Adding Excitement to Student Projects: Try Web-Based Industry Collaboration

Shan Barkataki, Tom Bolton
California State University, Northridge/ Northrop-Grumman Corporation, NSD

Introduction

This paper describes an experiment where university students collaborated with an industrial partner in the design and development of "real-world" software in an actual production project. Students were given real tasks that appeared in the project's Pert chart; therefore their assigned tasks had real deadlines, and late delivery of the student tasks had the potential for adversely affecting the whole project schedule. This required the students to have frequent interactions with the industry partners to resolve technical issues, report status, and participate in peer reviews. For all practical purposes, the students worked as members of the project development team, except that they worked in a laboratory at the university. However, we wanted the students to act like students, i.e., work part-time (they had other courses) and keep irregular hours (typically by starting in the afternoon and working late into the night and, of course, the weekends). Finally, we wanted to investigate the feasibility of conducting all collaboration activities through the Internet. Accordingly, all interactions with the industrial partner were done electronically, using e-mail, desktop conferencing, and the ordinary telephone. Except for a "getting-to-know-you" pizza bash at the start, there were no face-to-face meetings.

Industry-Academia Collaboration

There is a long history of collaboration between industry and academia to benefit training and education[6]. Such collaboration provides mutual benefit and serves as an excellent vehicle for closing the gap between industry and academia [14]. One dominant form of industry-academia collaboration is industry-sponsored research done at the universities. We did not pursue this avenue because our focus was on industry collaboration in the undergraduate and master's level classes.

In 1995, a Working Group for Software Engineering Education and Training (WGSEET), within the Software Engineering Institute (SEI), began to track industry-academia collaboration activities related to training and education [17]. The group conducted surveys and interviews to determine the number of universities participating in industry-academia collaboration and the various forms in which these collaborative activities occurred. The WGSEET reports indicate worldwide interest in industry-academic collaborative work. From these reports and other published work [8, 9, 10] it would appear that the predominant form of this collaboration involves universities delivering specialized training to industrial partners [4, 5]. Another popular form of collaboration involves students doing projects given by industrial partners as part of their academic program [2, 15]. These tend to be small projects that the students can do without requiring a great deal of interaction with personnel from the industrial partners. Yet another approach involves mimicking an industrial development environment within the classroom [16].

Our Approach

In contrast to the collaboration methods described above our method involved the students working in a real, live project. We found this to be a successful approach, providing a win-win game plan for all parties involved. Students are excited by the prospect of working on "real projects," with "real engineers." This type of work merits as professional experience in the resume. The industry partner is pleased to have a channel to influence the training and education of its potential employees and have the opportunity to recruit "job ready" graduates. It works for the university, because such collaboration aligns well with two of its goals—providing students with professional experience, and serving the local industrial community. Finally, faculty benefits by being able to interact with their industrial colleagues and from the opportunities for professional development.

Fair Compensation for Students and Funding

Our method should not be construed as a scheme to exploit cheap student labor. Funding for this pilot project was through a grant from our industrial partner. Although there was no contractual agreement for the university (and hence the students) to deliver anything, the students did professional quality work. We strongly believe that if the students are expected to deliver industrial quality products on production schedules, then they should receive comparable compensation. Accordingly, we arranged for the students to receive an hourly rate comparable to the starting salary of newly hired computer science graduates. This was well above the rates students earned in typical university jobs; however, we believe our *Fair Compensation Policy* was a significant factor in successful completion of the experiment. We would recommend the *Fair Compensation Policy* to others planning similar work.

Students submitted time cards for hours worked. This gave us an opportunity to teach them about ethical and responsible time keeping, an important professional practice. We were pleased to observe that although the students mostly worked unsupervised, they did keep honest records of the hours worked.

