and disciplinary concerns depicted in figure 1.

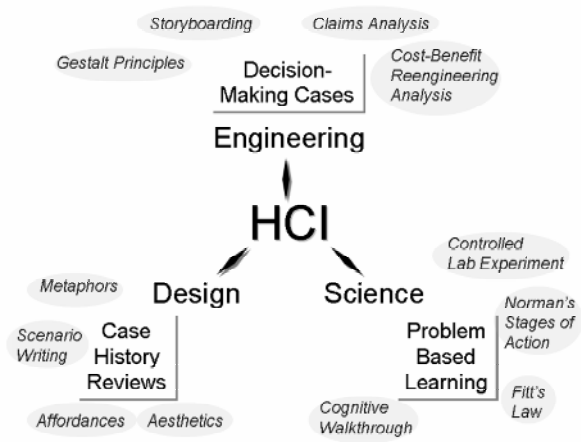

Figure 1. Case methods as a unifying pedagogy for HCI. Shaded regions show HCI topics addressed by each method.

The instructor for the course, while an experienced researcher with close to a dozen publications to his name, was a new instructor teaching the course for the first time in a compressed summer session. However, he was supported by an experienced instructor who had taught the course several times before and an additional researcher who assisted in the planning and preparation of materials. Together, this team constructed a course plan to meet the objectives and served in different roles as activity leader (leading the students in the method at hand) and observers (collecting data on student performance and evaluating the instructor's execution of the technique). Several notes were made during the activity sessions, including the portion of students making thoughtful contributions, student mastery of concepts through individual assessments, collection of anonymous student self-reporting on preparation time, and collection of general student feedback. Prior to each activity, the instructor was reminded of the specific goals of the technique, as well as the methods that would support activity management. After each

activity, the observer would review successful and unsuccessful aspects of the actual activity execution in preparation for the next instance of the technique. Also collected was comprehensive student feedback in an end-of-course survey.

3. EXPERIENCES WITH CASE METHODS

We note the following outcomes, which are exemplary of our experiences in using each of the case method techniques to teach an undergraduate HCI class, derived through direct observation.

3.1 Case History Reviews

As a case technique to focus instruction on design-related concepts within HCI, such as the role of scenario creation within an iterative design process, metaphors, aesthetics, and affordances, we employed case history review from architectural education [1]. As with all the case activities, a lecture was given by the instructor in the previous class session that introduced the topic within the larger context of HCI and an interface development lifecycle. Students prepared for these cases by studying reading materials, such as professionally prepared cases in an online case library or conference papers that detailed system design features. In one instance, the reading materials were supplemented with a live demonstration of an interface in development, which walked students though the iterative prototype versions and design rationale.

Using this case discussion technique, the primary role of the instructor is to guide students toward extracting and synthesizing facts from the case history that exemplify dimensions of the abstract design concept, moving them to a deeper understanding of the case events. As literature on this technique points out, this should be done in a very non prominent manner that forces students to draw their own conclusions and debate critical points with each other, rather than looking to the instructor for answers. In our experience, we struggled with several instructor challenges that came up in the early adoption of this technique:

- *Redirecting questions*. Students often ask the instructor direct questions to clarify facts from the case or aspects of the concept being studied. Instructors must be able to resist the tendency toward establishing their own expertise and challenge the students to answer their own and each other's questions.

- *Facilitating participation*. Should an instructor call on students not volunteering comments? While it is compelling to try to involve as many students as possible in the case discussion to assess learning, effective use of this method should encourage students to participate on their own. Instructors must develop techniques that inspire participation through relevant knowledge and personal experiences, rather than using more coercive methods of eliciting participation.

- *Creating controversy*. Especially at the beginning of the semester when students are new to the subject and the discussion technique, they tend to agree with each other and anything that the instructor says. This tends to result in a communication pattern of instructor question, student answer, instructor follow up, student answer, etc., rather than student-to-student interaction.

