Metz, M. H. (2001) Intellectual border crossing in graduate education: A report from the field. *Educational Researcher*, *30*(5), 12–18.
Newell, W. H. (1992). Academic disciplines and undergraduate interdisciplinary education: Lessons from the School of Interdisciplinary Studies at Miami University, Ohio. *European Journal of Education*, *27*(3), 211–221.
Schwab, J. (1969). The practical: A language for curriculum. *The School Review*, *78*(1), 1–23.
Vygotsky, L. (1978). *Mind in society* (M. Cole, trans.). Cambridge, MA: Harvard University Press.

Chapter 8

Applying the "Studio Model" to Learning Technology Design

Charlie Cox, Steve Harrison, and Christopher Hoadley

Introduction: An Unforeseeable Affordance

Studio classrooms (e.g., laboratories, workshops, and ateliers) can produce disconcerting and even revelatory learning experiences. Studios seem to be catalysts for such events, as students confront learning outcomes that seem to pop out of nowhere. In this chapter, we'll discuss how the studio model can be applied to teaching learning technology design.

Let's examine one such encounter and its results within the studio context that enabled it. To do so, we turn to architectural design education, wherein students are required to participate in studios. One of us (Steve Harrison) has this story to tell about a studio course he took as an undergraduate:

> It was a long time ago. Arch 102 is the second "upper division" undergraduate architectural design studio class. The final project: design a three-unit apartment building on a hillside. The drawings were to be done in ink.
>
> At about midnight on the night before the design review, one student wandered over to another's table. The student was painstakingly laying down ink lines. Drawing in ink was really hard to do without having it smear under the T-square or the triangles. Looking over the scheme, the wandering student asked, "Where's the living room of the top unit?"
>
> "Oh shit!" In transferring the many pencil sketches to the final boards, the living room from the upper-most unit had gotten dropped. The place where it had been intended to go was now a carefully ink-drawn mass of walls. In fact, just about the only elements of the plan drawing not yet inked were the property lines and landscaping.
>
> By 2:00 a.m., a living room was added to the plan, appended to the side of the building, extending over the property line. It was there because it was the only open space on the board and there had been a few un-inked lines to work with.

At 7:00 p.m., we gathered for the design review. As was the custom, outsiders were brought in for the design review. Class faculty sat as a sort of host for the evening, framing the design problem and the issues that the class had addressed. Students (having turned in the projects at mid-morning) had returned after napping, eating, and changing their clothes.

The jury of outside design critics consisted of a couple of other senior design faculty members and one local architect not associated with the school. The most well-known person on the jury was (the late) Joseph Esherick, both a practicing architect and a senior member of the faculty. He was very well respected among designers and students. He had been a central figure in the expansion of the school and even had designed the very controversial building it was housed in. He held a fair amount of political sway in the department, and he ran a firm that most students wanted to work for.

About halfway through the evening, it was time for the over-the-property line scheme to be presented. Most of the students became visibly uncomfortable as this obviously embarrassing design was presented. The student-architect stammered a bit, but pressed on. It made me so embarrassed to watch, I remember turning red.

Joe Esherick looked at it carefully then, and said, "You know that's what I'd do. The view is better from that location, so I'd tell my client to buy the lot next door."

In that one moment, everything changed.

The Big Guy had said that it was OK, clever even! And that we should do out-of-the-box thinking. He turned what was a widely embarrassing moment into one of challenging: the assumptions of the problem statement, constraints-as-immutable, and the collective wisdom of the students who had been present in the studio the night before.

Over the years, I have come to think about the complexity of that situation. Who was Joe Esherick talking to? How was it being heard? What effect did it have? Was he suggesting that this "out of the box" thinking rated a very good grade?

His observation undercut the pedagogy of the entire studio, since (especially in the first few design studio classes) dealing with constraints was the central theme of design education. Were his words intended that way? (We can assume that he did, since the faculty of the class was present.)

He had been invited in from the outside to review the work. What were the implied limits of such a review? And then there is the matter of Joe reading something into the intent of the designer that was not there. Did it matter? Was Joe aware that many in the audience were uncomfortable?

It is impossible to know if the student remembered what had been said in the review -- or if it made any impact on her approach to design.

This is also a story about a rite of passage. It has lots of canonical elements of the tradition of getting through design school: the preceding "all-nighter"; the camaraderie, community and competition of working together in a studio; the desperate rescue of an idea; and the presentation in front of a Big Name Architect.

All of this out of one little incident in a design review with very limited consequences. None of the presented designs would be selected to go forward with construction and the grade would not make or break a student's career. While I remember my reaction and the quality of the presentation after all these years, I cannot even remember if the student mentioned that the crossing of the property line had been a late-night fix, or not. I don't even remember either my own project or the comments made about it, but this event has been of great value to me, nonetheless. I have often used the story to explain the limitations of constraint-based reasoning (often used in artificial intelligence), and as a totem of problem re-formulation when I feel stuck.

