Journal of Information Technology Education Volume 9, 2010

A Tale of Three Classes: Case Studies in Course Complexity

T. Grandon Gill and Joni Jones
University of South Florida,
Tampa, FL, USA

ggill@coba.usf.edu; jjones@coba.usf.edu

Executive Summary

This paper examines the question of decomposability versus complexity of teaching situations by presenting three case studies of MIS courses. Because all three courses were highly successful in their observed outcomes, the paper hypothesizes that if the attributes of effective course design are decomposable, one would expect to see a large number of common attributes emerge in the characteristics of all three courses. Instead, radical differences in course design and delivery are observed across all three courses.

To explain how such different approaches can lead to successful outcomes, the paper draws upon the concept of a rugged fitness landscape (Kauffman, 1993), first introduced in evolutionary biology and later applied in informing science (Gill, 2008), wherein high levels of interactions between entity attributes necessarily lead to multiple fitness peaks. To support the proposition that the courses described exist on such a landscape, the courses (and the evolution of their designs) are examined for qualitative evidence of interactions between characteristics. Looking at four general areas—the instructor, the course content, the design/delivery method, and the students—evidence for the presence of interactions is observed. Thus, the three courses appear to confirm the hypothesis that the fitness of a particular course exists on a rugged landscape.

The paper considers how landscape ruggedness may impact research in the area of course design. Informing science research has demonstrated, for example, that when entities on such a landscape individually attempt to maximize fitness, they tend to cluster on peaks. As a consequence, statistical approaches to explaining entity fitness, such as multiple regression analysis and structural equation modeling, may vastly exaggerate the significance of observed relationships (Gill & Sincich, 2008). The huge number of potential interactions between characteristics in even small models may also require huge numbers of observations to perform such tests (as is commonly the case in medicine). Thus, qualitative approaches to understanding the course fitness may become the only rigorous tools that can be applied. Arguably such research is likely to take a very different form—both in terms of length and descriptive content—than much of the past research published in the area of course design.

Keywords: complexity, programming, decomposability, case method, research methods.

Editor: Ann Monday

Introduction

Pedagogical research is unusual within academic research in that nearly all the researchers in the area are also practitioners, which is to say they teach as well as research teaching. For this reason, interest in the answers to the research questions is personal as well as professional. Will distance learning teaching be as effective as face-to-face techniques? Is the case method really more effective than lecture? Should laptops be allowed in the classroom? The number of questions that might be posed is essentially unbounded.

No one would dispute that research on teaching and learning can be challenging. After all, there are many variables that must be considered. Who could plausibly argue, for example, that factors such as the experience of the instructor, the characteristics of the students, the form of content being presented, the method of delivery, and the setting of the class are irrelevant to learning? Nonetheless, often for causal relationships that involve many variables, the individual effects of specific factors can be teased out using techniques such as regression or structural equation modeling (SEM). In such cases, the underlying process can be described as nearly decomposable (Simon, 1981). Additionally, sometimes the interrelationship between variables is so great that such decomposition is impossible. In such cases, the relationship is complex. Where such complexity exists, the research strategy needs to be reevaluated, since an individual variable's impact on overall effectiveness can be highly dependent upon the values of other variables.

A particularly significant implication of complexity relates to the value of quantitative analytical techniques, such as those just mentioned. Recent research has demonstrated that, under the reasonable assumption that individuals continuously attempt to improve fitness, complex underlying relationships can produce statistically significant yet entirely misleading results (Gill & Sincich, 2008). Thus, the assumption of decomposability needs to be carefully tested prior to applying these techniques. At the present time, such tests require qualitative analysis of the process; quantitative tests for this form of complexity have yet to be devised (Gill, 2008).

The present paper considers the question of the decomposability of teaching situations by presenting a qualitative analysis of three case studies of MIS courses. The cases themselves are intrinsically interesting—all three illustrate innovative teaching techniques, 2 of the 3 were winners of the Decision Science Institute's (DSI) Innovative Curriculum Competition, and all demonstrated substantial evidence of learning and student satisfaction. The research also finds that by comparing the three cases side-by-side considerable insight is gained into the complexity of the relationship between teaching approach, course setting, and outcome.

The paper begins by introducing the concept of a rugged fitness landscape, taken directly from a model proposed in evolutionary biology (Kauffman, 1993). Then the research design is presented, which involves a qualitative search for interactions across four key areas of the course context: instructor characteristics, content characteristics, design/delivery characteristics, and student characteristics. Each class is presented, with details provided in two appendices, and the key interactions that were observed are identified. Because the first of these courses—referred to as Ism3232.A—evolved dramatically over time, it is presented in both longitudinal and cross sectional terms. The remaining two courses—Ism3232.B and Ism6155.A—experienced relatively few design changes from the time they were first offered. Both are therefore presented only in cross sectional terms. By comparing time slices and cross sectional observations, observational evidence of high levels of interaction between areas is acquired. The paper concludes by considering how this evidence might change the conduct of future research into IT education.

Rugged Fitness Landscapes

The underlying model used in this study is the rugged fitness landscape, as popularized in evolutionary biology by Stuart Kauffman (1993). The concept of a fitness landscape is relatively simple, and its basic form—that some desirable dependent variable is a function of one or more independent variables—should be familiar to nearly any researcher in the social sciences. It begins with some value to optimize or improve—often presented as the dependent variable in a mathematical or statistical model—that is referred to as the "fitness" value. The remainder of the model consists of variables that can be controlled or observed which may contribute to fitness. For example, a linear equation, such as:

$$Y = c_0 + b_1x_1 + b_2x_2 + \ldots b_Nx_N$$

can be viewed as a very simple form of fitness landscape, where each of the N variables contribute independently to fitness. What makes a contribution x_i independent in the above equation is that the value of b_i does not depend on the values of the other variables (i.e., $x_1 \ldots x_{i-1}, x_{i+1} \ldots x_N$).

Interactions between characteristics are said to occur when the impact of a particular variable, x_i, on fitness cannot be determined without knowing the values of certain other variables. Where a very limited number of such interactions between variables occur, there are statistical techniques (e.g., using the product of the interacting variables as an additional variable) for accommodating the effect. As the number of variable interactions grows, however, these techniques become impractical. For example, if a model consisted of 30 dummy variables (i.e., binary variables that can only have 0 and 1 values) and they all interacted with each other, you would need to determine 2^{30} (over a billion) coefficients in order to capture every interaction.

In order to capture the degree of interaction in a particular fitness space, Kauffman (1993) uses N,K notation, where N is the number of variables that determine fitness and K is the number of other variables each variable interacts with. Thus, the two extremes become:

> **N,N-1**: Every variable interacts with every other variable. This is sometimes referred to as the *chaotic* case.
>
> **N,0**: Every variable exerts an influence on fitness that is independent of every other variable. This is referred to as the *fully decomposable* landscape in this research.

The term *rugged* is used to describe landscapes that are not fully decomposable or *nearly decomposable* (where interactions can be captured with a few specified interaction terms). The reason that such a term is apt is that such landscapes will nearly always exhibit local fitness peaks—which is to say combinations of values where any change to a single variable value results in a decline in fitness. These peaks present a formidable obstacle to maximizing fitness since moving from one fitness peak to another (higher) fitness peak one variable at a time necessarily involves a reduction in fitness during the period of transition. Conceptually, navigating such a fitness space is like traversing a mountain range. By taking a path that always travels upwards, you are guaranteed to reach a peak. There is no guarantee, however, that you will reach the highest summit. Instead, you may find yourself at the top of a foothill.

A rugged landscape also presents a significant challenge to researchers seeking to apply statistical methods, such as regression and structural equation modeling, for purposes of hypothesis testing. In particular, unless one assumes that entities on the landscape are *not* trying to increase their fitness (e.g., instructors do not try to make changes to their courses in order to improve their effectiveness), entities on rugged landscapes will tend to migrate to peaks. This migration can, in turn, produce serious errors in statistical models that assume the landscape is decomposable (Gill & Sincich, 2008). Thus, before observational data from a particular landscape is analyzed using such techniques, it is critical that the assumption of full or near decomposability be verified. Unfortunately, little (if any) existing research into course design undertakes such an investigation.

There are a variety of types of evidence that could support the hypothesis that a particular landscape is rugged. These criteria include:

1. Highly dissimilar examples of high fitness can be identified; this would suggest the presence of multiple local fitness peaks across the landscape.
2. Incremental changes to fitness—resulting from manipulating the same variable in the same manner—are observed that differ significantly in different situations; this suggests that the variable's effect cannot be established independent of the values of other variables. It is also possible to observe large changes to fitness resulting from of individual variable changes, since interactions can effectively magnify the impact of such changes. For example, omitting the baking powder from a cake recipe may drastically reduce the fitness of the resulting cake, even though the quantity of the ingredient is small and its impact upon taste negligible. This differs from decomposable landscapes, where the impact of a particular variable is always the same and if many variables participate in determining fitness, the average incremental impact of each will be relatively small.
3. Fitness behavior in a particular setting that varies significantly from findings well supported by previous research; like the second, this suggests a situation-dependence that implies interactions between variables. (Throughout this paper, our use of the term setting is equivalent to "context" or "situation.")
4. Sensitivity to small changes in variables. When a landscape is decomposable, changes in most variables exert a predictable (and usually small) impact on fitness. Where the underlying landscape is complex, variable changes can act through interaction and a small change (e.g., omitting half a teaspoon of baking powder from a cake recipe) can dramatically change fitness.
5. The researcher's interpretation of the fitness landscape, based on observation or experience, may supply a logical basis for arguing that such interactions are to be expected.

The last of these, combining perception-based logical arguments with observed data, may be somewhat unsettling from an empirical research perspective. Traditionally, the assumption is that the needs of objectivity are best served when the characteristics of the observer appear to exert minimal impact on the observational data that are employed for hypothesis testing—an assumption justifying author anonymity and double blind peer review processes. Unfortunately, as previously noted, a high level of interactions between variables dramatically increases the coefficients that must be determined when standard statistical methods are employed. More coefficients, in turn, can easily raise the number of observations required to determine their values to levels that are impractical (e.g., tens of billions of observations in the case of 30 highly interacting variables). Thus, further progress is likely using techniques that allow for deep study of fewer situations, such as case-based research. In such methods, the use of many sources of data, mixed with a liberal amount of interpretation by observers whose expertise must be demonstrated, is encouraged as part of a process referred to as triangulation (Yin, 1994). Indeed, some of the most influential case research has relied heavily on researcher-interpreted analysis (e.g., Allison, 1971).

