4

DISCIPLINE-SPECIFIC INSTRUCTION IN TECHNICAL COMMUNICATION

W. J. Williamson
Philip Sweany

This chapter explores possibilities for pedagogical innovation offered by interdisciplinary ventures in professional development. Introductory technical communication courses offer their instructors and host departments opportunities to link conceptually and pedagogically with representatives from disciplines across campus. Too often, however, such opportunities are not acted upon. In this chapter, we discuss our participation in a two-year project that linked technical communication and computer science courses for students entering computer-related professions. We discuss the impetus for this venture, describe the linkage of courses that resulted, and explore the advantages, disadvantages, and challenges posed by the venture. We close the chapter by offering advice to others who may be interested in launching similar projects or redefining whole programs with an interdisciplinary focus.

Our final recommendations focus on two key elements of the problems posed by interdisciplinary collaboration: identifying institutional constraints and potential sources of support and engaging in thoughtful reflection on the limitations and expertise of the faculty who hope to bring such projects to fruition. We assume that most interdisciplinary ventures will, like ours, be limited to revising specific courses. However, the project of reimagining courses leads necessarily to consideration of full curricula and opens up at least the possibility for transforming entire programs.

THE IMPETUS TOWARD INTERDISCIPLINARY PEDAGOGY

Courses that introduce students to the fundamentals of technical or professional communication are staples of college-level curricula. Whether such courses are required components of a general education curriculum or offered as electives, they have two characteristics in common: many students do not look forward to taking them, and many faculty do not look forward to teaching them. We foster a different attitude toward the course. We see the introductory course in technical communication as an ideal site for connecting disciplines and emphasizing professional development.

Introductory courses in technical communication vary in content and strategy, but at their core, they focus in some way on the communication of information related to scientific, technical, or other special knowledge areas. Communicating information has been a perennial struggle for scientific and technical professionals. Likewise, science and technology are not the typical centers of expertise for disciplines strong in communication. Simple logic suggests that some mutual growth may result from any collaboration between these groups. Nonetheless, simple logic in this case defies institutional tradition. We believed that an interdisciplinary linkage of some sort could help students better understand both computer science and technical communication. And in the process of developing an interdisciplinary classroom venture, we hoped to learn more about our disciplinary differences and commonalities as well, especially with regard to pedagogy. As we show later in this chapter, interdisciplinarity emerged in this project in many ways, including the initial course design, the selection of a course project, the execution of our course design and, to some extent, the makeup of our project teams in the course itself. And although we recognize that the core concept of offering a discipline-specific course of this kind is not new, even at Michigan Tech's campus where we were both teaching at the time, we would argue that deep-rooted divisions among colleges, departments, and disciplines, coupled with short institutional memory for pedagogical innovation, makes projects such as ours innovative.

We agreed from the beginning that a course linkage between computer science and technical communication presented an opportunity to push students toward thinking about themselves less as students and more as developing professionals. Emphasizing professional development became our conceptual link between the subject matter of our courses. We argued that students may be able to succeed by compartmentalizing their knowledge into discrete structures (such as classes, tests), but working professionals could not. (Of course, we attempted to challenge the myth of separation in the academy as well, but that attempt is a discussion worthy of its own narrative.) The emphasis on professional development emerged in the way we presented the course to students, drew discussions back to issues of responsibility, responded to their work, and involved students in a project that asked them to serve a real client. Throughout, we addressed students as developing professionals and, in doing so, asked them to raise their expectations of themselves and their classmates to a professional level.

The time and energy we invested in this project was in part an attempt to address the negative attitudes students bring to their work in introductory technical communication classes. To some extent, we were successful in altering those attitudes. Students responded favorably in course evaluations, suggesting that we continue the practice of linking computer science to technical communication. Although many students commented that the linkage finally made the introductory course in technical communication relevant and worthwhile (and did not recognize as readily the importance of technical communication to their work in computer science), some students did understand and appreciate the full impact of our pedagogical venture. To understand the students' preconceived notions about technical communication, we need to frame the introductory courses institutionally.

