Self-Organized Teams and Leadership in an Agile World

> **More and more organizations around the world are adopting Agile software development practices. In these organizations, many managers are struggling to find their place in a world with self-organized teams.**

This white paper will explore how to begin building self-organizing teams. The first half of the white paper will provide some background on the characteristics of these teams and why it is desirable to create them in your organization.

The second half will discuss the different conversations managers can have with their teams in order to make the creation of self-organizing teams a bit easier.

What is a self-organized team?

The term "self-organized team" can be very loaded in organizations. When managers do not take the time to agree with their teams on a meaning, the danger is that these teams will create their own definition for it.

To set a baseline for discussion in this white paper, here is high-level interpretations of a self-organizing team:

- A team that has a certain level of decision-making authority. This level may change and evolve over time but there is a clear sandbox defined where teams can make decisions.

- A team that is working toward meeting its emerging vision. Without a clear team vision, self-organization can sometimes lose meaning because the team members forget why they are doing what they do.

- A team that takes ownership of how it works and that continuously evolves through having a continuous improvement mindset. In other words, where it is today is not where it will be six months from now.

Why do we want self-organized teams?

To better understand why we want self-organized teams, it can be useful to first take a look at the meaning of the term "management." According to Peter Drucker, management is about human beings and its task is to make people capable of joint performance.

When we describe what a manager actually does, we may think of tasks such as:

- Assigning and planning team workload;
- Delivering project estimates;
- Hiring new employees;
- Doing conflict resolution;
- Managing a budget;
- Doing performance reviews;
- Solving problems and removing roadblocks;
- Managing team delivery capacity;
- Identifying and setting objectives;
- Making decisions.

Assuming this is about eighty percent of a manager's tasks, which items in the list above you would NOT want a self-organized team to do?

The truth of the matter is that management is just way too important to the success of the enterprise to leave completely in the hands of the managers. What we really want is for managers to share these tasks with their self-organized teams.

The other reason we want self-organizing teams is to increase the motivation and engagement of employees by delegating authority and empowering them to take initiatives and make decisions.

Understanding different stages of group development

When building self-organized teams, there are multiple models that can be used as references. For managers, it can be useful to pick one, get very comfortable with it and try to measure where their teams are currently at.

When using models such as the one in this section, it is important to understand that all models are imperfect

and interpreted based on the point of view of the person using it.

One useful model, shown in table 1 below, is Tuckman's Stages of Group Development[1]:

1 https://en.wikipedia.org/wiki/Tuckman%27s_stages_of_group_development

When teams are evolving through Tuckman's stages of group development, managers should try to understand where their teams are currently at and adapt their behaviours accordingly. Another important consideration is that teams may fall back one or many stages when changing the team members.

Table 1 – Tuckman's stages of group development

Stage name	Key Characteristics	Leadership style
Forming	• Team members are on their best behaviour. • No one expresses their opinion. • Team members are trying to understand what needs to be done.	• Managers and leaders may need to impose their ideas at this stage because no one on the team is speaking up. • Managers and leaders should encourage people to speak their minds.
Storming	• Team members start expressing their opinions and preferences on how to carry out the work. • Clashes of opinions and personalities are occurring between team members. • It's necessary to resolve the team's conflicts in order to exit this stage. • Teams can stay caught in this stage for a very long time.	• Managers and leaders may need to sell their ideas instead of imposing them as the team is already in conflict. • Managers and leaders should support team members in seeing the perspectives of others. • Managers and leaders should support and help their teams to align toward a common goal.
Norming	• Team members have a common goal they want to work towards together. • Team members tolerate and accept one another. • Team members are learning to work together.	• Managers and leaders should begin to encourage and engage their teams in decision-making. • Managers and leaders should support team members in seeing the perspectives of others. • Managers and leaders should encourage collaboration on the team.
Performing	• Team members are competent and autonomous in their work. • Team members can make decisions without supervision. • Team dissent and conflict are handled in a positive way.	• Managers and leaders should act in ways that encourage the teams to maintain and grow their autonomy. • Managers and leaders should not hinder the decision-making capability of the team.

