Maintaining Knowledge Currency in the 21st Century

Paul Ammann and Jeff Offutt

{pammann, ofut}@gmu.edu
703-993-1660 703-993-1654
Department of Information and Software Systems Engineering
George Mason University
Fairfax, VA 22030-4444

Abstract

Software engineering is a rapidly changing discipline, and will continue to be so for the foreseeable future. This pace of change brings both problems and opportunities to universities that teach software engineering. Engineers are no longer satisfied with one or two initial university education experiences, but by necessity are becoming lifetime learners, with frequent trips back to educational providers. This recurring education is needed to update engineers' knowledge with new ideas and concepts, and to update engineers' skills. In this paper, we take the position that universities can and should respond to this situation with a new model for graduate software engineering education, which we call professional currency certificates. These courses should offer the depth of knowledge and university academic credit that traditional academic courses offer, but with the convenience and practical nature of corporate training courses. This hybrid model results in a new kind of course that more closely meets the needs of lifetime learners.

`Keywords`: Certificate, training, graduate education, Master's degree

10th SEI Conference on Software Engineering Education and Training, Virginia Beach, Virginia, April 1997.

1 Introduction

It is no secret that universities in this country are going through major changes, and software engineering is both an early result of the change process and in the forefront of the wave of change. The authors of this paper have been involved in software engineering education, as students, researchers, and educators, for more than 10 years. We have had the opportunity and challenge of teaching undergraduate students [2], traditional graduate students [5], professional graduate students [1], and professional training courses. We are currently faculty in a department that offers a Master's degree as well as a Certificate in Software Engineering*.

We are currently teaching our graduate courses at a local corporation, in response to a request to offer our certificate on-site to employees of the company. This experience, together with the current and growing pressures on universities and professionals, has led us to develop a new model for how software engineering education can be delivered in the future.

*The ISSE web page at `http://www.isse.gmu.edu` and our previous paper [1] have more information on the Software Systems Engineering program at George Mason University.

Others have said many times that educational needs for professionals and engineers will be very different in the next century. As far back as 1978 [4], it was noted that software professionals need skills and training, in addition to deep knowledge. Some changes are obvious and do not need new educational models – managers will have to be computer-literate, professionals will be rewarded for creative, analytical decisions as opposed to "doing it the way it's always been done", and workers will be functioning more and more like today's professionals. A difference that offers particular challenges to educators is the oft-quoted prediction that professionals and engineers will need to spend their entire lifetime learning new knowledge and skills to stay current. This is already true in the software engineering field, and is having a direct impact on our students, of which over 90% work in local industry. It is clear that we should expect 21st century engineers to return to the classroom every ten, five or even three years for more education. These classes will be a continuation of the students' initial degree. That is, an initial degree is just that, initial.

This brings up some important questions. What are the educational needs of these lifetime learners? What are the different needs, both in delivery of knowledge and in service, that these students have? It seems clear to us that *it is impractical for engineers to continually take traditional semester-long academic courses*. These students already have a college education, and in many cases will have a full Master's degree. This means they do not need as much evaluation as traditional students. Neither do they need as much practice (that is, homeworks and projects) as traditional students, and indeed, do not need as much of the traditional "hand-holding" services like office hours and advice about studying. Most fundamentally, they do not need a degree, but a specific amount of knowledge that will help them to once again become current with the field.

What they do need offers us challenges that will stretch the traditional university model. They need classes that do not interfere with work (that is, at night or on the weekend), and courses that in addition to providing them updates on knowledge, also provide opportunities to learn specific new skills. Because of time constraints, they need a more efficient delivery system. They will also often be in a position where time is more important than money, so they will, for example, be willing to pay for services like child care and couriering of course materials.

At George Mason, we see some of these needs already. Our physical office hours are sparsely attended, partly because our students do not have the time to come to school as often, and partly because our students already know how to study by themselves. On the other hand, "electronic office hours" are becoming more and more popular. Most of our courses are offered in the early or late evening, and recent offerings of courses on Saturday have been well attended. The University operates a money-making child care service, which in addition to a normal 7:00 AM to 6:00 PM schedule, offers a "drop-in" evening program for parents who are attending classes. We occasionally get requests from our students' secretaries to fax syllabi and course notes – responding to these requests strains our already burdened support infrastructure, but a student who has the resources to have a secretary ask for a fax probably has the resources to pay a little extra so that the university can provide such service.

