Steps towards a Scientific Approach to a Database Course Transformation: Data Collection and Analysis

Benjamin Yu, Edwin M. Knorr
University of British Columbia
Department of Computer Science
Vancouver, BC Canada V6T 1Z4
{benyu, knorr}@cs.ubc.ca

ABSTRACT

Instructors modify offerings of their courses in response to changes in emphasis, curriculum, student preparation, resource limitations, and problems with previous offerings. Changes also involve assessment instruments such as assignments and exams, ensuring that students are indeed *learning* the material, rather than relying on "information" from previous offerings. However, instructors are seldom guided by meaningful and useful data in making these changes. Instead, most rely on their "gut feelings", and many changes are made in an ad hoc manner, with minimal supporting data to assess whether the changes contribute positively or negatively to student learning. This paper reports on our experience with the transformation of an undergraduate database course at the University of British Columbia (UBC), using best practices from education research. We provide examples of the types of data that can be obtained through various instruments, and can be used in an objective analysis to affect course changes. If such data collection methods are put into place *before* changes to a course are anticipated, instructors will be better prepared to assess how students are affected by those changes.

This paper is organized as follows. Sections 1 and 2 provide background information about a process model for course transformation. Section 3 describes the course being studied. Section 4 shows different types of instruments that can be used for data collection. Section 5 provides examples of the data collected, and the analysis that can be performed using that information. Section 6 describes our plans for course transformation based on the data collected.

Categories and Subject Descriptors

K.3.2 [**Computers and Education**]: Computer and Information Science Education – *Computer Science Education, Curriculum.*

General Terms

Measurement, Human Factors.

Keywords

Data collection, data analysis, course transformation, teaching, learning.

1. INTRODUCTION

Changes are inevitable, especially for a conscientious teacher who constantly desires to improve his/her courses, and particularly in the fast-changing field of Computer Science. At UBC, under the Carl Wieman Science Education Initiative, significant effort has been put into evidence-based course transformation. The objective is to improve student learning, especially in the areas of problem solving and critical thinking. Another objective is to increase student interest in science education. Under this initiative, our Computer Science department has been active in making a number of changes to undergraduate courses [1,10].

While much effort has been made in educational research, it is often difficult for many instructors to know where to begin and even then, to know whether the changes make any difference to the students. In this paper, we present the initial stage of a scientific approach to course transformation of a database course. As in any scientific experiment, it is important to collect relevant data. This paper focuses on the types of data that an instructor can collect using various instruments, so that transformations can be evaluated to determine their effectiveness for student learning. Even though some of the data and results presented in this paper are specific to a database course, most of the ideas can be applied to other courses or disciplines.

2. PREVIOUS WORK ON COURSE TRANSFORMATION

To provide an overall *process* for course transformation, Wieman et al. [9] provide a case study of a physics course at the University of Colorado. The entire process involves the following steps:

1. Setting learning goals
2. Choosing presentation material
3. Creating assessment tools to evaluate student mastery of specific concepts
4. Evaluating resources available and resource needs
5. Structuring course details such as grading, choice of textbook, lecture format, and schedule
6. Preparing lectures
7. Getting student buy-in
8. Preparing assignments
9. Preparing exams
10. Hiring TAs
11. Evaluating learning and general aspects of the class
12. Passing the course on to a new instructor

This paper focuses primarily on Steps 3 and 11, with particular emphasis on the instruments that have been developed to collect student attitudinal data, as well as data to measure conceptual learning gains in the course. We maintain that these instruments should be put in place, and data be collected from students, *before* any significant transformation takes place.

WCCCE '10, May 7-8, 2010, Kelowna, Canada.

3. COURSE BACKGROUND

This study focuses on the first of a sequence of two upper-level database courses offered by UBC Computer Science. It is a third-year elective course that most of our undergraduates take. During the Fall 2009 session, 91 students were registered in the course, with 61 students being in a CS program (e.g., major, honours, or combined program with another discipline), 16 coming from Applied Science (Engineering), and 14 from other backgrounds. Of these 91 students, one was in 5th year, 32 were in 4th year, 49 were in 3rd year, eight were in 2nd year, and one was in 1st year. The students (77 male, 14 female) come from ethnically diverse backgrounds.

