Adapting Scrum to Managing a Research Group *

September 18, 2010

Department of Computer Science Technical Report #CS-TR-4966

Michael Hicks Jeffrey S. Foster

Dept. of Computer Science
University of Maryland, College Park
{mwh,jfoster}@cs.umd.edu

1. Introduction

Working with and mentoring Ph.D. students is the central activity in running an academic research group, with two broad goals: (1) to collaboratively produce high-quality research results, and (2) to help students to become independent researchers capable of working at research labs or academic institutions. The two of us are examples of successful mentoring, having fulfilled these goals; as Ph.D. students we published regularly, and are now tenured associate professors. Now our job is to mentor our own students to achieve a similar, or superior, level of success.

At first, we followed a simple, fairly typical approach to mentoring: we met once or twice per week with each student in roughly half-hour or hour-long slots. Meetings covered the gamut of interactions, from status reports to brainstorming to solving technical problems. When we needed more frequent or lengthy interaction, e.g., due to an impending deadline, the fact that we had few students and few outside commitments made additional meetings easy to arrange.

Unfortunately, as the number students grew from two or three each to six or seven each, and as our outside commitments steadily increased, our simple approach reached its limits. The sheer quantity of meetings was interfering with their quality, and our days became driven by scheduling constraints rather than by the importance of research. Because meetings were infrequent and our schedules were full, we contributed to research haphazardly, and were slow to react to difficulties. These problems were compounded by a lack of cohesion among our students. Most were working in isolation, and there was little sharing of ideas or collaboration among them.

In this paper, we describe how we have largely solved these problems by adopting a management approach we called SCORE, adapted from the Scrum agile software development methodology.[1] The Scrum methodology has several elements, but the central one (and the one from which it draws its name) is a 15-minute, daily, all-hands meeting (called the Scrum) in which developers update everyone in the group on (1) what they did since the last meeting, (2) what problems they encountered, and (3) what they plan to do for the next meeting. In-depth technical meetings are scheduled on an ad hoc basis. In SCORE, like Scrum, we hold 15-minute all-hands *status meetings* three times per week in which students report their progress, any obstacles encountered, and their planned next steps. Whenever we need to have a longer, technical discussion, we schedule an *on-demand* meeting—and since the only scheduled meeting is the Scrum, such meetings are far easier to arrange. To help increase group spirit further, we have a weekly lunch, and we also hold a reading group one day per week.

Though SCORE is conceptually simple, its benefits to us have been significant. It is now easy to keep up-to-date with students' progress, and when students are struggling, it often only takes a day or two to realize something is not right, and to begin to address it. Our time is spent far more effectively, because the status meetings themselves are low overhead, and we only hold the longer one-on-one meetings when they are needed. On-demand meetings have a clear purpose and are therefore much more productive than our weekly meetings used to be.

SCORE has also significantly benefited our students. We conducted a survey of our students' opinions about SCORE, and found that every student reported that his research experience is significantly better than it was before. Students say they are more productive, more enthusiastic about research, and have better interactions with other students and with their adviser. Students reported that there is now a real sense of community in the group that was never there before. Students know what other students are doing, in substance and in approach. Students are interacting more among themselves and with faculty other than their adviser, and are more

* This paper expands on a two-page paper by the same authors that appeared in the October 2010 issue of *Communications of the ACM* entitled "SCORE: Agile Research Group Management" (Hicks and Foster 2010).

[1] SCORE stands for "Scrum for Research"

supported and supportive. There is a shared sense of success. As a natural consequence, senior students have started to mentor junior students more directly, which in turn better prepares them for faculty jobs. A number of colleagues at other institutions have also begun to adopt SCORE, and they have reported to us similar benefits.

Retrospectively, we believe SCORE's benefits have arisen from two principles embodied in the Scrum/SCORE approach:

1. *Keep status separate from research.* The scrum meeting is for status, and the on-demand meetings are for solving unforeseen problems. Keeping the two activities separate allows them to be undertaken more efficiently.
2. *Meaningful group contact creates community.* By giving each person a window onto the activities of others, the scrum meeting lets participants learn from others' successes and failures, and helps create a group sense of momentum and accomplishment.

Over the last several years, our group coalesced into a strong research *community*, where research is now more fun and interesting for everyone.

In the main body of this paper, we go into more detail about our old approach, describe SCORE more fully, present some qualitative survey results on SCORE, and discuss some potential variations. We hope this paper contributes to a broader discussion of how to best mentor Ph.D. students and run a successful research group.