These changing needs will also necessitate changes in the faculty. Most importantly, in order to teach our students current knowledge, we must stay current. It is our hope that forward-looking universities will find ways to encourage and reward faculty who spend time and effort keeping up to date with the rapid pace of change. We will also need to be flexible in how we teach, when we teach, and where we teach. For students who are well accustomed to learning, we need to become more efficient and organized in our development.

We are not suggesting that the traditional college experience, or even the traditional graduate experience will disappear. It is our belief, and the position of this paper, that we must do more, and if universities do not do it, other organizations will. This would be unfortunate for several reasons. University expertise lies in delivering education in a deep, knowledge-intensive, manner. This is precisely the kind of education that is needed. Additionally, many students will want university credit for their courses, which, in most cases, requires that courses are taken through a university.

In the rest of this paper, we elaborate on our position with regard to graduate education. Then we

present a detailed, analytical discussion of two traditional models for graduate education, and present our proposal for a new model to deliver software engineering education. We have had preliminary experience with many aspects of this model, and our experience indicates that it will be quite successful.

2 Statement of position

We are interested in

- delivering post graduate education
- primarily to practicing software engineers
- on a continuing, **lifetime**, basis.

Recently, we have had the experience of offering university courses at a local corporation. Specifically, George Mason University is offering its Certificate in Software Systems Engineering at the corporation's site. The requirements for the Certificate are the completion of five courses from the Software Engineering program with an average grade of B or better. The corporation requested the following five courses to be taught on-site over a five semester period:

1. SWSE 619: Software Construction
2. SWSE 620: Software Requirements and Prototyping
3. SWSE 621: Software Design
4. SWSE 625: Software Project Management
5. SWSE 637: Software Testing and Quality Assurance

SWSE 620 was taught in the spring semester of 1996, SWSE 619 taught in the fall semester of 1996, and SWSE 625 is being taught in the spring of 1997; all courses have between 35 and 40 students (we cap our enrollment at 40). As presently structured, the course offerings are open to any George Mason University student with the appropriate prerequisites. Our position, based on the experience described above, is that it is possible to be much more innovative and effective than simply altering the physical location at which courses are taught. This paper elaborates our view of what is possible.

We first discuss the differences between two traditional models of post graduate education; a traditional academic model, and a traditional corporate training model. Although most of the experience of both authors is in the academic model, both authors have also taught courses in the corporate training model. After explaining the differences in these models, and why they do not exactly fit the needs of lifetime education, we develop a hybrid model that incorporates the most desirable aspects of both models.

Corporate training courses differ from academic courses in several very basic ways. The most obvious has to do with time. Traditional academic courses meet for 14 to 18 weeks for short periods of time (one to three hours at each meeting), whereas corporate training courses typically meet for two to five consecutive eight hour days. This is a superficial difference that leads to many fundamental differences. The pace of an academic course allows the students to do quite a bit of work outside of the lectures, primarily reading the texts and doing homework and projects. Homework, projects, and exams serve several purposes. The most obvious, of course, is evaluation for a grade. But they also provide practice and allow individualized feedback from the instructor, which are important vehicles for learning.

This leads to a deeper difference; academic courses allow students to get a deeper level of knowledge about the material than a fast-paced corporate training course does. The time between classes means the contact between teacher and students is not continuous. The students have time to read outside of class and think about material between classes. This is not to say that corporate training classes do not have

value. Whereas the fast pace and continuous nature of the training classes make it difficult to absorb deep amounts of knowledge, they do work well for skills and information that do not require a large depth of understanding.

An example from today's software engineering education is that of object-oriented software development. Many companies are switching from a functional programming language (such as C), to an object-oriented programming language, usually C++. They often accomplish this by sending groups of programmers to a three or four day course on C++ programming, then using the language in the next project. This has often led to disaster. In this scenario, the programmers often come back and use the same design principles, program construction methods, and coding styles that they used in C – this is commonly called "C programming with a C++ compiler". If object-oriented features such as generic typing and inheritance are used at all, they are typically misused. Not surprisingly, many of these projects fail.

To effectively integrate an object-oriented language into a development process, the entire process needs to change. The design should be different, and the software construction process should be modified. To do this requires a depth of knowledge that goes beyond what can be gained in a three day course on C++ programming. Effective object-oriented software design and development can be taught in a semester-long academic class (and is), but many practicing engineers, some of whom already have Master's degrees, cannot afford to take a traditional semester-long course.