Research Design

The central research question being investigated is the degree to which the fitness landscape for course effectiveness can be characterized as rugged. As noted in the introduction, qualitative methods appear to be the only approach suitable for addressing this question at the present time. For this reason, a multi-case design—incorporating in-depth process observations—was employed.

Unfortunately, the five criteria for demonstrating ruggedness in the fitness landscape present a formidable research design challenge—since evidence needs to be gathered from nearly opposite sources. For example:

- Criterion 1 (existence of widely separated high-fitness peaks) is best served by observations that *differ from each other in a many ways as possible*.
- Criteria 2, 3 and 4 (variables that behave differently in different situations or which differ from widely observed behaviors) are best served by controlled experiments where *manipulations limited to a single (ideally) or very small number of variables are observed*.
- Criterion 5 (investigator interpretation) is best supported by archival data gathered prior to searching for ruggedness, thereby reducing the impact of investigator bias.

As it turns out, the cases investigated in the present paper simultaneously meet these criteria to a reasonable extent. The research described in the present paper involves three classes (representing two distinct courses) and two different instructors. Details on the instructors and classes are provided in Appendix A and Appendix B. The classes can be summarized as follows:

- **Ism3232.A**: A section of an undergraduate introductory programming course taught by Instructor A. Including the instructor's time at a previous institution (where the course had the same number and description), the course had been taught for over a decade. During the period from 2003 to 2008, extensive data had been collected through surveys and other means each semester, providing a continuous series of data points that could be used to assess the effectiveness of the course.
- **Ism3232.B**: A section of the same undergraduate programming course taught by Instructor B for the first time in the fall semester of 2007.
- **Ism6155.A**: A graduate capstone class for the university's MS-MIS degree program taught by Instructor A. This objective of this course was to help students in the MS-MIS program develop a greater appreciation of how business functions and MIS work together to determine organizational effectiveness.

The data that was accumulated from these courses was not the result of a research project. Rather, the research endeavor stemmed from the interesting—and sometimes seemingly inexplicable—results we encountered as the courses were taught over time. Naturally, the qualitative character of the findings presented was a result of process by which evidence accumulated. As Hambrick (2007) points out, however, detection and reporting of anomalies in existing data can lead to important additions to our knowledge even where the findings are not the product of theory-driven design. Thus, our goal is to explain the results that had already been accumulated, particularly where anomalies inconsistent with existing theory and practice were detected.

In looking for evidence of ruggedness, attributes expected to impact fitness were classified into four categories: instructor characteristics, course content, course design/delivery, and student characteristics. These categories were chosen because they map, respectively, to the *sender*, *message*, *delivery system* and *client* components of Cohen's (1999) original conceptual model used to define informing science. Here, the sender maps to the instructor, the student maps to the client, the delivery system (also referred to as the *channel*) maps to the instructional design/delivery, and the message maps to the course content.

Viewed from an overall design perspective, the three classes investigated varied by instructor and content as shown in Figure 1. The columns in the figure represent the instructor. Key elements of the background of the two instructors relative to their experience teaching the courses are presented in Appendix A. The rows represent content. The two courses investigated, an introductory programming course taught to undergraduates and the capstone course of the university's MS-

MIS program, were very different in their learning objectives. This fact, combined with substantial differences in the backgrounds of Instructors A and B, provided a diversity of settings suitable for testing criterion 1.

The multiple boxes in the Ism3232.A cell indicate the presence of longitudinal observations. As previously noted, this particular class had evolved over time through a series of incremental changes that had been tracked with a comprehensive survey instrument. The evolution of the course and its outcomes had also been documented extensively in previous research (e.g., Gill, 2005a, 2006a; Gill & Holton, 2006). Taken together, these offered a plausible basis for testing criteria 2, 3, 4, & 5. Many of the major elements of the context of these courses were outside of the investigators' control. Notably, in the case of the longitudinal study, none of the variables observed could be controlled for experimental purpose, so we were constrained to using a natural experiment.

Probably the greatest challenge in this research was coming up with a measure for course "fitness". Particularly when comparing courses with highly different content, such as Ism3232 and Ism6155, there is no obvious way to compare the amount of learning that has taken place on any objective scale. For this reason, once again, triangulation from a variety of sources was employed:

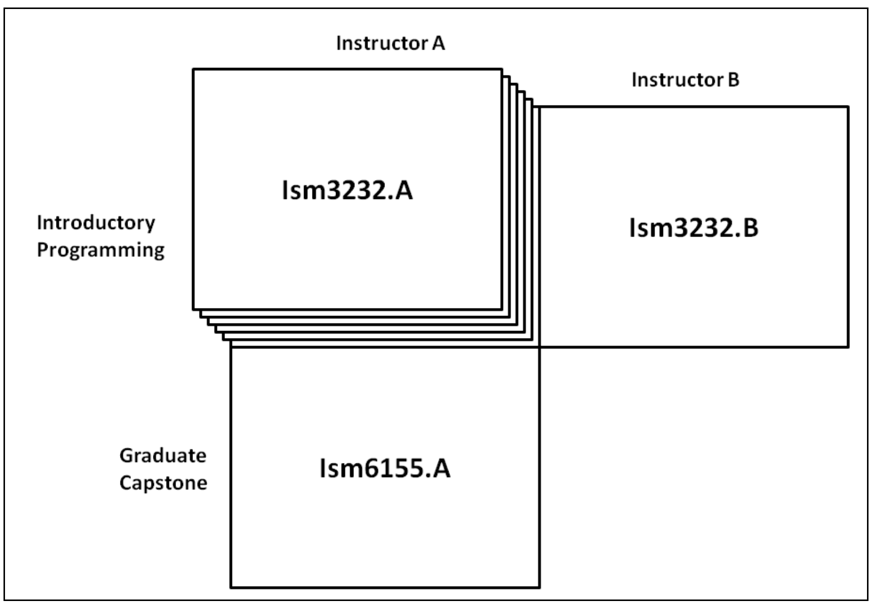

Figure 1: General Research Design.
Instructor is held constant for two forms of content. Content is held constant for two instructors.
A series of longitudinal observations hold instructor and content constant,
while delivery technique varies.

1. *Instructor evaluations*. Although frequently disparaged as an indicator of student learning, these measures often correlate with other indicators of student learning (McKeachie, 1990). They also provided two other advantages: they were the only comparable measure available across courses/instructors, and—from the instructors' perspective—they were a particularly impactful measure of fitness, inasmuch as annual evaluations used for pay and promotion relied almost exclusively on these measures for the assessment of faculty member teaching performance.

2. *Course attrition*: For programming courses in particular, high attrition rates are often encountered. Ism3232 was graded on a letter scale, going from A to F. Grades of A, B and C were sufficient to allow students to proceed with the program. Thus, the percentage of students receiving D, W (withdrew from course) or F (failed the course) grades is a reasonable indicator of course fitness for such courses. In the graduate course studied, attrition was almost non-existent and almost no C grades were awarded, so the measure was not particularly useful in this context.
3. *Course performance*: For Ism3232.A, a fixed curve and fixed set of materials was employed for a sustained period of time (from late 2004 to early 2006). As a consequence, student grades and GPA data were directly comparable during this period and could be used as an estimate of relative learning across sections. When Ism3232.A changed format in Fall 2006, this comparability was broken, however. Similarly, direct comparisons between Ism3232.A and Ism3232.B could not be made, owing to different grading and testing strategies.
4. *Peer review assessment*: Reviews of course fitness by other faculty members. In this context, outcomes of two of the classes (Ism3232.A and Ism6155.A) were both reviewed extensively as part of the *Decision Science Institute's* (DSI) Instructional Innovation Competition. Less formally, Ism3232.B was evaluated by Instructor A during the semester when it was first offered.

While none of these measures individually can be viewed as a truly reliable indicator of course fitness, when they converge a strong case for fitness can be made. Yin (1994) advocates the use of such an approach as part of data collection.

Results

For the sake of compactness, results are presented in the form of a series of tables, subsequently analyzed in the discussion section. In parallel with this, narratives that describe the classes more fully are presented in Appendix B. These should serve to further clarify the results tables and should also be of interest to the reader seeking to better understand the teaching approaches employed in the three classes, perhaps as a basis for adapting them for his or her own teaching use.

Fitness of the Three Classes

In order to demonstrate ruggedness according to criterion 1 of the previous section, it is necessary to identify multiple fitness peaks within the domain being studied. Proving a particular class represents an actual peak (i.e., the combination of attributes is locally "optimal") is likely to be impossible given the subjective nature of many of the criteria used to assess fitness, yet the three classes all demonstrated very high fitness and informally the term peak is used to describe any position in the region of high fitness close to the formal peak. Using the triangulation measures of fitness specified in the research design section of this paper, evidence of the fitness of three courses (using their December 2007 offering, the semester during which all three classes were offered) is presented in Table 1.

On all four dimensions of fitness, each of the three classes performed very well. As noted in Appendix B, the student evaluations of the instructors for Ism3232.A and Ism3232.B represented the highest and second highest scores in the history of the course. The Ism6155.A evaluations continued a long trend of high scores for the course. In all three courses, student comments were highly positive. The DWF rate for both Ism3232 sections was under historical averages of 30-50% (see Table B.1 in Appendix B). (The 0% DWF rate for Ism6155.A was fairly typical for MS-MIS classes and was, therefore, less noteworthy.)

Table 1: Measures of course fitness

	Ism3232.A	Ism3232.B	Ism6155.A
Instructor Evaluations	• Extremely High (4.89/5.0) • Highly positive student comments	• Extremely High (4.79/5.0) • Highly positive student comments	• Very High (4.64/5.0) • Highly positive student comments • Most positive mentions in program exit interviews
DWF	< 3%	21%	0%
Student course performance	• Large body of content covered • High self-reported work load	• Large body of content covered • Strong exam performance	• High level of student interaction
Peer review	• DSI competition winner in 2007 • University teaching award in 2007	• Reviewed by Instructor A	• DSI competition winner in 2007

Student course performance was more difficult to assess across sections and courses, since the material being conveyed and instructional techniques employed were so dissimilar. Nonetheless, as pointed out in Appendix B, the amount of content being conveyed in both of the Ism3232 sections was unusually large for an introductory programming course (e.g., at Instructor A's previous institution, the decision had been made to spread the same content over two courses) and the self-reported workload of Ism3232.A students was roughly twice that reported for other undergraduate courses in the MIS major. For Ism6155.A, performance assessment was subjective and was based largely on instructor observations related to the quality of work and level of class participation.