This is an illuminating story of how studios provide learning experiences different from those of, say, lectures—but it happened in architecture school and dealt with students learning about designing built form. How does it speak to courses on designing technologies devoted to learning? Would students better learn technology design in a course taught as a studio?

Organization of this Presentation

The topics we propose for discussion run in sequence from general to specific. First, we compare some traditional studio domains (especially architectural design) with technology design. Then we'll recount what happened when we set up technology design courses to resemble studios, both physically as far as was convenient and pedagogically.

Bear in mind that the implementations we tried were not literal transfers of technology design students into a studio course that architectural design students would find familiar. Design has to be pried apart from the domain in which it takes place. Budgen (1995, p. 301) reinforces our "distinction between learning about design practices and notations and learning how to design software" when he reviews software design as studied

and practiced. Hazzan & Karni (2006) have also distinguished between architectural design and software design. Nonetheless, Kuhn (1998) argues for the advantages of treating software designers like architectural designers, saying, "[T]his analogy helps us focus on usefulness, usability, user needs and practices, and other technical and non-technical aspects of good software design. It highlights concerns about people's lives and work practices and how people 'inhabit' systems" (Kuhn, p. 65). Elsewhere on the same page, she lists the key characteristics of studios as applied to software design as:

- project-based work on complex and open-ended problems
- very rapid iteration of design solutions
- frequent formal and informal critique
- consideration of heterogeneous issues
- the use of precedent and thinking about the whole
- the creative use of constraints
- the central importance of design media.

In short, a studio can be used for both learning to design in general and learning to design software and other technologies. A studio allows for iteratively demonstrating development and sophistication not only to a student's peers but to faculty and experts in the profession. And the feedback received as a result of each demonstration informs the next iteration. Social interaction, as noted earlier by Steve Harrison, is the crux of studio practice.

Studio Structure

What do we see when we look at studios? One characteristic is that the activity of posing and solving problems is less constrained than in other content delivery schemes, and more paths can be taken to reach those solutions. Studio time is spent iterating and refining approaches to problems and crafting artifacts to communicate solutions. It is common for a studio instructor to require students' participation in defining the scope of the problems to be addressed and then to revise that scope or even the outcomes and objectives pursued. Students sculpt problems and solutions, sometimes informally with one another, and at other times while bandying arguments with instructors or critics from outside the course. They take the feedback and start another iteration, perhaps even redefining the problem and the solution in order to recommence. Thus, it should not be expected that any two students arrive at similar interpretations or conclusions, even when those students have agreed to

the same initial problem conditions. Every studio leads somewhere, just not to the same somewhere for every student.

Which is not to say that there are no constraints for studios, just that the constraints are more malleable than in other learning environments. Sometimes an apparent constraint turns out to be a valuable affordance. In fact, Chris Hoadley makes a point of telling students that, when confronted with an either-or path of competing solutions, a designer should seek to manipulate the constraints of the problem to find a solution that incorporates both of the desirable features. This new solution to the redefined problem should transcend the trade-off.

Two immutable constraints are time and space. Outside of class meetings, the hours spent defining and redefining a problem and then justifying a formative or summative solution are beyond the resources available to many students. The amount of time just isn't anticipated nor set aside for a technology design course the way it is in architecture, wherein the behavior and structure and function of studios are givens. Furthermore, studios typically have dedicated spaces. These serve not only for the co-location of students during projects but for storing prototypes and models and sketches.

Finally, there are particular social processes that allow this type of learning to take place. Among these, the most salient for design education are spatial ownership and persistence; review and critique; and overhearing.

Spatial Ownership and Persistence

In Steve's architecture studio story, the protagonist got valuable feedback simply because he was in the same space as his fellow students at the same time. The studio environment encourages many such collegial interactions: advice, materials, arguments, and even distractions are all readily available. It is through encounters with other students who are continuously available that disciplinary conventions are learned. Answers emerge with time and iteration, not because a student makes fewer and fewer errors but because what begins as an amorphous, ambiguous proto-solution grows not only in specificity but also in justification. And this growth is in direct relationship to the dialogues that have taken place in the studio.

The space of the design studio where these dialogues occur can be said to be owned by the class. Yes, the classroom for the course becomes a project room, as we have already indicated, but there are other social consequences of ownership that are part and parcel of design learning. To enter the studio is to enter into the space of the group or, rather, of the project that concerns the group.

What does this teach about designing? It communicates strongly that the project comes first. As we further discuss assessment and the relation of project ideas to ego, we will see how important this is. The "project comes first" is a value that is widespread in design practice, but it is fundamentally a socially constructed value rather than, say, a fundamental ethic of designing. It is often learned by informal social interaction: "You're going home NOW? It's only 11:30!" (We can just imagine hearing that going on all around the student who was struggling to add on the missing room.)

