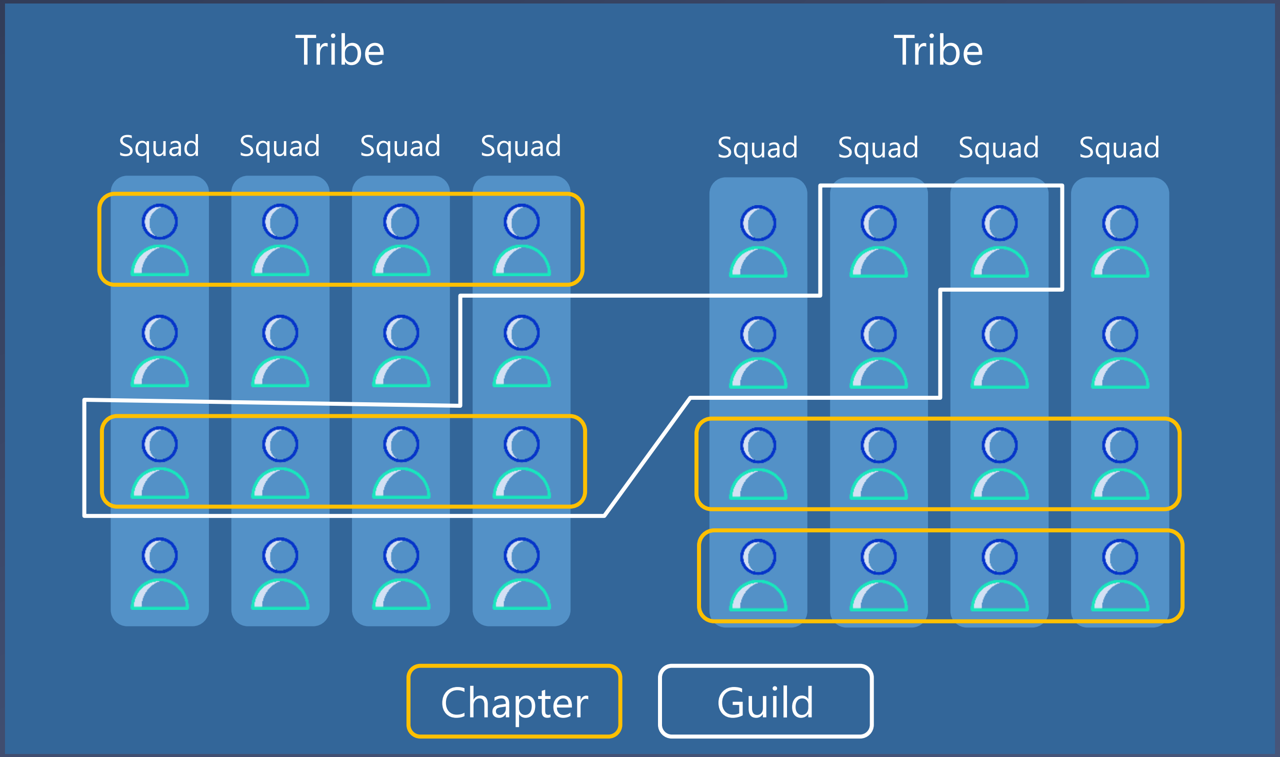
The Spotify Model



V1.0

Source: PULZ, 2022

The Spotify Model

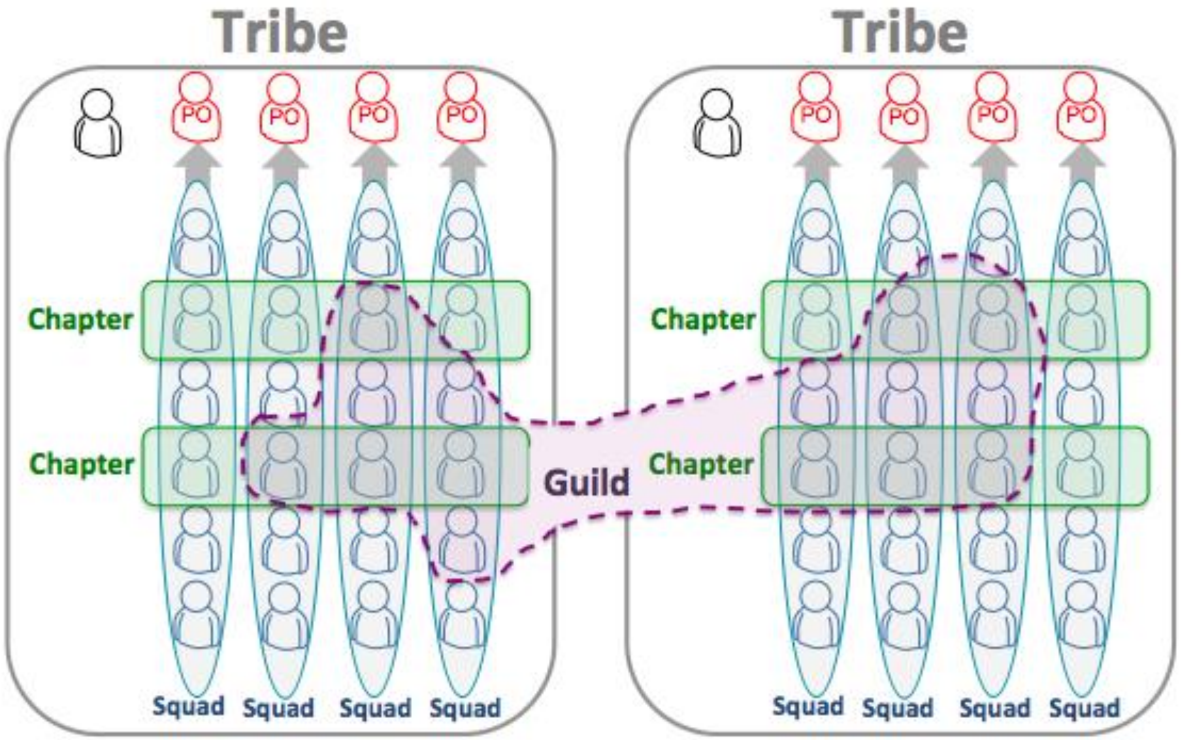
Spotify was growing very fast and that meant **multiple (Scrum) teams had to work together to create one product.** This created a lot of complexity and challenges because Scrum was designed for individual teams, with the result that delivery of releases was less frequent and less good. And that compromised user value.

This situation of rapid growth in the number of development teams led to a new way of working, which was later named **the Spotify Model** (which you may thus think of as one of the many Scaled Agile Frameworks that would later emerge).

The Spotify Model promotes a very **high degree of autonomy of the teams,** in the sense that one is even free to choose how to work! Each team (or Squad) chooses its framework (e.g., Scrum, Kanban, Scrumban, and so on). Squads are organized into Tribes and Guilds to keep people aligned and to spread knowledge among peers across teams.

The Spotify Model dates back to 2011 when **Henrik Kniberg & Anders Ivarsson** published a white paper on behalf of their colleagues: Scaling Agile @ Spotify With Tribes, Squads, Chapters, And Guilds.Kniberg is often cited as the "inventor" of the model, but he's the first to tell you that it's not. And he's also happy to tell you that **it was never intended that people would copy it** or implement it in their own businesses. **"It's just an example of how one company works,"** he says.

Spotify Model: structure



A **Squad** is similar to a Scrum team. A Squad is a **self-managing** and **multidisciplinary** team that has all the knowledge and competences to deliver a product or service from design to final product.

A **Tribe** is a **group of Squads working on** the same or related products or services.