With respect to peer review, both the Ism6155.A and Ism3232.A had been reviewed as part of the DSI's Innovative Instruction Competition. The initial entry consisted of a 30 page report documenting course effectiveness. Three finalists each year gave a live 30 minute presentation before judges at the DSI annual meeting. Both of the classes won the competition, in 2005 and 2007 respectively. A document detailing Ism3232.A's effectiveness was also submitted as the basis for Instructor A's university award for excellence in undergraduate instruction, received in 2007. Although Ism3232.B did not undergo formal peer evaluation of this type, Instructor A—in his role as course coordinator for all Ism3232 sections offered by the university—did perform a thorough inspection of the course and, after previewing the course examinations, predicted average test scores far lower than those actually achieved by Instructor B's students. Given Instructor A's decades of experience in teaching the same course, the surprisingly strong performance of these students can be characterized as objective evidence of student achievement.

Collectively, then, it is reasonable to conclude that all three classes were high fitness offerings. In order to support the contention that they resided on separate peaks, the characteristics of the three offerings are now compared.

Characteristics of the Three Classes

The strength of the assertion that the high fitness classes occupy on separate fitness peaks will be determined by the degree to which the classes differ in characteristics (independent variable values, in the terminology of multiple regression analysis). To assess this separation, key attributes of the class offering are presented in Table 2. From this table, the following conclusions can be drawn:

- If an attribute has strongly different values across the three classes AND its effect is decomposable, then it does not materially contribute to fitness. This follows because, in the previous section, it was argued that all three classes seemed to be close to fitness peaks.
- Since all of the attributes in Table 2 have different values, it follows EITHER that none of them has a material impact on course fitness (assuming that their effect is decomposable) OR that many or all of their effects involve interactions with other characteristics.
- If interactions with other characteristics are an important contributor to fitness, then the three classes do, indeed, occupy separate fitness peaks.

Table 2: Cross-Course Comparison, based upon Fall 2007 semester. With all three courses existing at high fitness levels, the fact that no attribute values are consistent across all three classes implies that the attributes are either of minor importance (decomposable assumption) or that their impact on fitness is through interactions with other variables (complex assumption)

Category	Attribute	Ism3232.A	Ism3232.B	Ism6155.A
Design/Delivery	Classroom Lectures	No	Yes	Minimal
Design/Delivery	Multimedia Lectures	Yes	No	No
Design/Delivery	Moderated Classroom Discussions	Optional	No	Yes
Design/Delivery	Paired Student Problem-solving	No	Yes	No
Design/Delivery	Student Presentations	No	No	Yes
Design/Delivery	Deadline Flexibility	Yes	No	No
Design/Delivery	Mandatory Attendance	No	Yes	Yes
Design/Delivery	Examinations	No	Yes	No
Design/Delivery	Outside Class Projects	Yes	No	Yes
Design/Delivery	Level of Performance Feedback	High	High	Low
Design/Delivery	Grade Subjectivity	Low	Low	High
Design/Delivery	Source of course organization	Evolved	Designed	Designed
Student	Student Level	Undergraduate	Undergraduate	Graduate
Content	Topic	Programming	Programming	Capstone
Instructor	Instructor	Instructor A	Instructor B	Instructor A
Instructor	Instructor Experience with Course Subject Matter	High	Low	High

Given the number of the attributes presented—including many aspects of course design/delivery as well as instructor and content characteristics—it seems unlikely that most are not material contributors to fitness. As a consequence, the diversity of values in Tables 1 and 2 provide strong support for the presence of ruggedness, based upon the likely existence of multiple peaks (criterion 1).

The cross course comparison is most compelling with respect to peaks existing in the Design/Delivery areas. Evidence for interactions involving student characteristics, instructor characteristics, and content characteristics are somewhat less compelling, since it is harder to identify specific attributes for classifying students, content, and instructors. There are, however, some observations that can be interpreted as evidence for interactions across areas. For example:

1. The fitness of Ism3232.A and Ism3232.B in Fall 2007 appeared to surpass all previous fitness levels for the Ism3232 course (even prior instances of Ism3232.A). Both Instructor A and Instructor B attributed part of this to the fact that during the first week of class, each urged students to consider carefully the different designs of the two sections and to switch if they wanted a more self-paced (Ism3232.A) or structured (Ism3232.B) experi-

ence. Some switching (4 or 5 students) did appear to take place and, perhaps as a consequence, the small number of negative comments on the Ism3232.A organization—the only section for which comparative data was available—that were routinely encountered in prior semesters of Ism3232.A were absent. Such strong negative reactions can have a significant impact on overall evaluation averages (explaining the improvement) but would also indicate a significant interaction between student and design/delivery.

2. Instructor B's comments regarding her discomfort with the pure self-paced structure of Ism3232.A suggests a significant interaction between instructor and design/delivery characteristics.
3. A number of students took both Ism3232.A and Ism6155.A, since students who needed to take a programming course as a prerequisite sometimes were admitted into the MS-MIS program. In their comments to Instructor A, they reported a very different reaction to the two courses—some preferring the former and some the latter. Since the instructor was the same in both courses, this suggests the presence of a two or three way interaction between student, content, design, and delivery.

Longitudinal Analysis

As more fully described in Appendix B, the evolution of Ism3232.A between Fall 2001 and Summer 2006 provides an interesting source of insights on ruggedness because, during that period, both instructor (Instructor A) and course content remained virtually unchanged (e.g., with the exception of adding a major assignment in Spring 2002, the same assignments and grading scale were used). Thus, variability in course fitness was necessarily attributable to changes in design/delivery, students, and the interaction between the two.

As suggested by the variability of course evaluations during the period (e.g., see example student comments in Appendix B, Period 1), the student population cannot be viewed as homogeneous. Rather, such variation seems to point to both student-instructor and student-design/delivery interactions. Equally interesting, and as noted in a previous study (Gill & Holton, 2006), some effects that have been widely observed in the literature—those between programming course performance and both gender and prior programming experience (e.g., Goold & Rimmer, 2000; Hagan & Markam, 2000; Holden & Weeden, 2003; Roberts, 2000)—were entirely undetectable in Ism3232.A. This falls under evidence criterion 3—behavior substantially different from widely reported findings in the literature.

Even more compelling evidence for interactions can be found when the evolutionary narrative (see Appendix B) is summarized, as shown in Table 3. Over the 6 year period (from before 2001 to summer 2006), there were roughly three periods where a strong case could be made that the class had achieved high fitness (identified using the point numbers specified in Appendix B). In comparing Period 0 with Period 2 and Period 6, only one attribute (online support through asynchronous discussion groups) remains consistent. (Moreover, by the time Point 6 had been reached, the degree to which these discussions were actually used had dropped precipitously from earlier sections, where several hundred postings per assignment were not unusual.) Thus, the choice is either to conclude that none of the conflicting attributes are material to course fitness or to conclude that they interact.

Table 3: Longitudinal Comparison of Ism3232.A sections

Attribute	Period 0	Period 1	Period 2	Period 3	Period 4	Period 5	Period 6
Classroom Lectures	**Yes**	Yes	**Yes**	Yes	No	No	**No**
Multimedia Lectures	**No**	No	**No**	Yes	Yes	Yes	**Yes**
Multimedia Walkthroughs	**No**	Yes	**Yes**	Yes	Yes	Yes	**Yes**
Asynchronous Discussions	**Yes**	No	**Yes**	Yes	Yes	Yes	**Yes**
Self-Paced	**No**	No	**No**	No	No	Yes	**Yes**
Progress Monitoring	**No**	No	**No**	No	No	No	**Yes**
TA support	**Yes**	No	**No**	Yes	Yes	Yes	**Yes**
Examinations	**Yes**	Yes	**Yes**	Optional	No	Yes	**No**
Fitness	**High**	Low	**High**	Medium	Medium	Medium	**High**

The adaptations that took place as the course evolved provided many further examples supporting the assertion that interactions between characteristics were material to fitness. For example:

1. Instructor A had the strong belief that student's should not be forced to come to class once they reached the college level. Attendance was never a problem in Period 0; however, since it was nearly impossible for students to complete course projects without attending class, they came. This is an interaction between the instructor and design/delivery.
2. When multimedia assignment walkthroughs were introduced in Period 1, completing assignments without lectures became much more feasible. Attendance fell dramatically, yet outside support from other sources (such as an online discussion groups or teaching assistants) was very limited. Thus, students who decided to take full advantage of the additional materials often found themselves isolated from the class when they were confused. This would be an example of a criterion 2 effect: adding multimedia walkthroughs would be expected to positively impact course fitness (or at least not hurt it) yet when interacting with the absence of a mandatory attendance policy—preferred by Instructor A but not by all instructors—it actually appeared to exert a disproportionately large negative impact. Not all students were thus affected, however, suggesting a further interaction between student characteristics and design/delivery.
3. When additional structure was added to the course, through a change to content (flow charting assignment), earlier oral exams, and the incorporation of online asynchronous support, the degree of disconnect between some students and the course was reduced. The course then returned to higher fitness, confirming the previous interaction.
4. As more and more material became available through taped classes and, later, web-based lectures, the potential disconnect between students and the class again grew. With virtually no students attending classes, the decision to eliminate live lectures was made. This led to an interaction between students and assignment due dates. Students who were comfortable with the format handed in assignments on time and therefore received full credit. Students who had trouble adapting to the format tended to procrastinate and therefore had late points deducted from their assignments. This led to a paradoxical situation where the students having the greatest problems ended up having to do more work to get the same grade as students having an easier time. Realization of this led to Instructor A's decision to go entirely self-paced. Here, student characteristics again interacted with course design/delivery and instructor characteristics.
5. Although the self-paced format seemed to improve fitness with respect to retention (see Table B.2 of Appendix B), it made it virtually impossible for Instructor A to ascertain the overall

progress of the class. To rectify this, a progress monitoring system was implemented (a tool made necessary solely as a result of the course's self-paced structure). Providing an additional source of communication with students, class fitness returned to high levels. Once again, a three-way interaction (student-design/delivery-instructor) seems evident.

