Position Paper: The Importance of Experience with Industry in Software Engineering Education

Rayford B. Vaughn, Jr. and Jeffrey Carver
Department of Computer Science and Engineering, Mississippi State University
vaughn/carver@cse.msstate.edu

Abstract

This paper describes experiences and lessons learned at an ABET accredited software engineering program with respect to students at the undergraduate and graduate levels studying and/or researching the application of software engineering techniques. Examples of successful interactions are provided as well as the rationale for including industry experience. Disadvantages of such interactions are also included when appropriate.

1. Introduction

Mississippi State University (MSU) achieved the first ABET accredited undergraduate Software Engineering degree at a public institution in 2003 after having adopted an ABET SE curriculum in 2001. Early in curriculum discussions, the software engineering faculty recognized the need for industrial experience as a part of this program as well as the difficulties inherent in acquiring it. This paper addresses several of the reasons that such experience was deemed necessary, the difficulties that had to be addressed, experiences in incorporating it, and some of the advantages and disadvantages observed. MSU's program includes a bachelor's degree in software engineering (BSSE) as well as graduate study at the Masters and PhD level in computer science with an emphasis in software engineering. The SE faculty is largely comprised of individuals having industrial experience and a tendency toward empirical research. Over the past few years, we have placed some of our graduate students in industry for the purpose of gathering data relevant to their research and we have allowed undergraduates to work with real customers in an industrial setting so that they could experience real world concerns.

2. Why industrial experience is necessary.

We believe that some of what a software engineering student must learn, is not available in text books and even if it were, the lessons would not be remembered or understood as well as they are with experience. Industrial experience in the areas of process focus, team dynamics, planning, performance evaluation, customer management, delivery schedules (time and budget), time accounting, and product delivery is generally a longer lived lesson than when such topics are only discussed in class. These ideas were previously presented in [1,2]. Essentially the lessons learned in our program were that the stress of time pressures, the unpredictability of customers, the technical language barrier, cost considerations, and a mix of team skills are important to experience and that the experience stays with students a very long time. We also found that the ability to make mistakes in a controlled environment (i.e., with a faculty member present) can be more easily overcome and corrected than in an actual industrial environment when the student's job may be at risk.

A few years ago, we surveyed a number of organizations with an informal survey directed toward senior executives currently employed within the software services industry. Companies contributing to this survey included EDS Federal, Harris Corporation, AmerInd Incorporated, Pragmatics Incorporated, and Business

Communications Incorporated. All five companies were heavily involved in building or maintaining software systems and all discussions that contributed to this survey were with individuals at President, CEO, or Vice President level. Responses were collected informally and without attribution. As a synopsis for this paper, the responses indicated that industry was looking for students that had the ability to work as a member of a team; a solid work ethic; the ability to work under stress; a professional demeanor; good communication skills; the ability to follow processes; technical proficiency; a desire for continuous learning; project management skills; a customer focus; an appreciation for the dynamics of requirements; and *good citizenship* qualities [1]. Based on this feedback and our own software engineering experiences, we developed a one semester course for all software engineering degree students in our BSSE program known as "the practicum" where students were required to practice those qualities listed above while delivering a software product to a customer over the course of two semesters. During this practicum, students from diverse backgrounds (business, computer science, and software engineering) are placed together in a team which must win work (with a proposal), develop cost and development schedules, design and code to requirements, manage a customer, document, and test and deliver a product. The student team must follow processes taught to them previously in software engineering courses and they are often managed by a graduate student (who may be from the College of Business). The experiences gained have been invaluable to these students. An email from one Management Information Systems graduate student who had served as a project lead and went on to work for a major corporation in a software development role provided us with excellent feedback. It read (in part) as follows – "*The experiences I encountered by working with students from the CS department have proven useful to me on a daily basis. I am fortunate to work with a wide variety of people from various backgrounds and training, which was exactly what was given to me in the CS course. I was amazed at the difference in personalities between the CS and BIS students in the ... team, and I am still observing those differences today in the teams that I am working with. ... I would not have been as knowledgeable about how to approach tasks without the experiences encountered from working with people from different backgrounds and curriculum. Everything from the course proved valuable to me; from the deadlines imposed to the "enology lab dinner" to celebrate the hard work that we did to get our proposal accepted. I have encountered the same work related situations in my first year of work and was prepared for them thanks to the experiences gained during that semester.*"

3. Experiences in our program

Our experiences working with industry must necessarily be broken into two categories – undergraduate and graduate. We address these separately in the sections that follow. At the undergraduate level, our focus is teaching students software engineering process and development skills that they will need in the active workforce. At the graduate level, our focus is slightly different in that we have tried to have our students collect real world data to support their research efforts and to determine if their research seems to have value in the commercial world. Experiences with both categories of students have been interesting.

3.1 Experience factors with undergraduates

Our undergraduates generally gain industrial experience in their senior year with a two semester senior project class. The classes are listed in the university bulletin as follows:

CSE 3213. Software Engineering Senior Project I: Six hours laboratory. Software requirements elicitation and specification, cost estimation, scheduling, development or project management and quality assurance plans, reviews.

CSE 3223. Software Engineering Senior Project II: Six hours laboratory. Team work, software design, construction, implementation of project management and quality assurance plans, and configuration management.