The sum of the co-location of colleagues in a dedicated space is a persistence of resources, and one important thing about these resources is that the need for them is largely unpredictable. When a question arises, being able to look at information provided by the instructor or other authorities is often not enough; it can be necessary to verify an interpretation with others who have parallel goals.

Let's turn now to sketching, another studio activity and tool for communicating intent. Having someone to show one's sketches to is as much an affordance of co-location as any other dialogue. It iteratively increases in sophistication as it is informed by successive rounds of debate within the studio. It grows from initially imparting conceptual direction, to faithful facsimile prototyping of final product.

Such persistence produces disconcerting and revelatory encounters, as in drawing a living room that violates a property line and unexpectedly finding that to be acceptable. Face to face with the consequences—among these the train wrecks where their old schemata used to be—students must rely on reflection and retracing their paths in order to explicate the lessons learned and to sort those for future reference. It is a small leap to presume that the recognition of design process is accelerated by having others on hand with whom to discuss how sense was, or could be, made from such an encounter.

Review and Critique

The studio milieu includes reviews performed by instructors and students in formal and informal situations known simply as "crits," being short for critiques. These already have been taxonomized (Dannels, 2005), but deserve revisiting here because the crit might be the most controversial of the studio model's aspects. There are variations, but the consistent points in the studio model are that students present a design solution and receive an explanation of what was found to be problematic, often accompanied by suggestions.

A source of the controversy is that many students carry the memories of vicious critiques directed at their studio work. Wood (2003, p. 5) compares some critiques to hazing rituals, and Stead (2003, p. 10) alludes to their "sadistic overtones." But Willenbrock (1991) recounts one such episode in detail:

> One day, a professor approached for a mid-project desk crit and pointed to the model I had constructed to transform a two-dimensional drawing into three dimensions. "Is this you?" he asked. Hoping to build a casual rapport with this rather stern young teacher, I responded jokingly, pointing to myself, "No, no this is me," then to the model, "This is my model." "No!" he replied firmly, putting his hand on my model, "This is you and this is shit!" It was an incredible high when the unity between self and work brought us praise, but quite devastating when our efforts were insulted. Frustration was heaped upon hurt when the critique was vague, or we simply could not understand.

That this sort of attack was not only allowed but perpetuated is a problem with the field. When left unexamined, instruction devised from craft, such as that for architecture studios, carries harmful habits along with the beneficial ones (Clark & Estes, 1998). And, because architecture education has employed problem-based learning for over a century, some of those unexamined habits have become reified as traditional methods of instruction in the meantime. Worse, questionable habits might even be recognized but yet not dealt with, as in, "It didn't kill me as a student, and it won't kill these students either."

Anyway, the disconnection of personal commitment is central to this form of designer-to-critic relationship. It is hard to learn and difficult to perform, especially for students entering graduate design programs with a background in engineering or other disciplines wherein results are more deterministic and evaluation seemingly more objective. That is one of the reasons why architectural design education often involves many years of studio classes beginning with fairly gentle forms of critical feedback—and argues that one or two studio classes might not be sufficient to build the confidence of students with regard to detaching from their design and yet being able to argue convincingly for it.

Then, too, the mere relationship between critique and assessment can be disorienting all by itself. Critiques that one receives in studio might not affect one's grades at all. They may be seen as a demonstration of qualitative discovery rather than an application of quantitative measurement. When delivered without personal attack, critiques are often seen merely to constitute differing viewpoints to be collected in the student's expanding catalog of expertise. Thus, one can imagine the confusion from the students' point of view: Is what they get in the form of feedback

"just one opinion" or something agreed upon by the field (or at least the instructor) that applies to one's grade?

In fact, grades often depend less on meeting a particular, fixed set of standards in the traditional assessment model (e.g., passing the final exam) and more upon what the instructor is seeking. To introduce further complexity, what is sought can vary from student to student within the same course: product or solution, process in arriving at the solution, presentation of the solution, or just plain old progress in that a student leaving the studio at the end of the semester understands more than when entering.

Overhearing in Studios

In addition to the deliberate mechanisms of feedback, studios permit overhearing. In Steve Harrison's story, the events did not happen to him, yet they affected him profoundly. He overheard the informal critique late at night and was an observer of the formal review by Joseph Esherick the next day.

The value of overhearing in design learning cannot be overstated. Lave and Wenger have written about legitimate peripheral participation, of course (Lave & Wenger, 1991). And though the value of peripheral participation in learning concepts is hard to identify, it is clearly worth much in project-based learning (Barrows, 1986; Hmelo-Silver, 2004).

Design is a wide purview, and it is unlikely that students by themselves can run the gamut of all the opportunities for learning presented in the typical design space. Even incidental contact with others' design moves and outcomes is just as important as learning from one's own design moves and outcomes. Note further, though, that overhearing in Steve's story goes beyond conceptual framing or adding of details, it is the central figure of the recollection and reported reflection. (Also notice that other than recollecting some embarrassment, his own activities are completely unremarked.)