2. The Old Way

During our first several years as assistant professors, interactions with our students centered around weekly, individual meetings. We would meet with each student one-on-one to discuss research ideas, work out technical problems, and plan research tasks. With around three students each, this approach worked well. We saw students regularly during their prearranged time slot(s), and as problems arose, new results came in, or paper deadlines approached, we had the flexibility to meet with students more often, as needed.

As time progressed, though, this simple approach became less effective. The number of students between us grew to about ten, and our service commitments (reviewing, program committees, etc) also increased. As our schedules became busier, it became hard to have impromptu meetings with students, which meant that most interaction took place during the weekly (or twice-weekly) scheduled meetings. As a result, several problems emerged.

First, our meetings became extremely inefficient and ineffective, due to internal fragmentation. Generally speaking, the time needed for a one-on-one meeting can vary significantly: a brief status report can take 10 minutes, while working out solutions to a technical problem can take more than an hour. With so many students, a prearranged half-hour or hour long meeting slot became the worst of both worlds. When the student only had status to report, the meeting slot was too long, and even if we ended quickly it was hard to make use of the remaining meeting time for work. On the other hand, when the student had a deep technical issue to explore, there was not enough time, and a follow-on meeting might have to wait a day or two, slowing progress and increasing overhead.

Second, students could fall through the cracks between meetings. If a student was struggling, we might have to wait a week to find out. Then typically we'd give the student some advice, and another week would go by, and sometimes there would be no improvement. At that point we could intervene more strongly, but two weeks had gone by, and it could take another week of frequent meetings to get back on track.

Third, we began to notice that we had built up a set of great individual students, rather than a collaborative research group. Each student was working on an individual project, and the students' principle interactions were with his or her adviser. This meant that when one of us was away traveling, the students' progress would often drag, because the only person who knew enough to help them out—their adviser—was not available. Moreover, we were building up isolated islands of knowledge within the group, and students did not know what the other students were doing. This meant that we were the conduit—more often, a bottleneck—of dispersing knowledge throughout the group. Even when one student could be helped by talking to another student, neither of them would know it until they spoke with us. This also decreased opportunities for collaboration across students and their exposure to ideas, since students would tend to focus solely on their own work.

Clearly, something needed to change.

3. The New Way: SCORE

While at a research meeting in September 2006, the first author happened to chat with an officemate from graduate school, Jon Moore. During the conversation, Jon mentioned that in his software development job he was really impressed with the "agile" methodology they used, which was called Scrum. As he was describing the system, the light bulb went on: We could adapt Scrum to running our research group to solve many of the problems we were having.

3.1 Scrum

Figure 1 illustrates the basic Scrum process for developing software.[2] Development tasks are divided among several teams, each consisting of approximately seven members, one of whom also plays the role of the *scrum master* who organizes the group. Teams implement product features in a series of four-to-six week *sprints*, each of which culminates in a working prototype. At the end of a sprint, the development team and management hold a half-day or all-day planning meeting to decide what tasks should be carried

[2] `http://www.controlchaos.com/images/diagram/flow.gif`

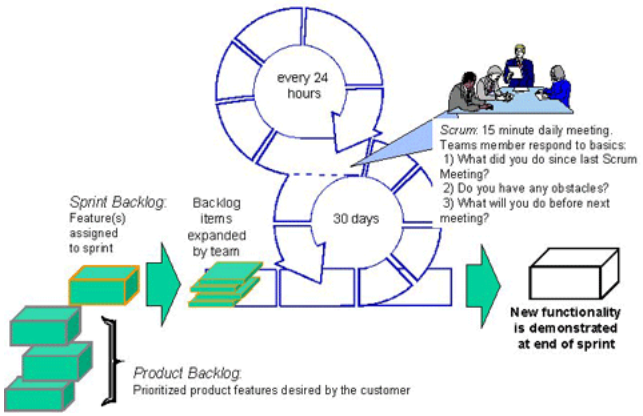

Figure 1. Scrum workflow

out during the next sprint; this set of tasks is called the *sprint backlog*.