As far as the student outcomes gained through the use of this technique, it was somewhat disappointing that about two thirds of the students could recall the concepts that these case reviews demonstrated at the end of the semester, and very few students cited this type of case activity as their favorite method or the method they felt best facilitated learning. We also observed the lowest in-class participation rates in these types of activities, with rates about 10 percentage points below average, despite the fact that students prepared longer on average for these activities by about 15 minutes.

3.2 Problem Solving Cases

To provide instruction on the hypothesis testing methods important within HCI, we used problem solving cases adapted from other scientific disciplines [5]. Although we employed the variety of techniques discussed earlier (issue case, single subject focus, concept application, lab-based experiment simulation, and resolution of conflict), they all had in common the presentation of a problem, the requirement for students to generate a hypothesis, the use of a problem solving method, and the presentation of small group results to the class as a whole. To prepare for these case activities, students studied materials that detailed the problem solving method or context specific to the problem situation. Topics reinforced using this case technique included identification of target user groups, discovery of usability problems with a cognitive walkthrough, evaluating information design with Norman's stages of action, and performing a controlled lab study to identify interaction shortcomings.

- *Providing templates*. As these groups develop their approach to the problem, it becomes difficult for the instructor to review the groups' mastery of the method without seeing evidence of their thinking processes. A critical element in planning the activity becomes the creation of templates for students to fill in, e.g. on an overhead transparency or a blackboard, to facilitate presentation to the class as a whole.

- *Focusing problem discussion*. Successfully employing a problem solving method in a short amount of classroom time requires strict attention to keeping the discussion focused, however students were often observed straying from the topic at hand or defaulting to other more familiar methods. Again, the idea of a solution template is helpful in keeping students focused on the procedure to be learned.

- *Learning from others*. While it is often helpful to have selected groups present results near the end of the activity period to assess and promote responsibility, students in other groups often see little value in these presentations, missing a valuable opportunity for students to learn from each other. Instructors can facilitate this learning experience by requiring groups to contrast their findings with other groups, employing peer review in grading, or selecting the best group other than their own.

Problem based case activities usually resulted in the highest levels of student recall at the end of the semester, although some of the specific techniques were not as effective. The strongest cases (issue case, single subject focus, lab-based experiment simulation) were ones that involved significant hands-on elements, such as preparing and executing an experiment or filling out a form that guided an analysis, while the activities that did not promote as much learning (resolution of conflict, concept application) tended to be purely analytical tasks that required synthesis of multiple sources of information. With lab-based experiment simulation most frequently chosen, more than half of the students selected a problem-based case technique as their favorite type of activity.

Participation and preparation were also strong in all problem based case activities.

3.3 Decision-Making Cases

To provide students with deeper insight into the processes of employing claims analysis, using Gestalt principles, evaluating storyboarding techniques, and choosing reengineering approaches, we used decision-making cases. For each of these activities, students were provided with enough background information to understand a dilemma in an interface development process. In the activity, they were to identify possible courses of action, develop the pros and cons of each option, and specify criteria and weights by which to evaluate their decision. To achieve this, students had to reach consensus through group discussion about the potential consequences from each course of action, drawing upon the topic at hand. The usual pattern in which this progressed was guided by filling out a large matrix of action options and decision criteria on the blackboard.

- *Using formal decision-making processes.* This type of activity only runs smoothly if students thoroughly understand how to use a common formal decision making process. Although the classroom setting presents an opportunity to teach such a process, students should have the decision-making mechanics in mind as they prepare for the case assignment.

- *Enforcing justification of decisions.* A decision-making case approach must rigorously integrate the topic at hand by requiring students to justify the scores they assign to different courses of action. Having students record rationale for the scores and weights within their decision-making process allows less apparent tradeoffs and contradictions to emerge, in turn allowing deeper appreciation of the topic.

- *Encouraging consensus formation.* By far, the decision-making case activities generated the most interactive student discussions with lots of controversy. However, the downside of this consequence is that students will easily lose focus on the decision-making task and belabor the discussion of small points if the instructor does not help students capture key differences and reach a final conclusion.

