

Coaching Via Cognitive Apprenticeship

Ray Bareiss
Carnegie Mellon Silicon Valley
Building 23
NASA Ames Research Park
Moffett Field, CA 94035
011-650-335-2801
ray.bareiss@sv.cmu.edu

Martin Radley
Carnegie Mellon Silicon Valley
Building 23
NASA Ames Research Park
Moffett Field, CA 94035
011-303-818-3283
martin.radley@sv.cmu.edu

ABSTRACT

At Carnegie Mellon's Silicon Valley campus we employ a learn-by-doing educational approach in which nearly all student learning, and thus instruction, is in the context of realistic, team-based projects. Consequently, we have adopted coaching as our predominant teaching model.

In this paper we reflect on our experience with the nature of teaching by coaching using a framework derived from Cognitive Apprenticeship, and explain how we employ the techniques it suggests in our teaching. We also discuss a range of instructional tensions that arise in teaching by coaching and present a survey of student attitudes regarding the effectiveness of our approach.

Categories and Subject Descriptors

K.3.2 [Computer and Information Science Education]: Active Learning. Pedagogy. Software Engineering.

General Terms

Human Factors.

Keywords

Software Engineering Education, Active Learning, Learning by Doing, Story-Centered Curricula, Cognitive Apprenticeship

1. INTRODUCTION

Carnegie Mellon's Silicon Valley campus offers an MS in Software Engineering with Technical and Development Management tracks targeted at working software professionals [2]. We also offer an MS in Software Management intended to be a "technical alternative to an MBA" for mid-career software professionals.

The Carnegie Mellon Silicon Valley Software Engineering curriculum comprises six semesters worth of project-based courses. Working both in teams and individually, students act as employees of a fictional company hired to architect, design, and implement or manage multiple releases of a complex application on an aggressive schedule. Within this structure, and with support from faculty members, students confront realistic technical and business problems, including conflicting requirements, limited

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGCSE '10, March 10–13, 2010, Milwaukee, WI, USA.
Copyright 2010 ACM X-XXXXX-XXX-X/XX/XXXX...\$5.00.

time and resources, team leadership challenges and ineffective communication.

Given the nature of the knowledge and skills being taught and Cognitive Science research suggesting that the most effective teaching is delivered during active problem solving (see, e.g., [3]), we have adopted coaching as our predominant teaching model (Our teaching model is actually somewhat more complex than we present in this paper, employing subject-matter experts and role playing faculty in addition to coaches. Please see [2] for a more complete discussion.)

Our early approach to teaching through coaching emphasized open-ended questioning (cf. [6], [10]), encouraging students to figure out how to complete assignments on their own with coaches providing guidance almost exclusively by asking questions. While this appeared to produce good learning outcomes, we observed that students became frustrated with this approach and wanted more direct guidance on how to complete their work successfully, balancing the pain of discovery and the amount of effort required to master the subject matter. We began a search for a more varied and balanced model of teaching through coaching and found one in Cognitive Apprenticeship.

2. COACHING VIA COGNITIVE APPRENTICESHIP

Cognitive Apprenticeship transfers the instructional techniques of apprenticeship from their traditional realm of physical activities, such as carpentry or metal work, to mental activities which have historically been taught via traditional classroom techniques:

"While there are many differences between schooling and apprenticeship methods, we will focus on one. In apprenticeship, learners can see the processes of work: They watch a parent sow, plant, and harvest crops and help as they are able; they assist a tradesman as he crafts a cabinet; they piece together garments under the supervision of a more experienced tailor. Apprenticeship involves learning a physical, tangible activity. But in schooling, the "practice" of problem solving, reading comprehension and writing is not at all obvious -- it is not necessarily observable to the student. In apprenticeship, the processes of the activity are visible. *In schooling, the processes of thinking are often invisible to both the students and the teacher. Cognitive apprenticeship is a model of instruction that works to make thinking visible.*"[4] (Italics are ours.)

Cognitive Apprenticeship suggests that problem solving skills can be taught in the context of scaffolded problem solving by observation of modeling by expert practitioners, appropriate removal of the scaffolding (fading), and by reflection on student experiences. Collins, et al, [4, see "Method" beginning on page

13] provide a coherent and seemingly comprehensive catalog of teaching techniques. We have diverged from their model in one key regard which we found to be problematic: Collins, et al, identify “coaching” as an instructional technique, but its description subsumes the other techniques enumerated in the model, implying that coaching is indeed a superordinate category. We have, therefore, dropped coaching as a distinct category.