Every day during a sprint, teams hold a *scrum* meeting,[3] in which each team member answers three questions: (1) What did you do since the last scrum meeting?; (2) Do you have any obstacles?; and (3) What will you do before the next scrum? For any issues that cannot be immediately resolved, or if a team member seems to be having trouble making progress (whether they realize it or not), the scrum master sets up separate meetings or takes whatever other action is appropriate. Scrum meetings should last no more than 15 minutes, and to help ensure this, team members typically stand during the scrum.

3.2 SCORE: Adapting Scrum to research

We viewed the Scrum methodology as a way to address many of problems described in Section 2. The key attraction was the scrum meetings, because they had the potential to decrease schedule fragmentation—longer research meetings could be scheduled as the need became evident following a scrum meeting, rather than pre-allocating fixed slots for each student. We also thought scrum meetings could increase communication within the group, since every team member would hear the other members' progress.

SCORE has several elements (and is still a work in progress), but draws its two key elements from Scrum: three-times-per-week status meetings, and one-on-one meetings, scheduled on demand.

Status meetings. The centerpiece of SCORE is the *status meeting*, analogous to the scrum meeting. We currently hold status meetings on Tuesdays, Wednesday, and Fridays. We thought about holding daily meetings, as in Scrum, but everyone in the group felt this was too often (though we have never tried it).

[3] In analogy with the same term from Rugby, in which a mass of Rugby players vie for the ball.

During the meeting, everyone describes what they have done since the last meeting, the results achieved and the obstacles faced, and what they plan to do by the next meeting. The status discussion may be described in the context of a larger goal, e.g., working towards a paper deadline. We professors also describe our activities to the group, so that students learn more about our jobs, and to perhaps increase the chances they may take a faculty position when they graduate. Students sometimes participate in the status meetings by conference call. This helps people stay in touch even if they cannot be physically present.

Research progress can take many forms, and so students talk about a wide range of activities, such as implementing code, carrying out an experiment, reading a paper, working on a proof, writing up a result, or preparing a talk. It is not important that students stay rigidly focused on a short-term goal, but rather that they make *some* kind of progress in their research each day. We encourage students to present their status to the group, rather than just to the faculty, and we try to be understanding and encouraging in response. Students may also say there has been no change in their status, typically because of classwork or for personal reasons.

As with scrum meetings, we try to keep status meetings to 15 minutes, which gives each person roughly one or two minutes to talk. It requires some discipline to stick to such short status presentations and avoid delving into technical details—such in-depth discussions are delayed until the on-demand meetings (below). To help keep us more focused, we now stand during the status meetings, as with scrum meetings. Surprisingly (to us), this really works, and status meetings almost always complete on time.

We sometimes tack on a short (10 minutes or less) meeting at the end of a status meeting, when appropriate for the whole group. For example, we might hear a brief summary of a conference someone has just attended, discuss what papers to read in our reading group (below), or clarify administrative issues in the group.

On-demand Technical Meetings. Status meeting presentations by design do not go into much technical depth. To discuss research questions, methods, technical challenges and results, etc., we hold *on-demand* meetings between the student(s) and their adviser(s). Notably, these meetings are not regularly-scheduled; we always schedule such meetings when the need arises.

SCORE on-demand meetings are like those in Scrum for "removing obstacles" but are much broader in scope. Typically, we schedule on-demand meetings based on something that comes up in a status meeting. For example, a student might have some experimental results to discuss; be stuck on making progress and need some advice; or may just want to brainstorm about some research ideas. In the best case, students request these meetings, though for less experienced students we often are the ones doing the requesting.

In Scrum, planning takes place during end-of-sprint group meetings. In SCORE as we practice it now, we plan research activities during on-demand meetings. For example, we might discuss next steps once a paper has been submitted; meet to plan activities leading up to a paper deadline; or lay the groundwork for new research when we discover a promising idea. We suspect that exploring new research directions could be done better, as we discuss further in Section 5.

The frequency of on-demand meetings varies tremendously. When senior students are making good progress on their own, we may go one to two weeks without an on-demand meeting. In contrast, in the weeks before a paper deadline, we typically have on-demand meetings at least daily, even if only for a few minutes each. One of the strengths of this approach is exactly this ability to adapt to students' needs, so that those who need or want help get it quickly (and do not have to wait for a scheduled meeting).

In addition to one-on-one student-adviser meetings, we have recently begun scheduling short talks on demand, too, which are given to the entire group. Most often, a student will talk about a research milestone, e.g., as detailed in a paper just submitted. Such a talk is essentially a "checkpoint" of a research activity, as it presents the most relevant details and the end results of an effort witnessed in small bits over several weeks or months. In addition, a student may talk about a tool or language with which they are familiar that is of interest to others in the group, a fact that may have come up during a status meeting.

