Using a Course in Computer-Aided Engineering to Foster Life-Long Learning

William C. Lasher
Associate Professor of Mechanical Engineering, Penn State Erie, The Behrend College
Erie, PA 16563-1701 lasher@psu.edu

Abstract - **This paper describes how a course in Computer-Aided Engineering has been used as a venue for developing a life-long learning mentality in students. This course covers numerical methods and a variety of software packages, and due to the rapid rate of change in software is an ideal place to teach life-long learning using multiple techniques. The motivation for life-long learning is introduced through a class discussion regarding computers and software. Students are then asked to learn how to use three different software packages with minimal formal instruction. Additionally, a technique formerly developed by the author is used to show students how to interpret a technical text and learn how to learn numerical methods from the text. Survey and exam data show that these techniques have been successful in enhancing the motivation for life-long learning and providing students with some of the tools necessary to accomplish this.**

Index Terms – Life-long learning, self-learning, EC 2000.

INTRODUCTION

It has always been a goal of University education to develop in students a foundation for life-long learning. Critical thinking, problem solving skills, and a strong background in fundamentals form the basis for life-long learning, and these have historically been key components of engineering education. ABET's Engineering Criteria (EC) 2000 require that students must demonstrate "a recognition of the need for, and an ability to engage in life-long learning," which means that engineering educators must not only prepare our students for life-long learning, but measure that our graduates recognize its importance and are in fact equipped for it. In addition to accreditation requirements, the increasing pace of technological change, the emergence of new fields such as nanotechnology, and a global business climate demand that students continue their education well into their careers.

Much has been published on what life-long learning is and how to promote and assess it. Marra, Camplese and Litzinger [1] present a literature review on life-long learning, especially as it relates to EC 2000. They note that students require both the motivation for learning and the skills necessary to achieve it. This means helping students see education as relevant and something they can achieve, and also developing their metacognition (or their ability to think about their own learning). They also comment on the conflict between content and process, saying that "as long as we as educators see our main task as 'covering' the material, we will never pause long enough to help our students learn to learn on their own."

Other researchers support the idea that both motivation and thinking skills are essential for developing life-long learners. Wells and Langenfeld [2] report the results of a survey of graduated engineers and their supervisors, noting that they "rated the attributes of 'adaptability to change', 'desire to learn', and 'good thinking skills' as the most important personal qualities for life long learning." Healy [3] presents an approach to life-long learning based on student motivation and understanding the dimensions and styles of learning. Gandolfo [4] argues that the development of metacognition itself will positively influence student attitudes of and motivation for life-long learning.

A weakness in traditional engineering education is that the motivation of the need for life-long learning and development of the thinking processes required for it are implied rather than explicit. Engineering faculty frequently comment that courses such as computer programming and math develop problem solving and critical thinking skills; however, this is not clear to many students who see these courses simply as hurdles that must be overcome. Briedis [5] argues that proactive strategies are needed to "jump-start interest in and appreciation for lifelong learning."

Assessment of life-long learning attitudes is also addressed in the literature. Litzinger and Marra [6] discuss formal assessment instruments, but state there is a question as to whether they are a good measure of self-directed learning ability. They also note that researchers assert that self-directed learning ability is highly contextual, which suggests that assessment of life-long learning attitudes may be better accomplished in specific courses. While assessment is not the major focus of this paper, it is clear that assessment of life-long learning presents a challenge, and further research is indicated.

The author has been addressing life-long learning in several courses, but one of the most obvious places to do this is a course in Computer-Aided Engineering (CAE). This course presents an overview of numerical methods, as well as an introduction to several computer packages. Due to the rapid pace of change in computer tools, this is an ideal place to discuss the need for life-long learning and to develop self-learning skills. The approaches used in this course are consistent with the literature on life-long learning: motivate

the need for it, and develop critical thinking skills necessary to support it.

SETTING AND OVERVIEW OF THE COURSE

Penn State is a complex, multi-campus University spread over 24 separate locations, with the main campus located at University Park. Behrend College, located in Erie, is a four-year, primarily undergraduate college of the University, and offers 31 baccalaureate majors, including Mechanical, Electrical, Software, and Computer engineering. Engineering students can complete the first two years of their education at any Penn State campus, then transfer to University Park, Behrend, or the campus in Harrisburg to complete the final two years (or upper division) of their program.