The following sections present a discussion of how we have interpreted and implemented each technique in our teaching context:

2.1 Modeling

When modeling problem solving, a coach “thinks aloud”, verbalizing the thought process as he or she solves an authentic problem. For example, when teaching a Requirements Analysis course, one of the authors modeled analyzing the “whole product” required to solve a customer’s entire business problem by walking through Moore’s model [8, p. 113] using the example of the products produced by the author’s previous eLearning company to show how eLearning software alone does not solve companies’ training problems and to articulate what a complete solution would entail. Note that the students were being asked to develop requirements for quite a different application, a social software application for ridesharing. We have found that it is important to model by analogy so students cannot directly copy the answer, but rather they must process what they observe and transfer it to a new problem-solving context.

We have also found storytelling to be a compelling approach to modeling, one that makes coaching vivid and memorable for students [9]. For example, during our Requirements course, students complained about the difficulty of getting interview subjects to describe their work processes specifically during requirements elicitation. The coach responded with the story of an experience interviewing design engineers during which he encountered the same problem. He explained to the students that he solved the problem by having each engineer literally hold, during the interview, an artifact that the engineer had designed and focusing the interview on the specific process by which that artifact had been designed. True stories conveying the coach’s real-world experience are especially powerful as a source of guidance, and when we find good stories we tend to reuse them.

2.2 Scaffolding & Fading

Collins, et al, describe scaffolding as “the supports the teacher provides to help the student carry out the task.” We have implemented three forms of scaffolding: task design, direct guidance, and feedback:

2.2.1 Task Design

In the bulk of our courses, we define the scope the tasks and provide input material our students need to do meaningful work while focusing on targeted knowledge and skills. In doing so, we eliminate peripheral, time-consuming, and (sometimes) overly difficult aspects of the work to be done. For example, in our Introduction to Human-Computer Interaction (HCI) course, students are given a product vision and requirements documents (including well-organized interview data) to enable them to focus on interaction design rather than having to conduct the entire requirements process as a prelude. The coach may also decide to

adjust the students’ assigned tasks on the fly, relieving the burden on students who are struggling while further challenging students who are excelling. For example (again from the HCI course), an advanced group of students might be asked to storyboard a mobile version of the desktop application they were originally assigned.

2.2.2 Direct Guidance

The coach must constantly monitor a team’s progress to ensure that they are not “spinning their wheels” and, thus, become frustrated and losing motivation. In such situations, the coach will provide “just-in-time hints” to a team, or if a student team is really struggling, the coach may provide even more specific direction. In doing this, the coach must prioritize the key learning objectives to be mastered based on the team’s skills, capabilities, and available time.

2.2.3 Feedback

While coaching, we constantly provide verbal and written feedback to a team of students. In particular, we have found written feedback to be quite effective, and we employ a “limited mastery approach” in which teams submit a deliverable, receive detailed feedback, and are permitted to revise and resubmit it before it is graded. Feedback ranges from directly correcting minor errors to pointing out common errors, which are left to the team to correct, to asking open-ended questions to encourage them to think more deeply (e.g., “You specified that the system *must* respond to all queries within 2 seconds. What if it took the system 3 or even 4 seconds to respond to 25% of queries? Would the system be a failure?”). In one author’s courses, the density of comments is approximately seven per page on first drafts of a typical written deliverable. We have found verbal feedback to be especially impactful if provided by role playing faculty members [2] in the context of a meeting, such as a presentation to management, in which students’ work is evaluated as it would be in an industrial context.

As students gain mastery, scaffolding is gradually *faded*, i.e., removed (generally later in the curriculum rather than within a single course), and the students are expected to take more direct responsibility for planning and monitoring their own work. Scaffolding is almost completely faded when students undertake a real-world Practicum (i.e., capstone) Project at the end of their studies. During the Practicum, students are not provided with supportive curricular materials (although they can and should refer back to the material provided in previous courses), and the coach no longer gives directive guidance, instead reminding the team that they now own the entire process and its outcomes. That said, however, the coach does continue to observe the team closely and is ready to intervene when major problems arise (especially problems with real-world clients which may be particularly difficult for the team to address themselves).