Role of Teamwork and Industrial Practices in Education

The IEEE/ACM final report on Curriculum 2001 specifically recommends incorporation of collaborative team projects and industrial experience [1]. Not surprisingly, projects involving collaborative teamwork have become an integral part of engineering and computer science education at universities worldwide. The June newsletter of the Forum for Advancing Software Engineering Education (FASE) describes thirteen case studies involving project-based learning [12]. These case studies describe a variety of approaches for providing students with opportunities to do collaborative projects within the framework of university courses.

Student Participation in Industrial Production Projects

It is not a common occurrence to see students working on production projects as part of their academic program. Discussions with potential industrial partners revealed two major concerns for their reluctance to enroll students in a production team. The first concern was difficulty in communication. A manager responsible for delivering a software product needs to meet regularly with her team. This is necessary to keep track of progress, conduct reviews, and resolve a multitude of issues that may arise during project execution. It is difficult to do these things with students, especially when they are geographically separated from the main development site. The second concern was related to industrial security and protection of intellectual property. By working on this project the student and faculty might gain some insight into the company's product line and there was a concern that this information may be leaked to outsiders.

Web-Based Desktop Conferencing

We resolved the first concern with readily available web-based collaboration tools. Students participated in meetings, group reviews, and technical discussions using desktop conferencing tools. Previous experience had convinced us that slow bandwidth could be a real problem in desktop conferencing. We needed a fast connection and were pleased to have the college IT department provide us with a fast and dedicated Internet connection (approximately 1.2 megabits/second). Students used the typical desktop conferencing tools— whiteboards, remote program/desktop sharing, chat utilities, and fast file transfers. We tried out several different collaboration tools and found that Microsoft® NetMeeting® served our purpose quite well (NetMeeting comes free with Windows). Most desktop conferencing tools, including NetMeeting, provide audio and video conferencing over the Internet. However, even with our T1-like speed we found these features difficult to use. For audio connection we found speakerphones at both ends to be the ideal solution. We gave up on Internet-based video conferencing. Although it was interesting to see the participants through the camera, transmission delays and jerky pictures were distracting. During our desktop conferences we used NetMeeting's shared whiteboard, shared applications, and remote desktop control features for displaying diagrams and code listings.

Internet Security

Maintaining industrial security during desktop conferences and file transfers was a major concern for our industrial partner. Based on the advice from the college IT department and security experts from our industrial partner we placed the student PCs in a separate subnet, isolated from the main campus network. We shut down all non-essential TCP/IP ports on these PCs. Finally, we installed a security appliance supplied and managed by our industrial partner. This appliance provided a strong firewall protection to the PCs in the lab. There was a similar device installed at the industry partner's site and the two together created a highly secure tunnel between the PCs in the lab and the industry partner's intranet.

Protecting Intellectual Property

The problem of protecting intellectual property proved more difficult to solve. We addressed some of the problems by taking measures similar to those used by the company when engaging a subcontractor, such as signing non-disclosure agreements. Our industrial partner made sure that the students and faculty were not exposed to any proprietary information. At the university, we used physical isolation to protect all project-related data from unauthorized access. In the end, the scheme we used proved to be both effective and inexpensive. We gave each student a 40-

gigabyte removable hard disk that held the entire operating system, the development platform, plus all work products the student developed. Whenever a student wanted to work on the project she physically mounted the removable hard disk on the PC, booted up the system and did her work. At the end of the work session the disk was removed and locked away securely. The operating system on each disk had logons for only the student and the supervising faculty. It was similar to giving each student a dedicated PC but at a cost of around $100 per student instead of $2000 per student. We could have encrypted the data stored on the disks, however in the light of the physical security these steps were considered unnecessary.

Another challenging problem arose when the students needed access to a database, located in the industry partner's site, for testing their applications. We set up the software and appropriate procedures to access the database from PCs in the lab through the secure Internet tunnel. However, in the end, direct access to the company test database was considered a high security risk. For the same reason, the database could not be replicated in the lab at the university. The solution was for the students to create and populate their own test database, which they did with help from the industrial partner.