Prior to the course, the instructor (also the second author of this paper) created a set of detailed learning goals. Learning goals tell students what they are supposed to learn in the course, and provide them with a learning structure for exam preparation and self-feedback [8]. Essentially, they form a "contract" about what will be delivered or accomplished in the course.

4. DATA GATHERING ACTIVITIES

In this paper, we report the use of five different instruments to collect student data: (1) attitudinal surveys; (2) pre- and post-tests; (3) traditional assessments such as assignments, midterms, and final exam; (4) student interviews; and (5) data collected from a Learning Management System (LMS)—namely, Blackboard/WebCT Vista.

4.1 Attitudinal Surveys

Attitudinal surveys are used, at different stages of the course, to gain insight into student expectations and attitudes, including their motivations for taking the course, the amount of effort they perceive this course will require, and how much effort they are planning to put into it. It should be noted that this is not the same type of attitudinal survey that has been used in other science disciplines to characterize student beliefs about science and learning. An example of the latter is the Colorado Learning Attitudes about Science Survey (CLASS) [2] which captures student attitudes about science and connections to the real world, including students' personal interests, sense-making, conceptual connections, applied conceptual understanding, and problem solving capabilities. As far as we know, there is no similar attitudinal survey instrument in CS. Our survey is less specific and focuses mostly on the students' personal interests and expectations of the course.

We conducted three attitudinal surveys: at the beginning, middle, and end of the Fall 2009 term. Some of the questions we asked at the beginning of the term include each student's:

- main reason for taking this course
- expected number of study hours per week
- expected final grade
- areas that excite or worry them the most
- plans after graduation

In many schools, course evaluations are only performed at the end of the course—and this is obviously too late to help the students in the current course. The middle-of-term survey was used to provide feedback to the instructor so that adjustments could be made. The questions included:

- pace of the course (e.g., too fast, too slow, about right)
- effectiveness of the teaching methods, lectures, clicker questions, textbook, tutorials, and assignments
- areas that are working well, or need improvement

Lastly, the end-of-term survey was used to gauge the students' overall perceptions of the course. Questions included:

- actual number of study hours per week
- amount of effort required for this course, compared to other courses they took during the term
- frequency of attendance in lectures and tutorials, and the reasons for their attendance and absence
- percentage of completed readings from the textbook
- expected final grade

4.2 Pre- and Post-Tests

These in-class tests are identical tests to assess student learning gains from the course. The pre-test is administered on the first day of class, and the post-test is administered at the end of the course. They are not for marks, but include questions that are related to the concepts being covered. Their primary function is to serve as an indicator of the effect of changes in teaching methods on student learning. They are used to assess student learning *gains* in the course, as a better alternative to absolute (actual) marks because some students come into the course with some database knowledge or experience, and the only way to measure the incremental gain coming from the course (for such students) is to establish a baseline.

Most students are not expected to score well in the pre-test, and we mention this to them to ease the pressure. Several iterations of the pre-test are required to provide a good baseline, and we are already aware of some of the shortcomings of our initial attempt.

Our pre-test also had a *diagnostic* component that measured prerequisite knowledge. It is well-known that students come into a course with various degrees of mastery of prerequisite material. For example, some students come in as transfer students from other institutions, and others may have barely passed their UBC prerequisites. Thus, some students may already have obstacles to overcome in trying to master our database course. Instructors sometimes wonder if their students are adequately prepared for their course; but, without an appropriate diagnostic tool, all of this is speculation. Furthermore, it is often puzzling to know *what kind of help* to give students, or which areas to focus on before presenting them with more challenging material.

Coming into the course, most students have no awareness of the technical terms used in the course (e.g. serializability, foreign key, functional dependency, and normalization), even though they might have some familiarity with the concepts. A good pre-test should present material without technical terms, yet still be able to measure the conceptual knowledge that students already have. Then, we can determine learning gains due to the course. The bottom-line is that in order to evaluate the knowledge and shortcomings of incoming students, and to do something about these aspects, we first need to *measure* what the students know.

