Is Integration of Communication and Technical Instruction Across the SE Curriculum a Viable Strategy for Improving the Real-World Communication Abilities of Software Engineering Graduates?

Panel Moderator
Gerald C. Gannod
Computer Science and Software Engineering
Miami University
gannodg@muohio.edu

Panelist
Janet E. Burge
Computer Science and Software Engineering
Miami University, Oxford OH
burgeje@muohio.edu

Panelist
Paul V. Anderson
Howe Center for Writing Excellence
Miami University
Paul.Anderson@muohio.edu

Panelist
Andrew Begel
Microsoft Research
Redmond, WA
Andrew.Begel@microsoft.com

Abstract

Software engineering educators and trainers are acutely aware that software engineering graduates need strong real-world communication abilities. The National Science Foundation is supporting a three-year project in which industry professionals, CS/SE faculty, and communication-across-the-curriculum specialists are collaborating to develop curricula and teaching resources designed to improve communication abilities of CS/SE graduates by integrating communication instruction and assignments with the technical work in courses across the students’ four years of study. Our panelists—an industry practitioner, a CS/SE educator, and a communication specialist—will describe what has been learned in the project’s first half and invite comments, insights and advice from the audience.

Introduction

Software engineering educators and trainers are acutely aware that software engineering graduates need strong real-world communication abilities [1, 2, 3]. University efforts to provide their graduates with these skills usually involve requiring Computer Science and/or Software Engineering (CS/SE) students to take communication courses taught by another department or designating one or two SE courses as communication-intensive. Neither strategy has proven to be so effective that employers and universities have stopped searching for better ways to prepare students to communicate on SE topics in the real-world situations of the SE profession [4]. The National Science Foundation is supporting a three-year project in which industry professionals, CS/SE faculty, and communication-across-the-curriculum specialists are collaborating to develop curricula and teaching resources designed to improve communication abilities of CS/SE graduates by integrating communication instruction and assignments with the technical work in courses across the students’ four years of study.

This approach is bolstered by leading-edge communication research showing that well-designed, intellectually challenging communication assignments based in a course’s subject matter can enhance students’ mastery of technical content as well as build their communication
skills [5,6]. Such integration also motivates students by showing the direct connection in the workplace between technical work and the communication that makes the results valuable to the employer, clients, and other stakeholders. However, development and implementation of this strategy faces several significant challenges, including the belief among many CS/SE faculty that they are not capable of teaching or evaluating communication. Our panelists—an industry practitioner, a CS/SE educator, and a communication specialist—will describe what has been learned in the project’s first half and invite comments, insights and advice from the audience.

The project’s SE professionals have executive or managerial responsibility in large organizations, such as Microsoft, SAS, and Northrop Grumman, and small companies, such as Integrated Industrial Information, Inc (I3). The faculty are specialists in computer science and communication at 14 universities ranging from the homes of the PIs, Miami University and North Carolina State University, to MIT. The project began with a day-long session in which industry participants the communication abilities required by new college graduates for the faculty participants. The project’s goal is to develop teaching supports, assignments, and other resources that CS/SE faculty can use to create assignments and instruct their students in four areas: writing, reading, speaking, and teaming. While this project focuses on academic education, its strategy may suggest new approaches to communication training in industry.

Panelist 1: Andrew Begel

Biography

Dr. Andrew Begel is a researcher in the Human Interactions in the Programming Group at Microsoft Research, Redmond, WA. He studies software engineers at Microsoft to understand how they communicate, collaborate and coordinate, and how this impacts their effectiveness in collocated and distributed development. After conducting studies, he builds collaboration tools to help mitigate the issues that were discovered.