Introduction to Technical Communication at Michigan Tech

The Department of Humanities at Michigan Tech typically offers six to eight sections of Introduction to Technical Communication (Humanities 333) each quarter. Although a few sections are taught by tenure-track faculty, most sections are taught by graduate teaching assistants who have demonstrated classroom expertise in first-year writing and who have some interest in teaching technical communication. Sections tend to fill quickly to their twenty-five- seat capacity due to general campus demand; most majors on MTU's campus require the Introduction to Technical Communication course, and all majors can take the course as an elective. MTU students typically enroll in the university's general education technical communication course during the final year of their undergraduate curricula, when they are concurrently enrolled in upper-division courses in their majors. Students are not usually encouraged to overlap these experiences in any way. As a result, the introductory technical communication course is often viewed as the final writing course that students need "to get out of the way" before they can graduate.

The course is housed in Michigan Tech's Department of Humanities, which includes faculty from a variety of disciplines, including rhetoric, technical communication, literature, composition, philosophy, and modern languages. The department's disposition toward Humanities 333 has varied somewhat in recent history. At times, the course has been considered little more than a service course in the most limiting sense. At other times the course has generated interest from faculty and graduate students alike as a site for interdisciplinary investigation of communication practice. For a time, MTU's undergraduate Scientific and Technical Communication program was directed by a faculty member whose expertise lay outside traditional technical communication studies. In many ways, this administrator's direction introduced invigorating and transformational ideas into the overall conception of the undergraduate program. However, this administrator also questioned the significance of investing faculty time into supporting campuswide, service-oriented courses in technical communication. As a result, the Introduction to Technical Communication course was not formally administered by any subgroup within the department. Although this hands-off policy changed under the guidance of the following administrator, several years later the department still feels the lingering effects of that decision and has been forced to retool the course because it has been reclaimed into the department's teaching culture.

Despite documented early successes in writing-across-the-disciplines initiatives at Michigan Tech (see Young and Fulwiler 1986, ; Elizabeth Flynn et al. 1990), Humanities 333 has always been powerfully influenced (not always positively) by pedagogical tradition and institutionally imposed limitations. For example, due to the high concentration of engineering majors enrolled in these courses, students tend to address engineering concerns more than those of any other discipline on campus. Nevertheless, engineering and science faculty at Michigan Tech have expressed concern that Humanities 333 is too broadly defined, asking for courses that serve the particular needs of specific disciplines. Our interdisciplinary venture grew out of an initiative from computer science faculty who shared such concerns.

In the next part of our discussion, we outline the moves that took us toward establishing curricular links between computer science and technical communication at MTU.

A Conceptual Framework for Course Linkages

Our interdisciplinary venture was based on the assumption that linking introductory courses in technical communication directly to the professional development programs of other curricula seems a logical move toward more effective pedagogy and more fruitful relationships among disciplines and departments. We also felt that this linkage had potential for addressing more-general workplace concerns about communication abilities among college graduates: by linking communication to disciplinary content, we hoped to spark greater interest in the subject matter and to make it more central to professional development in computer science.

Certain evidence suggests that such curricular relationships have been beneficial for both students and faculty. For example, at Georgia Tech, mechanical engineering students are introduced to technical communication by writing faculty, who teach communication within the context of engineering courses (Donnell, Petraglia-Bahri, and Gable 1999). At the undergraduate level, technical communication is taught in four courses; the curriculum design emphasizes the simultaneous development of communication ability and professional engineering expertise (114). At the graduate level, students are returned to a novice level as professional communicators, at least with respect to their transition to the role of researchers; communication is linked explicitly to professional development in a series of courses and seminars (115–116). At Worcester Polytechnic Institute, John Trimbur (1997) teaches a course called "Writing about Disease and Public Health" to students from primarily biological and technical communication backgrounds. In that course, Trimbur asks students to evaluate the transformation of medical information in its travels from the academy to public spheres. For Trimbur and his students, understanding the delivery of information is critical to professional activity and responsibility in biological professions. The course emphasizes the awareness of professional responsibility in the presentation of information to a variety of audiences. Although these arrangements do not involve specific course linkages, as ours does, the overall motivation and impact for each course is much the same, emphasizing communication in the context of professional development.

There are good examples of course linkages as well. Brian Turner and Judith Kearns, for instance, describe the linkage between a first-year composition course and a first-year history course at the University of Winnipeg (Turner and Kearns 1996). This linkage was designed to help students adjust to the rhetorical demands of college-level writing and to help them understand the content of their history course (5). Dennis Lynch (1997), a Michigan Tech colleague, is involved with another first-year course linkage, this one between composition and biology. Among the goals of the linked courses, Lynch identifies the study of how biologists write and communicate and what it means to become a biologist (163). In each of these course linkages, as with the examples of the conceptual and pedagogical links discussed previously, there is a sense and a hope that by introducing interdisciplinarity into science, history, engineering, and communication instruction, there will result a greater understanding of each element. In effect, the hope is that the whole really will be greater than the sum of its parts.