Implementing the Penn State Learning Design Studio

We found the studio model so attractive and provocative that we applied it to technology design—in particular, to the design of software (and other artifacts) for learning. We created a course at Penn State's University Park campus, called it the Learning Design Studio, and taught it in Spring 2005 and Spring 2006. One semester it was offered not only to undergraduate students in the School of Information Sciences and Technology (IST)

but also to graduate students in the College of Education, and the other semester it was offered just to IST undergraduates.

The Learning Design Studio arranged the course activities to approximate those of a studio traditionally associated with domains such as architecture. Students shared concepts that they generated and moreover critiqued those concepts in a collaborative environment just as architecture students would. In the sections that follow, we discuss how we took the general studio model that we have described heretofore and incorporated certain attributes into what otherwise would have been a lecture-based course in user-centered design. In our case, the end users of the technologies to be designed were learners, so this could be more specifically described as learner-centered design.

Whatever the technology goal, we still had to resolve some crucial differences between the traditional studio model and what was available to us in actuality. For one, our courses lacked a physical studio to work in whenever or however long we liked. We taught the courses on a typical undergraduate course schedule for an hour of class at three days a week in a classroom that was empty for a scant 15 minutes before the next scheduled course met.

For classrooms, we were provided with computer labs: somewhere around 45-foot-wide by 40-foot-deep rooms with four long tables of 48 inches in width or so, and 12 networked stations (CPU, monitor, keyboard, mouse, internet access) at each table. Some rooms were populated entirely with Apple desktop computers, and others were equipped solely with those running Windows operating systems. One room was tiered with platforms such that each row of tables stepped up in height all the way to the back of the room, and only one side of each table faced the lecturer. In another room where the floor was all on one level, the tables were perpendicular to the front wall, and students could sit at both sides. The lecturer occupied the first 10 feet open at the front of a room, where the projection screen and chalkboards were mounted. That left approximately 18 square feet, excluding circulation paths, of usable space to each student at a more-or-less fixed seat determined by the equipment locked in place. This was not ungenerous by any means but was inflexible as far as accommodating any collaborative work where four or five students as a group might try to gather physically. Physical persistence for persons or artifacts simply was not accommodated.

That meant we had to come up with other means for creating, or at least approaching, what we would call an authentic studio experience. Though we couldn't pursue the lab as studio approach wholeheartedly, we could and did modify the lab with enough studio aspects to show the value of iteration and how some situations need to be approached by trial

and error because they don't always fall into any familiar classification from which we can draw established conventions.

Challenges from a Student Preparation Standpoint

We've discussed some physical challenges, and now it's time for some intellectual hurdles. Our resulting studio would seem novel to traditional studio instructors for many reasons, with the first three related to our students' being unfamiliar with studio goals and methods, to the extent of having a disconcerting encounter from the first day of classes.

First, the students had not been encouraged in an explicit manner in prior courses to attend to how persons other than they, themselves, would interpret their intent from the designs they produced. So part of this story's interest derives from students' encountering user-centered design and having to predict empathetically or discover through research and then accommodate the behavior of a target audience of users envisioned as learners (and, perhaps, instructors interacting with those learners).

Second, even merely approximating studios brings social interaction to the foreground, and the social interaction we were trying to foster didn't have a specific classroom to anchor it. Being social gatherings in a space used for no other classes, studios intentionally afford co-location for students to bounce concepts off one another and whoever else passes through. How does one supplement the standard classroom or computer lab, using what it can provide and substituting for what it cannot? Where else can the activities take place, and is there one best substitute or do different activities have to take place in different contexts?

Last, engineering students, for example, often are trained to think of design from a laboratory perspective whereby objects are specified, prototyped, and tested in an environment that approximates in situ placement as much as the available equipment will permit. This analysis-heavy approach lends itself well to quantification and the documentation of a design process as comparison and decision but doesn't explore the cultural and political institutions underlying those decisions. Its chief goal, in terms of process, can be to approximate an algorithmic process to replace trial and error.

Computer science has tended toward emulating this model, with structured software engineering processes, optimization through usability metrics and testing regimes, and a sense that one can, with enough proven principles and structured processes, optimize user interface design. Redefining a problem for which a solution already is in the works would seem a great waste of time, as would concurrently pursuing a second

or third solution before the first attempt had ended in either success or exhaustion.

What happens, then, when one dares, or is forced to dare, testing of the problem's very scope as laid out by the instructor? What happens when a student takes ownership of the direction in which the project is headed? After all, rejecting a program is not at all unheard of in studios (though when it is seen for the first time, as in Steve Harrison's story, its radical alienage can be frightening) but must be accompanied by a sense of responsibility for the student to establish an equivalence of effort and scope with the program that was laid aside.

Steve's architectural vignette can be read as a tension between pragmatic need and the standards perceived as appropriate design that precedes the choice or, in his protagonist's case, the realization that no choice is left but to reject the initial conditions. The student resorted to what seemed at the time no real solution at all for the design problem at hand—to push the missing living room over the property line—because there was no resource that he knew of left to him.