Position Statement (An Industry Perspective)

In an empirical research study observing newly hired Microsoft developers who had just received their university degrees, we found that the employees' most difficult challenges were not in applying their technical skills (e.g. design, programming, testing, debugging, etc.). Instead, they struggled to learn how to communicate appropriately in oral and written form with their manager and colleagues in a hierarchically organized team [3]. New developers at Microsoft are often assigned to fix bugs or work on non-critical side projects in order to learn the software. However, these tasks have additional objectives to provide opportunities for the new employees to get to know their manager and teammates and learn the rules and practices of the software methodology employed in their project. They learn their team's culture by asking questions, participating in team meetings, observing others' work, and by being observed and guided by colleagues who mentor them. All of these communication opportunities are fraught with the anxiety, stress, and self-doubt common to anyone who begins a new job, and are magnified to those with no prior work experience. As the newcomers progress, of course, their confidence grows, and their colleagues begin to accept them as equal members of the team. New college graduates’ transition to Microsoft and other CS employers would be much quicker and much easier if they received better preparation for workplace communication in college.

While most programs include instruction in communication, their strategies are not as effective as they might be. Communication skills are often taught in isolation from CS
technical work, leaving students without an understanding of how these skills apply to their future careers. When programs incorporate workplace communication instruction into their own courses, they sometimes employ inauthentic work scenarios. For example, in many team projects, students are all positioned as having equal authority, identical responsibilities, and equivalent experience. These projects would be more valuable if they taught the oral and written communication skills students need when working on the much more common kind of teams where their colleagues will have different levels of power (manager), degrees of knowledge (senior software engineers), roles (testing, gathering requirements, marketing, shipping, leading meetings), and depth of experience (engineers who already know the codebase and its history). Students need instruction and experience in developing these complex communication skills in many courses, not just those that lend themselves to team projects. I will describe and invite suggestions about the ways the NSF-sponsored project is taking advantage of this important opportunity for industry and universities to develop CS curricula that incorporate communication instruction that is extended broadly through students’ education and is based on real-world practice.

Panelist 2: Janet E. Burge

Biography

Janet Burge is an Assistant Professor in the Miami University Computer Science and Software Engineering department. Her research interests include design rationale, software engineering, AI in design, and knowledge elicitation. She is a co-author (with Jack Carroll, Ray McCall, and Ivan Mistrik) of the book “Rationale-Based Software Engineering”. Dr. Burge is a recipient of a NSF CAREER Award for her project “Rationale Capture for High-Assurance Systems” and is also the lead PI on the NSF CPATH project “Integrating Communication Learning Outcomes into the CS and SE Curriculum.” She has been at Miami University since 2005. Prior to that point, she worked for more than 20 years in industry as a software engineer and research scientist.

Position Statement (An SE Educator Perspective)

Most CS/SE programs culminate in some form of a senior capstone project. Ideally, students come to this project already having developed the communication skills needed to successfully complete the project. In this final experience before graduation, they should be extending and refining their professional communication abilities, not beginning to learn them. Unfortunately, in many CS/SE programs, students do not receive the needed instruction and extensive practice in communication before the end of their studies.

Often, instruction on writing and speaking is relegated to a technical communications course. Unfortunately, students may not take this course early enough in their studies to apply to their capstone course (since it is taught outside the CS/SE department and usually not a prerequisite). Further, it is difficult to know how to apply skills learned out of context. Most CS/SE programs require students work in teams and make presentations in various courses but do not explicitly teach teaming and speaking skills. Finally, reading skills are often assumed. No attention is paid to the many specific reading tasks that are essential in software engineering. Because there is no consistent emphasis on communication, students fail to recognize the importance of these “soft skills” and refer to them as “busywork.”

These three issues—context, instruction, and importance—can be addressed only by emphasizing communication in multiple core courses and providing explicit instruction, rather than assuming that students will pick up these skills somewhere else. By distributing
instruction across courses, programs can enable students to exercise and apply these skills in multiple contexts, they can emphasize that the skills are a critical part of being a computer scientist or software engineer, and they can reduce the burden that would otherwise be placed on one or two instructors and courses.