Other elements of SCORE The status meeting and on-demand meetings are the Scrum-inspired core SCORE. Apart from these elements, we try to foster a greater sense of community in two other ways. First, we co-locate our students in the same office space as much as possible. Greater proximity tends to foster greater levels of interaction. Second, we also have regular social outings. On Wednesdays, we all go out to lunch together, and every couple of months we go out for beers to celebrate various achievements. Finally, we have a weekly reading group that meets on Mondays. (Meeting on a day with no status meeting increases the days on which we have face-to-face contact.) We read one paper per week, and the responsibility for running the discussion rotates among the students. The reading group is optional, but most students regularly participate.

3.3 Benefits to us

For us, SCORE has several benefits, which redress the problems mentioned in Section 2.

First, the regular status meetings keep us up-to-date on how students are doing. Rather than having a week or two go by while a student struggles, we (and the student) can usually tell after a couple of status meetings that something is not right, and then set up an on-demand meeting to get back on track.

Second, our time is now spent far more effectively. The status meetings themselves are extremely efficient. With everyone in the room at the same time, we can very quickly share status information and hear about others' progress. Because the faculty have large blocks of time set aside for on-demand meetings, those slots can be filled at the last minute (e.g., right after a status meeting) according to actual needs, rather than being fragmented by pre-allocated meetings

Third, one-on-one meetings are far more productive than before. Almost by definition, on-demand meetings have a clear purpose, such as a particular problem to solve. Moreover, they allow us to contribute to actual research on a deep level. Since on-demand meetings are focused, we can hold anything from a 15-minute meeting to a 3-hour meeting, as appropriate.

All of these benefits can be attributed to SCORE's adherence to the idea that status should be separate from research. Because research is a process with goals and methods, we can observe it and follow its progress in regular status meetings. But because the core activities of research are creative and unpredictable, we must allow ample time for ideas to be explored, accommodating surprises and inspiration.

We have found that SCORE also benefits the students. For example, our less constrained schedules make it easier for students to have more frequent informal interactions with us—the chance they can stop by the office and we will be available to chat for a few minutes is much greater than before. But the benefits go far beyond this, and they derive from the sense of community engendered by SCORE. In the next section, we describe the students' perceived benefits.

4. What Students Think

As our group adapted to SCORE, we informally polled students to ensure they were satisfied. After roughly a year and a half, we conducted a more formal survey among the students that regularly participate in our research group. This totaled 13 students of varying seniority, ranging from two second-semester graduate students to four sixth-year students about to graduate. All students had participated in SCORE for at least 6 months.

The survey had two parts. The first part asked the respondent to rate the quality of his or her experiences with graduate study using SCORE. Answers were given on a 5-point Likert scale, where a response of 1 was "very poor" and a response of 5 was "excellent." Of the 13 total respondents, the 8 that had experienced our original research group structure were also asked the same questions about the old way. The second part of the survey asked the respondents to expand upon their answers for SCORE—what benefits they saw in particular, what drawbacks, and suggestions for possible improvement.

The responses were uniformly positive: SCORE improved students' experience in every way we considered.