The structure of Period 6 was largely adopted for the cross-sectional version of Ism3232.A (Fall 2007). The principal difference was that a new department chair preferred that the multiple sections of the course have different instructors, so that TA support for the course would no longer be necessary. Instructor A therefore replaced oral exams to validate assignments with random block multiple choice exams administered through Blackboard—an interaction between administrative requirements and course design/delivery—and used the SCORM facility on Blackboard (which allowed student access to online lectures and readings to be tracked) to enhance the progress monitoring system. This represents an interaction between technology and design/delivery.

Discussion

Beyond the specific sections discussed, the patterns described in this paper are consistent with our observations for 12 sections of Ism6155 (from 2003-2009), 8 sections of Ism3232.A in its post-longitudinal form (from 2006-2009), and 6 sections of Ism3232.B (from 2007 to 2009). In assessing our findings, it might first appear that great deal of effort was made to justify a point that most experienced instructors would instantly concede: that many good ways to teach a particular body of material exist and choosing the "best" in a given situation is likely to depend on the characteristics of the instructor, students, content to be conveyed, and tools available for teaching. Thus, the reader might well ask "what is all the fuss about?"

In fact these findings are more profound than they may, at first glance, appear to be. For example, if statistical analysis of observations from a rugged fitness landscape is to be performed, it is critical that material interactions between variables be included as terms (Gill & Sincich, 2008). In the examples presented, however, over a dozen different variables were used to describe design/delivery alone (and these variables would be far from sufficient for describing *all* design/delivery options). No attempt was made to postulate the variables necessary to adequately capture the attributes of instructors, content, instructional technologies, or students. To imagine the entire course setting could be adequately described with even hundreds of variables is highly optimistic (e.g., over 30 characteristics have been proposed to describe student learning style alone; Gill, 2008). With so many variables, however, the number of possible N-way interaction terms (roughly 2^N for N binary independent variables) rapidly exceeds the number of atoms in the universe. Thus, these statistical tools become useful only where you can adequately predict all the interactions in advance. Here the ups and downs of Instructor A's Ism3232 in periods 0 through 6 are instructive. Even a highly experienced instructor, teaching a course that he had taught for a decade, was unable to anticipate many of the interactions that occurred as new technologies and students were encountered. Thus, in the absence of near decomposability, it is doubtful that useful models of effective instruction can be developed that are sufficiently robust to handle even modestly novel instructional situations.

Research Design Issues

The research challenges presented by rugged fitness landscapes have been previously outlined (Gill, 2008). These include:

- When a set of observations are drawn from the portion of the landscape associated with a particular peak, any findings only apply to that peak. Or, stated in concrete terms, what works to improve fitness in a particular course setting (e.g., Ism3232.A) will not necessarily work in a course setting that is significantly different in its characteristics (e.g., Ism3232.B).

- When observations are drawn from multiple peaks—and interaction terms are not explicitly included—the results will often vastly overstate the statistical significance of terms when tools such as multiple regression and SEM are employed (Gill & Sincich, 2008).
- The nature of the theory that can be generated from such landscapes can be characterized as "ugly", which is to say it will not be compact, it will not generalize well outside of the observations used to develop it, and it will tend to grow in size as new observations are incorporated into it.

For these reasons, different research strategies are likely to be necessary to effectively research such landscapes. A number of these strategies are evident in the present paper:

1. *In researching a rugged fitness landscape, it is generally better to focus on identifying and studying peaks rather than on acquiring many observations.* As previously noted, the compact and rigorous theory that results from analyzing many observations drawn from a decomposable landscape simply cannot emerge from a truly rugged fitness landscape. Instead, gathering many observations without an understanding of the landscape structure can easily lead the investigator to fall prey to statistical illusions (Gill & Sincich, 2008). Identifying combinations of attributes that produce peaks, on the other hand, can provide useful guidance to other entities existing on the same fitness landscape.
2. *Understanding the history of an entity can provide important insights into the structure of a rugged fitness landscape.* Entities on a rugged fitness landscape will tend to adapt in a manner that continually seeks to increase fitness. This may either occur as a generational phenomenon (e.g., survival of the fittest) or through a process of intelligent adaptation. As a consequence, studying the changes leading up to a particular state will often provide useful information regarding whether or not a peak has been achieved and what types of interactions are being encountered.
3. *Transformational, rather than incremental, changes may be required to increase the fitness of an entity.* The challenge for entities operating on a rugged fitness space is local peaks. Once a local peak has been reached, any incremental change—no matter how well intentioned—leads to reduced fitness. To move to a higher fitness peak, it may be necessary to transform a whole collection of characteristics at once. To motivate that change, a clear understanding of the target peak (see item 1) is likely to be required. What this implies is that research describing *how* to get to a particular peak (i.e., implementation descriptions) may well be more valuable than any theory that helps to identify what peaks may exist. This further reinforces the value of historical data that describes the process by which other entities reached the desired peak (see item 2).
4. *Desirability of replications*. Where a fitness landscape is decomposable, the best research questions tend to be original; replications of this research, in different domains, are expected to yield roughly the same answer as the (properly conducted) original research, meaning their contribution to knowledge is limited. Holes in the literature therefore provide the most fertile ground for new discoveries and many prestigious journals specifically refuse to publish replications of prior research. In a rugged fitness landscape, on the other hand, important questions need to be asked over and over again since different regions of the fitness landscape can yield very different results. As was shown clearly in the case of Ism3232, where five quite different high fitness combinations were observed—three of which had a common instructor and nearly identical content (Period 0, Period 2 and Period 6).

A rather dramatic illustration of how the assumption of landscape ruggedness can impact the amount of data and replications needed to validate empirical findings can be found in the case of clinical surveys in medicine—a domain where interactions between independent variables are expected and routinely encountered (e.g., the warnings on broadcast advertisements for pharmaceuticals). Consider, for example, the (seemingly) simple question "Is coffee good for you?" The question has been considered in roughly 19,000 different studies (some of which had over 100,000 participants observed over a period of decades) and has yielded a number of results of high statistical and clinical significance—such as a huge drop in the incidence of Type II diabetes among males who drink 6 or more cups a day. Nonetheless, the only uncontested conclusion of that research seems to be that more research is needed (Kirchheimer, 2004).

The form in which research is presented is also likely to be impacted when the domain that is being researched is a rugged fitness landscape. As a consequence of favoring peak research over observation gathering and in light of the value of exploring the history of the observation, an in-depth case study is likely to be more useful than either empirical analysis of many observations or theory building involving the presentation of compact models. An unfortunate result may be an increase in length associated with such research, a common criticism leveled against case research (Yin, 1994).

Rugged Landscape Researchers

A further implication of the different approach to research necessitated by rugged fitness landscapes is increasing dependence on the observational expertise of the researcher. Typically, such landscapes involve hundreds of possible variables, some of which will have influence in a particular region and some that will not. In the classes discussed, for example, Blackboard discussion groups appeared to play a pivotal role in some sections (e.g., Period 2) and yet were almost entirely ignored in others (e.g., Period 6)—despite the fact that the topics, tools, and protocols were identical in both cases. Only a skilled observer will be able to distinguish the relevant from the irrelevant. Statistical analysis is likely to be of limited use in such cases; frequently, new variables will need to be invented to capture new situations, and the reliability of such measures will always be subject to question. Rather, they become highly dependent on the investigator's judgment as a source of rigor.

This dependence upon the researcher's expertise can become particularly problematic when studying areas related to learning. The demands of the research in these cases threaten both objectivity and anonymity. In virtually any educational setting, the characteristics and background of the instructors involved are likely to be important contributors to fitness. In higher education settings, however, there is a very high likelihood that any instructor involved is also a participant in the research and is, therefore, a co-author of any manuscript produced. Even in those rare instances where this is not the case, the reviewer is likely to assume that it is. Thus, researchers are faced with a choice: a) provide all relevant details regarding the instructors involved in a particular educational setting and risk the wrath of reviewers, or b) omit relevant details relating to the instructors and thereby potentially undermine the rigor of the analysis. This is not an easy choice to make, but it is an inevitable one if, as the present paper concludes, the domain of IT education is highly rugged.

Generalizability, Diversity and Adaptability

An obvious argument against the broader conclusions that have been presented is the appearance of doing the same thing that was cautioned against: using research conducted in a relatively small domain (i.e., two types of courses and two instructors) to generalize it to a much larger domain. It is conceivable, since only a tiny fraction of the course fitness landscape has been explored, that this research just happened upon the one particularly rugged portion of the landscape, and that the remaining area not examined can actually be described as a smooth, decomposable landscape with a single peak that responds well to conventional research approaches (and attitudes towards research).

Actually, it is likely that the course fitness landscape was once far more decomposable than it is today. Fifty years ago, for example, it would not be much of an exaggeration to characterize U.S. business education as a group of young white males, largely drawn from the privileged class, being taught by older white males, of similar background, employing classroom technologies that consisted of chalk and, perhaps, a slide projector. Under such circumstances, one might well find that only a few course-fitness peaks were accessible—perhaps one for lecture-oriented delivery and one for case discussions. With so few peaks, research conducted in each domain might generalize well from one instructional setting to another.

Today, obviously, the educational context is very different. There are a blend of genders and a mix of students drawn from many backgrounds and nations. A similar transformation in the characteristics of instructors is taking place, albeit lagging the change in students. The range of technologies available for instructional use in the classroom has exploded, as has the range of tools available for use outside the classroom, such as computers and the Internet. A similar transformation has occurred in the number of topics that can be incorporated into business and communications curricula. The work of Kauffman (1993) is based on the mathematically derived premise that landscape ruggedness results from the number of elements in the system (N) and their degree of interaction (K). The changes of the last 50 years have contributed to vast increases in both. Thus, the landscape should be *expected* to be rugged and to grow increasingly so.

