

The Addison-Wesley Signature Series



A MIKE COHN
SIGNATURE
BOOK
Mike Cohn

THE SCRUM FIELD GUIDE

AGILE ADVICE FOR YOUR FIRST YEAR
AND BEYOND

SECOND EDITION

MITCH LACEY



Forewords by Jeff Sutherland and Kenneth S. Rubin

THE SCRUM FIELD GUIDE

SECOND EDITION

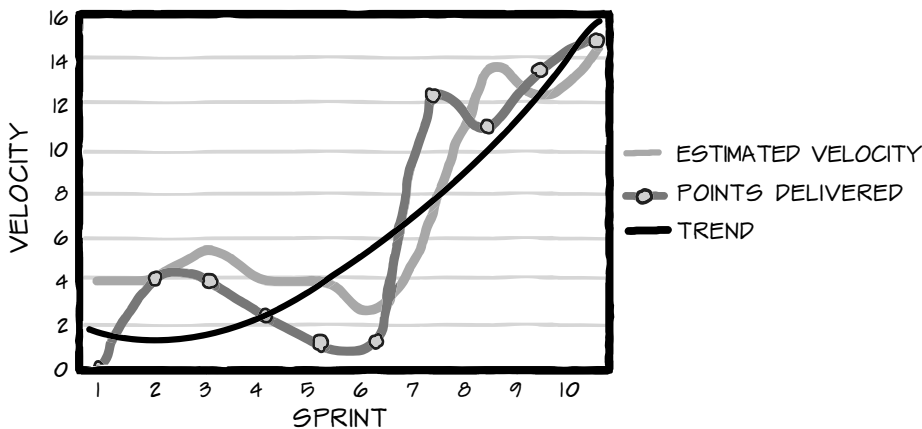


FIGURE 8-8 Actual numbers from a team that struggled

The real teams you see in these examples all had varying starting points and sustained velocities, but in each case you can see a quick rise, followed by a shallow dip (where the team was in danger of stagnating), followed again by a sharp rise and a gradual increase to high sustainable pace.

So why did these teams achieve these improvements while others sputter out? The reasons for the inability to grow are as plentiful as the number of agile teams. Some fail to implement engineering practices, others don't learn to work as a true cross-functional team, still others struggle with organizational blockers. While I cannot say for certain what caused the teams shown in these charts to achieve their results, I do know that all these teams had full-time ScrumMasters who were focused on clearing impediments and maximizing team efficiency.

Don't handicap your ScrumMasters. Staffing full-time ScrumMasters gives them the ability to realize their true role of team coach and advocate. And it allows teams the opportunity to reach their full potential as well.

Keys to Success

The job of ScrumMaster is real. It can have a big impact on costs, as illustrated previously, saving the company money—period. But what does a ScrumMaster do all day to justify a full-time role? The following list encompasses most, but not all, of the day-to-day tasks.

- Remove impediments/resolve problems
- Break up fights/quarrels
- Act as a team mom

- Report team data
- Facilitate
- Help out where needed
- Educate the organization
- Drive organizational change

In 2007, Michael James published a great checklist (see the URL in the references section [JAMES]). If the link is broken by the time you read this, search for “ScrumMaster Checklist Michael James” in your favorite search engine.

Remove Impediments/Resolve Problems

Removing impediments and dealing with problems are always at the top of the list because, especially early in a project, there are many issues to deal with, both external and internal. Many chapters in this book are about removing impediments—running a good daily scrum, having people show up on time, and so on—but there are more than just team-centric impediments. Impediments or problems found by teams in large companies could be that another team does not respond in a timely fashion, your team needs to understand how an interface works and the people who wrote it are not available, or you have someone out sick for a few days. You also have to interface with management, doing everything from convincing them that an agile process is worthwhile to reminding them of the fact that your job is full-time to paving the way for bonuses based on team, rather than individual, performance. Finally, ScrumMasters work to help the team solve its own problems and remove its own impediments. A good ScrumMaster should strive to work himself out of a job by creating a high-performing team. For some, this may be impossible to achieve, but it should always be the goal.

Break Up Fights/Act as Team Mom

People are human. Sometimes they get in weird moods, don’t get enough coffee, have a bad night’s sleep, or just see something that sets them off in a bad way. One day, my youngest daughter, Emma, who was six at the time, could not decide what shirt to wear and thought everything she picked out looked terrible—she lost it. My wife and I had to bring her back to reality.

I’ve had perfectly rational team members who come to work one day and lose it over a challenge to their ideas. Talking them off the ledge takes patience and people skills. Good ScrumMasters need to have the soft skills to intervene when necessary and the emotional intelligence to know when it’s better to stay out of it altogether.

Report Team Data

The ScrumMaster needs to help the team improve. One of the ways this is done is through reporting. A good ScrumMaster tracks the team’s historic velocities, burndown

rates, and many other project-related metrics. This is done not only to assist the ScrumMaster's own team in improving its estimation and smoothing the rate at which stories are completed during a sprint but also to help other teams (especially new ones) in the organization as well.