Among the efforts we highlight here, the consistent element to developing interdisciplinary linkages is the emphasis on professional development. That emphasis is central to our work as well. From the first, we felt that when framed as courses in professional development rather than as a series of exercises about writing forms or the mechanistic development of text, technical communication courses can become explorations of what it means to become a responsible, practicing professional. Thus, we presented communication as the social medium within which products such as software packages develop. This richer focus on communication as a social activity is likely to be more interesting and useful to people across the disciplines. It is also likely to be more rewarding for students. In the context of computer science, course linkages open opportunities to discuss the relationships in professional life among communication, product development, and customer service. We can open up opportunities to engage faculty and students in discussions about responsibility in professional life, including the ways communication knowledge provides a medium for other professional responsibility. These courses also provide opportunities for teachers to talk about the differing roles professionals may play with regard to communication, including the different ways engineering, science, business, service, or communication professionals might deal with the same body of information in different contexts.

The theme of professional development drove us to design a seamless curriculum whenever possible; that is, we wanted our two courses to feel like a single course. Therefore, instead of treating the courses as two separate but linked courses, we developed the curriculum as a comprehensive, six-credit instructional unit. In the early going, we both met with both classes. Although we maintained final instructional authority in our respective courses, we approached their design as if we were developing a single course. As a result, computer science and communication interests and issues were as fully articulated as we could make them across the two courses. We were driven in our general design by two concerns: (1) that we foreground the importance of communication in both courses, rather than encourage students to separate computing and communication; and (2) that we provide students with a "real" project that would challenge them in ways that an invented project would not. We drew on the guidelines for successful writing-across-the-curriculum initiatives offered in Toby Fulwiler's (1991) "The Quiet and Insistent Revolution." We engaged collaborative learning groups in open-ended assignments that posed real-world challenges. We addressed student writing as managers rather than as teachers, offering guidance rather than grade-oriented commentary. We shared our values as communicators, researchers, and educators by discussing our pedagogical and research goals directly with students (183–185). We also looked to service-learning scholarship for assistance in drafting our specific project goals. For example, in "Technical Communication and Service Learning," Randy Brooks (1995) suggests that "the most valuable service learning includes reciprocity of outcomes: (1) the doing helps the community solve problems or address needs, and (2) the thinking helps the student develop disciplinary skills, community responsibility *(ethos),* awareness of cultural diversity through the integration of theory and practice" (12).

Our planning culminated in the pilot course linkage, offered in the fall quarter of 1997. In the next section, we describe that venture in greater detail.

SOFTWARE DEVELOPMENT IN LINKED COMPUTER SCIENCE AND HUMANITIES COURSES

During the spring of 1997, Philip Sweany, who was a member of MTU's computer science department at the time, initiated discussion about a curriculum revision regarding technical communication instruction. In the wake of those discussions, we began developing a course linkage that brought together the introductory course in technical communication with an upper-division software management course. Students enrolled in the linked courses worked to develop educational software for a local middle-school math class. In one ten-week quarter, we asked students to design, develop, and document a prototype software package suitable for early- development field testing. Students worked in teams of four to six people to complete several project-related items:

software package
user manual
design document (software design proposal)
functional description (description of software capabilities)
documentation plan
software testing documents
software maintenance plan
several progress reports

Only a few of the project-related tasks were completed for exclusive credit in one course or the other. Most course products received developmental feedback and grades from both instructors. When we responded to projects, especially written documents, we tried to treat the encounters as shop scenarios; that is, we responded to students' work less as teachers and more as managers. And in keeping with the shop atmosphere, students showcased their work in a "software fair" held in an open computing facility at the end of the quarter.

We prepared students as well as possible within a tight time frame to undertake their projects. As part of the learning and planning process, we asked students to evaluate existing educational software packages and their accompanying documentation. During the early part of the quarter, we connected students with subject matter experts from three areas: educational software design, small-group dynamics, and middle-school mathematics pedagogy. A faculty member from MTU's education department led a series of discussions on educational software design, including special focus sessions on developing interactivity and appealing to young audiences. A graduate student from MTU's humanities department led a series of discussions on small-group dynamics, including sessions on roles and role-playing and conflict management. Throughout the quarter, students were in contact with the math teacher who served as the project client via the World Wide Web and email. We gathered questions from students about her teaching methods, the teaching curriculum she used in her classes, and her students' experience with computers. She responded by posting information to a special section of her Web site devoted to the project.