Project Description and Student Tasks

The project our students worked on was a large software system estimated to grow to over one million lines of developed code. The system architecture had many software components and a separate team was responsible for developing each such component. It was designed using object-oriented methods. Implementation used four different languages (C++, JAVA, C, and Ada-95) and many off-the-shelf software packages as reuse products.

Our industrial partner created the actual student assignments. We had agreed that the assignments should expose the students to the idiosyncrasies of a large-scale software development environment. Some characteristics of such environments are:

- Unclear requirements specification at the start of the project with requirements stability increasing as the project progresses through the development life cycle.
- Need for discussions and negotiations between the "developer" and the "customer".
- Attempting to develop solutions without a guarantee that a solution meeting all the requirements can be found within budget and schedule.
- Emphasis on reuse-based development.
- Performing trade studies and selecting design alternatives.
- High workload and compressed schedule.
- Need for formal planning, and monitoring.
- Regular status reporting.
- Frequent peer reviews and a good deal of systematic testing.
- Use of automated design and development tools.
- Professional and ethical behavior.
- Need to acquire skills outside of the classroom, for example, by reading books, articles, reference manuals, and by trial and error.

The actual assignments involved design and implementation of software in the Operator Interface component. Our students completed two major tasks. One involved design and development of Java applications to initiate and manage a number of different human computer interaction sessions. The other project involved development of web applications for updating a central database server. Discussions during and after the project revealed that the students did experience most of the issues listed above and had a good appreciation of the differences in the characteristics of academic and industrial work.

Conclusions

This paper has described a practical approach for introducing industrial experience into the curriculum. Rather than dealing with toy problems, this approach engages the student in real projects with real problems. In this approach, the students gain valuable professional experience. In addition to overcoming the technical challenges, they learn to deal with problems posed by unstable requirements together with issues such as security, professional ethics, status reporting and group reviews. Our specific project was completed successfully. Students were very pleased with the opportunity they had to participate in the web-based work environment and gain industrial experience while still in college. Our industry partner was also pleased with the results of this experiment. The work products created by the students have been put to actual use. Pleased with the success of this pilot project the organization is considering similar ventures in the future, although these will be funded with contracts rather than grants. We were encouraged by the success we had in web-based collaboration. Our experience validates the feasibility of students participating as team members in distributed work groups and establishes a new model for introducing industrial experience into an engineering curriculum. This has the potential of opening new doors in many areas of industry-academia collaboration, including the possibility of international collaboration.

Acknowledgements

The authors would like to thank Mr. Bill Allison, Vice President and General Manager of the Navigation Systems Division of Northrop-Grumman Corporation and Mr. Gary Mallaley, Chief Technology officer at the same organization, for their support in accomplishing this task. We also extend gratitude to the students in the COMP698 class who did the actual work and the engineers at Northrop-Grumman Corporation who participated in this experiment.

Bibliography

1. ACM, IEEE, (2001) "Computing Curricula 2001", http://www.computer.org/education/cc2001/final/cc2001.pdf
2. Almstrum, L. V., (1999), "A Course With Service-Oriented Project Work", Forum for Advancing Software engineering Education (FASE) Volume 9 Number 08 (115th Issue). http://www.cs.ttu.edu/fase/v9n08.txt
3. Beckman, K., Coulter, N., Khajenoori, S., and Mead, N., (1997), "Collaboration: Closing the Industry-Academia Gap", IEEE Software, V 14, No 6
4. Beckman, K., (1997), "Directory of industry/university collaborations with a focus on software engineering education and training, version 6", SEI Special Report, CMU/SEI-97-SR-018, Software Eng. Inst., Carnegie Mellon Univ., Pittsburgh, PA