The important thing to remember is that *reporting team data*, either internally to the team or externally to management, does not mean you are collecting data to hold against the team. Instead, you are providing visibility into data that the team needs to see and understand to improve.

Facilitate and Help Out Where Needed

Facilitate has its roots in the French word *facile*, which is to make things easy. The job of the ScrumMaster is to make it easier for the team to achieve its sprint goal.

Many ScrumMasters wrongly assume that the best way to facilitate is to just do the work, which is not the case. Facilitation is not solving other people's problems or doing it for them: it's making things run more smoothly. The goal of facilitation should be to help the team analyze and get a better understanding of the system and the problems at hand.

The job of the ScrumMaster is not to provide answers or to do everything for the team, but rather it is to help people get to the answers that they may already know. It's like teaching a man to fish so he can sustain himself versus just giving him a fish every time he's hungry. Good ScrumMasters practice a facilitative form of coaching called *servant leadership*. Jean Tabaka, in *Collaboration Explained*, sums this up nicely.

Simply stated, the servant leader as leader drives to serve the group first. Leadership evidences itself in servant leaders through their use of power; that is, in a position of strength, they determine that the greatest power they can wield is in service to their teams as leader. For technical leads, team leads, and project managers who have used power to control versus serve teams, this paradigm shift can seem antithetical to their role. Ultimately, however, as evidenced in the agile software development context, a shift to this altered style of leadership reaps the greatest rewards. [TABAKA, p. 18]

Now, there may come a time when you need to help out to complete a task. Helping out is okay on occasion as long as you don't forget that it can hinder you in your role as ScrumMaster. As I've mentioned in previous chapters, as the ScrumMaster you need to see the forest, not just the trees. Team members are looking at the tree in front of them. If you start to help out with tasks and stories, you too will begin to see only the trees, potentially causing you to miss larger things or impediments that are negatively impacting the team.

When you do help out (and I know you will), remember that doing so must be the exception rather than the rule. While the team members might kick and scream when

you refuse to help more often, remind them (and yourself) that you are *not helping* for their own good. If they start relying on you for too much and you impact their velocity, you have just done them a disservice by artificially inflating it. The line between helping a team and hurting it is a fine one. One example of this could be that the team has overcommitted. If the ScrumMaster “helps out” by doing some work items on the sprint backlog, thereby enabling the team to meet all or most of its commitments to the product owner, the damage is done. The team will not learn *why* it ended up in this situation and will likely repeat it again. The goal here is for the team to learn, not for the ScrumMaster to bail out the team.

Educate the Organization and Drive Organizational Change

“If you want to make enemies, try to change something.” That quote, commonly attributed to former United States President Woodrow Wilson, definitely applies to these two ScrumMaster tasks: educate the organization and drive organizational change. It’s tough to convince people to try something new because change is frightening. I find that when new concepts, policies, or methodologies are being introduced inside an organization, people’s first response is to wonder how it will affect them. Will their jobs be safe? Will they be able to provide for their families? Others openly rebel, refusing to consider the new way because the status quo, as bad as it might be, is better than the unknown.

A good ScrumMaster finds a way to introduce new ideas and initiate change without shocking people’s systems. For example, a friend of mine is running for the local school board. He’s not going to just wait and show up on election day and ask people to vote for him. Instead, he’s recruiting people, like me, to help with his campaign. That campaign will introduce the voters to him and to his ideas, and eventually attempt to convince people that he is the best choice for the post.

As a ScrumMaster, you should have a strategic campaign for educating the organization. Consider your timeline for rolling it out, where you want to be and when, and whom you can recruit to help bring in your message. Once the message is sent, hopefully in multiple formats, you will be able to bring people over to your way of thinking. As a result, they will be open to how you are working and what you are building. In the end, the job of the ScrumMaster is twofold: to help build teams and to help the organization become successful. To read more about getting people, and organizations, on board with Scrum, see Chapter 2, “Getting People on Board.”

In Summary

It’s tough to justify your job, especially when it might not be a job you’ve done before. When approaching your manager about making the ScrumMaster role a full-time one, be sure to have a cool head and a good deal of ammunition.

Come armed with as many facts as you can get your hands on. Bring data showing the improvement realized by teams in your organization. If you don’t have any from

your own company, bring data showing improvements in other organizations. Relate those improvements to the cost per point.

Be prepared to remind management of the reasons teams use agile processes in the first place: lower costs, reduced risk, happier employees, more satisfied customers, and an increased ability to respond to change.

References

[JAMES] James, Michael. 2007. “ScrumMaster Checklist.” http://www.scrummasterchecklist.org/pdf/scrummaster_checklist09.pdf (pulled on 15 November 2009).

[TABAKA] Tabaka, Jean. 2006. *Collaboration Explained*. Upper Saddle River, NJ: Addison-Wesley, p. 18.

Work Consulted

Benefield, Gabrielle. 2008. “Rolling Out Agile in a Large Enterprise.” *Proceedings of the Annual Hawaii International Conference on System Sciences* (1530–1605), p. 461.

PART II

FIELD BASICS