2.3 Articulation

Articulation involves getting students to verbalize their knowledge and reasoning. We continue to rely heavily on questioning for this aspect of instruction. For example, when students in the HCI class are beginning to think about conducting usability tests, the coach will encourage them to articulate their thought process by asking such things as: “Do you agree with Krug’s assertion in *Don’t Make Me Think* [7] that “the importance of recruiting representative users is overrated”? “When do you think this does and doesn’t matter?” “What types of user

attributes are most likely to matter in your project?” After completing their project, they are asked to articulate what they have learned in the form of a presentation to faculty roleplaying senior management in which each team proposes best practices in interaction design drawn from and justified in terms of their experience. (Note that this activity also promotes reflection.)

Many of our courses also require student teams to present their recommendations to faculty role playing as company executives. For example, in our Project and Process Management course, student teams are asked to recommend an appropriate software development methodology for a specific project and to articulate their knowledge by justifying their recommendations in a presentation which will be accepted or rejected based on the quality of their justifications.

2.4 Reflection

Reflection helps students to critique their own performance with the goal of extracting general lessons from a recent specific experience. Collins, et al, suggest this most often occurs in the context of comparing their work and thinking to that of others. For example, when a task in any of our courses culminates in a presentation, students may observe the presentations of other teams including the faculty’s feedback, which highlights strengths and weaknesses of the work. As previously noted, students receive comprehensive written feedback on all deliverables which provides another basis for reflection. Finally, at the end of most courses, the coach facilitates a “reflection session” during which the team discusses its work processes and their outcomes, as well as their performance as a team, and the team is encouraged to articulate general lessons learned which will be applicable to similar work in the future. We should note that we do not always promote reflection adequately, and we consider this to be one of the major areas for improvement in our teaching.

2.5 Exploration

Exploration involves pushing students to become independent problem solvers, requiring them to frame their own problems and to define and execute the processes for solving those problems. In our curricula, such exploration is largely restricted to the final Practicum Project in which student teams must interact with real-world clients to understand requirements and negotiate the scope of work and then to define and manage their own work processes, and as noted earlier, to own the results of their decisions. We are considering options for such things as independent study projects and using real-world projects earlier in the curriculum to enable students who are ready to engage in exploration earlier in the curriculum.

3. INSTRUCTIONAL TENSIONS

Over the past six years, we have encountered a range of tensions while teaching by coaching:

3.1 Does the Coach Need to be a Subject-Matter Expert?

In our experience, the coach does not need to be a subject-matter expert (SME); however the coach must be adequately versed in the subject matter to facilitate student learning.

The coach must exceed a threshold of expertise to get and maintain buy-in and respect from the students and to know if a

team is “fooling” him or her. This threshold can change depending upon the expertise of the students. For example, one of the authors has coached teams through a Foundations of Software Engineering course in which the students develop a software product using an agile methodology and tools. The author is not a hands-on software developer at the same level as many of the students; however the author has managed agile software development projects and is very familiar with agile methodologies. This has enabled the author (as a coach) to resolve the majority of student questions. Technical questions that the author cannot answer are forwarded to another faculty SME for resolution. We see this as being analogous to professional sports where many successful coaches cannot play the sport at the level they coach.

If the coach is not an SME, the coach and any SME faculty involved in the course must be on the same page in terms of the course learning objectives and the current thinking on industry best practices to avoid conflict between what the SME espouses and what the coach implements.

We support faculty coaches who are not SMEs by creating coach’s notes. Coach’s notes provide a description, by week, of the key topics, points and logistics that the coach must cover with their student teams. However, providing coach’s notes brings the overhead of maintaining the notes and may lead to “prescriptive coaching” (in which the coach leads the team to answers based on the coach’s notes) rather than “adaptive coaching” (in which the coach helps guide the team to an outcome which is the appropriate culmination of its own work).

3.2 Should Coaches Grade?

After a coach guides a student team in creating quality deliverables and pre-reviews those deliverables, it is difficult to objectively critique the final version of those deliverables. Of course, the coach is in the best position to know how and where the team has been guided and whether the team paid adequate attention to that guidance. At the same time, the coach wants the team to do well, and so may interpret what they read or hear in more positive terms. Correspondingly, students may lose trust and confidence in their coach if they perceive that their coach is not setting them up for success in the course.

3.3 Student Buy-in

Students, especially those new to this teaching approach, are used to faculty providing them with prescriptive answers through lectures and are expecting to be tested on remembering the answers. We have found that students can quickly become experts at getting coaches to provide “the answers” instead of learning on their own (for example, telling their coach: “I don’t get it...I don’t know what to do...You have to help me.”).

In addition, the goal for some students becomes completing the work as quickly as possible while still getting a good grade rather than maximizing learning.