His immediate consideration became one of how to minimize the hit he reckoned that he was bound to take. Perhaps he debated how to eliminate the living room and continue with his drawings as before, when the omission was still yet to be discovered, obscuring his offense and trusting to hope that he might get by undetected. Finally, though, he determined the problem to be satisficed by violating a property line while maintaining the program to include a living room and holding to the deadline as a time constraint. The acceptability of this design move seemed out of bounds to the other students because neither the form of the encroaching living room nor the method for locating it had been legitimized in prior critique.

The rejection as a design move was, in fact, an appropriation. That is, the reuse of an idea, reinterpreting it (citing the source as inspiration), in essence lays claim to the idea's ownership by shaping it. That this studio's culture did not yet include an agreement about manipulation of the property line made this design move risky. Not only did it not stem from an idea previously vetted through critique, it mitigated a constraint that students had reified to be inviolable as part of the studio's ethos.

This particular set of appropriations was enabled by the complex dance between the practical and the social, between the *praxis* of the problem (the deadline, the limitations of pen and ink, the rules writ large) and the *ethos* of the studio (the attitudes, ways of thinking, interpreting, postulating and, ultimately, designing). Being clever in a socially validated way in this case was actually a reasonable substitute for solving the stated problem by allowing the students to question the assumptions and redefine the task.

What the participants learned went beyond their own design successes or failures on the assignment. It gave the students a new way to think, to try to derive an affordance from the manipulation of constraints. And that is what we were after in the Learning Design Studio but found difficult to explain to students in terms of classroom situations with which they were familiar.

Other Issues Requiring Attention

Apart from physical classroom conditions and students' entry level behaviors, we anticipated five major issues in teaching the course: fostering persistence, managing students' discomfort with ambiguity, keeping student focus on multiple criteria concurrently rather than paring away criteria to be manipulated in isolation, preventing students from falling back on trite models of the domain, and avoiding problems associated with a "personality cult." We take each of these in turn.

Students in the Learning Design Studio formed their project teams around their individual schedules, such that each team had some possible out-of-class meeting time. This ensured at least some meetings in addition to regularly scheduled class time (which was itself partially devoted to group meetings). However, even for projects, such as those in the Learning Design Studio courses, that did not have a dedicated permanent and continuously accessible space, wherever a group of students meets to work is really owned by the project. So what could we provide, in addition to scattered physical meetings, that would accommodate varying schedules and still promote that influence? It just so happens that many benefits of co-location and persistence were achieved by students' use of a Web application developed at SRI and called the Gallery, Organizer, and Repository of Projects (Gorp; see Chapter 13).

Some class sessions were conducted by posting designs on Gorp. And Gorp provided a persistence of display and access to designs that students uploaded, and allowed others participating in the TRAILS courses across the country to take the designs for a spin and then critique them. Gorp helped to solve groups' difficulties in arriving at a consensus, or developing an ethos, toward shared criteria for good design in learning technology. Course brainstorms by combinations of information sciences and education students also yielded a list of various criteria for good designs, and we used Gorp to include these criteria as fields that had to be filled in for each design iteration in the project.

Managing the discomfort of ambiguity was largely accomplished through setting expectations. In introducing the course, Chris Hoadley placed in the syllabus a lengthy disclaimer that students could not expect single, correct answers and that they could expect to be more at ease with this fact by the end of the semester. In addition, students used to having a great degree of control over their grades were concerned by having vaguely defined criteria for success. This was helped by giving students formal (rather than solely informal) intermediate feedback on projects and ample opportunities for extra credit, so that they felt they could improve their grades in ways other than by "getting an A on the project."

To help students focus on multiple criteria, the flexibility of feedback from multiple instructors proved to be useful. As mentioned earlier, students helped construct a shared set of criteria of good design. This ran the danger of being merely a "checklist." Accordingly different criticisms, aimed at different criteria, levels of analysis, and aspects of the designs, were raised during the semester. As both Chris and a teaching assistant (in one case, Charlie Cox; in one case, Paula Bach) had background in the field and could provide feedback, students learned that their design might excel in one dimension of the criteria while simultaneously falling short in another. Another technique used to help students focus on multiple criteria was the formation of design teams around different predilections of individual students within a group, ranging from "project management" to "graphic design" to "programming." By having teammates to negotiate with, students were forced to confront multiple assessments of their designs.

On the subject of eliminating trite models of design, of technology, or of education, let us first consider that it is the rare individual who, dissatisfied with the perceived banality of automobile variety, manufactures a novel one. Why would an engineer "reinvent the wheel" (to resort to a cliché in describing the tendency to resort to clichés) when the well-known form is already available? Likewise, why wouldn't a student repackage a solution already known to be successful? In order to head off complacence of this nature, Chris introduced the concepts not only of revolutionary design but of evolutionary design as well. It was not a requirement of the course to propose new paradigms of structure, function, or behavior as solutions for every project. However, it was necessary at least to take an existing design and propose incremental improvements, as long as those could be supported with some evidence of thoughtful testing such as allowing access to the formative design to potential users and observing their reactions.