The CAE course is a required junior-level (3rd year) course in the Mechanical Engineering program offered during the fall semester. Due to a high number of transfer students from other colleges (both within the Penn State system and from other institutions) this course serves as an entry point into the upper division program and is a good place to bring students with different backgrounds to a common level. The course description is: introduction to the tools and techniques of computer-aided design, including CAD (Computer-Aided Design), spreadsheets, numerical methods, and finite element analysis. The outcomes for this course are listed in Table 1.

TABLE 1
COURSE OUTCOMES

1. Develop a spreadsheet to perform common and typical engineering calculations, and be familiar with the capabilities of MATLAB.
2. Create a solid model of a simple, single part, calculate mass properties, create fully dimensioned 3-view drawings, and use the solid model for developing finite element meshes.
3. Perform a linear finite element analysis of a simple, single part.
4. Create simplified analysis diagrams (Free-body diagrams, block diagrams or control volumes) of realistic systems or components and develop an appropriate mathematical model for the system or component.
5. Use an appropriate numerical tool and numerical method to analyze and optimize a mathematical model for design purposes.
6. Assess the validity of a numerical solution to an engineering problem, including numerical accuracy and appropriateness of modeling assumptions.
7. Learn how to use a new computer package with minimal formal instruction.

Some of the topics are redundant depending on where students started their education. For example, students who started at Behrend will have already been introduced to the solid modeling package Pro/Engineer, whereas students who transfer from another school may have learned other solid modeling packages. A similar situation exists with Matlab; some transfer students learned this package in their programming course, whereas students at Behrend are taught C++. As can be seen in the outcomes, this course does not focus on a few topics, but presents a broad range of topics, with the intent of providing familiarity rather than deep expertise. Most of the tools and techniques presented here are used in later courses, and by repeated exposure the students generally develop a strong ability to use the tools and techniques by the time they graduate.

LEARNING PACKAGES

One of the major problems facing engineers today is the proliferation and continual revision of engineering software. This provides a perfect situation for convincing students of the need for life-long learning. During the first class period this issue is addressed through a discussion entitled "the problem of packages." Students are asked to simply name CAD packages with which they are familiar, and the instructor writes the name of the packages on the board. This typically produces a list of 20-30 different packages. The question is then asked: which CAD package should you learn, and why? The students quickly understand that everyone has their favorite, and that wherever they end up working, it is unlikely that they will have learned the specific CAD package used there. The instructor then presents comments from the Mechanical Engineering program's Industrial Advisory Board regarding the question of what an engineer should know about computer software (Table 2).

TABLE 2
INDUSTRIAL ADVISORY BOARD COMMENTS REGARDING WHAT ENGINEERS SHOULD KNOW ABOUT COMPUTER SOFTWARE

1. Most engineers do not use CAD systems, although there are exceptions.
2. Engineers should know how to pull up a drawing and get dimensions.
3. The companies will provide the necessary training.
4. Some engineers do FEA, but mostly this is a specialty, and graduates will have access to an expert.
5. It is much more important for engineers to know when an FEA is appropriate than how to run the package.
6. Much drafting and FEA are being done overseas.
7. "Collaborative tools" are becoming much more important.
8. Most engineers to not write programs, although programming is a good way to teach structured thinking.

The importance of having a basic literacy of common packages with an understanding of package capabilities is discussed, as well as the importance of knowing how to learn new packages and revisions. Many students enter the course assuming that the major focus will be on learning a CAD package in a structured laboratory setting, and are disappointed to find out that this is not the case. To further explain why the focus is on self-learning, a story is told about something that happened several years ago. The author was ill on the day of a schedule computer lab, but had prepared a handout that explained to the students what they had to do. He called one of the office assistants and asked them to distribute the handout to the students and explain the situation. The students proceeded to work through the lab without an instructor present. At the next class period the author asked how the lab went. Several students commented that it was the best lab of the semester because the instructor wasn't there. This elicited a chuckle from the class (including the author), at

which point the students added that the absence of an instructor forced them to think through the assignment and work with other students instead of simply raising their hand and asking a question. They clearly understood that this was better for learning. From that point on the author decided to never again "teach" a computer package through a structured lab (although lab sessions are held to introduce students to the software and make sure they understand the user interface).

At this point the students have "bought in" to the motivation for self-learning. The instructor asks them if they would like a methodology for learning software (the answer is always yes), and the handout shown in Table 3 is distributed.