Pre- and post-tests are created based on the *learning goals* for the course and should be validated by extensive student interviews. After we perform a sufficient number of these interviews, we plan to revise the wording of the questions, or replace certain questions with better ones. Without student interviews, we can only speculate as to what our students' misconceptions are. During our course, the tests were only "semi-validated" through six student interviews, due to a lack of time. A more extensive set of student interviews, and refinements, is necessary. Some of the questions asked in our pre/post-tests involve:

- interpreting an Entity-Relationship (E-R) diagram containing various relationships and cardinalities
- modeling a many-to-many relationship (normalization principles)
- set notation (diagnostic component)
- B+ trees (diagnostic component)

4.3 Traditional Assessment Instruments

Traditional assessment instruments include midterm exams, assignments, and final exams. The key to creating meaningful assessment questions is the set of learning goals defined prior to the start of the course. Each question or problem should have a learning goal associated with it.

4.4 Student Interviews

Although student responses from the aforementioned instruments provide significant amounts of data about the students' knowledge and understanding, they usually do not provide insight into the students' mental processing in their derivation of solutions to specific problems. One-on-one student interviews in which students are requested to "think aloud" [5] as they work through questions, are especially useful. Students are encouraged to verbalize their thinking process as they encounter each problem. It is important that the interviewer not interrupt the student at any point during the interview, and not make any comment about whether or not the student is solving the problem correctly; otherwise, bias may be introduced. It is through these interviews that we gain insight into the student's thinking process, and in particular about how the student approaches a problem, relates the problem to something seen before, and proceeds when stuck.

To gain an overall perspective on how students solve database problems, students from different grade categories (e.g., A–, B, C+, D) were invited to interviews with an education researcher, after the first midterm. (These one-on-one student interviews are best done by someone other than the instructor of the course to preserve student anonymity and to avoid potentially biased answers.) Some of the problems used in our interviews involve:

- creating an E-R diagram from a description
- interpreting an E-R diagram, and extracting the context behind the diagram
- converting an E-R diagram to a relational model
- interpreting the relationships of a few entities, given a set of SQL statements

4.5 Learning Management System Data

The course uses an LMS to facilitate discussion via a moderated bulletin board about topics such as instructor announcements, pre-class readings, assignments, lectures, and tutorials. A number of reports can be generated by the LMS to track student usage of the system, including: number of visits to a given page, number of user sessions, average session length, most active days, most active times, etc.

5. DATA ANALYSIS

This section provides some of the actual data collected in the Fall 2009 offering of the course, and describes how the data was used in our analysis to provide insights for future changes.

5.1 Attitudinal Survey Analysis

Of the class of 91 students, only 39 students completed the online attitudinal surveys at *both* the beginning and the end of the course. (More specifically, 76 students completed the first survey, 58 completed the middle-of-term survey, and 43 completed the end-of-term survey.) Since we are most interested in changes in students' attitudes, the data analysis is limited to less than half of the class.

The attitudinal survey conducted at the beginning of the course shows that 54% of those students indicated that gaining a skill for employment was their main reason for taking this course, whereas 34% indicated that this was their second reason. 28% indicated that their main reason for taking the course was the course content, whereas 43% indicated that this was their second reason. 5% admitted that getting "easy" credits from this course was their primary reason for taking it! 53% of the students planned to find a CS-related job upon graduation, but 14% planned to continue into grad school. This data is consistent with the number of students who indicated that gaining employment skills was very important.

At the middle of the term, students gave their perceptions about the usefulness of certain aspects of the course with respect to learning. This data is summarized in Table 1.

Table 1. Usefulness of Course Activities to Student Learning

Course Activities	Percentage of Respondents
Lectures	86%
Clicker Questions	81%
Tutorials	58%
Assignments	76%
TAs' Office Hours	21%
Instructor's Office Hours	29%

This offering of the course was the first time that clicker questions with peer instruction [4] were used in the lectures. The students overwhelmingly felt that this contributed to their understanding of the material. This is consistent with research results on clickers and peer instruction [4,7].