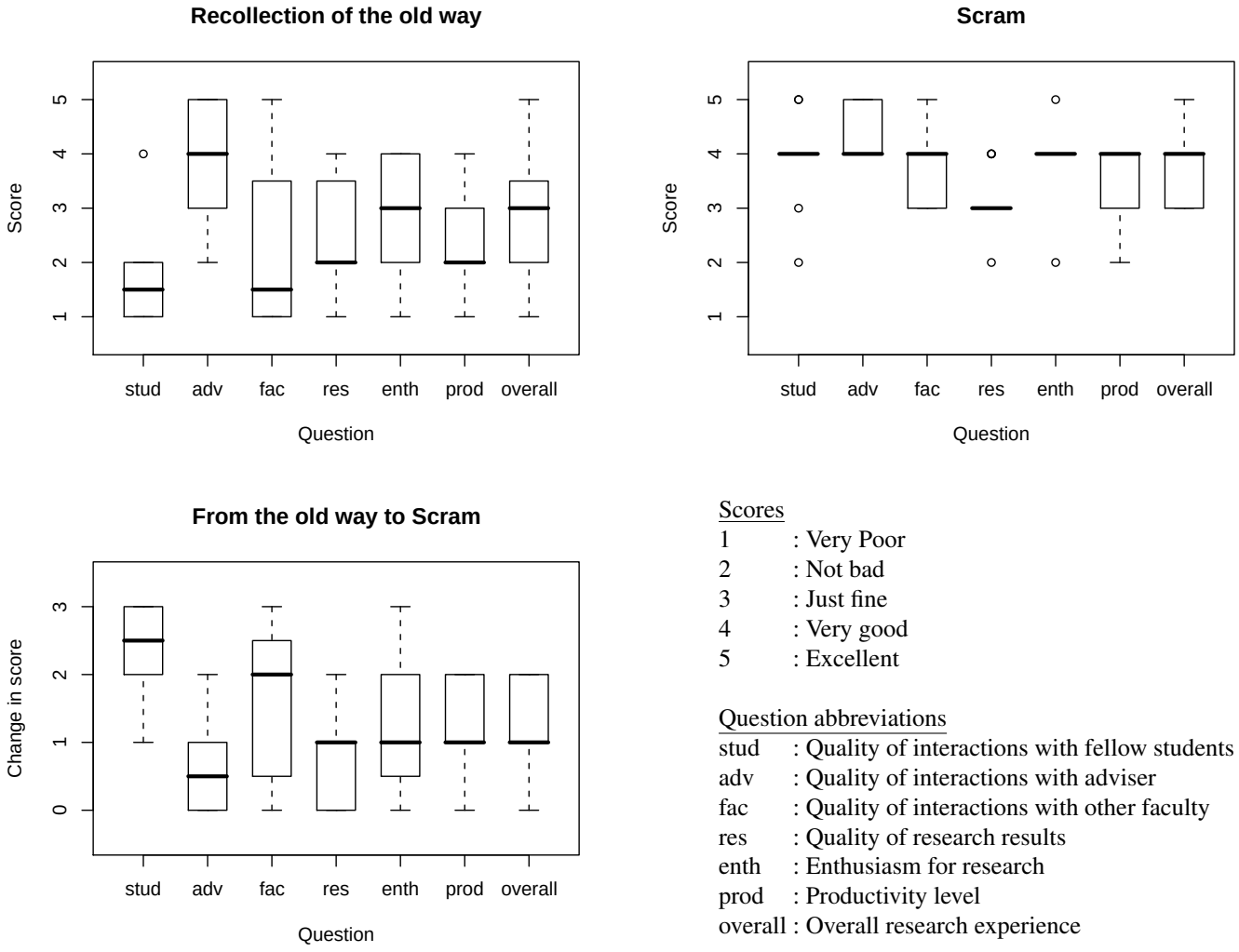

Figure 2. Survey Results, Numerically-scored Questions

4.1 Numeric Survey Results

Figure 2 shows the results of the first part of the survey. All the results are presented as box plots.[4] The boxplot in the upper left summarizes the recollections of the 8 students who experienced our original research method. The boxplot in the upper right contains the views of all 13 students who have experienced SCORE. The lower-left boxplot shows the difference in responses between the two for those 8 students who experienced both. The lower right corner of the figure summarizes the scale and the questions.

[4] In a box plot, the dark central line is the median, while the top and bottom edges of the box are the upper and lower quartiles, respectively. The difference between the two (the height of the box) is called the interquartile range (IQR). A response more than 1.5×IQR outside the box is considered an outlier. The smallest and largest non-outliers are indicated by a dashed vertical line or "whisker." Outliers are shown as open circle above or below these whiskers.

We observe several trends in this data. First, the change from our original approach to SCORE was, in the students' minds, unambiguously positive. As can be observed in the lower left boxplot, not a single response found SCORE to be lacking compared to our original approach. The most improved areas are student-student interactions (a median jump of 2.5 points) and interactions with other PL faculty (a median jump of 2 points). One would imagine these areas improved because of the increased face-time afforded by the status meetings, but we found the magnitude of the difference surprising. Section 4.2 lists some short answers that help explain the improvement.

Second, despite the fact that we no longer have weekly one-on-one meetings, student-adviser interactions slightly improved (by 0.5 points). Since starting SCORE, many students have taken advantage of our freer schedules, and meet with their adviser more regularly, for varying amounts of

time. So it's understandable they would find interactions better. Other students have been more resistant to meeting. In some cases they are just respecting our busy schedules, but in other cases they feel embarrassed that they have not made enough progress. Sometimes they solve their problems, but may also end up mired in some dark corner. With SCORE we can identify this situation reasonably quickly and set up one-on-one meetings to get the student on track.