Throughout the quarter, project-related discussions spanned issues in software and interface design, teaching and learning strategies, usability testing, and document design. We encouraged students to engage the theories they encountered in class discussions and in professional literature and, whenever possible, to extend those theories to their work. Although the software design project was the centerpiece of this linked curriculum, students participated in a variety of discussions and assignments that helped them develop the expertise they needed to complete their work.

Outcomes of the Software Design Projects

Because we did not initially have a long-term commitment to the project from our department-level administrators, we did not think it wise to plan to make the class projects an ongoing effort. That is, we insisted that project groups attempt to develop, test, and deliver complete educational software packages in eleven weeks. Given this tight time frame for software development, students were forced to focus on simple design plans and to establish relatively modest goals for the complexity of interactivity they designed into their software and manuals.

Predictably, students' attempts to design software for a middle-school audience were either too simplistic or too complex. Michigan Tech students (especially those enrolled in scientific and technological disciplines) tend to be adept at solving math problems. It was difficult for them to understand the relative simplicity of middle-school math curricula. Nevertheless, most groups managed to develop projects that the client said she would be willing to field test. Most of the software packages that resulted from that first run of the linked courses focused on presenting students with information from the curriculum and then quizzing them on the knowledge they were supposed to gain. In preliminary, informal field testing, students in the client's classes were often overwhelmed with the complexity of the problems presented in students' software. However, they responded favorably to the idea of integrating computers into their curriculum for any purpose.

One project group continued its work in an independent study that extended beyond the initial quarter. That group developed a game, *Lemonade Stand,* that presented users with the problem of maintaining a successful beverage stand in the face of daily weather changes and fickle market demand for their product. To add an element of drama, the project group allowed users to avoid natural and social disasters (such as tornados and bullies) by solving additional math problems. This project group focused more than others on interface design and developing interactivity.

The user manuals that resulted from the software development projects were in some ways better developed than the packages themselves and with good reason. In a sense, the documentation drove the development of the software. Project groups submitted detailed written plans for their software and manuals well in advance of the completed projects. Those plans formed the basis for all of their future development. Because of the tight time frame, groups tended to stick to their original designs whenever possible so they would not have to go back and rework their documentation to fit new project developments. Thus, project groups kept to their general plans, focusing their problem-solving innovations on methods for accomplishing their original design goals. In the end, there were still last minute software changes not properly documented, but these instances were few. Overall, the user manuals were targeted more appropriately to the focus audience than the software packages themselves. Although students tended to assume that their audiences would find the software intuitive in its design, they also responded well in general to our suggestions for developing the documentation further.

Perhaps more significant than student successes in terms of their learning experience, however, was their recognition that they had all misjudged their audience in some way. The software fair was a showcase for what students learned and what they could have done, as much as it was an opportunity for students to discuss the software they actually developed. Few students were satisfied with what they had accomplished during the quarter, although most believed that their experience was not adequately represented in their final products. Unfortunately, MTU's quarter system left us with little opportunity to follow up on and encourage detailed reflection from software groups once the course ended. For many students, we could only hope that we had established a solid groundwork for such reflection and that they would engage in that process independent of their classroom experience.

We stated earlier that we believed students would benefit more from both their communication and software design courses if they were able to make connections between courses previously disconnected and to work with a real audience and purpose. And as hoped, students understood the relationship between their technical and communication responsibilities more fully in both classes than either of us had experienced prior to linking them. For example, in previous software development courses, Sweany observed discussions among groups primarily about technical issues in software development. In our linked courses, project groups discussed meeting user needs as much as they discussed solving technical problems. They argued about the logistics and merits of usability testing for both the software and its accompanying documentation. Teams in general were more concerned from the beginning about the way all the elements of the problem affected software design decisions (including user needs, information design, content, teaching strategies, learning styles).

In the end-of-the-term evaluations, students expressed a general support for the course linkage. In addition to receiving good marks in the numerically based evaluations, the course received positive qualitative comments:

> The format provided an interesting and relevant way to learn the material. It gave everything a sense of connection and direction.

> I would definitely recommend this course. I would warn them that it does require a lot of work, but that they will learn a lot.