These problems lead to our proposal of a "hybrid" model for post-graduate education. We hope to combine elements of both models to create a new kind of course that suits the lifetime learners of the new software engineer. The primary goals of the two traditional models are different: traditional academic courses have **knowledge-based education** as their primary goals, and traditional corporate training courses have **specific skills training** as their primary goals. Of course, academic courses do teach skills, and training courses do convey knowledge, but these goals are not primary. In our new model, which we call **professional currency** courses, the primary goal is *tailored education*, and it includes knowledge and skills in equal measure. The content is tailored to the needs of the professional and the company. Professional courses are taught over a shorter time period, and involve less evaluation, but some practice. They should be offered at times convenient to professional engineers. We also expect that whenever possible, class assignments should be integrated with work.

Denning [3] discusses some of the needs of the new engineers, and although his paper primarily addresses undergraduate engineering education, the principles are intended to apply to graduate education as well. One of the major themes of his paper is that emphasis on abstract theory should be transferred to problem solving at the client level. The professional model's deemphasis of homework and exams and emphasis of projects oriented toward the corporate mission fits well with Denning's message.

3 Models of graduate education

This section presents an analytical comparison of three models of graduate education. We consider a number of attributes of graduate education, and consider how these attributes vary. This analysis is summarized in Table 1. In the remainder of this section, we discuss each attribute in turn.

3.1 Discussion of table attributes

Goal: The primary goal in a traditional academic environment is knowledge-based education as opposed to skills training. Academic education is intended to endure for a professional's entire career. Professionals should be able to adapt the theoretical principles learned in the university to their slowly evolving work environment. Unfortunately, today's work environment evolves in a manner that is far beyond the capacity of a single academic degree. Professionals now periodically need additional education tailored to the demand

	Traditional Models		New Model
Attribute	**Academic**	**Training**	**Professional**
Goal	Education	Training	Tailored Education
Time	Semester	Days	Weeks
Course Focus	Academic Program	Specific Topic	Corporate Need
Degree Credit	Yes	No	Yes
Prerequisites	Enforced	Not Enforced	Enforced
Homework	Yes	No	Less Than Academic
Projects	Generic	Not Feasible	Corporate Work
Evaluation	Exams	None	Exams and Corporation
Presentations	Blackboard	Slides	Slides
Notes	Up to Student	Supplied	Supplied
Interaction	High	Low	Moderate
Student Maturity	Less Mature	Not Relevant	More Mature
Location	University	Site Independent	Corporation
Motivation	Student	Corporate	Both
Cost	State and Tuition	Direct Charge	Company-supported

Table 1: **Attribute Summary of Three Models**

of their specific work environment. Examples abound among the students in our classes; since the time when most of them finished their initial education, the field has changed to use object-oriented development, reuse of software components, a greater emphasis on reliability and maintainability, graphical user interfaces, and client-server applications, among others. Such incremental education is the precise goal of the professional model.

Our experience is that companies are eager to provide specific guidance for tailoring a set of courses. For example, our software construction course is not specific to a particular programming language, but, for practical reasons, must be taught with a specific language. In the past, the course used Ada as an implementation language. The corporation we are offering our certificate to argued strenuously that the course be redesigned around C++. As a result, one of the authors (Offutt) devoted a month of summer to redesign the course.

Another example of corporate tailoring concerns the courses that make up the Certificate being offered. Less emphasis on theory and more emphasis on direct application led to a revision of the Certificate to replace our formal methods course with our software testing course.

These two examples support the argument that a corporation wants to be "in the loop" when it comes to furthering the education of its work force. Corporations want as close a match as possible between instruction and business. Our client corporation was not interested in the standard computer science argument that students should be able to switch programming languages with little effort. Their perspective was that they only wanted to spend effort (e.g., on learning a language) if doing so would directly help the company.