Finally, while several categories have only improved by 1 point when considering the median, there is far less variance in the distribution. There were no scores as low as 1 in the SCORE results, and other than the question of research quality (answered by the students with some humility!) the results are clustered around 4.

4.2 Short Answer Results

The short answer questions asked students to elaborate on the benefits and drawbacks of SCORE, and to suggest possible improvements; we defer discussion of possible improvements to the next section.

Benefits Nearly every student cited the short, frequent status meetings as the key to SCORE. There were several observed benefits. First, regular, short meetings create motivation and pace for research work. One student says:

> I like the frequency of the status meetings. Frequent meetings make incremental progress necessary: to have something to say at each meeting, you can't goof off for an extended period of time. Also, if you don't know where to go next, there isn't much time before another meeting, when you can get back on track. On the other hand, the frequency of the meetings means that, if something came up and you don't have anything to report for today, it's not a big deal; you'll have something for tomorrow or the next day. I also like the fact that meetings the are short, and standing up is a nice way to emphasize the importance of brevity. It makes it clear that we are here to take care of business efficiently. Nonetheless, the meetings still have a friendly atmosphere, which I think is important. I also like the opportunity to follow up with quick 1-on-1 conversations right after the status meetings.

Comments like these provide support for the substantially-improved productivity reported in the numeric portion of the survey. While we might imagine similar improvements to come with more frequent status reports via some other means (regular e-mail and/or several short one-on-one meetings), several students pointed out that group status meetings were particularly effective. One student liked hearing about how and to what extent other students were making progress:

> I can follow other people's research and "daily research routine." That helps because it's interesting and I learn things, but also because I can compare my productivity and have a better idea of how I fare.

Another student mentioned it was also consoling to learn that other students hit slow patches, too: "It helped me with the realization that everyone has rough patches and that it is not a big deal." Finally, in some cases, even with the fairly brief reports, another student can offer methodological or technical advice, e.g., about problems are issues with a software tool. (Advice is usually provided after the status meeting itself.) One student said:

> ... when you're having trouble, there are 10 other students who may have dealt with the same problem in the past. ... If I'm stuck with [something] and bring it up at the status meeting, I've got two or three people who know how to fix my problem. While I could probably, eventually figure it out myself, it's MUCH more time efficient when someone else knows.

Generally speaking, the group-oriented approach catalyzes a greater diversity of ideas. Whether due to the status meetings or the research report meetings, more people hear about problems or ideas, and so more feedback and further ideas can be provided.

More than half of the students specifically cited a PL "research community" or "sense of belonging" as an improvement due to SCORE. Students seem to genuinely enjoy hearing about what other people are working on, but more than that, they enjoy hearing about the process. Several students also said they found it interesting to hear what faculty are doing, in anticipation of taking a faculty job after they graduate. The students say they feel the joy of their fellows' successes, which then creates further motivation and enthusiasm for their own work. Several students also pointed out that regular social gatherings (weekly lunch, occasional "beers" celebrations) and proximate offices were also important in fostering this sense of community. One student said, "Status meetings and the office atmosphere etc. make it worth my while to come to school." Our feeling from observing the group over the last year or so is that people are just having more fun doing research.

Four students specifically cited the 15-minute average meeting length as important. For example, one student said:

> At one time our meetings were getting to be pretty long ... it's good that we're back to 15 minutes, with standing up etc. We're really pretty close to ideal with the status meetings at this point ...

In the Fall of 2007 our status meetings were approaching 30 minutes as students talked more with their adviser, during the meeting, about particular technical issues. While the longer meetings produced more technical information, they did not generate more group interest or contribution. To the contrary, the longer meetings became boring and tedious, and so we rediscipliined ourselves to keep the meetings short.

Drawbacks Of the 13 students that filled out the survey, 8 cited no real drawbacks.

Two of the more junior students complained that the status meetings were too short to learn much about what other students were really up to, and that the research meetings did not always fill the gap. One senior student conjectured that junior students might not be getting as much out of SCORE as he did out of the regular attention he got from his adviser back when there were few students (but he stated the current system works great for him now).