> During the first week of classes, I considered dropping. It seemed that the work would be hell. At times I came close but I'm glad I stuck it out. I found that the project was fun and a good learning experience.

These early successes not only gave us many things to think about but also left us hopeful that we were on the right track, working toward a useful, productive goal that satisfied students, administrators, and our colleagues in teaching.

Despite rich possibilities for professional and curricular success, we still faced political and financial challenges in our endeavor to reinvent the introductory technical communication service course at MTU. We found that institutional momentum does not easily shift to support customized interdisciplinary education. In the next section, we reflect on some of the advantages, disadvantages, and challenges to establishing linked, interdisciplinary courses.

ADVANTAGES, DISADVANTAGES, AND CHALLENGES OF INTERDISCIPLINARY PEDAGOGY

Although we believe this interdisciplinary course linkage was generally successful, it is nonetheless a venture rich with complications. In this section, we address some of the advantages, disadvantages, and challenges we have come to connect with our efforts. Although we believe that this interdisciplinary effort ought to be adopted more widely, we understand why such curricular innovations, without careful planning and energetic self-promotion in the early stages, are likely to remain isolated experiments in pedagogical design. Because we are aware of the benefits as well as the challenges for implementing this curriculum, we address the issue in our final recommendations.

Advantages of Interdisciplinary Pedagogy

Our initial attempt to link courses in computer science and technical communication was met by students with increased commitment to connecting communication and computer science in their thinking about professional development. We have gathered feedback through a variety of means, including standard course evaluations from each course, anonymous questionnaires, informal interviews with students, and word of mouth. And some students have gone on to use their course projects as professional portfolio material in the job market. The linked-course project has begun to acquire a favorable reputation among first- and second-year students, many of whom now look forward to participating in the project. From the outset, we focused on four aspects of the course linkage we felt would most likely serve student interests and needs: the advantage of a comprehensive curriculum; professionally contextualized communication instruction; real-client motivation; and interdisciplinary exchange between faculty.

Comprehensive Curriculum Design

We have argued here that the linkage of courses in other disciplines to the introductory technical communication course could in the best of circumstances result in a comprehensive learning experience for students. Further, linked courses offer an opportunity to focus in greater depth on the relationships among subject matter expertise, communication activities, and responsibility against a backdrop of professional development. We still see this advantage as the overarching benefit for such curricular innovation.

Professionally Contextualized Communication Instruction

A consequence of linked courses is the opportunity to contextualize communication instruction with the acquisition of subject matter expertise in scientific and technological disciplines. This contextualization is more than a side effect of such pedagogical efforts; it is a real advantage to be pursued in course design and execution. We agree with students that contextualized communication instruction is far more relevant to their professional development than a course offered to a general audience of students who represent a random sampling of campus disciplines and departments. When instructors have the luxury of making technical communication courses immediately relevant to students, there is far greater potential for meeting everyone's expectations in the course.

Real-Client and Real-Project Motivation

Although it is not a direct requirement that a comprehensive, contextualized venture such as ours be anchored by a real project with a real client, we recognized immediately that the cumulative contact time gave us the opportunity to introduce this element to the courses with some hope of success. The advantage of serving a real client is an extension of the other circumstances of our venture; nonetheless, we would recommend that anyone interested in forging such linkages pursue the possibility of connecting to a real project. Again, this helps students contextualize their work more easily in a framework of professional development and offers them added motivation to succeed. A client will see their work in the end.

We recognized the extra effort required from our client to participate in a project of our devising. To benefit from the project, she had to invest time in helping students understand her needs and the needs of her students, as well as her curriculum structure and teaching methods, and she had to make time to test prospective software packages once they were ready for their audiences. We worked hard to ensure that her efforts were rewarded regardless of the success of our project. In return for our client's participation, we offered technical support for the delivery and use of the software developed by students, the promise of revisions to potentially lucrative software packages, and the possibility of new computers from future project funding.

Interdisciplinary Exchange between Faculty

Regardless of the classroom success or failure of a project such as this one, we feel strongly that one of the benefits of the venture, both for ourselves and students, is simple collegial exchange. To make clear our scholarly, pedagogical, and professional assumptions and goals, we had to understand them ourselves and be able to articulate them effectively. A significant element of our exchange revolved around tracing the intellectual legacy of our practices in response to the most elegant and confounding question of all: why? By going into our courses with a greater understanding of the values each of us held, we were able to anticipate how to best complement one another and when to use our professional differences constructively and creatively to spark discussion with and among students. Such pedagogical acts require trust that comes only from contextualized interdisciplinary discourse.