This approach to communication instruction faces many challenges. In most CS/SE programs, faculty cannot assume that students have been introduced to certain skills in classes already taken because students can take courses in many sequences. There is also the add/subtract problem: How can programs add communication instruction without removing required technical topics? Most CS/SE instructors have not been trained in teaching about communication. It can be difficult to persuade some that these skills are important enough to warrant the time and effort required to adopt new forms of instruction that develop them.

To meet these challenges educators need the following: 1) identification of the key skills required of CS/SE graduates and expression of the skills in the form of student learning outcomes to be addressed in the CS/SE curriculum 2) example “roadmaps” for incorporating instruction in these skills in the core curriculum; 3) training for CS/SE faculty in teaching communication skills; 4) training for CS/SE faculty in developing communication-intensive assignments; and 5) sample assignments at multiple levels that combine communication and technical skills in contexts related to work students will do in their careers. I will describe progress the NSF project has made in developing these resources, identify challenges remaining, and invite reactions and ideas from the audience.

Panelist 3: Paul V. Anderson

Biography

Paul Anderson is Professor of Technical and Scientific Communication and Director of the Howe Center for Writing Excellence at Miami University. He is a Fellow of the Society for Technical Communication, the Association of Teachers of Technical Communication, and Miami University’s Institute of Environmental Sciences. His publications have won awards from the National Council of Teachers of English and the Society for Technical Communication.

Position Statement (A Communication-Across the Curriculum Specialist Perspectives)

The best way for students to develop disciplinary communication skills is by studying and practicing them within the context of the discipline [7]. Through instruction and assignments given in their technical courses, students learn how to deploy the genre, styles, and other conventions of their disciplines in ways that meet the needs and serve the purposes of their future employers. Recent studies also demonstrate that well-designed writing assignments in technical courses increase students’ mastery of course content [5,6]. Writing and other communication tasks reinforce technical instruction rather than detract from it.

For decades, specialists in communication across the curriculum have been developing strategies for helping disciplinary faculty integrate communication instruction into their courses. Yet, the NSF project presents some unique challenges. We’ve found that communication assignments can be built around existing technical assignments by placing the technical work in the context of scenarios that are based on the way SE professionals would communicate the technical results to someone else. However, some of the six core courses on which the NSF project focuses lack technical assignments that can be augmented in this way for all four communication modes (writing, speaking, teaming, and reading). Should all four modes be addressable in all of the courses? If not, how can CS/SE programs assure that
adequate attention is paid to each mode in their overall curriculums? We must also determine how to scaffold instruction so that the recurrence of writing, speaking, teaming, or reading instruction in advanced courses builds and broadens, rather than merely repeats, what students have already learned.

Additional challenges arise from the project’s goal of providing resources for CS/SE programs throughout the country. We quickly discovered that CS/SE programs vary much more than we had anticipated. Courses central to one program don’t appear in others. Courses with the same name can have much different content. Likewise, even in different sections of the same course at the same institution can use different technical assignments for projects. Our challenge is to devise highly flexible, yet still useful, assignments and instructional aids and then to arrange them in a depository (presumably online) that can be searched so easily that CS/SE faculty will be persuaded to go there more than once.

While these implementation issues are significant, the largest challenge is motivational: convincing CS/SE faculty that significantly revising their curricula and courses will better prepare students for SE careers by increasing their communication abilities and their technical expertise.

**Biography of Moderator: Gerald C. Gannod**

Gerald C. Gannod is a professor in the department of computer science and software engineering at Miami University. His research and teaching interests fall in the areas of software engineering, enterprise computing, digital humanities, applications for mobile platforms, mobile learning, and automated technologies for streamlining assessment of student learning. Dr. Gannod joined the Department of Computer Science and Software Engineering in 2006. In 2010, Dr. Gannod became director of the Miami University Mobile Learning Center, a joint venture between the School of Engineering and Applied Science, the Armstrong Institute for Interactive Media Studies, and Information Technology Services.

**Acknowledgments**

This work was funded by NSF CPATH-II Award CCF-0939122/CCF-939081. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation (NSF).

**References**