Although growing diversity may force a change in how research is conducted, such growth should not be viewed as undesirable. To the contrary, fitness landscapes involving living systems are generally not static. Rather, the location of peaks and valleys tend to be influenced by co-evolving systems (Kauffman, 1993). In IT education, these systems involve both business and technology. Many of the same forces that transform the teaching landscape—such as technological innovation and globalization—are increasing the ruggedness of these co-evolving systems as well. The problem with single-peak, decomposable, fitness landscapes is that they tend to lack the adaptability to survive major changes in how fitness is achieved (Kauffman, 1993). Thus, ensuring that collective teaching activities are widely distributed across multiple peaks may be the best way of ensuring continuing effectiveness. The need for research that explores these peaks in an impactful manner—e.g., one that encourages readers to consider adopting course design combinations that they had never previously thought of—has never been greater.

Conclusions

Whenever you teach a course, you are on a fitness landscape. Assuming you are motivated to improve your course—and nearly all of us who read this journal are—then you will welcome the insights that research can provide. If that landscape is decomposable, those insights can be in the form of a check list. Independently set each characteristic to its most desirable value and you maximize fitness. If the landscape is rugged, on the other hand, your needs are very different. To navigate a rugged landscape what you most need is a map that tells you where you are, identifies places that you may want to go, and provides some direction on how to move from one place to another in the least painful manner. The three cases that have been presented suggest a straightforward conclusion: that the fitness of a particular class is described by a rugged fitness landscape. The arguments supporting this include: 1) the existence of several high fitness classes with very different characteristics (e.g., classes reside on multiple peaks that were very different), 2) the presence of variables that appear to exert different impacts on fitness in different settings (e.g., the added availability of multimedia walkthroughs actually cause class fitness to decline in one setting), 3) the presence of fitness behaviors differing from those widely reported in the literature (e.g., lack of experience and gender sensitivity in one of the programming courses considered), 4) observed sensitivity to small changes, and 5) through qualitative interpretations of knowledgeable observers.

Given the small number of cases considered, it is reasonable to question the generalizability of these findings. Viewed in statistical terms, an N of 3+ is far from compelling. Yin (1994) argues, however, that it is often more appropriate to treat case studies as individual experiments, rather than as individual observations. How many balls, for example, would you insist that Galileo drop from his tower before you were ready to concede that large and small balls fall at the same speed? Viewed in this context, these findings clearly support the conclusion that the landscape being studied is rugged. Assuming this finding can be replicated—just as you would probably want to drop balls from another tower, just to be sure—then you are faced with two possibilities. Either you happened to come across an unusually rugged patch in a broader domain that is otherwise decomposable or the domain is generally rugged. The latter is a more reasonable initial hypothesis. Only additional research can lead to conclusive evidence.

Were we to accept the conclusion that the IT education research domain is a rugged fitness landscape, existing research priorities would need to be carefully re-examined. Much of the research that is published today seems far better suited to a decomposable world. Empirical statistical analysis and conceptual frameworks are of greatest value when the findings are likely to generalize well to settings outside of the narrow domain observed. If the landscape is unlikely to permit such generalizations, research should focus on enriching descriptions of the processes encountered and the variables found to be relevant. If a variable's impact is not expected to generalize, it really doesn't matter that its impact on local fitness is statistically significant unless that same impact is also substantial—in which case computing the precise significance is largely a statistical exercise, since impacts that are truly substantial are nearly always significant in the statistical sense. Indeed, in the unlikely event there is *any* doubt with respect to the statistical significance of a *substantial* impact, it would be much better to investigate that impact further (using whatever qualitative methods are available) as opposed to ignoring it because it failed to pass the test.

In a rugged fitness landscape, it also makes sense to spend far more energy on understanding implementation processes since the decision to transition from one peak to another is likely to depend heavily on how much it costs to travel the path between them. It is remarked that much of the research in IT education has little impact—a comment that applies equally to disciplinary research (Gill & Bhattacherjee, 2007) into the co-evolving systems that determine what needs to be taught. Until research is conducted in a manner that is harmonious with the properties of the fitness landscape that is being researched, this deplorable state will continue.

References

Allison, G. T. (1971). *Essence of decision: Explaining the Cuban missile crisis*. Boston, MA: Little, Brown and Co.

Cohen, E. (1999). Reconceptualizing information systems as a field of the transdiscipline informing science: From ugly duckling to swan. *Journal of Computing and Information Technology*, *7*(3), 213-219.

Deitel, H. M., Deitel, P. J., Hoey, T. R., & Yaeger, C. H. (2004). *Simply C#: An application-driven tutorial approach.* Upper Saddle River, NJ: Pearson Education.

Gill, T. G. (2005a). Learning C++ 'submarine style': A case study. *IEEE Transactions on Education, 48*(1), 150-156.

Gill, T. G. (2005b). *Introduction to programming using Visual C++.NET*. Hoboken, NJ: Wiley.

Gill, T. G. (2006a). The mystery of the self paced course (A). *Informing Faculty, 1*(3), 1-26.

Gill, T. G. (2006b). A learner-centered capstone course for a MIS master's degree program. *Decision Line, 37*(2), 4-6.

Gill, T. G (2008). Reflections on researching the rugged fitness landscape. *Informing Science: the International Journal of an Emerging Transdiscipline*, *11,* 165-196. Retrieved from http://inform.nu/Articles/Vol11/ISJv11p165-196Gill219.pdf

Gill, T. G., & Bhatacherjee, A. (2007). The informing sciences at a crossroads: The role of the client. *Informing Science: the International Journal of an Emerging Transdiscipline, 10*, 17-39. Retrieved from http://inform.nu/Articles/Vol10/ISJv10p017-039Gill317.pdf

Gill, T. G., & Holton, C. (2006). A self-paced introductory programming course. *Journal of Information Technology Education, 5*, 95-105.

Gill, T. G., & Sincich, A. (2008). Illusions of significance in a rugged landscape. *Informing Science: the International Journal of an Emerging Transdiscipline*, *11*, 197-226. Retrieved from http://inform.nu/Articles/Vol11/ISJv11p197-226GillIllusions.pdf

Goold, A., & Rimmer, R. (2000). Factors affecting performance in first-year computing. *ACM SIGCSE Bulletin*, June, *32*(2), 39-43.

Hagan, D., & Markam, S. (2000). Does it help to have some programming experience before beginning a computing degree program? *ITiCSE Helsinki, Finland*, July, 25-28.

Hambrick, D. C. (2007). The field of management's devotion to theory: Too much of a good thing? *Academy of Management Journal*, *50*(6), 1346-1352.

Holden, E., & Weeden, E. (2003). Software development: The impact of prior experience in an information technology programming course sequence. *Proceedings of the 4th Conference on Information Technology Education. Lafayette, Indiana*, October 16-18, 41-46.

Kauffman, S. (1993). *Origins of order: Self organization and selection in evolution*. Oxford: Oxford University Press.

Kirchheimer, S. (2004). Coffee: The new health food? *WebMD*. Accessed on 9/8/2008 from http://men.webmd.com/features/coffee-new-health-food.

McKeachie, W. J. (1990). Research on college teaching: The historical background. *Journal of Educational Psychology, 82*(2), 189-200.

Roberts, E. (2000). Strategies for encouraging individual achievement in introductory computer science courses. *SIGCSE '00. Austin, TX*, March, 295-299.

Sackrowitz, M., & Parelius, A. (1996). An unlevel playing field: Women in the introductory computer science courses. *ACM SIGCSE Bulletin, Proceedings of the Twenty-Seventh SIGCSE Technical Symposium on Computer Science Education*, March, 28(1), 37-41.

Simon, H. (1981). *The sciences of the artificial* (2nd ed.). Cambridge, MA: MIT Press.

Yin, R. K. (1994). *Case study research: Design and methods* (2nd ed.). Thousand Oaks, CA: Sage.

Appendix A: The Instructors

In this appendix, key elements of instructor background are summarized.

Instructor A

Instructor A, who designed and implemented the Ism3232.A and Ism6155 courses, was a tenured faculty member at the time of the case. He had entered academia, after 10 years in business and the military. With respect to Ism3232.A, his programming expertise was largely self-taught. His first commercial programming experience came between his first and second year studies for an MBA at a well known case method-oriented business school. Subsequent to graduation, he developed a thriving computer modeling consulting practice until, eventually, he returned to the same business school to get a DBA in Information Systems. While getting his DBA and during subsequent academic postings he continued to program commercially while, at the same time, developing an extensive library of self-authored case studies. He also authored a programming textbook, published by Wiley (Gill, 2005b). That textbook incorporated many of the innovations used in Ism3232.A.

With respect to Ism6155, the case-method approach that was central to the design of the course was highly familiar to him. He had completed his MBA at a well known university that relied entirely upon the case method for that program. Prior to and in the course of completing his DBA at the same university, he authored numerous case studies and employed the case method in his graduate teaching.

Instructor B

Instructor B was an untenured assistant professor with a strong research record who had joined the department in 2003. Although she had a substantial breadth of experience in teaching MIS-related classes, both at the university and prior to receiving her doctorate, Ism3232.B was her first programming teaching assignment. In addition, unlike Instructor A, she had never programmed commercially, although she had created a large application using the C++ programming language as part of her dissertation. To help her prepare for the course, she chose to sit in on Ism3232.A during the Spring 2007 semester. Although Instructor A encouraged her to use all the materials that he had created for her own course development (lectures, assignments, draft textbook), she felt uncomfortable doing so. In an email to Instructor A, she stated:

> The current design of the course is very self-paced and delivered primarily online. Remedial face-to- face lecture was provided to assist those who need personalized help with various topics. I discovered that the current structure was not in line with my teaching style and philosophy.
>
> My teaching style is active and interactive. Although, the current course [Ism3232.A] was active, I needed a more structured set of interactions with the students.

As a consequence of these feelings, she decided that she would redesign her section of the course to fit her own personal style.

Appendix B: Course Designs

Qualitative descriptions of the three courses are now presented. In the case of Ism3232.A, this description includes details on its evolution from a more traditional programming course, using the C and later C++ programming language, to its final form as a C# course.

Evolution of Ism3232.A

Unlike the other two cases to be discussed, Ism3232.A evolved into its unusual structure gradually, through a series of tentative introductions of elements. These are identified as a series of periods, each of which has represents either distinct characteristics or outcomes, summarized in Table 3 of the paper itself.

Period 0: Ism3232 at prior institution

The original course design, developed in the early 1990s was a traditional lecture course with exams and programming exercises. The first major modification to the course came in 1994 when a student—who had been receiving near perfect assignment scores and single digit exam scores—contended that his exam scores were not reflective of his knowledge. Although understandably suspicious, Instructor A orally quizzed him on the material. The unexpected result was that the student's overall course grade was immediately changed from a D to an A and the oral exam was made available for all other students as a means of compensating for poor performance on written tests.

