PERCOLATIONS

Lisa C.
Kaczmarczyk

Questioning Our Assumptions about Introverts and Computing

HOW OFTEN DO WE QUESTION our basic assumptions? In somewhat of a departure from my usual tack in this column, I'm going to challenge you to examine a few of our common assumptions, and percolate until you arrive at your own conclusions about how you view, value and teach future computing professionals.

Let's start with two commonly held beliefs about computer science and introverts. People in and out of the computing field often believe that a) technology is a field that is very attractive to introverts, and b) technology is a field that has traditionally been very amenable to introverts. As a result of holding these two beliefs we can (and generally do) conclude that a career in computing is a good match for introverts.

There is evidence in support of these two points. In her book, "*Quiet: The Power of Introverts in a World That Can't Stop Talking*" [1], Susan Cain cites a study showing that early computer enthusiasts were overwhelmingly introverts. On the same page she also discusses the open source movement as a form of team work highly attractive to introverts. Clearly, an open source development environment allows groups of people to develop large complex systems without having to interact face to face or even necessarily in any direct manner at all! If not for the open source movement, we would never have seen the likes of Linux and Apache. Bane of many writing teachers existence that it is, Wikipedia owes its birth and continued existence to a mostly anonymous mass of behind the scenes contributors—many of whom would likely call themselves introverts.

What about social media's effects on introverts? Cain claims that social media makes it harder to be an introvert. She reasons that because social media is less asynchronous and less anonymous than traditional internet communication it is less comfortable for introverts. Is this indeed the case?

Not necessarily, at least not for the technically savvy introverts we are often concerned about. We know that technically adept introverts find ways to remain anonymous, or to simply reduce the face-to-face team activities Cain suggests they dislike. In fact, there is an argument to be made here that social media provides introverts a 21st century way to comfortably communicate and work with a wider group of people than ever before. Thus, the first question for your percolation: *is interaction using social media hard for introverts?*

Moving right along: for some time it has been virtually obligatory to have our students work in teams. We do so in part because of educational research findings that show that group work improves learning outcomes. Fresh from the guilty secret department, I learned that many computer science faculty will confess (anonymously) that, personally, they hate working on teams. Yet they unquestioningly walk the talk of how preferable group work is over individual work in a corporate development environment. If you want to punch a hole in that argument, check out Cain's book for much convincing evidence that team work is not necessarily the most effective and efficient way to be creative or productive – even for extroverts. Much to percolate on there.

Recently I took part in a group discussion about introverts and careers in the tech industry. The participants were industry professionals ranging from recent entrants to experienced executives. In passing, someone said that working in an Agile/Scrum environment was very hard on introverts. At the time I nodded to myself as the conversation moved on. Later, I thought: is it? I wonder…

I formulated a question: *Does an Agile/Scrum environment favor extroverts; is an Agile/Scrum environment unattractive for introverts?* I polled three people I know well. Here are their responses (edited for length and to fix minor typos).

From a college professor who teaches software engineering:

> *"So your question doesn't have an easy straight-forward answer. To be sure, development processes like Agile and Scrum are less attractive for introverts. But let's really talk about software development in general. Is it really the case that you can be an introvert and isolate yourself from others and still have a successful career in software development? I don't really think that is very common these days. Software today is developed by large teams of people who have to be able to communicate and work together for the common task. In a traditional Agile environment, the introvert is not going to be as successful because he [sic] doesn't always want to speak up.*
>
> *Having said that, the statement isn't really that clear-cut because much development is done across different locations and with many developers working from home. These factors do not allow for traditional Agile development and Scrum. As a result, virtual scrum has*

> *become more common in this type of environment thereby tipping the scale back towards the introvert."*

From an experienced developer working in a global tech firm:

> *"I think in many cases it's true that development using agile favors an extrovert, because agile works best in smaller companies with relatively small development projects, where people will be expected to meet in person or on the phone every day for scrums, and where all stakeholders in the project will be involved, communicating with developers and testers on a daily basis. In small/medium companies, most of the staff is local - working in the same physical space with each other, and development with agile means that the developers are expected to actively talk to nearly everyone on the team every day - often in person.*
>
> *When agile is used in a large company, where much of the development and stakeholder staff is remote (typically in India and China), and where there are language barriers between team members, agile still means daily communication between many members of the team, but typically over a chat interface. This suits introverts in most cases, although agile (in my opinion) doesn't really work well in this scenario."*

From an early career developer in a small tech company:

> *"We've actually had this EXACT conversation at work a few times. Personally, I tend to be pretty social, and I don't mind the more team-oriented move as much. However, another thing that people are pushing lately is "pair programming" and a few of my coworkers and I view this much differently than our boss does.*
>
> *… while I don't believe [my boss] is the type who would actually like to pair program, he is the type to push it on the rest of us because he saw empirical data in a study that pair programming is really efficient and helpful to teams.*
>
> *So, we've gone back and forth with him a number of times on this. For example, I don't mind working with someone when I'm working through a difficult problem or out of ideas on something, but if I'm just blasting my way through a task, I don't see the point in having someone else there. Especially if it's code that's going to be fairly easy to understand (linking a UI to the back end, things that you do a lot of). But our boss read this article and started trying to get us to pair on everything.*
>
> *I work well with someone else if it's sometimes, but if it's constant, it can be way too much. I actually do better if I have little distractions, like Facebook, emails, IM, just little things that help pull me out of the deep hole that I'm in. If I stop focusing on a problem for a minute or two, I find that I usually get a better idea for how to do it, if I try to push it I won't get there until I go home and do something else for a bit. I find that if I'm pair programming, I don't have those distractions and my work actually gets worse because I spend half my energy forcing myself to focus really hard on the problem, and that's not what works best for me.*
>
> *Another coworker has argued this several times with my boss as well. He is an introvert, and while he's not bad to work with, he gets easily frustrated and he finds working with someone else constantly to be a very draining experience. … They've gone through this several times, and generally my coworker agrees to try it again, it works for a day, then our boss disappears to his other project and we all go our separate ways until our stand up meeting because it works better for us that way and our boss isn't around to notice, lol.*
>
> *I think developers tend to be such logical people that a lot of the time, especially the men (sorry, but it's been so true in everything that I've seen) see something in black and white in a study, and think they have to do it exactly that way forever for it work. They completely throw out the human part of the picture without watching what's actually happening to the team. I'm not saying you shouldn't try it the way it's written, but I've seen people say that you can't deviate even when the team members are unhappy for long periods of time, and that drives me crazy. They end up creating a problem that never existed before.*
>
> *… I would say that the biggest issue I've seen with "agile" and "scrum" is more the fact that people take them as gospel instead of recognizing what works best for their own people. So I could definitely see that agile methods which push for more team work, especially things like pair programming, are causing some introverts to shy away from the field.*

There you have it. You are now tasked with stepping away from your current assumptions about Agile/Scrum and team work in general within the development environment. After that, your final task is to deliberate on how you translate your assumptions (old or new) into the classroom. Ponder, ruminate, investigate. Percolate alone or in groups, via whatever medium you are most comfortable with. If you are one of the secret cadre of faculty who dislike group work but nonetheless enforce it, rest assured that you are not alone and your instincts may be good ones! Ir

References

[1] Cain, Susan. *Quiet: The Power of Introverts in a World That Can't Stop Talking.* (New York, Crown Publications, 2012): 79.

Lisa C. Kaczmarczyk
12710 Torrey Bluff Drive #176
San Diego, California 92130 USA
lisakacz@gmail.com

DOI: 10.1145/2537753.2537763