Activities that employed decision-making cases also resulted in high levels of student retention and participation and lower than average student preparation, although few students selected this technique as their favorite case method. On the other hand, two thirds selected a decision-making activity as the worst. In our employment of the decision-making case methods, we announce the dilemma at the start of the in-class discussion, but based on student feedback, presenting the dilemma with the background material would have improved execution of these activities.

3.4 General Feedback

The results captured in the sections above are based on student feedback on individual cases. However, we also surveyed students about the three specific case methods (case history reviews, problem solving, and decision-making cases). Problem-based learning and decision-making approaches were found to be about equally effective, and far superior to the case history reviews in facilitating student interest in the learning topics. About half the students responded that the problem-based learning techniques were the best methods to inspire participation, consistent with observed results, with most of the remaining half of the students selecting decision-making cases. Case history reviews showed a definite advantage in assisting long-term learning over decision-making cases, as did problem-based techniques.

When asked to respond on a five-point Likert scale to broader questions about the case method, general student consensus (considering response means and standard deviation) is described in the table below:

Definite agree	• good for teaching design and problem solving • enjoyed more than reading the textbook • useful for guiding project actions • enhanced understanding of HCI
Agree	• good for training decision-making processes • enjoyed more than lectures • eager to do again
Somewhat agree	• good for illustrating abstract concepts • enjoyed more than other teaching approaches
Neutral	• enjoyed more than course design project

To apply these techniques smoothly, instructors should expect similar challenges as noted throughout this section, although case-based learning approach will present an exciting and active learning environment.

4. SUGGESTIONS FOR CASE LEARNING

From our experience, we can suggest several additional considerations that are important for employing various case method learning techniques:

- *Case history review.* The ability to recognize and analyze the decisions made by designers depends on a thorough understanding of the design problem constraints, requiring deep familiarity with the case write-up. Often the problem that we find is that students do not prepare at anything more than a cursory level, leaving it up to the instructor to find a way to motivate them to read and think about the case as they would a project of deep importance to them.

- *Problem-based learning.* This approach requires students to apply specific methods to well-defined problems. Often the methods are new to students, and the predictable result is that they default to more familiar methods, straying from the point of the activity. Instructors must ensure that students maintain a focus on both the methods to be learned and the problem at hand, perhaps reserving later time for comparison of alternate methods.

- *Decision-making cases.* This complex approach requires students to complete several steps of a formal decision-making process, develop a personal understanding, and still achieve a common, shared decision with their classmates. Depending on the curriculum, many students may have had a course that teaches them a formal decision-making process, but in most cases students will be ill-prepared to complete this case analysis and will require several iterations before accomplishing the technique successfully. As such, unless students are experienced in decision making, it seems best either to dedicate a large number of case-related activities to decision-making cases or to avoid using them as the overhead of learning the technique will yield minimal success.

4.1 Integrating Cases and Design Projects

One result we hoped to achieve with case-based learning was to gain the benefits of design experience without devoting resources required for hands-on interface implementation. The course in which we tested the case method program included a substantial semester-long design project. Perhaps as the most resounding aspect of student feedback on the case method was that they enjoyed and felt like they got more out of their design projects. However, even with very specific, phased submissions to allow frequent instructor criticism throughout the design process, our feeling is that student design projects did not consistently demonstrate a strong level of conceptual mastery, while the case activities did provide such demonstration. Deliberately designing the case activities to directly influence thinking related to the design project may be a more complete solution—providing students with a large amount of valuable and enjoyable experience in an active learning situation with rich focus on essential HCI design, science, and engineering concepts.

To address this in the next step in our research, we will investigate how in-class, collective case activities can more closely parallel out of class, small-group design projects. We are also intent on narrowing the types of case methods used to provide a more consistent and predictable learning environment with the case activities. This may suggest adapting the various methods into a single method for case learning that is most conducive to HCI learning objectives—we are investigating different approaches that we will compare with the results obtained in this initial effort. Studies of other active learning approaches to HCI may also yield other promising methods for educating students on the diverse aspects of HCI.