A **Chapter** consists of employees with **the same expertise within a Tribe.**

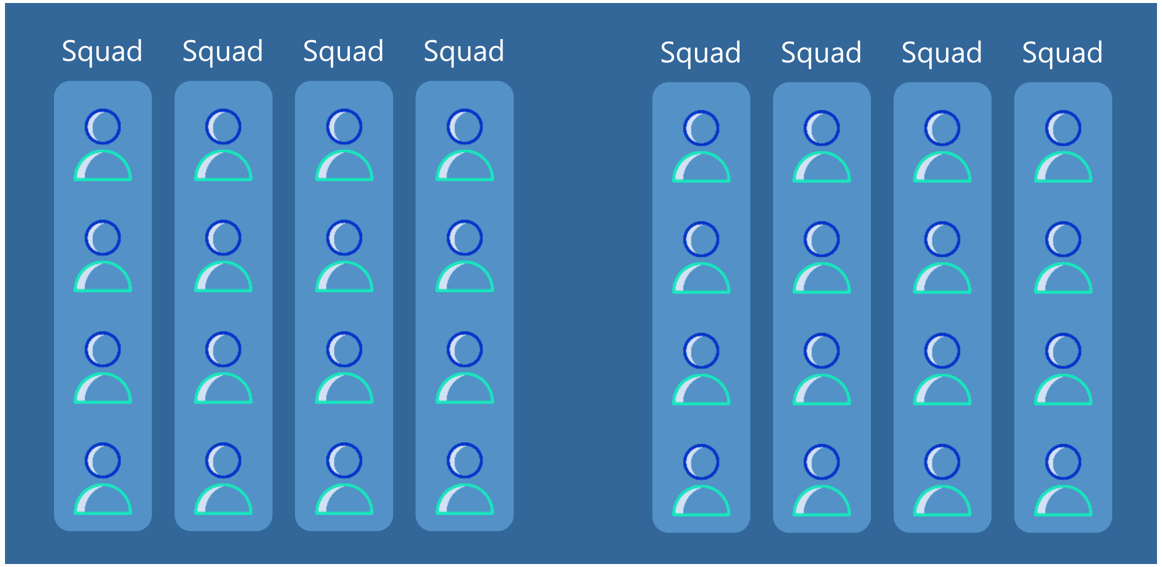
A **Guild** is a group of employees with **the same expertise and/or interests** that are not limited to the Tribe but scattered throughout the **organisation.**

Source: www.agilescrumgroup.nl, 2023

Structure: Squads

A Squad can be compared to a Scrum team, a group of 6-10 people who combine different areas of expertise to create an (interim) product. The team is **self-organizing** (and in some situations self-steering) and focuses on **one functional area.** Thus, there is a well-defined domain, but collaboration with other squads is inevitable. Team members are empowered to decide for themselves how to design and create products.

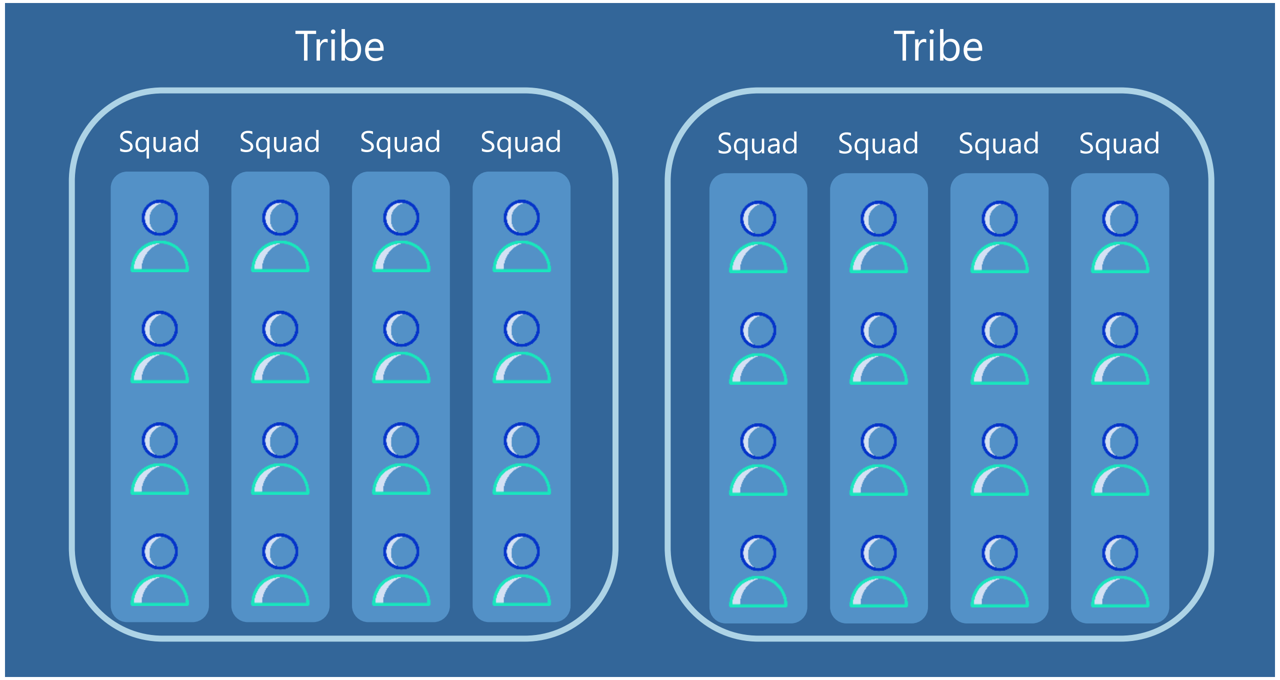
A Squad has a mission that is part of the product strategy. Each Squad has a product owner and an Agile coach.