During the nine months of project experience, the students experience (many for the first time) the actual process of “winning work”. In most SE jobs our students will graduate into, they will become involved in writing competitive proposals to win work for their company. In order to expose them to this process, our students are often given a proposal solicitation for software engineering work that they must respond to. This response is not only a technical response – but also a management plan for product delivery and a cost proposal. The students must organize themselves to win the work, seek assistance in developing a cost plan that includes overhead rates, fringe, and fee – yet remains competitive with their perceived competition. Often the students will find that they need and value the experience of their management information systems colleagues in this effort in that SE students are generally ill prepared for this necessary part of the software engineering “real world” process. As the students progress and successfully win the work, negotiate their price, and work with a real customer on exact requirements – they begin to understand the importance of solid configuration management processes (for requirements as well as other artifacts) and the futility of trying to discover requirements very early in the life cycle. Team dynamics always play a crucial role for the students as they begin to experience the need to work with (and rely on) students with varied backgrounds and as they discover the need to modify their team roles. Inevitably, the students organize themselves incorrectly at the beginning of the course and evolve into a more efficient organization as individual strengths become apparent during the development process (e.g., coding, customer management, configuration management). Students are managed by a student project manager (not a faculty member) and must learn to work with the PM under pressures of time, technical achievement, personalities, and the stress of delivery. The faculty overseeing the course works with the PM to watch progress and to make sure that resources are available as needed by the team. Performance evaluations (peer evaluations) are accomplished periodically during the semester and reviewed jointly by the faculty and PM. Real world turnover of personnel is built into the experience by allowing students to register for the two course sequence in any order – that is, a student can take CSE3223 first followed by CSE 3213 second or the other way around. Students often speak of the long hours, stress, customer difficulties, team conflicts, and other such lessons far more than they mention the pure technical nature of their task. Such experiences, we believe, place them in far better stead as a graduate than would a normal class experience.

3.2 Experience with graduate students

Giving graduate students an industrial experience is, for us, more difficult. Our university is not located in an area rich with software engineering companies – so our students must travel to gain this experience. The primary objective of involving these students in an industrial setting is to validate their research as useful to the software engineering community (to the maximum extent possible) and to acquire industrial data supporting their research. We have found several key factors that must be addressed to make this involvement successful. First, a non disclosure agreement (NDA) with the industrial component is essential. Industrial cooperation can only occur if the company involved feels comfortable that no harm will come to their customer base or their corporate reputation. Secondly, there are often concerns expressed with publications resulting from the work. We often address these concerns with agreements to include the industrial entity in the publication process – that is, as a co-author or editor of the work before it is published. We generally agree to credit the organization for their support of the work and to not identify by name any customer without permission. Strong positive relationships with senior management in the company is always helpful – we have found that mid-level managers tend to be risk averse and often are not supportive of academic collaboration. Last, we have found that it is especially helpful to have the student spend at a minimum, a whole semester with the company. Short duration experiences do not seem to result in the kind of acceptance and cooperation needed for the student to accomplish their work. A student working several months in the organization tends to be able to develop insider relations that foster cooperation, trust, and eventual success. Over the past few years we have managed to support student's participation in an industrial setting for research in the areas of software reuse, industrial practices and requirements elicitation/development [3,4,5,6,7].

4. Summary

It appears to us that industrial experiences are useful to the student, the faculty, and the industrial partner. The experience necessary is different between undergraduates and graduates. Certain key obstacles need to be overcome for graduate students to include NDAs and publication concerns. With undergraduates – the key problem is to find real customers and involve key industrial practices, stresses, and environments in the experience.

5. References

[1] Vaughn, R., "Teaching Industrial Practices in an Undergraduate Software Engineering Course", Journal of Computer Science Education, Swets & Zeitlinger, vol 11, no. 1, pp. 21-32, 2001.

[2] Vaughn, R., " A Report on Industrial Transfer of Software Engineering to the Classroom Environment", Thirteenth Conference on Software Engineering Education and Training, 2000.

[3] Kudikyala, K. and Vaughn, R. "Understanding Software Requirements Using Pathfinder Networks", CROSSTALK, The Journal of Defense Software Engineering, vol 17, no 5, May 2004, pp. 21-25.

[4] Kudikyala, K. and Vaughn, R. "Software Requirements Understanding using Pathfinder Networks – Discovering and Evaluating Mental Models", Journal of Systems and Software volume 74/1 pp. 101-108 Jan 2005

[5] Vaughn, R. and Lever, J. "Third Party Walkthrough Inspections: A Joint Navy/University Empirical Software Engineering Project", Fourteenth Annual Software Technology Conference, Salt Lake City, UT, Apr 29 - May 2, 2002.

[6] Vaughn, R. and Wolfe, L., "Teaching Software Engineering - Implementation of the Industrial View", Twelfth Annual Software Technology Conference, Salt Lake City, UT, Apr 30 - May 4, 2000.

[7] Vaughn, R. and Fonash, P., "The Practice of Software Code Component Reuse - DoD and Commercial Perspectives", Tenth Annual Software Technology Conference, Salt Lake City, UT, April 19-23, 1998.