By the late 1990s, Instructor A had made oral examinations an option for the last two project assignments in the course and allowed those project grades—if validated by oral exam—to be substituted for a final exam. By that time, nearly all the students in the course began taking the oral exam option—for those who did not, it was usually a consequence of not having completed the assignments. When individuals did not pass the oral exam, they received no credit for the assignments. They were, however, given the opportunity to take a second oral exam with the instructor.

Throughout period 0 Instructor A's course evaluations were considered excellent by the standards of programming courses, winning him the university's undergraduate teaching award in 1996. Retention rates were comparable to other MIS courses. At no time throughout the period were any doubts expressed regarding the effectiveness of the course. Thus, course fitness can be characterized as high during this period.

Period 1: Fall 2001

In Fall 2001, Instructor A joined another institution. Upon arriving, he transferred over the complete course design from his previous institution, which was quite similar in demographics and a member of the same state university system. To that design, he added one major improvement: narrated multimedia animated screen captures that were supplied to students on CDs. These segments demonstrated the use of Microsoft's Visual Studio programming tool and walked students through some the course exercises. In a teaching case study describing the course (Gill, 2006a, p. 7), the results of the class were described as follows:

> [Instructor A]'s first… classes were among the quietest he had ever encountered. Students virtually never initiated questions and seemed reluctant to respond to the numerous questions… posed to them during lectures. Nonetheless, their median score on the midterm was in the high 50s, slightly above what he was used to seeing at [his prior institution]. Student course evaluation forms were handed out very early, in mid-November. The reason for this was that the last three class meetings covered topics not related to assignments—so designed to give students time to complete their last two projects—and

> [Instructor A] knew they would be sparsely attended. Starting right before Thanksgiving, and continuing through the end of the semester in mid-December, [Instructor A] administered well over a hundred oral exams on the last two assignments. The general atmosphere of these exams seemed to be quite upbeat, with relatively few retakes being required. It seemed to [Instructor A] that the students had finally gotten their act together—although he naturally wished that they had started earlier.
>
> When student evaluations came back in January, [Instructor A] was in a state of shock. His overall instructor rating average of 2.63 (on a 1 to 5 scale, with 5 being best) was so far below anything that he had ever received in the past that he initially thought he was reading the scale backwards.

Another remarkable feature of the course was an extraordinary variability of student reactions to the course and instructor. These ranged from (Gill, 2006a, p. 19):

> I thought the course was wonderful. [Instructor A] made information for the class accessible in many, many ways. The CD for the class is the greatest thing. I wish I had other classes like this one. My overall evaluation of [Instructor A] is perfect. I have not had a better teacher at USF.

to:

> Up to this point I am still wondering why this monster became a professor. He is a self-righteous person. He needs to go back where he came from.

Retention was comparable to other sections of the course. Some argument might be made that the amount of material covered should count positively in assessing class fitness. In addition, based upon his observations and many face-to-face contacts with students during the oral exam period, Instructor A believed that the low evaluations were largely a result of the evaluation forms being distributed more than a month prior to end of the semester—before the oral exam process began—and therefore may have been strongly influenced by student grade anxiety. Nonetheless, the low teaching evaluations seemed to call for an overall course fitness assessment of low, particularly when contrasted with other semesters of the same course.

Period 2: Spring 2002

Gambling on his belief that the course was fundamentally better than its instructor evaluations suggested, in Spring 2002, Instructor A modified the course in relatively minor ways diametrically opposed to the many student comments indicating that the course workload was too high. He added another project with a required oral exam to the course due just prior to the course midterm. He also set up a discussion board on the university's Blackboard course management system to answer student questions. The result of these efforts—which did not fundamentally alter course content or design—was an extraordinary rise in student evaluations between Fall 2001 and Spring 2002, from 2.63 to 4.47 on a 1 to 5 scale (Gill, 2006a). With comparable retention to other sections of the course and strong student demand for Instructor A's subsequent sections of the class, the fitness of the Spring 2002 class could be reasonably characterized as high.

Period 3: Fall 2002 though Fall 2003

During the period from Fall 2002 to Summer 2006, Ism3232.A continuously evolved while, at the same time, being buffeted by changes in MIS enrollments that echoed national trends. At the time when Instructor A had joined his university, in Fall 2001, the principal concern had been on how to deal with the explosive growth in MIS majors, whose numbers had swelled from the mid-200s in the mid-1990s to over 1100 in 2001, making it the largest major in the university's College of Business. This growth had two practical consequences. First, high retention was not necessarily viewed as an important contributor to course fitness; some faculty instead argued that Ism3232 should be viewed as a gatekeeper course, screening out individuals not suited for the major. The second consequence was that Instructor A's department chair had encouraged the development of a more efficient delivery system. In Fall 2002, Instructor A began holding lectures for his three Ism3232.A sections (with combined enrollments of nearly 120) in a TV studio for one section, with undergraduate teaching assistants replaying the tapes in classrooms for the other 2 sections.

During this period, course evaluations dropped into the 3.5 to 3.9 range out of 5 (a drop also experienced by sections of the programming taught by other instructors). DWF rates of nearly 50% were also comparable to other sections not taught by Instructor A. In addition, what roughly constituted a peer review of the course took place in the Fall 2002-2003 period, when a major publishing house agreed to use the course content as the basis for a programming textbook (Gill, 2005b). As a consequence course fitness during this period could be characterized as medium.

Period 4: Spring 2004

By Spring 2004, drops in MIS enrollments had become highly noticeable and Instructor A became the sole instructor. During that semester, for the first time, Instructor A placed alternative multimedia versions of his lectures, as well as tutorial materials, on Blackboard. One immediate side effect of that action was described as follows:

> …median attendance at the live lecture section had dropped to 2 students (out of 35) and TAs reported that taped lecture replays were faring little better. At that time, the decision was made to eliminate traditional lectures altogether (Gill, 2006a, p. 6).

In addition, because nearly all students were validating assignments through oral exams in preference to taking written exams, the decision was made to eliminate written exams altogether. Because of the large number of changes being implemented, and overall course ratings continued their gradual decline to the 3.2 to 3.8 range (on a 1 to 5 scale), Instructor A also developed and began to administer an extensive (250+ question) survey of student perceptions of the class, provided in parallel with the university's 8 question survey. To provide incentive for survey participation, a flat grading scale (i.e., no plusses or minuses) was adopted, with individuals filling in the survey (submitted to a departmental secretary, so as to avoid any possible impact on grading) getting a + appended to their grade. This survey, derived from 3 separate instruments developed under the auspices of the National Science Foundation, allowed a clearer picture of the course's strengths and weaknesses to be developed. On the strength side, students reported:

- Satisfaction with the course's flexibility
- Enthusiasm for the degree of group work allowed
- Great enthusiasm for the availability of online content and support

In addition, course performance appeared to be entirely independent of student gender and prior programming experience (the latter being highly diverse, with roughly 50% of the class never having taken programming before, 25% having taken one course, and 25% having taken 2 or more courses). Both these results were highly unusual and desirable, since gender and experience-based performance differences were widely reported in the computer science educational literature (e.g., Goold & Rimmer, 2000; Hagan & Markam, 2000; Holden & Weeden, 2003; Roberts, 2000). The former, in particular, was also a source of great concern in the field (Sackrowitz & Parelius, 1996).

Less positive aspects of the survey included the fact that only about 15% of students anticipated that they would be programming at some point in their career, a general distaste for programming-related activities was indicated, and the self-reported weekly work load of the course was extreme (typically ranging from 15-20 hours/week). While Instructor A was highly suspicious of the precise self-estimates, there was a consistency in ratios across surveys. Specifically, students reported working twice as long on the course as they did on other MIS courses and three times as long as they did on other business courses (Gill & Holton, 2006).

Balancing the decline in evaluation scores was a gradual improvement in retention that began to appear in Spring 2004 (see Table B.1). In part, this was due to increased priority being placed on retention—since the department's principal challenge had rapidly changed from having too many majors to having too few majors. Making materials available to students in many different forms was part of this. Taken together, then, the overall fitness of the class could be characterized as medium.

Table B.1: Retention by semester [1] (from Gill & Holton, 2006)

Value	Spring '05	Fall '04	Summer '04	Spring '04	Fall '03
Enrollment	71 [2]	79	34	93	116
Passing	65%	63%	68%	54%	50%
D & F	13%	20%	11%	24%	19%
WD	19%	18%	22%	22%	31%
DWF	35%	37%	32%	46%	50%
Completing class (count)	57	66	29	72	80
A grades	38%	33%	0%	17%	26%
B grades	22%	23%	21%	17%	25%
C grades	24%	20%	66%	36%	21%
Passing %	84%	77%	86%	69%	72%
D & F	16%	23%	14%	31%	28%

[1]: A chi square test confirms that the pattern of passing and not passing students observed across semesters is different from what would be expected by chance ($p \leq .05$). These differences are notable in light of similarities in the student body across semesters: MANOVA detects no significant between group differences in age, pre-university programming course work, MIS and software development work experience, or MIS and software development work aspirations.

[2]: For the purpose of the analysis, excused incomplete ("I") grades were omitted from the enrollment figures

Period 5: Summer 2004

In Summer 2004, Instructor A instituted two new course policies. First, he required a personal individual meeting with each student so as to ensure each student understood the course's policies. Second, Ism3232.A became entirely self-paced, with all late penalties being eliminated. Prior to that time, each assignment had a specified due date, with late submissions resulting in a 10%/week late penalty. The logical problem with this arrangement was it meant that a chronically late C student needed, by the end of the semester, to earn enough points for a B in order to get a C grade. Given that there were no set test dates—oral exams were given on demand—there was no logistical justification for exacting such a penalty. Moreover, with the drops in MIS major enrollments, there was a strong incentive to provide positive experiences for students and not lose those students by virtue of their not fulfilling the requirements of the major (C grade or better), withdrawing, or failing—collectively referred to as DWF—in situations where they could ethically be passed.