> Many managers fall in the trap of believing that because the team is "self-organized", they no longer have a say in anything at all.

Initiating self-organization

When beginning to work with a new self-organized team, it is important for managers to spend some time with the team to set the stage and the expectations.

The first conversation should revolve around the meaning of self-organization and the expectations of the manager. It is also important to discuss the areas where team members are expected to make decisions alone as well as areas where they still need to involve their manager, and how they expect to be involved.

Many managers fall in the trap of believing that because the team is "self-organized", they no longer have a say in anything at all. There are two key things to remember here:

1 The team may not have the knowledge, nor the desire, nor the ability to make certain decisions.

2 Based on their history in the organization, team members may not actually believe they are truly empowered to make decisions or may fear blame if they make a bad decision.

Self-organization does not mean the team can go off and do whatever it wants. The role of the manager is to ensure that the team has a clear purpose and objectives in order to better understand why members are together as a team.

The next thing is to define the decision-making sandbox of the team and from there, the manager has a nice base to begin guiding it on their journey towards self-organization.

Creating an emergent purpose

In many organizations, teams are put together for an intrinsic purpose such as producing documents, code or other types of deliverables. They also have an extrinsic purpose that is assigned by the organization they work for, such as generating revenue from their deliverables for example.

What is missing for many teams is an emergent purpose, a clear reason to be together, collaborate and work towards something. When there is no such purpose, whatever change the leaders are trying to bring may not stick. Team members will eventually begin to wonder what the change is about and why they are working on it.

The second step to building a self-organizing team is to facilitate conversations to help your team create an emergent purpose together. The keys to a good emergent purpose for a team are:

• It needs to align with the current overall goal of the organization.

• It needs to come from the team members and be something that inspires them.

• It needs to be a stretch for the team, meaning that it is something they can achieve in a six to nine-month timeframe but not something impossible for them to do.

Once the team has defined its purpose, it needs to find a way to make it visible in their team area and use it as their North Star to guide their decisions. The role of the manager in this context is initially to regularly remind its team of its purpose and support it in working towards it.

There are some good tools to guide these discussions on the LeanChange.org[2] page. You will find a Strategy Canvas for organizations and a Team Change Canvas for teams. The value in tools such as these is in how they allow managers to structure conversations with their teams. The idea is not to reach a final version of any of these tools but rather the alignment that is created with everyone that is part of the conversation.

Identifying the sandbox for self-organization

To build a high-performing self-organized team, it is important for managers to learn to invest in delegating some of their decision-making authority to their teams. Imagine for a moment a team member that needs to go up the hierarchy chain in order to get approval to do something. Is this approval really necessary? What is the impact of getting this approval on the team?

For self-organization to succeed, its sandbox must be clear. Good practice is to start with an initial smaller sandbox in order to help the team learn and build capacity. A few weeks or months later, as the team is becoming better at self-organizing, managers can slowly expand the

2 http://leanchange.org/2014/02/how-to-create-alignment-for-agile-transformation-with-canvases/

sandbox to include more accountability.

Delegation boards[3] and Delegation Poker[4] are two of the key tools introduced in the book Management 3.0[5]. They enable teams and their managers to have a conversation to start identifying the sandbox in which the team can self-organize.

Delegation poker is a card game that creates conversations around the seven levels of delegation introduced in the Management 3.0 course[6]. The seven levels are:

TELL: The manager imposes a decision to the team.

SELL: The manager makes a decision and sells it to the team.

CONSULT: The manager makes the decision but has conversations with the team to get its input.

AGREE: The manager and the team make the decision together.

ADVISE: The team makes the decision but the manager is part of the conversation and can provide input.

INFORM: The team makes the decision and informs the manager after making it.

DELEGATE: The team makes the decision and life goes on.

Using the seven levels of delegation allows managers to delegate some of their personal decision-making authority to the team. There are some decisions in organizations that only a manager is allowed to make but he can decide how much (or how little) he wants to involve the team. Using this tool makes the boundaries clearer for both the manager and the team members.