Two students felt that while frequent meetings are an overall win, they do have drawbacks. One student felt that frequent meetings encourage steady progress on a single project, but do not really encourage concurrent exploration of new ideas (though this is not precluded):

> The current format is fairly regimented. Once on a project, I need to keep making constant progress each day. Earlier, with a goal for each week, I would have the option of setting aside a day or two for other vague research thoughts that I might want to explore that could lead to future projects. In the current format that appears hard. Despite the insistance [sic] that we could indeed state that as our status in the meeting, other than the senior students, most others will be unwilling to do that.

Frequent meetings may also be hard to schedule other events around. One student felt that meeting times could be improved, e.g., Monday, Wednesday, and Thursday could be better than Tuesday, Wednesday, and Friday, to "jump start" the week following the weekend.

5. Variations and Possible Improvements

5.1 The limits and pitfalls of SCORE

One important limitation of SCORE is that the group cannot be too large. With too many students the status meetings would become less useful. Another drawback of a large group is the greater demand for one-on-one meetings. While SCORE makes these meetings more efficient, eventually one hits real limits. At its peak our group had 14 students at various stages in their careers, with 12 or so people attending a typical status meeting. While SCORE worked at that size, we have found that a smaller group (right now we have 10) makes for easier, more intimate meetings.

Another question is the minimum group size. Scrum can be employed with a single manager and as few as three team members, but we have not explored the lower limits of SCORE. One student pointed out that having multiple faculty involved is particularly useful:

> Having at least two professors who are both cached-in on each person's research allows far more continuity when schedules are hectic. My guess is that if Jeff or Mike had tried to do this alone, it would have been harder.

Over the last two years, each of us has taken a substantial sabbatical. While we typically attended at least one status meeting via Skype during that time, the burden fell on the local person to help students with immediate needs. Attempting to carefully advise 10 students is difficult (at least for us) under any circumstance; SCORE made the task more efficient and more manageable.

Because SCORE only prescribes status meetings, it is the responsibility of all involved to not neglect other meetings. For example, it would be theoretically possible to go a long time before a student met one-on-one with his or her adviser. One possibility for avoiding this would be to create thresholds, as suggested by one student:

> If I have to suggest something, we could have thresholds, i.e. at least one presentation every so many weeks, or at least one one-to-one meeting every so many days, etc.

With outside collaborators who do not attend the status meetings, we often schedule a weekly research meeting; we find such meetings are complemented quite well by status meetings.

Finally, SCORE itself says nothing about what takes place in on-demand meetings. This depends on the project and the student, and lots of advice on this topic can be found elsewhere, e.g., desJardins (2008).

5.2 Some variations we reject

There are many ways to manage a research group. Here we briefly comment on some other approaches we thought about and why we chose not to adopt them.

Weekly group meetings Some research groups have weekly all-hands meetings, which often consist of status updates and/or talks by one or more students about their work. The goal of having students and faculty learn about what the others are doing is a good one. However, we see two problems with scheduled weekly meetings, based on our experiences as grad students and on the failure of the regular monthly meetings mentioned in Section 3.

First, if the group meetings consist largely of students presenting their current work, then there is often a strong temptation to have students present something, even if their work has not matured yet. We feel it is better to wait until one really has something to say, making the time better spent.

Second, if the meeting mostly involves status updates, a single weekly meeting will take too long. The key to SCORE is that status meetings are short and happen several times a week. In particular, we think three 15-minute meetings per week are better than a single 45-minute or hour-long meeting. Multiple meetings increase the pace of research, because students realize they are expected to make some progress each day. For example, a common pitfall for students is to wait until the day before a weekly meeting to work hard. Several status meetings a week encourage stu-

dents to make their progress smoother and more consistent. More frequent meetings also tend to be more information-rich, because it is easier for people to concisely summarize a few days worth of work than a week's worth. Shorter meetings keep participants' attention more effectively.

Email status reports. We occasionally use email status reports when scheduling issues preclude us meeting face-to-face. However, while e-mailed status report satisfy the "keep status separate from research" principle, we do not think they would make a good permanent solution.

The main problem is that email status does not create a sense of community in the same way an all-hands meeting does. Each individual emails their status out at some time of their choosing, changing a group activity into an individual activity. It is very easy to simply ignore email status reports from others—in the group status meetings, people pay attention because they want others to pay attention when they speak. Email status has no such social controls, and hence people feel less accountable. Finally, SCORE status meetings are high-bandwidth and provide an efficient broadcast mechanism to keep everyone informed. It is easy for someone to ask a clarifying question about a status report, or toss out an idea they just had. People are more reluctant to do this over email, because each message gets sent to the whole group, which feels like sending spam.