Coaches need to proactively acquire student buy-in to the instructional approach and to shift the discourse from grades to the value of learning outcomes. One effective technique has been to invite alumni to talk about their learning experiences during new student orientation to the program. Another is to acknowledge that a fictional learning context can never match reality, and to ask the students to “suspend their disbelief” about

the fictional project scenarios and act fully in their fictional work roles.

3.4 Individual Mentoring

Coaches uncover a remarkable breadth of student needs, ranging from addressing lack of confidence to career mentoring to resolving personal problems. Such issues are outside the scope of the curriculum, and time for coaches to address these issues is not an “official” part of the coach’s workload, so it is up to each coach to determine how much time they spend on these issues each semester. For many faculty the prospect of working personally with the students is what makes teaching rewarding, even if he or she has to use personal time to do so.

3.5 Team Issues

Coaches also uncover a remarkable breadth of student teaming issues, from lack of interpersonal skills to students who don’t pull their weight to (in rare occasions) students who are verbally or psychologically abusive. Unlike individual issues, resolving these issues is part of the curriculum; however it is still up to each coach to determine how much time he or she spends on these issues each semester.

One author calls team members individually once in the middle of each semester just to see if the student has something they might share privately but will not share in a group session. In most cases, no new information emerges, but occasionally a student will open up and provide information that enables to the coach to provide the student with specific guidance, address team issues, or take other actions as warranted.

3.6 Coaching Remote Students

Thirty to forty percent of our students are remote (i.e., they live far enough away that they cannot regularly come to campus). Having remote students on teams is an instructional virtue for our programs since virtual work is increasingly the professional norm, and our students can develop their virtual teaming skills as part of their studies.

Coaches must be careful to not favor local students, who participate more visibly, to the detriment of remote students. Coaches can level the playing field between local and remote students by:

- Holding coaching meetings by phone (and collaboration software), whether students are local or not.
- Proactively querying the students in all team conversations. Do not let students sit silent.
- Learning about virtual teams and virtual body language [5]
- Watching for bias in the evaluation and grading of local students over remote students.

3.7 Is a Coaching-Based Approach Scalable?

In our experience, a single faculty member can both coach and provide subject-matter expertise to a small course (25 or fewer students); however some of our courses have 60 students with even larger classes predicted in the future. Instead of attempting to hire multiple subject-matter experts to teach different sections of these large courses, we have used one faculty member to act in the roles of the supervising faculty, fictional role player (e.g., the VP of Engineering), and subject-matter expert, and we have used

other, sometimes less expert, faculty as coaches to work with as many sections of students as needed.

To validate the viability of this model we collected faculty effort data during the offering of a 58-student course that used 5 faculty working with 12 student teams over 7-weeks. One faculty member coached 4 teams, while the other faculty members coached 2 teams each.

Two faculty averaged less than 1 hour per week per student (.72 and .94), while the other three faculty averaged much higher (1.67, 1.81 and 1.92). If we assume that 75% of the coach’s preparation work goes away the second time they coach a course (this was the first time that this course had been taught), then all coaches would be under 1.5 hours per student per week. We have thus set a goal of spending a maximum of one faculty hour per student per week (since greater efficiency has been proven possible).

Of course, gathering teaching time metrics puts pressure on the coaches to limit their “face time” with students and, in fact, may cause coaches to be more directive in order to reduce student time demands. To avoid this unintended consequence we regularly reflect with the coaches in our larger classes about their interactions with students and other faculty, framing the discussion of time spent in terms of “high quality” (focused on facilitating key learning outcomes) versus “low quality” hours. Our goals are to achieve a crisper definition of good coaching practices and to build camaraderie and rapport among the faculty.

4. DOES OUR COACHING APPROACH WORK?

During June 2009, we surveyed our current students and *all* alumni (since 2003) to ascertain their attitudes towards our teaching methods, and 116 of 375 responded. Overall, 75% believe our teaching methods are more effective than what they have experienced in other university education; 13% believe they are neither better nor worse; and 12% believe they are worse. With respect to teaching specific types of skills:

- Technical skills: 65% believe they are effective, while 19% are neutral, and 16% believe they are ineffective
- Business skills: 82% believe they are effective, while 11% are neutral, and 7% believe they are ineffective
- Professional skills (e.g., teamwork, communication, negotiation): 88% believe they are effective, while 9% are neutral, and 3% believe they are ineffective.

The most frequent reasons respondents cited (via free text comments) for believing our methods are worse than traditional methods include:

- Too much emphasis on what the students think and too little on what the faculty think during discussions
- Faculty do not (typically) present an overview to provide a conceptual framework for learning new material
- Limited individual feedback (in contrast to team feedback which is copious)
- Faculty do not track teams carefully to ensure balanced contributions by all members.