The students' perception of the usefulness of the tutorials is disappointing. Only about 45% of the class regularly attended the tutorials. This data is an indicator of where effort should be placed in course transformation—in fact, that is where we are currently placing our efforts. Indeed, when marks are not awarded for tutorials, and when deliverables are not required, busy students tend to avoid attending tutorials that would otherwise help them. We have found that awarding marks for *both* participation and deliverables during the tutorials, in another course we are currently involved in, appears to make a difference. In that course, attendance at tutorials has tripled from earlier offerings; but, we still need to validate claims about attitude and ability. In our opinion, time spent in putting together well-structured, relevant tutorials that students take seriously, is one of the best investments we can make.

Another comment that we wish to make about Table 1 is that, typically, only a very small percentage of students attend office hours, even though many more should! Research has shown that one-on-one instruction (i.e., personalized or customized tutorials with immediate feedback), is one of the most effective ways of improving learning [3]. We are usually very generous with the allocation of office hours. Office hours are scalable: if more than a few students show up in the instructor's office, we look for a spare meeting room with a whiteboard, and continue with group office hours. Students often benefit from listening to each other's questions, and many stay for the entire session (e.g., 1.5 hours).

Expectations of final grades were tracked at the beginning and end of the term. The data can be used to tell whether students have been receiving sufficient and accurate feedback during the course about their progress. Table 2 summarizes the data collected.

Table 2: Student Grade Expectations and Actual Final Grade

Expected Grade:	50-67%	68-79%	80-100%
Start of term	0%	44%	56%
End of term	8%	49%	44%
Actual Grade	10%	28%	62%

Students often come into a course with high expectations, but gradually refine their expectations as feedback from midterms and assignments comes in. Our experience with other courses has been that many students underestimate the amount and difficulty of work that upper-level courses in CS entail.

The average final grade for this course over the past dozen offerings is about 74%. In our survey, nine students lowered their final grade expectation at the end of the term, one increased her final grade expectation, and the rest remained the same. However, 10 of the 39 students received a final grade higher than their end-of-term prediction, and only four received a lower grade than their end-of-term prediction. (Only 44 out of 91 students actually received a grade of A– (80%) or better. Thus, the students who completed both surveys seem to be the better students.) We tried to make all of the exams for the course about the same level of difficulty. Further study is needed to gain additional insights into students' self-assessment about their learning abilities and instructor/student expectations.

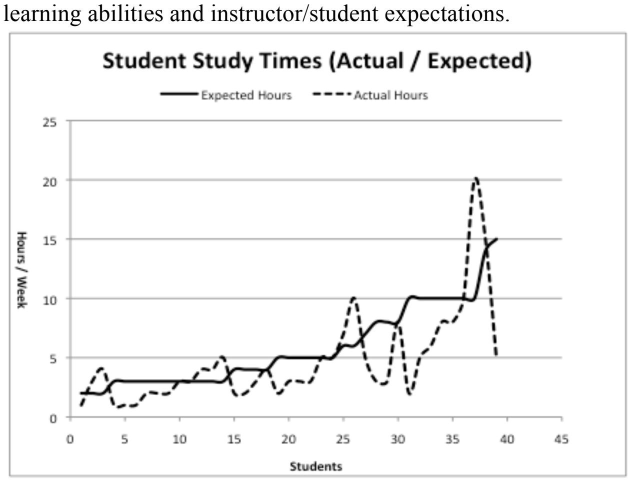

Figure 1: Students' Expected and Actual Hours Spent

The graph in Figure 1 (student # on x-axis) shows that the *actual* number of hours that students self-reported for the course (dashed line) can be significantly less than the number of hours that they *expected* to spend (solid line). Numerous explanations are possible. Perhaps: (a) they had less time to devote to the course than expected; (b) the workload was less than expected; (c) their learning styles were more in sync with the way the course was run; (d) the lectures were better than expected; (e) the tutorials were better than expected; (f) the course resources (textbook, online materials, sample problem sets, etc.) were better than expected; or perhaps (g) other courses were simply more "challenging" in one form or another, than this course. Further study is necessary before any conclusions can be reached. Nevertheless, such survey data should be used as a feedback mechanism to ask additional questions that can be included in future surveys, or on an individual basis via interviews.