In sum, SCORE status meetings provide regular, personal contact, both student-to-adviser and student-to-student, and we believe this is essential to creating a collaborative, robust group dynamic.

5.3 Next steps

While SCORE works pretty well, it could be better.

The biggest limitation of SCORE as we practice it now is the longer-term administration of the research process is fairly ad hoc. In Scrum, planning takes place at all-hands meetings prior to the start of each sprint. Currently we make plans during on-demand meetings between a student and his adviser. While this generally works, it does not take advantage of the diversity of views and ideas within the whole group.

In an attempt to model Scrum more closely, we have experimented with longer meetings that share research results, plans, and broader progress with the whole group. At first we held these longer meetings roughly monthly, inspired by Scrum's sprint planning meetings. During the meeting we expected every student to give a 15 minute presentation about their current progress and results, with some description of their near-term plans. After a couple of months, these meetings became tedious, and we ultimately abandoned them. Retrospectively, the problem was that students were being forced to report on results prematurely, which decreased their interest to the whole group. This disinterest was compounded by the length of the meeting—while one might sit through one 15-minute talk of mild interest, it is much harder to sit through ten mildly-interesting talks.

As mentioned in Section 3, we now schedule talks on demand, and this works much better. While these presentations allow students to hear more technical information about others' work, which is good, they do not particularly enhance collaboration, since by the time students are ready for a presentation they have *completed* a result.

On the survey, students zeroed in on this problem. They cited the following high-level goals more than once as desirable: (1) more collaboration among students, and (2) more emphasis on generating ideas for new projects, rather than solely solving problems presented by current projects. For example, one student stated

> I still find collaboration lacking. Even though I know a whole lot of intricate details in the projects being pursued by others, I still feel that we are not attempting to collaborate together (ie. more students on each project). Most projects are being led by one (and having an optional new entrant in the group following). I find that mostly the barrier to entry is fairly high and having other people come in is difficult because of that.

Suggested solutions could address both concerns. One student suggested brainstorming sessions and 10-minute "idea" talks on a particular idea or problem. The goal is to get students thinking about new problems—whether in a new area or as part of a current project—and to share those problems with other students to encourage feedback and engender collaboration. Another student suggested a kind of "continuing education" to learn new skills and problems in a particular area. The student proposed having the reading group focus on one area, rather than the current format, which covers a wide breadth of topics during the semester. We tried this during Fall 2007, when our reading group focused on mechanizing programming language metatheory (an area that is gaining some momentum now in the PL community), and we found it to be quite effective in learning a new area. We also recently tried a "code reading group" in which we dug into existing programs to understand, as a group, how they work. Another thought was to formalize the idea of a "side project," like Google's "20-percent time" (Google 2008), and report on it once per week. Such a side project could support another student's project or be independent. We already encourage new students to get involved by working on a small part of a more senior student's project, so we could imagine extending this idea throughout a student's career.

6. The Future

We think SCORE is a successful research model, but there are many exciting new directions for the future. We welcome comments, questions, and spirited discussion about SCORE

or other ideas for building and managing successful research groups.

Acknowledgments

Many thanks go to the current and former students and faculty of the PLUM research group:[5] Jong hoon (David) An, Nat Ayewah, Brian Corcoran, Mike Furr, David Greenfield-Boyce, Eric Hardisty, Chris Hayden, Khoo Yit Phang, Martin Ma, Iulian Neamtiu, Pavlos Papageorge, Nick L. Petroni, Polyvios Pratikakis, Bill Pugh, Elnatan Reisner, Saurabh Srivastava, and Nikhil Swamy.

References

Marie desJardins. How to succeed in graduate school: a guide for students and advisors. *Crossroads*, 14(4):5–9, 2008.

Google. Google jobs. `http://www.google.com/support/jobs/bin/static.py?page=about.html&about=eng`, 2008.

Michael Hicks and Jeffrey S. Foster. Score: Agile research group management. *Communications of the ACM*, 2010. URL `http://www.cs.umd.edu/~mwh/papers/score-short.pdf`. To appear.

[5] `http://www.cs.umd.edu/projects/PL`