(And three of 116 thought that the use of conventional lectures would improve teaching.) The first two reasons are most notable because they imply that some faculty do not adequately balance

the use of various teaching options; in particular, some faculty may continue to overuse the technique of open-ended questioning and underuse techniques such as modeling and providing direct guidance when appropriate to scaffold a struggling team. The second two reasons primarily reflect a limitation, we recognize, of available faculty time.

In contrast, students who believe the methods are effective cite the initial frustration but ultimate benefit of faculty not simply providing answers. Our goal as teachers is perhaps best illustrated by this quote from an alumnus:

The student seems to absorb the material. It becomes very much a part of you without you realizing it until you find yourself in a situation that requires application of some aspect of what you learned.

5. CURRENT THINKING ON THE IDEAL COACH

We have established a mature curricular framework; however we still leave many teaching decisions to the intuition of individual coaches, resulting in a variety of styles and range of effort, with corresponding variations in effectiveness. That said, our current thinking about the definition of an “ideal coach” is that ideal coaches:

- Adjust to use different teaching methods based on their assessment of a student team’s specific situation and needs
- Allow students to diverge from course planned content if/as warranted
- Balance use of the various techniques of Cognitive Apprenticeship, including open-ended questioning, appropriately given a team’s situation, abilities, and mental state
- Have an adequate level of subject-matter knowledge
- Ensure that the students develop and use appropriate work processes and apply what they are learning to their work
- Are flexible with their scheduling and availability to students
- Are willing to engage with the students to address and resolve individual as well as team issues
- Are experienced in facilitation
- Are able to establish an empathetic relationship with individual students and teams
- Are comfortable with virtual teamwork and distance learning.

Are coaches born or produced? We argue that smart, empathetic people can be taught to be good coaches assuming we can codify which teaching techniques are likely to be most effective in which situations and we can train coaches in areas such as:

- Teaching smart people how to learn [1]
- Providing a Cognitive Apprenticeship

- Modeling and coaching effective leadership, teamwork, communication, etc. as well as technical knowledge and skills
- Handling problem students and teams
- Gaining student buy-in to our educational model and teaching approach.

Our program initially employed part-time industrial practitioners as coaches; however because of the wide variability in quality and high administrative overhead, we have evolved to a small group of full-time faculty. This might be the end of the story if the plan was to not grow the student enrollment. Given planned enrollment growth, our ability to codify and impart our teaching methods to new coaches is critical to our successful growth as well as to the replication of these methods to other institutions.

6. ACKNOWLEDGMENTS

Our thanks to the Carnegie Mellon Silicon Valley faculty, past and present, for their dedication, energy and innovation that have made this program a success.

7. REFERENCES

- [1] Argyris, C. Teaching Smart People How to Learn, Harvard Business Review. May-June, 1991
- [2] Bareiss, R. and Griss, M. 2008. A Story-Centered, Learn-by-Doing Approach to Software Engineering Education. Proceedings of SIGCSE 2008 (Portland, Oregon, March 12-15, 2008).
- [3] Bransford, J. D., Brown, A. L., and Cocking, R. R. (Eds.). 2000. How People Learn: Brain, Mind, Experience, and School. National Academy Press, Washington, DC.
- [4] Collins, A., Brown, J.S., and Holum, A. 1991. Cognitive Apprenticeship: Making Thinking Visible. American Educator (Winter 1991), pp. 1-18.
- [5] Croasdell, D., Fox, A. and Sarker, S. 2003. Systems Development by Virtual Project Teams: A Comparative Study of Four Cases. *Annals on Information Technology, Volume 5*, pages 447 – 463.
- [6] Hmelo-Silver, C. and Barrows, H. Goals and Strategies of a Problem-based Learning Facilitator. *Interdisciplinary Journal of Problem-Based Learning* 1, 1 (Spring, 2006), 21-39
- [7] Krug, S. 2006. Don’t Make Me Think, Second Edition. New Riders Publishing, Berkeley, CA.
- [8] Moore, G.A. 2002. Crossing the Chasm, Revised Edition. Harper Business, New York, NY.
- [9] Schank, R.C. 1995. Tell Me a Story: Narrative and Intelligence, Northwestern University Press, Evanston, IL
- [10] Staff, Harvard Business School. 1998. Case Method Teaching, Report 9-581-058 (November 6, 1998).