4.2 Professional Case Generation Practices

We also see an important avenue of future work in the investigation of professional HCI practices that result in the generation of cases. Given the success we observed with cases as primary material in HCI education, as educators, we have a definite interest in a continuous source of case material that inspires reflection on late-breaking design practices, guidelines, and methods. Wixon presents a recent argument for case generation processes as an integral part of usability testing, based on his recognition during involvement in Microsoft product development that case writing provides analysts with a richer mechanism to express usability findings and a means for documenting lessons-learned within industry [11]. As procedures are refined for creating high quality interface design cases and as openly accessible collections grow within the HCI profession (such as the stories collected though the conference on Designing for User Experiences (DUX) and the AIGA Design Case Study Archive), academia can also benefit if we have developed complementary pedagogical methods.

As a final note, we also recognize the importance of developing techniques for design case meta-analysis, another point in Wixon's argument for cases [11]. Professional HCI practices result in few examples of or opportunities for interface design meta-analysis, perhaps since design, testing, and reporting methods are largely non-standard and often proprietary. However, if cases can be developed in a way that facilitates comparison of design efforts between systems, this can provide very valuable material for education. Toward this end, we are working on visualization techniques, usability testing methods, and knowledge reuse approaches to assist in case meta-analysis and education efforts. A broad perspective on the impact cases and case-based learning can have on HCI education will increase the potential for infusing state of the art research in HCI design, science, and engineering activities into our classrooms.

ACKNOWLEDGEMENT

Mary Beth Rosson and John M. Carroll's 2003 Case Studies Summer Workshop at Virginia Tech provided inspiration for this research effort.

REFERENCES

[1] Akin, O. Case-based instruction strategies in architecture. *Design Studies 23* (2002), Elsevier Science, 407-431.

[2] Barnes, J., Bryant, R., McCracken, D.D., and Reiser, S. Teaching human-computer interaction: Reports from the trenches. in *Proceedings of the Technical Symposium on Computer Science Education (SIGCSE 2003)*, 125-126.

[3] Bieron, J.F. and Dinan, F.J. Case studies across a science curriculum. Department of Chemistry and Biochemistry, Canisius College in Buffalo, NY. Available at http://ublib.buffalo.edu/libraries/projects/cases/curriculum.html

[4] Douglas, S., Tremaine, M., Leventhal, L. and Wills, C.E. Incorporating human-computer interaction into the undergraduate computer science curriculum. in *Proceedings of the Technical Symposium on Computer Science Education (SIGCSE 2001)*, 211-212.

[5] Herreid, C.F. Case studies in science: A novel method of science education. *Journal of College Science Teaching 23* (2), February 1994, NSTA Publications, 221-229.

[6] Hill, J.M., Ray, C.K., Blair, J.R.S., and Carver, C.A. Puzzles and games: Addressing different learning styles in teaching operating systems concepts. in *Proceedings of the Technical Symposium on Computer Science Education (SIGCSE 2003)*, 182-186.

[7] Kardos, G. Engineering cases in the classroom. in *Proceedings of the National Conference on Engineering Case Studies (1979)*, Available at http://www.civeng.carleton.ca/ECL/cclas.html

[8] Rodger, S.H. An interactive lecture approach to teaching computer science. in *Proceedings of the Technical Symposium on Computer Science Education (SIGCSE 1995)*, 278-282.

[9] Sears, A. and Williams, M.G. None of the above: What's really essential in HCI education? in *Extended Abstracts of Human Factors in Computing Systems (CHI 1997)*, 109-110.

[10] Winograd, T. What can we teach about human-computer interaction? In *Proceedings of Human Factors in Computing Systems (CHI 1990)*, 443-449.

[11] Wixon, D. Evaluating usability methods: Why the current literature fails the practitioner. *interactions 10* (4), July/August 2003, 29-3.