TABLE 3
METHODOLOGY FOR LEARNING NEW SOFTWARE

1. Get an overview of the package
   - Overview seminars, classes, someone who knows the package
   - Need a good idea of the capabilities of the package
   - Need to understand how the package "thinks"
   - Play around with the menu structure
   - Need printed manuals (usually provided in electronic form) and/or tutorials – on-line help files are of little value for overviews
   - Find or develop your own summaries of procedures for the package
2. Find examples similar to your application
   - Examples manual or tutorial
   - Need to understand strategy for modeling your problem
3. Tackle a problem you want to work on
   - Start with simplified version of problem and progress to more complex version
   - Make up mini-problems to understand how to do specific tasks
   - Don't hesitate to throw out the current solutions and start over using a different approach
4. Use repeatedly and as much as possible until you know the package
   - Learning comes with time and practice
   - Do things you might not normally do with the package just to increase your exposure to it
5. When and how to get help
   - Make sure you read the manuals (or help pages) before asking for help
     - you might solve the problem on your own
     - you will better formulate the question you have
     - you will better understand the answer you get
   - Contact local experts next
   - Locate web pages or sign up for exploders
   - Searchable tech support databases are available to maintenance subscribers
   - Contact help line as last resort

Learning a new package is not difficult, but it does take work. Expect to spend a significant amount of time dealing with problems.

The three packages described earlier (Pro/Engineer, Matlab and ANSYS) are introduced at various points through the course, in parallel with the numerical methods topics. Pro/Engineer is the first package students learn. They purchase a tutorial and are simply asked to work through it. A special lab section is scheduled to make sure students get off to a good start. Since many of the students have already used Pro/Engineer in their freshman course (but may have forgotten the details), students generally have little trouble or are able to work things out with help from a fellow student. It is not unusual for a section of 30 students to be able to complete the first lab without asking the instructor a question. At the end of the lab session the class is asked if they feel that the presence of an instructor is necessary, and the answer is always no. By using a simple introductory tutorial on a topic where some students already have some familiarity, confidence in self-learning is built, and for the most part a feeling of the need for someone to help them through the material is dissipated (although not completely eliminated in all students). Asking the question as to whether an instructor's presence is necessary reinforces this in the students' minds.

Students continue to work through the Pro/Engineer tutorial for the rest of the semester at the rate of about one tutorial per week. They are given an assignment for each tutorial lesson, and are expected to complete the assignment on their own time as homework. Students are encouraged to ask questions during class or via email, and occasionally issues do come up that require some discussion. Sometimes the problems are nuanced and the instructor must explain how to solve them, but frequently they are instructed to refer to the help pages and are able to resolve their problems with little direction, or by discussing it with other students in the class.

The second package the students learn is Matlab. While some students have used this in previous courses, for most students it is new. Matlab is taught through the normal weekly homework assignments that students do to learn numerical methods. No lab sessions are held, but a few minutes are spent in class to show students how the user interface works. Students are typically asked to do hand calculations for a particular technique, followed by an Excel spreadsheet calculation and a Matlab calculation (using Matlab routines). The Matlab calculation is explained either in the text or in a Matlab supplement that comes with the text. The text and supplement are generally sufficient for students to be able to figure out how to solve the problem in Matlab, although there are occasional issues that require further discussion.

ANSYS is the final package introduced to the students. Finite element theory for truss analysis is presented, and students perform complete hand calculations for a simple 3-element truss. The students are given a handout written by the author that leads them through the solution of the same truss problem during a schedule lab period. Due to the complexity of the user interface some interaction with the instructor is necessary, but by the end of the session most students have a good sense of how to use the package. The students are then given a truss design problem where they reinforce basic ANSYS commands and learn new ones (sometimes with the support of help pages). Students work through a few other types of analyses with handouts as homework assignments.

Results and discussion

In general this approach has worked well. Student comments collected at the end of the course are mostly positive, although

typically about 20% of the class recommends a few more schedule lab periods. There are always some students who do not buy in to the philosophy of self-learning; however, over the past three years 87% of the students indicated on an end-of-semester survey that they "agree" or "strongly agree" that outcome 7 (learn how to use a new computer package with minimal formal instruction) has been met.