With regard to a personality cult, let us first consider that from the vantage of a student, an instructor can be seen as possessing some desirable expert knowledge that he or she passes along. In the case wherein such knowledge is perceived to be exotic or esoteric (as design might be, when compared by novice students to a more comfortably straightforward subject), then imitation of the instructor can be interpreted as a satisficing strategy for passing the course. As it happens, this was one issue that didn't manifest itself, so we never had to resolve it.

"Wrecking the Train" As a Strategy

When the editors of this volume first read our chapter's content, they asked if "train wrecks" might be architectural jargon. We submit to the audience that the underlying concept of disequilibration or a disorienting dilemma that causes one to question foundational beliefs heretofore considered axiomatic is indicative of upheaval in any domain (e.g., see Mezirow's perspective transformation as learning, 1978). That is, no, train wrecks are not peculiar to architecture, and, yes, they are examples of attention getters, albeit of the more grotesque sort.

Attention getting is fairly common instructional design practice, an emphasis identified by theorists such as Gagne (1965) and Keller (1983). However, the question of how to get students to notice incidents of what they're going to be learning about, to bring previous designs to their attention, and to get them interested in interpreting a designer's intent is not answered simply. Then, following that attention capture, students have to be given the vocabulary to describe what they attend and the encouragement to sketch what they intend (whether by hand in two dimensions or by computer-aided drafting or through a three-dimensional model). Follow that with scaffolded practice in designing, itself, and the necessary feedback until an expert is no longer required to stand by them during the process, and you have a skeletal overview of a Learning Design Studio course.

Keeping in mind the list of students' entry level behaviors that we reviewed previously, one might derive from it why teaching design is depicted here as the wrecking not of a single train alone but of several, à la Charles Addams. And the first eye-grabbing collision concerns why certain annoying designs just, well, suck.

Irritation is easy both to recognize and to recall, so making the discourse available and familiar is accommodated by applying it first to what is disliked about an existing design. Dislike is expressible even to those who can do no more than point and grimace (the reaction to a train wreck), so one starts by pointing and stimulating others to point. The instructor's follow-up is demonstrating or otherwise modeling how designers go beyond the pronouncement of merely good or bad, like or dislike, and really begin to critique what makes a flaw and then what makes it noticeable, being careful to avoid pronouncements of absolute good or bad.

Gauldie (1969, p. 16) suggests a taxonomy of ugliness based on a correlation of how disturbed by an observed object the observer is with the observer's perception as to what degree the object is "inimical to well-being." When a novice observer is given the criteria to sort an object as confusing, or monstrous, or ungovernable, as Gauldie does, this introduces

a discourse that could be used by expert designers to communicate their perceptions about design with one another. As a result, there is then a foundation for students to analyze the product of design and speculate about a design process that would lead to the observed object as such a product.

Chris Hoadley (Hoadley & Cox, 2005; also see Chris' chapter, this volume) created a design ontology to make the presentation and internalization of discourse more orderly, facilitating students' fashioning of schema related to design. Among other features, the ontology gave students categories to be used in sorting aspects of design processes such as stages, roles, values, principles, and patterns. To have a group of collocated (even if only in virtual space through Gorp) students trying on new ideas and being able to name those and categorize them for the first time in their lives is an exciting social exchange. Getting the fit right requires being able to talk about the concepts with other people, in this case sharing an annoyance and being able to describe it as never before, such that others concur or debate about the nature of the annoyance.

User testing was another benefit from the Learning Design Studio's method that was found both disconcerting and revelatory to the students. This is because their prior knowledge of design was exactly the sort that produces annoyance and irritation in a user: the result of the designer's conflation of his or her own satisfaction regarding a product with the satisfaction of a user. Put crudely in train wreck terms, if you think you have a solution and your user testing says you haven't, then you need to take another track because the one you were on isn't negotiable any longer. Or, as this student said:

> Two weeks ago at work, another student (not IST, an engineer) was implementing Microsoft SharePoint...I made a suggestion to "clean up" the interface by removing some of the unneeded items that were placed along the right hand of the screen. I also suggested having only necessary icons on the "home page" so that users would not get confused when using it. Before this class, I would not have seen it that way! I also learned a lot about user testing. I never really thought of it before this class. I figured, design something; if it works for you and your friends—cool—it's done. That's not the case at all; you must do user testing to see how they will use the system.

Iteration is usually seen as a wreck because it disrupts the interpretation of efficiency as being the sooner one arrives at a single direction, the less resources will be spent in getting to the end. Maybe that interpretation has some utility for analysis, but not really for design. If, as in design, the end hasn't existed before, then there could be multiple paths to equally

acceptable solutions, none of which is inherently superior to another. Here the studio concept beneficial to novice students is to practice the communication of intent again and again to accelerate the ability to iterate, whether through sketching or other prototyping involving variable fidelity to the final product. In the words of one, "I learned that iteration is more than just a word that sounds impressive. It is the most efficient way to output the highest quality idea."