The transition to a pure self-paced organization was not without excitement. During Summer 2004, with less than 10 days left in the abbreviated 10-week semester, only 2 of the 34 registered students had accumulated enough points for a C grade (Gill, 2006a). Instructor A's teaching assistants had begun to notice this phenomenon in early June (three weeks into the course) and, despite numerous emails requesting an explanation from the students, were at a loss to explain it. The bloodbath Instructor A had feared did not materialize, however. Instead, during the final 10 days of the course, the students submitted and validated sufficient assignments to pass. In fact, the 14% DWF rate for that summer was the lowest the course had ever experienced, and the 3.9 instructor evaluation was the highest than had been received for more than a year. Moreover, these results were achieved without a single A grade being awarded that semester.

With improved evaluations and retention, the fitness of this section could be characterized as medium/high. Working against it was the weak overall student performance. This can be verified by the grade distribution not having any A grades, since the structure of the course's grading curve placed a much higher weight on the number of modules completed and validated than it did on individual project grades (which tended to be close to 100% for most projects). Thus, overall fitness might be described as a high medium.

Period 6: Fall 2004 – Summer 2006

Buoyed by the dramatic improvement in retention and the slight improvement in instructor evaluations, Instructor A decided to keep the self-paced design. Not wishing to repeat the uncomfortable feeling of imminent disaster that he had experienced towards the end of the summer, however, he developed a novel progress monitoring system for the course. Under that system, each week students could acquire 5 points (out of 1000 for the entire course) by either filling in a web-based form (at a link Instructor A provided) or by making an entry to a personal journal hosted on the LiveJournal web site—where students could sign up for free accounts. Instructor A also developed software that allowed him to consolidate Blackboard grades, weekly check-in forms, and RSS feeds from LiveJournal automatically into personal progress summaries for each student. These progress summary reports were then inspected by Instructor A and emailed to each student weekly. After these changes, the retention gains persisted (see Tables B.1 & B.2) and, significantly, the amount of material completed the typical student grew. Instructor evaluations improved to the 4.0-4.3 range, comparable to other undergraduate courses in the major. Thus, fitness during this period could be characterized as high.

Table B.2: DWF versus passing grades, counts of students (from Gill & Holton, 2006)

	Self-paced	Not self-paced
Pass	119	108
DWF	65	101
p<0.01 likelihood that self-paced and not self-paced came from same distribution, using chi-square test.		

Ism3232.A: Final Version

By mid-2006, there were a considerable number of forces that favored making changes to the Ism3232.A design. Among these:

- The department's MIS enrollments continued to plummet (from 1100 MIS majors in 2001 to around 200 in 2006). With programming being a relatively unpopular element of the major, Instructor A had spearheaded an effort to remove a second, more advanced, required programming course from the major. That had, in turn, made introducing object-oriented programming in Ism3232.A important, since students might not encounter programming again in their studies.
- The 2005 version of the Microsoft Visual Studio tool used in the course had been released, meaning that online lecture and assignment content needed to be redeveloped.

Based on these factors, Instructor A decided to change the language taught in Ism3232.A from C++ to C#, the latter being inherently object-oriented, easier to explain, and more supportive of MS-Windows programming. The last of these was particularly attractive to Instructor A, since it would allow students to create programs that seemed more “real world” than the text-based programming exercises that he had required of his C++ students.

Based upon his student survey findings—where a strong minority had always expressed an interest in learning how to program video games—Instructor A completely redesigned the course around a game-based metaphor. In addition, during the transition period (Fall 2006), he realized that he was likely to have little or no teaching assistant (TA) support, primarily because he wouldn’t have any past students who knew the C# language or who had completed the newly designed exercises. For this reason, he developed a new assignment validation system which employed proctored multiple choice exams, delivered on Blackboard, as a substitute for oral exams (although the latter remained available upon request).

Another important change that occurred in Fall 2006 was a fundamental shift in how class time was employed. Prior to the new design, class time had revolved around a TA playing a recorded lecture (also available online) and answering questions. A fundamental weakness of this approach in a self-paced course was the wide range in student progress that emerged as the semester progressed. How interested would a typical student be in a lecture that related to an assignment that he or she would not be starting for another month? Perhaps even more significantly, how interested would a student be in coming in to watch a lecture that he or she could view from home? To address these, Instructor A eliminated in-class lecture session altogether, replacing them each week with:

- 75 minute interactive problem solving sessions, where the instructor posed problems to students who then attempted to solve them (using Tablet PCs provided to Instructor A through a Hewlett Packard teaching grant), after which the efforts were discussed.
- 75 minute help sessions, conducted in a computer lab, where students could take proctored validation exams or get help from the instructor in completing their assignments.

These sessions were optional, with attendance typically ranging from 15-25% of the entire class. An example of the software used in problem solving session, taken from an actual class, is presented in Figure B.1 showing student submissions to an assigned task.

The content for the course was selected to meet three key goals: A) to motivate students by presenting them tasks that they actually wanted to accomplish, B) to emphasize topics that would be of use in subsequent courses, including non-programming courses (e.g., databases), and C) to engage them in activities that would resemble the type of activities they might be expected to perform in an IT work environment. To meet goal (A), programming activities were designed around a game theme, with the three major course projects involving: 1) building a simulation of the institution's MIS major (designed around a fantasy game theme), 2) creating a Bingo player/caller application, and 3) creating an aquarium simulation, where the player needed to click on fish to keep them in a simulated aquarium. To meet goal (B), numerous topics from other courses were embedded into the projects. For example, the Bingo server incorporated programmatic links to an external database (highly relevant to the database required course). The MIS major simulation incorporated code to manage an embedded web browser (relevant to data communications) and media player (relevant to the multimedia design elective). Finally, to achieve goal (C), the course focus was on understanding and modifying large bodies of code (supplied by the instructor), as opposed to creating code from scratch. Such activities would be far more reflective of the types of tasks that an entry level IS employee would be assigned than would be creating complete, but very simple, programs.

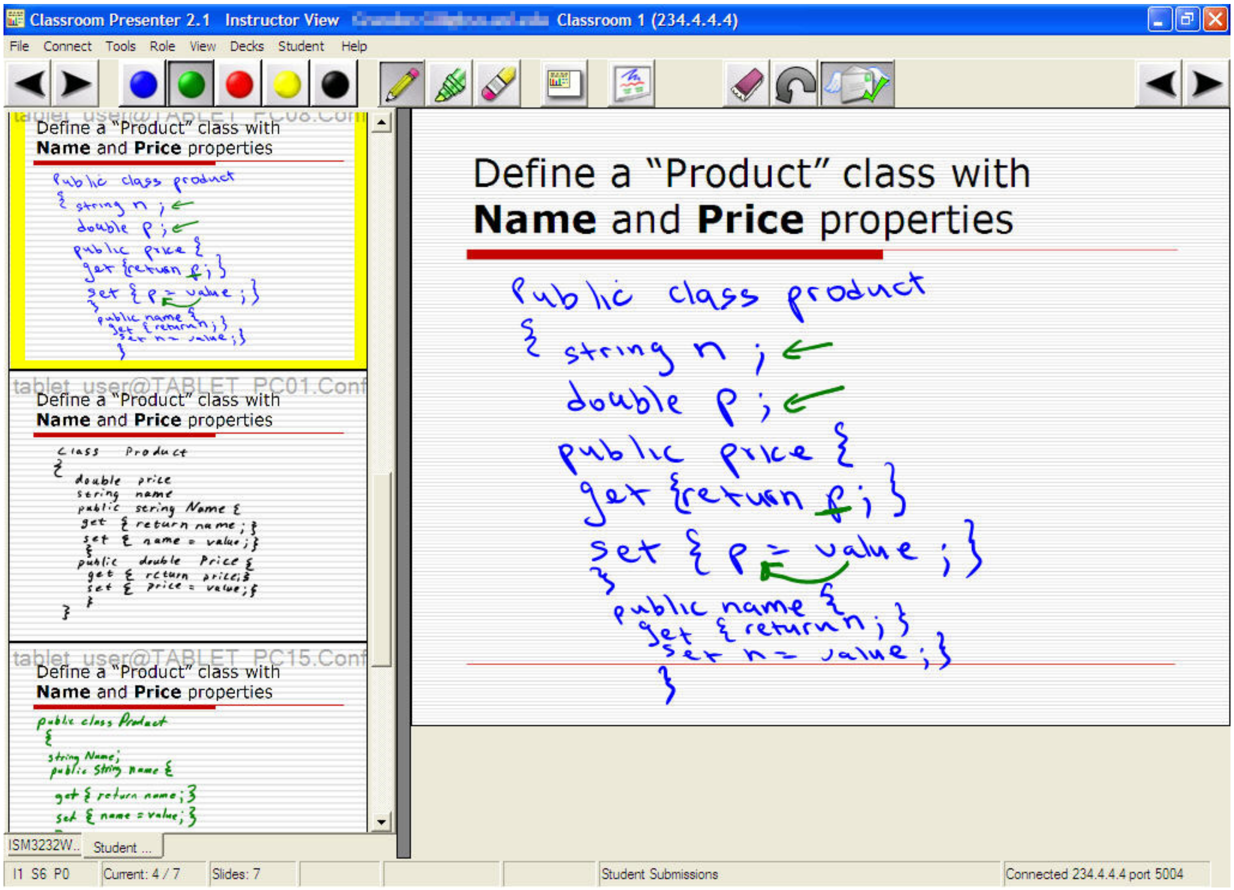

Figure B.1: Classroom Presenter allowed students to answer questions by writing on their Tablet PCs. Students submitted answers electronically to the instructor (see left column), who could then decide what answers to project and discuss (see right column, with instructor annotations [arrows], which is projected on the overhead without the left column being visible).

The results of Fall 2006 were highly positive, as summarized in Table B.3. In addition to the items summarized, student evaluations scores showed dramatic improvement on all 8 items. These positive changes persisted over the next 3 semesters. In Summer 2007, for example, the class experienced a DWF rate of 0%, for the first time in its history. In Fall 2007, the overall evaluation of the instructor climbed to 4.89 out of 5, the highest in the history of the course (and also the highest that Instructor A had ever received for a programming course) with a 2.5% DWF (1 F, in a class of 40 students). But in Fall 2007, another change occurred as well. A second version of the same class was offered, taught by Instructor B.