The two factors important to self-learning discussed in the introduction – motivation and skill development – are addressed in this approach. Student motivation is enhanced because they understand that they will not be taught everything they have to learn, especially when it comes to software. They generally accept that some self-learning is inevitable, and realize that it is in their best interest to build skills that will help them with future self-learning. Most students who entered the course with the belief that it was the instructor's job to teach them the software leave the course understanding that it is their responsibility to learn it. This is an important and significant transformation in student attitudes towards learning. In addition, student success in going through the tutorials enhances their self-confidence, and application of the methodology listed in Table 3 provides specific skills and techniques that they can use in the future.

LEARNING HOW TO INTERPRET A TECHNICAL TEXT

The second approach used to support life-long learning in this course is a technique which helps students learn how to interpret a technical text. This technique was developed previously by the author and used in several courses, and has been shown to be effective. A brief summary of the technique will be presented here; readers interested in the background and more details should refer to [7].

The initial motivation for this technique came from out of frustration over traditional lecture methods and a belief that most students do not actually read their engineering texts. The application of this technique to the CAE course makes sense due to the broad range of numerical methods described in the text and the limited amount of time to cover them. Early versions of the courses tended to present a long list of numerical methods; however, the course was redesigned several years ago to focus on the *principles* of numerical methods (such as accuracy and iteration), using several methods as examples. There is simply not enough time to learn how to use every technique in the book (just as there is not enough time in four years to learn everything an engineer needs to know), yet students may find a need for many of the techniques in subsequent courses, their capstone design project, or after graduation. It is the classic education dilemma of teaching students how to learn vs. teaching them material.

Research on text comprehension

There is a wealth of research regarding best practices in text comprehension. Marzano et. al. [8] discuss strategies for summarizing, and conclude that to effectively summarize, students must delete, substitute, and keep information, which requires that they analyze the information at a fairly deep level. Tierney et. al. [9] present a number of reading strategies, geared primarily for either general literacy or K-12 students. Some of these strategies, however, are broadly applicable, such as discussion groups, encouraging students to develop their own questions and adopt an active and inquiring attitude to reading, and vocabulary development. Simpson and Nist [10] discuss strategies specifically for students at the college level. One of their suggestions is to provide activities that force the students to examine their personal theories about reading. By this they mean that students arrive at college with pre-conceived notions about how to read effectively. For example, students who took high-school biology classes may have found that they were successful by simply isolating and memorizing key terms – a surface-level strategy that will not be effective in upper-level college courses. They also suggest that instructors use direct instruction over time rather than simply assessing and evaluating – that is, model the reading process for the students, provide examples, then have the students practice and evaluate the use of the strategy in their own reading.

Whatever strategy is used, it is clear that one cannot simply assume that students know how to read effectively; they must be taught. There also must be specific actions that the students take that force critical thinking, rather than simply reading the material at a superficial level. It is in the processing of the information that learning takes place, not in the receipt of that information. By asking students to summarize, provide definitions, compare and contrast, and discuss important issues, they will necessarily have to think about what they have read, which should result in better comprehension and learning. The development of these skills is essential to improving the ability of the students to self-learn. In addition, improved reading skills should positively influence student motivation for self-learning, as it will help them see self-learning as something they can do.

Description of the technique

The technique described in this paper is both simple to implement and consistent with the research described above. The procedure for the instructor is summarized in Table 4, with additional details in the text that follows:

TABLE 4
SUMMARY OF INSTRUCTIONAL PROCEDURE

1. Read the text, making notes and highlighting important issues.
2. Identify 5-10 key issues or questions.
3. Write questions for the students.
4. Give questions to the students at least one class in advance.
5. Call on individual students to answer specific questions, making sure the response is a paraphrase, not an exact quote.

Instructor preparation for a class consists of selecting a section of the text (usually 10-15 pages), reading the text,

making notes and highlighting important issues, and writing questions for the students to answer. It is important to keep the reading assignment short enough so that students do not feel overburdened by the reading. When reading the text, the instructor identifies the five to ten most important issues raised by the reading, as well as any questions that come to mind. These issues and questions are then written in the form of questions for the students. This process usually takes less than an hour to prepare for a 50 minute class period – well below the time typically required to prepare a traditional lecture.

As an example of the questions that are developed, students may be asked to explain definitions, such as the difference between *implicit* and *explicit*, or *converge* and *diverge*. Sometimes students will simply quote a section of text, in which case the instructor will usually ask someone else in the class, "using your own words, what did he (or she) just say?" Students quickly understand that they need to talk about their understanding of the text, rather than simply quoting it.