Time: As stated before, a major difference in academic and training courses is the time in which they are taught. To satisfy the goals of teaching knowledge and including outside projects, a professional currency course needs to encompass more time than a training course, and needs intervals of time between courses that are sufficient for the student to absorb the concepts learned in lecture, study the text, and practice the techniques. A typical semester course might have 16 weeks of classes, with three hours of class time per week; this totals to 48 hours of class time. There are various ways in which the same amount of class time can be provided at a faster pace. If we assume one eight hour day per weekend, then such a course could

be delivered in six weekends. Of course, an eight hour day is not equivalent to eight one hour lectures, so much of the delivery would have to be redesigned. If the company could devote time during the day, then 12 four-hour days of class would suffice, which could be provided over the course of one month. Both of these provide time spans that are realistic, and offer balance between the leisurely semester-long time span in an academic course and the furious fast pace of a training course.

Course Focus: The courses students take in an academic environment are typically constrained to satisfy the requirements of a degree. The result is a coherent degree. On the other hand, training courses are, to a large degree, independent of one another, and instead focus on specific topics. The professional model allows specific courses to be tailored to match the corporation's current needs. Both the student and the corporation benefit from this arrangement with more immediate applicability of course material to work products, but without sacrificing the ability to transfer the knowledge gained to future projects.

Degree Credit: A degree formally recognizes the accomplishment of an academic program. If carefully structured, the professional model can retain the coherence of a degree program. For example, students in our current program can extend their Certificates to full degrees by completing the remaining courses necessary to receive an MS. In this regard the professional model can complement, rather than replace, the traditional academic model, and students can proceed from a sequence of professional currency courses to a full Master's degree.

Prerequisites: It is difficult to require prerequisites for corporate training courses, which means there is a great variation in student background. In the academic model, prerequisites are enforceable and are reasonably effective at ensuring a uniformly prepared set of students. In principle, the professional model can follow the academic model with respect to prerequisites. In practice, enforcing prerequisites is more difficult for the professional model due to pressure from the corporation, which, for business reasons, wants to have a given set of employees in its courses at the same time. Enforcement of prerequisites, however, is essential if a professional currency course is to receive degree credit. It then becomes crucial to be able to convince the students and corporate administrators that prerequisites are essential.

Homework: Homework is a traditional way to achieve proficiency in materials presented in lecture. The professor works small, self-contained, often artificial exercises to illustrate a lecture, and after class the student works similar exercises to master the material. The compressed schedule of training courses makes homework largely impractical. Homework has its role in the professional model, but that role is smaller than in the academic model. The primary reason is that mastery of material can be achieved through projects, which tend to be larger than homework assignments, and have the added advantage of being more realistic. Another reason is that the intellectual maturity of the students means that homeworks are less necessary. Still another is that the evaluation component is less necessary; professional currency courses have more focus on teaching and less on evaluation.

Projects: Projects are very effective at teaching the skills needed in the workplace, in that they can be modeled on the workplace, either as it is currently or as it is intended to evolve. The professional model enjoys an enormous advantage over the other models with respect to projects. Projects often have a generic or artificial flavor in academic courses, and projects are simply infeasible in the training model. In a corporate environment, course projects can be tailored directly to work projects, and even be part of work projects. For example, Offutt teaches a course on human computer interface design, and allows students to choose a semester project. The best projects are often developed in combination for work and class.

Evaluation: Although often onerous for both professor and students, the evaluation process is critical to learning; students study for exams, and learn much from feedback on exams and outside work. The professional model opens up an additional possibility. Exams clearly retain a role, but corporate reward structures such as performance reviews can also be used to motivate students to learn.

Presentation: Although a variety of presentation mechanisms are used in all models, academic classes traditionally use blackboards, and corporate training classes traditionally use overhead transparencies. Blackboard presentations have the advantage of being easily adapted to student feedback during a lecture or dis-

cussion. Blackboard presentations are also easily adapted to incorporate new research results and changing technology. Nonetheless, our experience clearly shows that the corporate expectation is for transparencies, and the professional model might need to adopt this approach. The implication is significant: a transparency-based course tailored to a corporation's needs takes more time, effort, and money to prepare.

Notes: In traditional academic courses, students are expected to take notes during the lectures to record what the professor teaches and to use as study notes. The process of taking notes can enhance the learning process, but also takes considerable effort and time, both in out of class. Students in training courses typically do not take notes, but rely on instructor-supplied notes, usually in the form of copies of the transparencies. In the professional model, we expect copies of the transparencies to be supplied, and students to complement these with additional notes taken during the class.