Table B.3: Results of Fall 2006 change to Ism3232.A

	Fall 2006	Spring 2006	Fall 2005	Spring 2005	Fall 2004	Spring 2004
Overall Evaluation (1=Poor, 5=Excellent)	**4.53**	3.94	4.13	3.38	3.88	4.00
Students Enrolled	**71**	70	77	86	82	91
Students Surveyed	**41**	28	34	39	36	34
Retention (% of A, B and C grades for combined sections)	**72%**	63%	61%	52%	61%	56%
Missing student % – Percent of students accumulating no points	**1%**	11%	13%	9%	10%	11%
Average grades of retained students (not DWF)	**3.27**	3.32	2.90	3.06	3.14	2.50
Average points accumulated (out of 1000) of students surveyed	**703**	725	640	713	681	585
Satisfaction with type of assignments (1=very dissatisfied, 5=very satisfied)	**4.00**	3.85	3.50	3.38	3.30	3.37
Satisfaction with multimedia content (1=very dissatisfied, 5=very satisfied)	**4.63**	4.25	3.78	3.75	3.75	3.66
Self-paced format (1=not helpful at all, 3=moderate help, 5=extremely helpful)	**3.78**	N/A (survey inst. error)	3.20	2.53	3.00	2.80

Ism3232.B

As Ism3232.A evolved and MIS enrollments plummeted, Instructor A had become the only faculty member teaching the course. By Fall 2006, a high priority of the incoming department chair was to ensure faculty redundancy for all required classes. Thus, he had requested that Instructor B teach a section of Ism3232, starting in Fall 2007. The experience of the section (termed Ism3232.B) is the second case to be discussed.

In developing her own version of Ism3232, Instructor B incorporated a mixture of elements, some quite traditional, some quite innovative. On the traditional side, she chose to use an established textbook (Deitel, Deitel, Hoey, & Yaeger, 2007), administer both midterm and final examinations (accounting for 40% of each student's grade), and enforce a strict set of deadlines for student work. In explaining her choice of textbook, she stated:

> …programming skills cannot be learned without spending a great deal of time experimenting with and writing code. I selected a textbook that taught the concepts by providing the students numerous examples and working applications that the students develop interactively using the various constructs. Each module contains fairly simple exercises as well as challenging tasks. I routinely choose a combination of these exercises as the basis of the labs. Depending on the students' confidence level I incorporate easier exercises to build self-assurance or more difficult exercises to challenge and extend their skills.

On the innovative side, nearly all instruction was conducted in a lab setting. Lab sessions would begin with a short lecture, followed by a programming activity. During the programming activity phase, students would individually complete problems provided at the end of each chapter of the textbook. Although the students knew in advance what chapter would be discussed, they did not know which problems would be assigned. Moreover, by about the third week of class—according to Instructor B—students determined that they needed to study the assigned chapter in advance of coming to class if they were to have any chance of completing the activity. Each lab was graded and the student's top 10 grades from the 12 sessions represented 60% of their final grade.

In explaining the philosophy that guided her course design, Instructor B made the following comments:

> I believe that true learning occurs through exposure to, experience with, and reinforcement of concepts. It is essential that all three steps are present to ensure ownership of the knowledge being transferred. I emphasize the latter steps as they support active learning. Students are more engaged when they are actively, rather than passively, involved in learning. I therefore chose to incorporate weekly in-class labs as a primary learning tool. I felt the labs would accomplish a number of goals:
>
> - Allow the students to practice the skills demonstrated in the text and during lecture.
> - Require students to stay on pace with the planned syllabus of topics. During my observation of the current course I found myself procrastinating and struggling to finish the projects at the end of the semester. Because of this I did not assimilate and retain the material as well as I could or should have. I was not incentivized with a grade for the class but feel that if I had fallen victim to procrastination some of the students would have as well.
> - Gain confidence and a sense of accomplishment by completing a functioning application with the instructor available to assist with problems. The labs were designed to be complex enough to be challenging yet be completed within the time allotted in class. The decision to stage the labs in class rather than as homework was motivated by a desire to prevent or reduce the frustration induced by hours spent trying to overcome a simple syntax error. By observing the students efforts in lab I would be able to provide guidance to possibly decrease the time spent struggling with a problem that a hint might resolve.
> - Prevent cheating. In-class labs avoid the opportunity for students to inappropriately collaborate or cheat on the homework. It is unfortunate that we must account for this possibility but the reality is that cheating is prevalent, especially at the undergraduate level.

For purposes of comparison, the lab exercises in Ism3232.B were very different from the programming projects employed in Ism3232.A. Whereas a typical Ism3232.A project could take 3-6 weeks for a student to complete and involve the student writing several hundred lines of code (in addition to the hundreds or, in one case, thousands of lines provided by Instructor A), a typical Ism3232.B lab assignment—which normally needed to be completed in under 2 hours—would rarely exceed a hundred lines of student-authored code. Similar to Ism3232.A, many of the labs consisted of extending fairly complex applications for which some of the code was previously written. While Ism3232.A focused on completing 3-4 projects, the Ism3232.B labs required students to complete 19 applications during the course of the semester. The core objectives of the two courses also differed slightly. Since this was an introductory programming course Instructor B's primary objective was to teach the core constructs of object-oriented programming. Namely, variables, memory concepts, algorithms, various visual controls, event handling, repetition constructs, choice constructs, collections and arrays, methods, and class concepts. Similar to Ism3232.A, the Ism3232.B labs reflected applications that could be encountered in a business environment, albeit on a smaller scale.

In assessing the outcomes of Ism3232.B, there is only a single data point—the Fall 2007 semester—where both instructors taught the course at the same time. Based on those results, however, the course design would have to be characterized as a spectacular success. Among the indicators considered:

- Instructor B's course evaluation of 4.79 was the second highest in the history of the course (with Instructor A's evaluation during the same semester being the highest). The numerical result was supported by highly positive student comments. It was also above Instructor B's average for other courses, although that was also very high.
- The DWF rate of 21% (4W, 1F out of 24) was well below the historical course average.
- Student performance on examinations indicated a high level of comprehension.

The last of these assertions is supported by Instructor A's own observations of the Ism3232.B exams. Prior to the administration of the Ism3232.B Fall 2007 midterm, he inspected the test that Instructor B had developed. Based upon the difficulty of the test and his extensive experience with undergraduate programming students, he confidently predicted a median of around 40% with somewhere around 20% of the class scoring at the level of random guessing. The actual exam results forced him to eat his words. With a median of nearly 80%, he conceded that Instructor B's students had scored substantially higher than his own students would have.

Ism6155.A

The final case to be considered is that of Ism6155.A, Enterprise Information Systems, the capstone course for the department's MS-MIS program. The course was introduced in Fall 2002 and its basic design remains unchanged to the present day. Combining many innovative aspects, in 2005 it won the DSI Innovative Curriculum Competition. Because the class was taught by Instructor A, including it in the analysis allows us to explore the relationship between instructor and course design.

Ism6155.A was organized around three activity streams: case discussions, debates, and a multi-semester research project. Although use of the case method in business education could hardly be described as ground breaking, the course introduced a number of new variations. These include: a) an instructor-developed case detailing a classroom uprising to introduce the case method to students, b) incorporating a classroom response system into case discussions, and c) experimenting with three different modes of discussion: classroom, asynchronous online, and synchronous online.

The debate pedagogy, nearly absent from the business education literature, facilitated focused discussions on topics of current interest. Topics were loosely synchronized with the cases being discussed, and each week about one third of the class was assigned to the panel—presenting the pro and con sides prior to opening the debate to general class discussion. Although students were given some choice regarding what topics they would prefer to present as panelists, they were given no choice of side—often forcing them to look at issues from new perspectives. A research project required each student to trace the evolution of two strategic information systems, chosen from an instructor-developed list, that were introduced somewhere between the late 1970s and early 1990s.

The project activity was intended to build research skills and foster an appreciation for how MIS has evolved. Over a scheduled three year period, each system was researched at least three times (using a data gathering instrument designed by the instructor). The ultimate goal of the project was to establish system histories sufficiently rigorous so as to be useful to the MIS community. A more complete description of the course can be found in Gill (2006b).

The case for the effectiveness of Ism6155.A was presented in the 2005 DSI competition entry, which stated identified the following indicators of effectiveness:

- *Student evaluations* of the course and instructor are far above college averages. The most recent set of evaluations [Fall 2004], with a 74% response rate, awarded both the course and the instructor perfect (5/5) scores—an event so noteworthy the department chair circulated a memo to the faculty.

- *High quality of student-prepared work*, with both debate preparation and research papers far exceeding the instructor's original expectations. Anecdotally, it is a rare debate where the instructor does not learn something material about the topic. Also, one manuscript—written by a doctoral student and inspired by observations made in project reports—is already under review.
- *High levels of effort,* with students reporting spending more time on the course than on their average MS course. By way of supporting evidence, the Fall 2004 consolidated research logs of 18 students came to 309 single-spaced pages (when imported into MS-Word).
- *End-of-semester survey items relating to course design* not only show students are satisfied with each course activity, but also show complete lack of consensus regarding any alternative design direction.
- *Enthusiastic participation in course activities,* such as the online class day—offered up by the instructor as a possible voluntary activity in late January 2005. (Amazingly, 17 of 19 students surveyed anonymously afterwards opted for a second online day, despite the extra effort required).

Biographies

Grandon Gill is an Associate Professor in the Information Systems and Decision Sciences department at the University of South Florida. He holds a doctorate in Management Information Systems from Harvard Business School, where he also received his M.B.A. His principal research areas are the impacts of complexity on decision-making and IS education, and he has published many articles describing how technologies and innovative pedagogies can be combined to increase the effectiveness of teaching across a broad range of IS topics. Currently, he is Editor-inChief of *Informing Science: The International Journal of an Emerging Transdiscipline* and an Editor of the *Journal of IT Education.*

Joni L. Jones is an Associate Professor of Information Systems and Decision Sciences at the University of South Florida. Her research interests include electronic commerce; specifically Internet based exchange mechanisms such as electronic auctions and marketplaces. Her work explores the efficacy and uses of auctions, market mechanisms and other complex, electronically enabled, variable pricing mechanisms in business settings. She is currently applying prediction market concepts for software engineering estimates and industrial decision making. Her publications include research articles in *MIS Quarterly*, *INFORMS Journal of Computing, Production and Operations Management, Journal of Management Information System, Decision Support Systems*, and other MIS and information science publications.