As a second example of a question, the instructor might ask if a particular equation in the text is correct. Usually it is, although there are times when there are errors in the text. The point of the question is to make sure the student understands where the equation came from, and can generally follow the math, physics and assumptions used in its derivation.

The final example question involves Newton-Rhapson for solving nonlinear equations. Students are asked to be prepared to use a geometrical interpretation to derive the iteration formula from memory on the board in class. In cases such as this, if a student can correctly derive the formula, then they generally will know how to use it to solve a problem, and also have an understanding of the pitfalls of the technique (such as using an initial guess where the first derivative is zero). Frequently a significant percentage of students cannot do this, but having read the material and thought about the question better prepares them for the discussion in class.

There are several reasons that it is necessary to develop questions for the students. It highlights the critical issues, which can be difficult for students to do on their own; it forces the students to think about the issues that are raised; it provides a basis for classroom discussion; and it allows the instructor to assess the students' level of understanding of the text. In the context of the research on reading comprehension discussed earlier, the instructor is modeling reading for the students, forcing them to actively engage with the material, and assessing how well they have engaged.

Once the questions have been written, they are given to the students at least one class period prior to the class in which they will be discussed. The students are instructed what to read, and are sometimes told that certain sections can be skimmed. They are told to be prepared to discuss the questions in class, and do not have to hand in answers. They are also asked to develop a question of their own for each section. Students have indicated that it usually takes between 30 and 60 minutes to read the text and prepare answers to the questions, depending on the material covered, and have also indicated that this is reasonable.

During the class period in which questions are discussed, the instructor will ask specific students to answer specific questions. It is important to do this to make sure the students do the reading. In larger classes some method of keeping track of who answers questions must be employed. The author has used check sheets, and while they ensure participation from all students, the process of tracking students on a form tends to distract from the discussion. The technique that has been used most recently has been to mentally keep track of student participation, making sure to identify the student by name, which seems to work well. Correct answers are not necessary – the point is to get the students to read and think about the text. In fact, incorrect answers can be more educational, since a discussion about why the answer is incorrect can ensue.

Results and discussion

In previously published work [7], the author presents a comparison of exam scores in an individual course from two different semesters (one using this technique, one not), as well as survey data from several courses where the technique has been used. The exam data were collected during a course in which this technique was first introduced (a 4th year technical elective in Aerodynamics). The first two exams were identical to those given the previous year, when traditional lecture was used. The technique was not employed until after the mid-semester break, so the material on the first exam had been taught using traditional lecture, and most of the material on the second exam had been taught using this technique. The scores on the first exam were very close over the two semesters – the average differed by less than 4%, and the median was identical. On the second exam, the average was 11 points higher and the median 12 points higher for the course taught using the present technique (see Figure 1), and the entire grade distribution shifted up. While the sample size was small (eight students in each course), the improvement was significant enough to be a strong indicator that learning had improved.

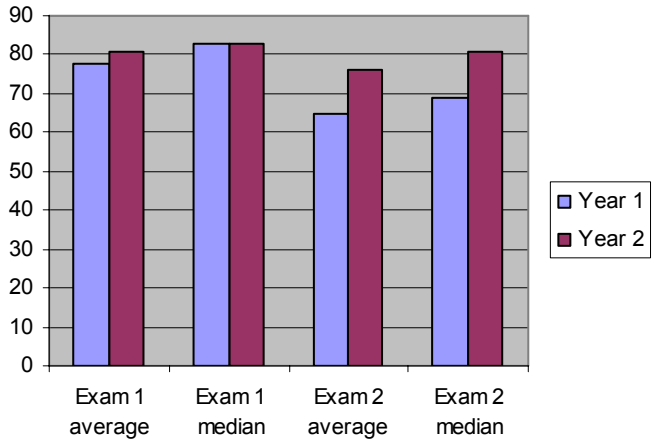

FIGURE 1
COMPARISON OF EXAM SCORES

The survey data from several courses are generally (but not uniformly positive). Typical student comments are that the "open discussion on reading helps to create multiple views on topic leading to a better understanding of material," and the "idea of giving us advance questions to answer during class discussion was very helpful in understanding the material." Students see learning in this way as being more fun and something they can do, even without an instructor. On the other hand, some students have learned how to be very successful in a traditional lecture course, and can in fact learn quite well in that mode. A discussion format can be frustrating to these students, who want to get to the material as quickly as possible. Some students also have difficulty with the lack of structure. One advantage to a traditional lecture format is that by coming to class, students end up with a clear summary of what is important in the course, and have the "correct" answers in their notes. In a discussion-based class, students can give incorrect answers to questions, and other students might not fully realize that the answers are incorrect. To mitigate this issue the author posts lecture notes on the course Web site.