Tactical Pedagogy in the Learning Design Studio

When it became apparent from time to time that projects were stagnating, charrettes appeared as short one- or two-day bursts of design activity on a matter related to the main project but addressing a much smaller scale, to bring new inspiration to a group, prodding iterations in different directions. The charrettes we used were groups designing for a shared problem, usually in a short period of time and so more intensively than for the larger longer projects.

By sprinkling charrettes into the Learning Design Studio when needed, we were able to focus on particular elements of the designing. For instance, to reinforce the need to develop comprehensive design descriptions, we had students describe the elements of the interface of an alarm clock in great detail and then later revisit the task with an eye toward their shared criteria, developed in the course.

That leads to the next train wreck and beneficial outcome: pinning up work for others to see. Secrecy born of competition is difficult to overcome in normative grading, even in light of the praise given to collaborative learning and repeatedly extolled to the students.

This is not unexpected, as design as a collaborative process doesn't seem to be broached in many curricula at either the secondary or higher educational level. Studios tend to replace this narrowly aimed and internecine competition with increased breadth of production as in following more than one path of investigation concurrently, with user testing to validate the direction taken.

One way whereby Chris Hoadley overcame this habituated reticence was by importing "speed dating" exercises. Given only enough time to make an "elevator talk" (a presentation given during an elevator ride), circulating groups peddled their designs to other stationary groups and listened to a reciprocal presentation before moving on. In effect, this was a dynamic version of the crit pin-up technique, wherein students present their work to date for comments from one another.

There were other more formal group presentations to the class as a whole, and sometimes to outside experts who had acted as clients. And the

tenor of these crits, being an experience without baggage to the Learning Design Studio participants, was refreshingly civil, especially as seen from the perspective of architectural crit memories.

Projects tended toward creation of interfaces for tools to accelerate or ease the performance of a given instructional function. When the College of Education graduate students were part of a team, there were few enough of them to assign one to each group of IST undergraduates, with the graduate student acting as a "chief learning officer" charged with informing the project with a learning theory basis. Furthermore, because the technology involved in the applications was meant to be educational, the users for whom students were designing were, themselves, other students. As opposed to the more general concept of user-centered design, the problem became specifically one of learner-centered design.

In that case, the project happened to be development of the interface of an application for supplemental instruction in math or science, with grade-school children as the target audience.

During the second course iteration, when only IST students participated, the problems addressed were less pedagogical and more aimed at learning life skills through HCI, such as helping the visually impaired to read the signage they might encounter in their environment. But the tools, themselves, were not the only focus of the course and, as such, were not developed much beyond the interface of tool with user. As stated previously, what was supposed to share primary interest to students were the discovery and development of their own design processes, especially as those affected probable users of the design products they were creating.

How the Learning Design Studio Afforded Reflection

The Learning Design Studio aimed to let students introduce themselves to their own processes for designing and then to explicate the pieces for themselves through reflection. Thus, recognition and continued awareness of one's individual process and an ability to manipulate that process held higher priority than any specific problem solution. Chris Hoadley introduced the course by saying, "I can't teach you to design in a semester, but I can teach you how to reflect on your designing and listen to the data. If you do this, you'll be able to become a great designer on your own long after you've left Penn State."

This led in turn to a definition of design emerging from process-oriented strategies: the search for a problem's structure, identifying and inquiring about users and other stakeholders, concurrent pursuit of multiple paths toward solutions, and iterative refinement along those paths. Of course, these were accompanied by the anxious discovery that

linearity was unlikely to appear in the process. It cannot be overstated how foreign this way of working is for most engineering and technology students, and indeed, for undergraduates generally. Even the education graduate students found the course somewhat disorienting in that there was neither a concise body of knowledge to master nor individual latitude to synthesize and opine; instead, everything—design outcomes, processes, even individual goals and requirements—had to be negotiated.

Students responded favorably to this approach. One commented,

> During this project we had two class days when half the class went to the library and [the] other half attended the class for a small, one-day paper. This is when it first dawned on me. This small paper required our group to think out our own design process. After writing this out it was simple to see that our group has developed a process specific to our train of thought. From now on, I had an official design process. Project two started with our group writing out our design process and working on step one... I am glad to say that I enjoyed learning about interface design to the extent that I had my position changed for my internship. I will be working on the research and testing part of integration for the first time and I am pretty excited to be able to put what I have learned in this class to real world application.

Conclusions, Being Mostly Favorable

As stated earlier, the examples laid out here do not comprise an argument for how design ought to be taught. If the audience feels differently about design processes and products, then a shift is encouraged with blessings for better alignment.