Other findings from the attitudinal survey data are: (a) less than half of the class attended tutorials seven or more times (out of 10); (b) almost 90% of the students own the textbook (or have easy access to a copy); (c) about 70% thought the textbook was useful; (d) surprisingly, about 50% believe there is a "geek gene" that is needed to be successful in computer science; and (e) about 90% have little to no difficulty with English (we mention this because the majority of our CS students come from ethnic minorities).

We believe that rapid feedback is important to students, and to instructors. That is why we used clickers with peer instruction, actively responded to bulletin board postings, tried to get midterm exams and assignments back reasonably quickly, and posted solutions to theory-based homework in time for the midterms and final exam. We provided sample exam problems and solutions, so that students were better prepared for exams. Also, we had two two-stage midterms [10], each having an individual and a group component. The group part had the same questions as the individual part for Midterm 1 (providing feedback within minutes for each group of four students, but we also included isomorphic and new questions for Midterm 2 (providing less feedback, but still encouraging groups to work together to solve the problems).

In future terms, we would like to have shorter and more frequent assignments, so that students can get feedback even sooner, and so that the course is divided into more manageable chunks. Furthermore, based on student feedback provided by the three attitudinal surveys and the pre- and post-tests, we believe that the tutorials should be more focused, perhaps having marks for both participation and deliverables.

5.2 Assessment Data Analysis

It is often useful to track how students perform on various assessment instruments throughout the course to gauge their understanding. Correlations of pre- and post-test scores with traditional assessment instruments are summarized in Table 3.

Table 3: Correlation of Assessment Instruments

	Pre-Test	Mid-term 1	Mid-term 2	Post-Test	Final Exam
Avg. Score:	53%	70%	71%	68%	74%
Minimum:	12%	24%	26%	32%	44%
Maximum:	85%	97%	100%	85%	98%
Correlation					
Pre-Test	1.00	0.47	0.48	0.48	0.43
Midterm 1	0.47	1.00	0.78	0.49	0.77
Midterm 2	0.48	0.78	1.00	0.56	0.81
Post-Test	0.48	0.49	0.56	1.00	0.61
Final	0.43	0.77	0.81	0.61	1.00

The data shows that the first midterm has a high correlation with the second midterm (0.78) and the final exam (0.77). Similarly, the second midterm has a high correlation (0.81) with the final exam. This suggests that student learning, as demonstrated in the midterms, correlates with their overall understanding of the material covered in the final exam and the course overall, and also explains the expectation that many students had of their final grades (as discussed earlier).

Pre- and post-test data show moderate student learning gains in the course, although there is little or no correlation with traditional assessment instruments. This may indicate that the questions in the pre- and post-tests differed from the types of questions asked in other assessment instruments. We plan to make several iterations to fine-tune the pre- and post-tests, and we plan to conduct more extensive student interviews to validate those tests.

5.3 Student Interview Data Analysis

All students who were interviewed followed a similar procedure in creating an E-R diagram from a description. After reading the description, they were asked to underline the key concepts. Not surprisingly, stronger students could identify the attributes from the entities a lot faster, and more completely, than the weaker students. The stronger students were better able to distinguish between a conceptual model and a relational model. This was seen when the weaker students attempted to (incorrectly) include foreign keys in their E-R model. Stronger students better understand the notation and abstraction of an E-R diagram, especially when asked to read and interpret one that they had not seen before. Stronger students are also able to extract essential information, explicitly and implicitly stated in SQL statements that involve referential integrity. This correlates with the "*explain in plain English*" problems studied by the BRACElet group which studied students' abilities to read and explain a piece of code [6].