It has become clear to the author that this technique can be very effective, but must not be overdone. Different students have different learning styles, and it is probably too much to expect a student to be prepared for every single class period, since they do not control schedules for exams, assignments, and projects for other courses. Based on several semesters of experimentation, the author believes that an optimum use of this technique is about once every two weeks.

SUMMARY

According to the research, students must be motivated as to the need for life-long learning, and provided with the necessary skills to accomplish it. The "problem of packages" provides an excellent framework for discussion of the importance and necessity of self-learning, and for the most part students understand and accept this as motivation. The use of tutorials for learning software is nothing new, but by themselves they will not foster an attitude of life-long learning. They need to be put into context so that students understand they are a tool to help them reach their goals. There must also be the right amount of support – not too much handholding, but enough so that students don't get frustrated. Much of the development of a student attitude for life-long learning depends on building their self-confidence. They need to know that they can succeed without having an instructor present to back them up every time a problem arises.

The development of metacognition skills can be enhanced by helping students learn how to comprehend technical works. They need to be shown what it means to understand an engineering text, and learn what kinds of questions they need to think about when reading. The structured technique presented here has been shown to be effective, and has gained broad support from the students. They view it as being more fun than traditional lecture and they also realize it's something they can do.

The techniques described in this paper do not address all of the issues important to self-learning. Due to the structured nature of the course, the students are given the learning objectives and provided with the necessary materials. They do not have to figure out what they need to learn or where to go to find the answers; these skills are better addressed in open-ended courses such as the capstone design course. These techniques do, however, get the students thinking about the importance of self-learning and start to build specific self-learning skills. This is an important change in student attitudes that can be built on through the remainder of the program.

While the approaches used here have not been universally accepted by students, they have been successful in increasing the students' appreciation for life-long learning. Course outcome surveys and end-of-semester student comments indicate the course has had a positive impact on their perceptions. In a graduating senior exit survey, one student commented on the program outcome regarding life-long learning that the author's "one day lecture on self teaching techniques has left a lasting positive impression on me."

REFERENCES

[1] Marra, R.M., Camplese, K.Z., Litzinger, T.A., "Lifelong Learning: A Preliminary Look at the Literature in View of EC 2000," *Proceedings of the 29th ASEE/IEEE Frontiers in Education Conference*, November 10-13, 1999, session 11a1, pp. 7-12.

[2] Wells, D.L. and Langenfeld, G.P., "Creating an Environment for Lifelong Learning," *Proceedings of the 1999 ASEE Annual Conference*, 1999, pp. 1395-1403.

[3] Healy, T., "An Approach to Teaching Lifelong Learning," *Proceedings of the 2004 ASEE Annual Conference*, 2004, pp. 565-570.

[4] Gandolfo, A., "Teaching Lessons Learned," *Journal of Professional Issues in Engineering Education and Practice*, July, 2001, pp. 93-97.

[5] Briedis, D., "Jump-Starting Lifelong Learning," *Proceedings of the 1998 ASEE Annual Conference*, 1998, session 2213, 4 pp.

[6] Litzinger, T.A. and Marra, R.M., "Life Long Learning: Implications for Curricular Change and Assessment," *Proceedings of the 2000 ASEE Annual Conference*, 2000, pp. 4083-4090.

[7] Lasher, W.C., "Using active learning to improve technical text comprehension and increase student participation," *Ethos*, http://pennstatebehrend.psu.edu/academic/lrc/ethos/, 9 pp., 2004.

[8] Marzano, R.J., Pickering, D.J., and Pollock, J.E., "Classroom Instruction that Works: Research-Based Strategies for Increasing Student Achievement," Association for Supervision and Curriculum Development, Alexandria, VA, 2001.

[9] Tierney, R.J., Readence, J.E., and Dishner, E.K., "Reading Strategies and Practices: A Compendium," Simon and Schuster, Needham Heights, MA, 1995.

[10] Simpson, M.L., and Nist, S.L., "Encouraging Active Reading at the College Level", in Comprehension Instruction: Research-Based Best Practices, C. Block and M. Pressley, ed., The Guildord Press, New York, NY, 2002, pp. 365-379.