Interaction: Graduate academic courses typically involve a high degree of interaction between the student and the professor, both in and out of class. This interaction allows the student to clarify confusing points, catch up on material that other students know, correct presentational errors that the professor makes, and lets the professor tailor details of the material to interests and needs of the particular class being taught. Training courses typically have very little interaction; the instructor presents and the students listen. In the absence of evaluation, the students are less motivated to request additional clarification, and the type of skills being conveyed in such classes usually need less tailoring to specific needs of students. Additionally, the fast-paced speed of the classes and the often large class size makes interaction problematic at best. In a professional currency course, the interaction should be moderate; enough to clarify points that are confusing to the students, but the material will probably seldom be modified during a lecture.

Student Maturity: The average age of graduate students is rising. At George Mason, the average age of graduate students is 35 years. Older students tend to be significantly more mature than their younger counterparts. They tend to have better study habits, and they have to balance the demands of work, school, and family. They tend to bring a professional attitude from the office to the classroom. The professional model can take great advantage of this maturity. Evaluation on homework, projects, and exams can be directed more towards learning and less towards discipline and thinking ability.

Location: In the professional currency model, courses are taught at the sponsoring corporation's site. Location might at first seem to be a minor consideration, but our experience at George Mason is that the change in location has significant effects on both practical matters and student attitudes. Primarily, it emphasizes that the course is different from a traditional academic course, and it offers a significant convenience to the professional students. It also creates practical difficulties for the professor. The most obvious is travel; the company must be expected to provide some reward function for the added inconvenience of teaching off-campus. Beyond that, having a class off-campus means that things like the library, computer labs, professor's office, teaching assistants, and bookstore are less convenient. Welcome services would be to deliver books and course notes to the corporation, have the professor and TA hold office hours on-site, and handle student registration in a "batch" fashion. These services are, of course, not free.

Motivation: The motivation for pursuing advanced degrees in the academic model comes primarily from the student. Although management might provide tuition reimbursement and study time, the student provides the primary direction. In contrast, the courses in the training model are motivated primarily by the needs of the corporation as perceived by management. The professional model offers the opportunity to combine these different sources of motivation. The overall structure of a program under the professional model is determined by management, but the student can still be motivated by a degree.

Cost: We have saved cost for last because it is the most difficult issue to resolve. A professional course will incur costs that a traditional academic course does not. The professor will have to travel and spend more time than on a traditional academic course. The university will require extra administrative overhead and administrative support to coordinate an off-campus class. The courses will require more pre-semester preparation time, and more frequent course modifications. Supplying notes, books, and office hours are expensive. Even coordinating mundane issues such as admission and registration outside of the

usual bureaucratic channels adds an extra burden on the university. For the professional model to succeed, new bureaucratic channels will need to be built and paid for. While these costs must be passed onto the consumer, we believe that the costs can still be significantly less than for a training course, and the advantages of professional currency courses will more than make up for the additional cost.

4 Conclusions

In this paper we advocate a new kind of graduate university course, a *professional currency course*, which can be a stand-alone course, part of a coherent certificate program, or part of a graduate degree. We have presented these ideas to many of our current graduate students at George Mason who are working as professional engineers, and the most common reaction is "yes, that is what I want, much more than a Master's degree!". Our perception is that the current traditional master's degree programs are often imperfect fits for what our graduate students need. They settle for a Master's degree, often a second or even third degree, because their real needs are not being met.

4.1 Lessons learned

Although we have not implemented a professional currency program as we describe here, we do have experience with many of the pieces. As stated, our students are mostly professionals in the computing industry, returning to school part-time for an advanced degree and currency. Additionally, we have taught two complete, regular degree courses at an off-campus location, and courses to complete a certificate program are planned.

This initial experience has made us realize that there are many **hidden costs** of teaching in this new way. Registration procedures, delivery of course materials, and preparation of lectures must change, with the result being added costs. The faculty must travel to the off-campus location, which costs travel expenses and, more importantly, lost work time for the faculty.

We have come to realize that giving partial control of the course to a corporation has side-effects. Prerequisites are harder to enforce; in the past two semesters, we have had more pressure from students (and their managers!) to accept them into the class without satisfying the prerequisites. The instructor no longer controls the class room. For example, in the fall of 1996, class was delayed three times, twice because corporate meetings preempted the designated room until after the class was scheduled to start, and once because a white board could not be located.