Although we have discussed these advantages already in this chapter, we have not addressed in any detail the drawbacks and challenges to interdisciplinary course linkages. We do that in the next two sections.

Disadvantages of Interdisciplinary Pedagogy

We would be remiss if we did not discuss the downside of interdisciplinary ventures such as the one in which we engaged. Although we discuss only two such disadvantages here, we round out the discussion by addressing challenges in the section that follows. Here we discuss problems of time investment in curriculum development and the negative consequences of extended student-teacher contact time.

Interdisciplinarity Demands Significant Time Investment

A course linkage such as this one demands a far greater investment of time than a standard university course preparation. Our discussions began several months prior to the first class meeting. During that time, we invested time in discussions about big-picture conceptual issues, logistical details, textbooks, guest speakers, assignment development, and other issues, including funding sources. Some of our efforts were supported by outside funding. Grants provided an alternate means of accounting for our time investment. Even so, this venture represents significant allocation of time above and beyond our typical course development.

Extended Student-Teacher Contact Can Have Negative Consequences

We already said that the extended contact time with students was one of the advantages of a comprehensive course linkage. However, that extended contact time can also become a disadvantage for both students and faculty. Students who struggle for any reason struggle in two courses, not one. If their struggles are confined to their peace of mind in the courses, then the disadvantage is personal to them. However, such is rarely the case, especially in a team-driven, project-oriented course. The course linkage represents a significant investment for students as well, and any problems are thus magnified. Time can be an opportunity or a prison.

Challenges of Interdisciplinary Pedagogy

In addition to the specific disadvantages of interdisciplinary course linkages, there are several obstacles to be overcome both to establish and maintain the venture. We classify them as challenges rather than disadvantages because they refer more to institutional difficulties than to pedagogical drawbacks. Each is also a call to respond innovatively to a dynamic professional context. We discuss five such challenges here: institutional obstacles; institutionalized disciplinarity; inconsistent rewards; funding; and sustaining commitment. After raising these challenges for consideration, we discuss several strategies for meeting such challenges and launching successful interdisciplinary teaching ventures.

Institutional Logistics Create Obstacles

Institutions do not always respond favorably to special limitations placed on course enrollments. Standard course prerequisites are easily accounted for by scheduling personnel and software, but our course linkage created problems for the campus registrar. The introductory course in technical communication is in high demand at MTU, and students from other disciplines were frustrated that some of the sections offered were closed to people outside computer science. The registrar's office did not share our enthusiasm for the linkage experiment and so resisted our efforts to have special limits placed on our sections.

Institutionalized Disciplinarity

Universities seem to be set up to encourage disciplinary isolationism, not interdisciplinarity, especially when working with undergraduate students. Students are required to take courses that range across several departments and disciplines as part of their studies, but rarely are individual courses cotaught by faculty from different departments or disciplines. We explored course linkages rather than a single cotaught course in part because we could not find a way to satisfy the university's need to assign credit for the course offerings to an individual faculty member.

Further, university faculty do not tend to encourage interdisciplinary thinking in their classes. Thus, students are not generally encouraged to build connections among courses from the complete range of disciplines they encounter during their studies. Institutionalized disciplinarity results in struggles to link courses and to get students to make connections between even similar subjects from different courses. Students are not trained to make interdisciplinary connections in their course work and thus struggle to do so when asked.

Inconsistent Rewards

Pedagogy is not scholastically significant in all disciplines. Rewards for pedagogical innovation are therefore uneven. Incentives for such activities are also uneven. Engineering and science disciplines do not value pedagogical research as highly as industrially connected forms of research. Consequently, interdisciplinarity may not be rewarded for those involved, if their tenure areas are narrowly defined. English faculty may not earn any returns for publications focused on pedagogy or on interdisciplinary studies. In some cases, this obstacle may be a significant barrier presenting real concerns, especially for tenure-track faculty.

Funding

Universities are not well set up to accommodate special allocations of resources. That is, when a special course is offered, such as Technical Writing for Computer Science Professionals, questions arise about who will accept financial responsibility for the course. Should responsibility remain with the department that hosts the faculty member or with the department whose special interests are being served? In our case, compromise was necessary. In university settings, funding allocations are highly political, and the issue of who pays is one best addressed early and with as much awareness of institutional politics as possible. We explored sources inside the university—through conversations with department chairs, deans of colleges, and upper-level administration—as well as outside sources for funding. Although we were able to secure sufficient support in the short run to launch the project, our results were mixed overall.

