

Table of Contents

| | |
|---|----------|
| What Kinds of Problems Do You See When Agile Teams Churn Team Members a Lot? | 3 |
| Want to Know More? | 4 |

What Kinds of Problems Do You See When Agile Teams Churn Team Members a Lot?

A while ago I did some analysis that talked about both the [positive and negative impacts of churning Team members](#). Since that analysis I've seen that:

- The data generated there about the negative impact of velocity has been confirmed in other organizations and is surprisingly consistent. For example, last analysis showed that

Adding or subtracting 1 or 2 Team members will result in an average reduction of 24% in velocity for the next 3 Sprints

- There are a lot of organizations where “high churn” is actually the norm, but there is a limited understanding of the more soft issues problems that this creates. In these organizations churn is generated by, for example:
 - Career development strategy: As part of a corporate development program people move between assignments every few years or so.
 - Corporate initiatives: Such as the out-sourcing or work relocation strategies. This means new people taking on new (to them) work in a new context. If we build entire new Teams (best approach) it basically means you are starting from scratch with that Team and will need to train, mentor, coach, integrate (on-board) accordingly.
 - Internal organizational optimizations: For example, Teams making local decisions on the appropriate structure to deliver value. Most of this churn is hidden as there is no organizational tracking, but it can be a significant source of churn.

So what are the impacts on a people and Teams if they are subject to high churn?

- We impact our ability to deliver value. Just repeating it hear in case you missed it above. In many ways this effect is much worse than this short term downturn as we end up having ...
- No high performing Teams: All of this leads to the biggest impact of this churn; we are losing the opportunity to create high performing Teams, where Teams can produce the equivalent of 5X over new Teams.
- We reduce predictability: We should not expect our Teams to meet commitments they have made. Teams are expected to be personally committed to outcomes they planned, but new Team Members were not involved in that commitment. Teams can work this by building extra buffer into their commitments, but note that today Teams are actually over-committing on average. The result? The team won't meet their commitments unless they significantly cut corners on the expected result (usually achieved by reducing quality of deliverables - technical debt).
- We impact our products: We are potentially impacting our ability to implement and support our products. Sometimes we don't have a smooth transition plan in place to ensure that impact on the product that person is working. In the worst of cases the change just seems to happen to the Train which can represent a severe disruption to product plans.
- We change the interface for stakeholders: We are making it harder for people to interact with our Teams and Teams. For example, we often see that Project Managers have troubles trying to figure out how to input and progress work through the new Agile structure. This is not made any easier as

we churn people on these Agile structures.

- **Change fatigue:** We are probably increasing “change fatigue” for the organization. Change fatigue happens when change is imposed from on high without a clear understanding of the reason for the change (why) and what is in the change for the individual. This is why retrospectives are effective - motivation for change comes from the people doing the work themselves. Constant change leads to learned helplessness and a “just tell me what you want me to do” attitude.
- **Respect (and trust) is hard to develop.** Respect takes time to build and is fast to destroy. When a part of the Team feels as if they waste their time continually retraining other members of the Team, they stop treating the new people with respect.
- **No high-functioning named roles:** It takes time for someone to become a great Scrum Master, Product Owner, Release Train Engineer, etc. This is not a surprise. There is a significant difference between our expectation of a local amateur football coach versus what we’d expect from a pro - football coach. That expectation is set by experience. Similarly most of the learning you need to be a Scrum Master is not from a book or a training course. To be a great Scrum Master you have to coach. You have to learn from that experience. It will take a couple of years to become into a great Scrum Master, and that is only the beginning of the journey. This experience is lost, not only to the Team, but also as a key role, when that Scrum Master takes on a new role. Further, it impacts our ability to sustain and improve our Agile practice.
- **Increased demand for coaching:** We significantly increase our need for coaches. Significant churn events result in significant requests to train, coach, and mentor “new” Teams. The effect is particularly pronounced as the churn events also effect named roles (e.g. Scrum Master and Release Train Engineers). We would normally rely on the named roles to work integration of people onto Teams but this assumes that we have significantly developed the role so they are comfortable working the issue. In reality, we don’t have enough run time for the named roles to establish skills needed.

Want to Know More?

- [What Is The Effect of Changing Team Members on Velocity?](#)
- [How Do We Reduce the Effect of Team Churn on Our Ability to Deliver Value?](#)
- [The Impact of Agile Quantified](#) - A set of data looking at all types of metrics from a dataset of thousands of Teams

[FAQ](#), [team](#), [structure](#), [organization](#), [churn](#)

From: <https://www.hanssamios.com/dokuwiki/> - Hans Samios' Personal Lean-Agile Knowledge Base

Permanent link: https://www.hanssamios.com/dokuwiki/what_kinds_of_problems_do_you_see_when_agile_teams_churn_team_members_a_lot

Last update: 2021/09/16 08:38