5.4 LMS Data Analysis

According to the reports generated by the Learning Management System, the number of page accesses—both in the Discussion groups (bulletin board) and otherwise—ranged from 18 to 3095. The student with the highest number of page accesses was almost twice his nearest "competitor" and three times his next nearest competitor. Surprisingly, there does not seem to be any correlation between page accesses and the final grade!

Other than the Assignments page and the Main page, the Tutorial discussion pages are the next highly accessed pages. As reported earlier, less than half of the students attended 70% or more of the tutorials. It was not clear whether the tutorial material presented in person was not useful, or if students got all the information they needed from the online resources and the textbook. The LMS report indicates that students refer to the different tutorial discussion pages frequently. The report also provides insight on specific topics that generated a significant number of discussions. These, in turn, provide some hints about the areas in which students were having problems. These include technical problems of accessing Oracle from a remote site, specific assignment problems, writing Java/JDBC code, and accessing SQL*Plus.

6. FUTURE WORK / CONCLUSION

The work presented in this paper is intended to describe the initial stage of a scientific based transformation of an undergraduate database course. We maintained at the outset that prior to any course transformation, a good set of data collection instruments should be put in place to: (a) understand what areas of the course need improvement, and (b) measure whether or not the changes made any difference. Based on the information derived from the data analysis, subsequent work will focus on: (a) how the course is transformed, (b) how the experiments are designed to assess the usefulness of these changes to student learning, and (c) how the data from the transformed course compares with existing data.

In this paper, we have implemented a number of data collection instruments and collected a significant amount of data. We have identified several areas where changes can be made, *and will be made*, such as designing better tutorials. We plan to include material that will enhance student employability upon graduation, without compromising or sacrificing other parts of the course. Also, we plan to provide more accurate feedback to students on their understanding of course material throughout the course.

The authors would like to acknowledge funding provided by the Carl Wieman Science Education Initiative for this study, and for Carl Wieman for his input.

7. REFERENCES

[1] Acton, D., Voll, K., Wolfman, S., and Yu, B. (2009). "Pedagogical Transformations in the UBC CS Science Education Initiative". *Proc. WCCCE 2010*, pp. 3-8.

[2] Adams, W., Perkins, K., Podolefsky, N., Dubson, M., Finkelstein, N., and Wieman, C. (2006). "A new instrument for measuring student beliefs about physics and learning physics: the Colorado Learning Attitudes about Science Survey". *Physical Review Special Topic - Physics Education Research*, 2(1). Survey available at: http://class.colorado.edu.

[3] Bloom, Benjamin S. "The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring". (1984). *Educational Researcher*, 13(6), pp. 4-16.

[4] Crouch, C., Watkins, J., Fagen, A., and Mazur, E. (2007). "Peer Instruction: Engaging Students One-on-One, All at Once". *Research Based Reform of University Physics,* 1(1).

[5] Ericsson, K.A. and Simon, H.A. (1998). "How to Study Thinking in Everyday Life: Contrasting Think-Aloud Protocols with Descriptions and Explanations of Thinking". *Mind, Culture, and Activity.* 5(3), pp. 178-186.

[6] Lister, R., Simon, B., Thompson, E., Whalley, J., and Prasad C. (2006). "Not seeing the forest for the trees: novice programmers and the SOLO taxonomy". *ACM SIGCSE Bulletin,*38(3), pp. 118-122.

[7] Ribbens, E. (2007). "Why I Like Clicker Personal Response Systems". *Journal of College Science Teaching*, November 2007, pp. 60+.

[8] Simon, B. and Taylor, J. (2009). "What Value are Course-Specific Learning Goals?" *Journal of College Science Teaching*, 39(2), pp. 52-57.

[9] Wieman, C., Perkins, K., and McKagan, S. (2007). "Course Transformation Case Study". Retrieved on December 24, 2009 from http://www.cwsei.ubc.ca/resources/files/Course_transformation_case_study.pdf.

[10] Yu, B., Tsiknis, G., and Allen, M. (2010). "Turning Exams into a Learning Experience". *Proc. SIGCSE 2010*, pp. 336-340.