Sustaining Commitment

Sustaining interdisciplinary commitment to a project such as this one is difficult. Faculty in English or humanities departments may not feel qualified to teach general courses in technical communication. It is an added burden (and source of insecurity) to ask faculty to teach such courses to a single discipline. On the other side, not all members of technical or scientific departments are likely to support the interdisciplinary effort we embarked on. Neither are all faculty from either side of these relationships likely to see the benefit of contextualized communication instruction. Thus, personnel issues are a problem with any discipline-specific pedagogy because it is difficult to sustain commitment. Once a team commits to a venture, as we did, there may be no one else in either department interested in or willing to continue the project. Thus, a good idea, a successful venture may fail ultimately due to inconsistent commitments to the idea.

Despite the drawbacks and challenges we present here, we remain firmly committed to the idea of interdisciplinary course linkages. Thus, we provide some suggestions in the next section for increasing the likelihood of success in such ventures.

DEVELOPING SUCCESSFUL INTERDISCIPLINARY COURSE LINKAGES

Different strategies for sustaining interdisciplinary education will likely work better at different institutions, but there are a few things that we have found at Michigan Tech that seem to give us plenty of room for rethinking our relationships to other disciplines, while also rethinking the goals and strategies driving our introductory technical communication courses. To be of use to all stakeholders involved (students, faculty, administrators), there must be professional benefits from participating in the project. In rethinking the role of the technical communication service course for computer science majors, we engaged in several months of exploratory discussions, following a plan that might best be described by M. Jimmie Killingsworth (1997) as "pragmatic," focused in local, immediate, and serviceable needs (244–245). Although the specifics of our model may not work for all curricula, the process should be of interest to people looking for strategies to strengthen their present scientific, technological, or communication pedagogy, at either individual or programmatic levels.

In "Linking Communication and Software Design Courses for Professional Development in Computer Science," we (1999) attributed our success to seven strategies we adopted and that we think might be helpful to others who embark on similar ventures:

Plan curriculum development time
Plan faculty development time
Find a real client and project

Visit each other's classrooms
Plan an ongoing project cycle
Promote departmental consistency
Engage in vigorous self-promotion

Plan Curriculum Development Time

Prior to entering the classroom, we invested significant time to discussing our individual goals, project goals, and pedagogical values. We discussed external funding sources and possible project clients and how we might approach them. We outlined course syllabi, ideas for inviting guest speakers, discussion topics, and readings. We discussed the linkages between assignments, strategies for responding to student work, and creation of opportunities for encouraging client input. For our professional development, this time produced some of our most valuable and rewarding efforts. Our vision of the project emerged from hours of negotiations about the general approaches pedagogical and managerial to our courses. In the process of discussing what we could and would do together, we were forced to lay out our pedagogical theories and practices and work on crafting a compatible whole. Our discussions about the courses and the overall project helped us define and refine our teaching and researching roles and goals, including our personal, professional, and pedagogical goals, as well as our individual and collective measures of success.

Plan Faculty Development Time

We invested significant time early in the project developing shared expertise in a variety of project-related issues, including educational software design, service-learning design, Java programming, and collaboration. Each area played its part in preparing us to enter into the project as a teaching team. Critical in these discussions was the ongoing process of assignment expertise. We did not want students to feel as if any one person involved in the project had exclusive expertise; we wanted to present as much as possible that our ideas, goals, and expertise with different project elements were complementary and overlapping. For example, we did not want to project a sense that the technical writing activities were secondary to the success of the overall project. But neither did we want to project that only technical communicators understand sound communication principles. We fostered a spirit of collaboration as much as possible in our development and wanted to portray that same spirit to and for students.

Find a Real Client and Project

This strategy seems obvious, but projects can really vary. Although we focused on educational software, anything that gives students the opportunities to apply their talent and knowledge while helping the community can create a more enthusiastic work environment. Even simple projects can promote this commitment.