Fostering continuous improvement

Another challenge for leaders is in allowing their teams to experiment with things and help them see and learn lessons from these experiences instead of getting paralyzed looking for the perfect solution.

Organizations run at a fast pace and management teams are often looking for quick results. For some, just the idea of doing multiple experiments to incrementally learn can seem like a waste of time. When leaders are unable to see the value in experimentation, how can they truly promote this to their teams?

> Teams need to learn that they do not always need to find the perfect solution to a problem.

Viewing new experiences and ways of doing things as experiments can be a good way of creating permission for people to try them. The important thing is to frame the experiments properly.

Table 2 below demonstrates how it is possible to frame an experiment, keep track of it and the lessons learned in the process.

Experiments can come from different places such as retrospectives or team meetings. Teams should clearly define a new experiment in a few sentences and be clear on what they are trying to learn before even starting it. Clarifying these two points is the key to experimenting in the spirit of continuous improvement.

Table 2 – How to frame an experiment

Experimentation Stages	Key Questions
Before starting	1. **What is the experiment?** 2. **What are we trying to learn?** 3. **How long is the experiment?**
Experiment in-progress or ended	1. **What happened?** 2. **What did we learn?** 3. **What is next?**

3 https://management30.com/practice/delegation-board/
4 https://management30.com/product/delegation-poker/
5 https://management30.com/product/management30/
6 http://pyxis-tech.com/fr/formations-agiles/management-30/

Another way to foster self-organization is to encourage smaller experiments. Teams need to learn that they do not always need to find the perfect solution to a problem. They can try something, see what they can learn from it and then do another experiment to get them closer to their end goal. This means that some experiments will be successful and others will fail but the important thing is to focus on what the team is trying to learn and what it is actually learning from the experiments.

Servant Leadership, Collaboration and Conversations

One of the most misunderstood concepts for managers of Agile teams is servant leadership. There are many different beliefs and interpretations of the term. Some managers believe they no longer provide any direction at all for the team, others believe it means their teams need to decide everything and they have no say in anything anymore.

Here are some examples of servant leadership in action:

- Leaders help and teach their teams to solve problems together instead of solving these problems for them.

- Leaders delegate decision-making authority to their teams.

- Leaders encourage and support their teams to become autonomous, make decisions together and self-organize.

- Leaders are consistent between what they say and how they act with their teams.

- Leaders collaborate with their teams by being a voice among others in the conversation.

In many organizational cultures, many of the above examples are difficult for managers to do. In these cultures, the leaders often need to be strong and have all the answers for their teams or there is a constant pressure for immediate results which goes against many of the behaviours associated with servant leadership.

Leaders of self-organize teams are comfortable creating conversations that help their teams move forward even when these discussions may initially be uncomfortable. They create a secure environment that allows team members to learn and grow.

Conclusion

In a world with self-organized teams, managers need to change their leadership styles in order to share management tasks with their employees and become more effective servant leaders. Teams will make their own meaning of self-organized team when managers do not take the time to talk with them and reach a common understanding of the term.

Learning about different group development models can provide managers and leaders with reference points on where their teams are currently at and what they can do to help them reach the next level of collaboration.

Delegation poker and delegation boards are key tools to help define a clear self-organization sandbox for teams. It is important to start with a small sandbox to let the teams, as well as their managers, experiment and learn about delegating authority. Once a team develops the skills and capacity to handle the current sandbox, it can work with its manager to expand the sandbox further.

Fostering continuous improvement and teaching teams how to experiment in a structured (but not rigid) way will help build a culture of learning by doing instead of continuously looking for a perfect solution before trying anything.

One of the key roles of a manager in an environment with self-organizing teams is to be a servant leader through the facilitation of conversations and through coaching his teams to become autonomous, make decisions together and be more self-organized.

Who are we?
We are developers, coaches, trainers, architects, software engineers, Scrum Masters, Product Owners, designers and much more... We share one passion: that is to transfer our know-how and deliver remarkable organizational solutions.

pyxis-tech.com