What the Learning Design Studio demonstrated was that technology design courses, in this case aimed at technologies for learning, could draw inspiration from studios in a successful manner. This involved the identification of poor design (and providing justification for such an evaluation), introducing discourse and an ontology of design practices, practicing rapid communication of intent as part of a dialog with others, user testing, iteration, and reflection on the accomplished process in order to inform the next performance.

To be sure, there was a lot of prerequisite groundwork such as finding substitutes for the studio affordances that didn't transfer to our classrooms and, following that, some possibly anxiety-producing conflicts for students. However, once some comfort with the ambiguous and collaborative aspects of a design process was achieved, students responded by applying their discovered abilities toward developing more expert performances.

How this performance development directly affected the design of learning technologies is, as yet, a result both ambiguous and deserving of iteration. It's ambiguous because maybe what students learned is better than what they would have otherwise, or maybe it was just different. It deserves iteration because the results were promising and indicated enthusiasm on the part of students. The real test, of course, will be when users actually get to try the products of these design processes, as when one of the Learning Design Studio classes was able to elicit testing and critique by public school students.

Bibliography

Barrows, H. (1986). A taxonomy of problem-based learning methods. *Medical Education*, *20*(6), 481–486.

Budgen, D. (1995). "Design models" from software design methods. *Design Studies*, *16*(3), 293–325.

Clark, R., & Estes, F. (1998) Technology or craft: What are we doing? *Educational Technology*, *38*(5), 5–11.

Dannels, D. (2005). Performing tribal rituals: A genre analysis of "crits" in design studios. *Communication Education*, *54*(2), 136–160.

Gagné, R. (1965). *The conditions of learning*. New York: Holt, Rinehart and Winston.

Gauldie, S. (1969). *Architecture*. London: Oxford University Press.

Hazzan, O., & Karni, E. (2006). Similarities and differences in the academic education of software engineering and architectural design professionals. *International Journal of Technology and Design Education*, *16*, 285–306.

Hmelo-Silver, C. (2004). Problem-based learning: What and how do students learn? *Educational Psychology Review*, *16*(3), 235–266.

Hoadley, C., & Cox, C. (2005). Educating reflective learner centered designers. *Proceedings of ED-MEDIA: World Conference on Educational Multimedia, Hypermedia & Telecommunications 2005*. Montreal: Association for the Advancement of Computing in Education.

Keller, J. (1983). Motivational design of instruction. In C. Reigeluth (Ed.), *Instructional design theories and models: An overview of their current status* (pp. 386–434). Hillsdale, NJ: Erlbaum.

Kuhn, S. (1998). The software design studio: An exploration. *IEEE Software*, *15*(2), 65–71.

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

Mezirow, J. (1978). Perspective transformation. *Adult Education*, *28*(2), 100–109.

Norman, D. (2002). *The design of everyday things*. New York: Basic Books.

Stead, N. (2003). *Producing critical thinkers, designing critical objects: Re-examining the role of critique in architectural education*. Paper presented at the Second International Conference of the Association of Architectural Schools of Australasia, September, Melbourne, Australia.

Willenbrock, L. (1991). An undergraduate voice in architectural education. In T. Dutton (Ed.), *Voices in architectural education: Cultural politics and pedagogy* (pp. 97–120). New York: Bergin & Garvey.

Wood, P. (2003). *The critique: An investigation into the critical role of the design critique in architectural education*. Paper presented at the Second International Conference of the Association of Architectural Schools of Australasia, September Melbourne, Australia.

Chapter 9

A Learning Technology Design Course, Deconstructed

Chris Quintana

Introduction: Teaching and Learning Design

Authentic design activity is a complex enterprise that entails a range of analytical and constructive activities spanning different domains (Lawson, 2006). These activities are multifaceted and open-ended, which can pose difficulties for students (Quintana, Eng, Carra, Wu, & Soloway, 1999). Designing learning technologies is complex because of the need to integrate design, learning, and content issues (Quintana, Soloway, & Krajcik, 2003). For example, typical design activities involve:

- Analytical activities to develop technology requirements, specifications, and constraints. This involves determining the audience for the technology, the learning goals and activities for that audience, and the types of necessary learning support.
- Conceptual activities to develop effective solutions for the learner audience by using descriptive tools such as scenarios, storyboards, and prototypes to describe the design.
- Implementation activities to turn the conceptual design into a tangible product. This includes iteration through activities of analysis, conceptualization, and implementation to refine the technology.

Furthermore, collaboration is needed in the design process between a range of parties, including designers, programmers, content area experts, teachers, and even learners themselves. Each team member brings unique theoretical backgrounds, areas of expertise, and content knowledge to the problem, making it challenging to develop a shared understanding between the parties and synthesize their knowledge.

The complexity of design activity directly impacts the way learners can learn design, especially if we consider pedagogical approaches where learners learn by engaging in substantive design projects. Project-based pedagogies are well established in engineering, computing, and artistic disciplines, and their use has recently been explored in other areas such as