We have found that the students are reluctant to come to campus, and, as a result, are relatively unfamiliar with university facilities and resources. Many students do not bother to buy parking passes. Last fall, less than two thirds of the class came to the campus to buy an required course packet. The students are less familiar with the computer facilities offered by the university, making it more difficult to assume a common level of computing ability. Several students made appointments in the professor's office but did not come. In two cases, the students said they could not find the office. On the other hand, "electronic office hours" are more popular. Offutt recently taught a course in which there was more student interaction than in any previous course he taught, mostly via email.

Students with full-time jobs have different priorities than full-time students. These differences have a significant effect on how professional model courses can be conducted. Many full-time students rank school before work or family, which is reasonable, since the typical full-time student does not work and has neither a spouse nor children. The typical student with full-time job does have a spouse, and is likely to have children. Therefore, part-time students often must prioritize school after family, home, and jobs. Far more flexibility is required from faculty than in the traditional academic model. Business travel must be accommodated, deadlines must be adjusted, and homework assignments and projects must be carefully timed to avoid

overloads. In the academic model, a professor would be very surprised to receive a call from a student's manager requesting that an exam be rescheduled due to a corporate deadline. In the professional model, such a request might be routine.

4.2 Summary

We believe that traditional, deep-knowledge undergraduate and graduate degree programs should remain the core and staple of the university, but that more resources should be devoted to corporate-oriented certificates. Certificate programs should in large part be based on the core degree courses, but should be modified to reflect the needs of lifetime corporate students.

Our initial experience has indicated that corporations should, and are willing to, pay directly for *effective* certificate programs. Applying standard tuition rates to these courses is unrealistic. New costing methods should be developed that are based on actual costs. Certificate programs should not be supported by state taxes, and should be expected to make a profit.

Faculty members must be monetarily supported to remain current and redevelop courses and curricula. Part of the actual work-load of faculty should be to learn new skills in their area. The university must find opportunity and incentives to faculty to keep current with the rapidly changing industries, be more flexible in how courses are run, and become more efficient at delivering material. Some of these goals can be supported through technology, if used appropriately. Incentives cannot be simply verbal suggestions, but must be reflected in flexible work-load assignments and in evaluation for promotion and tenure.

We have intentionally left out discussions of the impact of technology on delivering courses. This is partly because the use of technology is a very large topic that is beyond the scope of this paper, and partly because while technology can be effectively used to make delivery of courses more convenient, it seldom brings about essential changes. For example, a televised course is simply an alternative way to offer a course on-site. Electronic mail (which the authors currently use very heavily for student interaction) is simply a communication medium that is more convenient than personal office-hour visits. And a primary use of the web in many classes is to avoid the expense of copy machines and to provide asynchronous delivery of course materials.

One possible technique could be to have certificate courses developed by full-time faculty, and taught by adjunct faculty. The adjunct faculty can be a mix of full-time teaching faculty and part-time professionals who are able to provide the benefit of their experience and wisdom to the corporate students. This technique has been successfully applied by the Georgia Institute of Technology.

Universities have been under increasing pressure to better serve the needs of corporate America, find novel ways to pay for education, and become more efficient and innovative in how education is delivered. This model of professional currency certificates provides a way for universities to help software engineers keep their knowledge and skills current in an efficient, flexible, and economically viable manner.

References

[1] P. Ammann, H. Gomaa, J. Offutt, D. Rine, and B. Sanden. A five year perspective on software engineering graduate programs at george mason university. In *7th SEI Conference on Software Engineering Education*, pages 473–488, San Antonio, Texas, January 1994. Springer-Verlag Lecture Notes in Computer Science Volume 750, Jorge L. Díaz-Herrera (Ed.).

[2] C. L. Bullard, I. Caldwell, J. Harrell, C. Hinkle, and A. J. Offutt. Anatomy of a software engineering project. In *Proceedings of the 1988 SIGCSE Technical Symposium*, pages 129–133, Atlanta GA, February 1988.

[3] Peter J. Denning. Educating a new engineer. *Communications of the ACM*, 35(12):82–97, December 1992.

[4] Fred A. Gluckson. A treatment - Professional development. In *Proceedings of the National Computer Conference*, pages 1161–1166, Anaheim, CA, June 1978.

[5] A. J. Offutt and R. H. Untch. Integrating research, reuse, and integration into software engineering courses. In *1992 SEI Conference on Software Engineering Education*, pages 90–98, San Diego, California, October 1992. Springer-Verlag Lecture Notes in Computer Science Volume 640, C. Sledge (Ed.).