5. Beckman, K., (1999), “Directory of industry/university collaborations with a focus on software engineering education and training, version 7”, SEI Special Report, CMU/SEI-99-SR-001, Software Eng. Inst., Carnegie Mellon Univ., Pittsburgh, PA
6. Beckman, K., O' Mary, G., Lawrence, J., Cynthia L. Parish L. C., (1999), “Tracking and Evaluating Industry/University Collaborations for Software Engineering Education and Training”, Proceedings of 1999 IEEE 12th Conference on Software Engineering Education and Training, New Orleans, Louisiana
7. Beckman, K., Lawrence, J., Mead, N., O'Mary, G., Parish, C., Unpingco, P., Walker, H., (2002), “Industry/University Collaborations: Different Perspectives Heighten Mutual Opportunities”, Draft to appear in Journal of Systems and Software, http://www.sei.cmu.edu/collaborating/ed/indust-univ-collabs.html
8. Ellis, J. C. H, Nancy R. M., Moreno, A., McNeil, P., (2001) “Can Industry and Academia Collaborate to Meet the Need for Software Engineers?” Cutter IT Journal, June 2001
9. Ellis, J. C. H, Moreno, A., Mead R. N., Seidman, B. S., (2002), “Reeducation to Expand the Software Engineering Workforce: Successful Industry/University Collaborations”, SEI Special Report, CMU/SEI-2002-SR-001, Software Eng. Inst., Carnegie Mellon Univ., Pittsburgh, Pa.
10. Ellis, H.J.C., Mead, R. N., Moreno, A., Tanner, D. C., Ramsey, D., (2002) “Characteristics of Successful Collaborations to Produce Educated Software Engineering Professionals,” Computer Science Education Journal, Swetz & Zeitlinger publishers.
11. Hilburn, G. T., (2001), “So Let's Make That Software Project Course ‘Real-World’”, Forum for Advancing Software engineering Education (FASE), Vol. 11 No 04 (135th Issue) http://www.cs.ttu.edu/fase/v11n04.txt
12. Jarzabek, S., (2001), “Teaching Software Project Courses”, Forum for Advancing Software engineering Education (FASE), Vol. 11 No 06 (137th Issue) http://www.cs.ttu.edu/fase/v11n06.txt
13. Last, M., (2001), Survey Results on Virtual Team Practices in CS Education, St. Edward's University, Austin, TX, http://acad.stedwards.edu/~last/virtual_team_results.htm
14. Mead, R. N., (1996) “Industry-University Partnerships: The Wave of the Future?” Proceedings of the IEEE *1996* Australian Software Engineering Conference, Melbourne, Australia.
15. Walker, W. H., Coleman S., Gaberell, M., (1999) “Incorporation of Project-Based Learning in an Environmental Engineering Course at The Ohio State University”, Proceedings of the 1999 American Society for Engineering Education Annual Conference & Exposition, Charlotte, NC
16. Weinstein J., Gilchrist IV G., Hebsch, K., Stevens, J., (2002) “Providing a Real World Experience in the Teaching of Computer Technology”, Proceedings of the 2002 American Society for Engineering Education Annual Conference & Exposition, Montréal, Quebec, Canada.
17. WGSEET, (2002), Working Group - Software Engineering Education & Training, Software Engineering Institute, Carnegie Mellon Univ., Pittsburgh, Pa, http://www.sei.cmu.edu/collaborating/ed/workgroup-ed.html

Dr. SHAN BARKATAKI is a professor of Computer Science at California State University, Northridge, California. He specializes in object-oriented software development, UML, and e-business/e-commerce technologies. He has over 24 years experience in aerospace and defense industries. He received a Ph. D in Computer Science from University of Bradford, England.

Mr. TOM BOLTON works at Navigation Systems Division of Northrop-Grumman Corporation, Woodland Hills, California. He graduated in Computer Science from California State University Northridge and has over 20 years experience in software development both as an engineer and as a manager responsible for development of various software systems. His current interests are in human-computer interfaces and JAVA application development.