Clients themselves may come from a variety of settings—anyone who needs a project done or a problem addressed is a potential client. In our case, we set out to connect with another educator, and the local middle schools and high schools provided us with a long list of potential clients. In other courses where we have engaged in similar relationships with clients, we have sought out nonprofit organizations, community service organizations, and other faculty and staff from educational institutions. The difficulty is finding a client who recognizes that the success of projects often relies at least in part on the continued participation of the client. Absentee clients are too often surprised by the end result of student work. Clients who contribute to the success of projects without attempting to control them are more often satisfied with the results of their participation in service-learning environments.

Visit Each Other's Classrooms

For the first part of the term, we were regular participants in each other's courses. This participation helped promote a spirit of collaboration and connectedness we felt was important to display to students. They take the courses more seriously knowing that we do too. As the courses progressed, we held regular faculty meetings to keep each other apprised of developments in our classes as well as to share observations of student performance. We were careful to communicate our collaborations frequently so students would get the sense that the collaboration continued even after the start of the semester.

Plan an Ongoing Project Cycle

In this experiment, we carried only one project beyond the scope of the first quarter. We recruited one project group with a promising software concept and package to continue their work in an independent study. Ideally, we would have given every group the opportunity to work independently, but in this case this was not practical. Another possible source of project continuity would be to assign projects from one set of courses to groups in later sections of the same courses, asking them to contribute to the ongoing development of concept, software design, and documentation.

Promote Departmental Consistency

We promoted this project in our home departments to encourage other faculty who teach these courses to either adopt our approach or promote similar pedagogical values. Our hope was that we could recruit others to participate and thus ensure that even without our direct participation the concept of our pedagogy could continue. It was our hope to create a model for others to follow, to offer an opportunity to reconsider the strengths and weaknesses of the whole computer science and technical communication curricula. Conversation is healthy, we feel, and well-considered change can be healthy as well. However, this effort is an ongoing struggle.

Engage in Vigorous Self-Promotion

Part of our attempt to recruit others to the cause—other teachers, researchers, students, administrators, or funding sources—came in the form of self-promotion. We talked about the project with everyone, in meetings, in hallways, and via email. We submitted conference papers and publications. We submitted proposals for additional funding. Vigorous self-promotion is key to the success of any project and is particularly important to projects that ask people to rethink, revise, or reenvision their work if they are going to participate. Potential movement of any kind starts small and is sustained only by communication.

We recognize that interdisciplinary course linkages such as ours are not likely to become the model curriculum for computer science, for technical communication, or for professional development curricula. Unfortunate as this might be, we still feel the need to struggle to create and sustain an institutional atmosphere where such ventures are at least welcome experiments. That struggle opens up a trio of challenges for continuing investigation of questions beyond those that grow out of the discussion we have presented to this point. First, as a professional teaching community, we need to reinvent our pedagogy to fit the potential of new workplace relationships among professions. Nonacademic workplaces have often seemed more interdisciplinary than academic ones. Many projects in industry require the contribution of many disciplines and professions. Education ought to be a more interdisciplinary venture. But significant challenges remain before interdisciplinary education becomes an institutional reality. We need to identify those barriers and devise strategies for addressing them. Second, we need to create interdisciplinary places for discussion where values can come together, sometimes to collaborate, sometimes to complement, sometimes to conflict. Some conferences provide opportunities for interdisciplinary discussions, but too often our opportunities to engage with colleagues from other departments and disciplines are passed up. Certainly, our disciplinary barriers are in part a product of our disciplinary isolationism in curriculum design. Third, program administrators need to consider the short- and long-term ramifications of encouraging this interdisciplinary venture. In the short term, such projects open up pathways for focused reflection on pedagogical goals and practices. In the long term, such reflection can result in significant rewards. However, someone has to decide what resources, and in what amounts, are appropriate to commit to this project and whom among the many colleagues available it would be best to approach with this project. Again, how can we address this seemingly enormous set of challenges? This question bears further investigation, but we suggest some beginning places here in this chapter.

Interdisciplinary pedagogy of the kind we describe here is not new, but it is innovative. The present structure of colleges and universities does not make interdisciplinary work easy to develop or to succeed with. We met this challenge in the short run with hard work and hope for serving students well. Our early successes and failures provide fuel for future exploration in our work. And we end this discussion knowing that if teachers of technical communication and other disciplines can manage the potentially difficult logistics of a venture such as ours, there are real benefits for both students and faculty.

ACKNOWLEDGMENTS

The authors would like to acknowledge the contributions to this project of Fritz Erickson, Margaret Falersweany, Karla Kitalong, and Jennifer Williamson.