Source: PULZ, 2022

Structure: Tribes

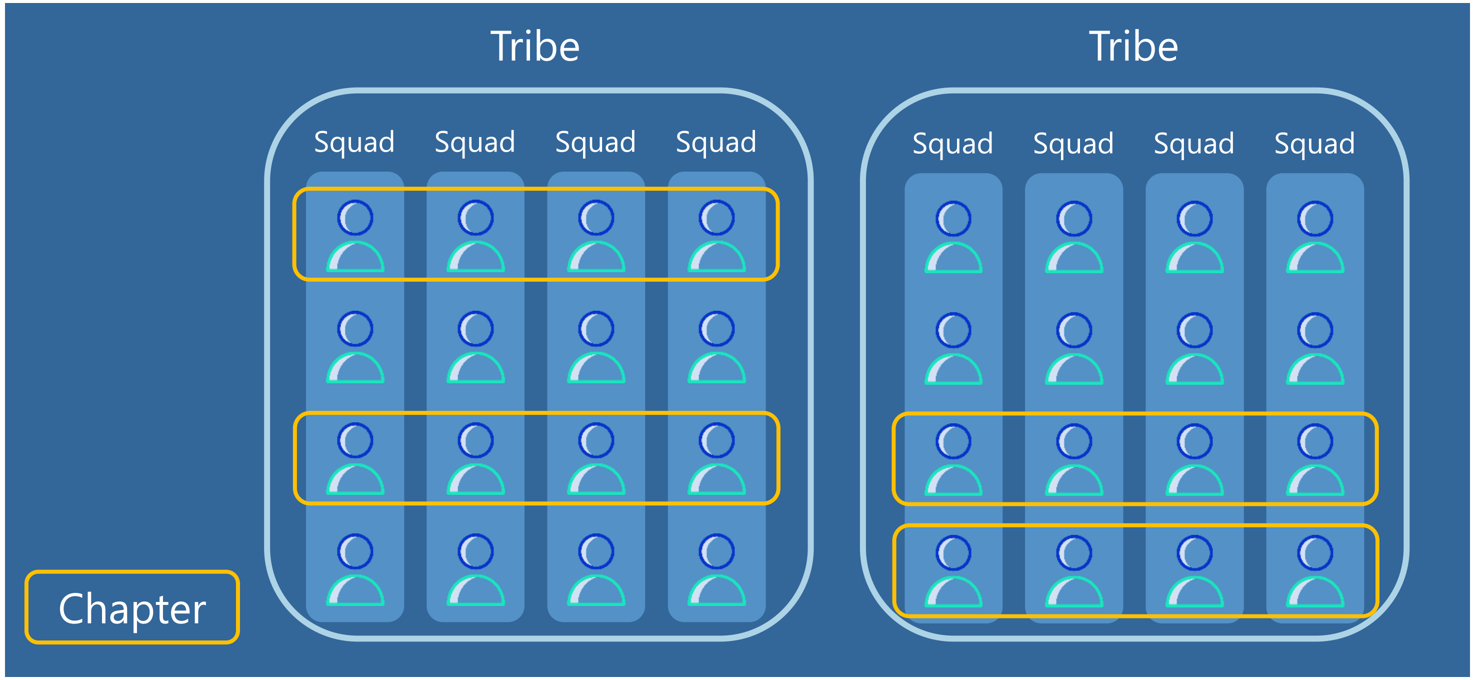
When several Squads work together in the same functional area, they form a Tribe. In theory, the size is up to 100 people, but larger is possible as long as the team members have the "single Tribe feeling". Tribes help achieve **alignment between Squads** and have a **Tribe Lead** who is responsible for helping coordinate between the Squads and encouraging mutual cooperation. A distinct role because **autonomy is the norm.** The Tribe Lead will have to act as a sort of respected tribal elder addressing the seniors of the Tribe.



Source: PULZ, 2022

Structure: Chapters

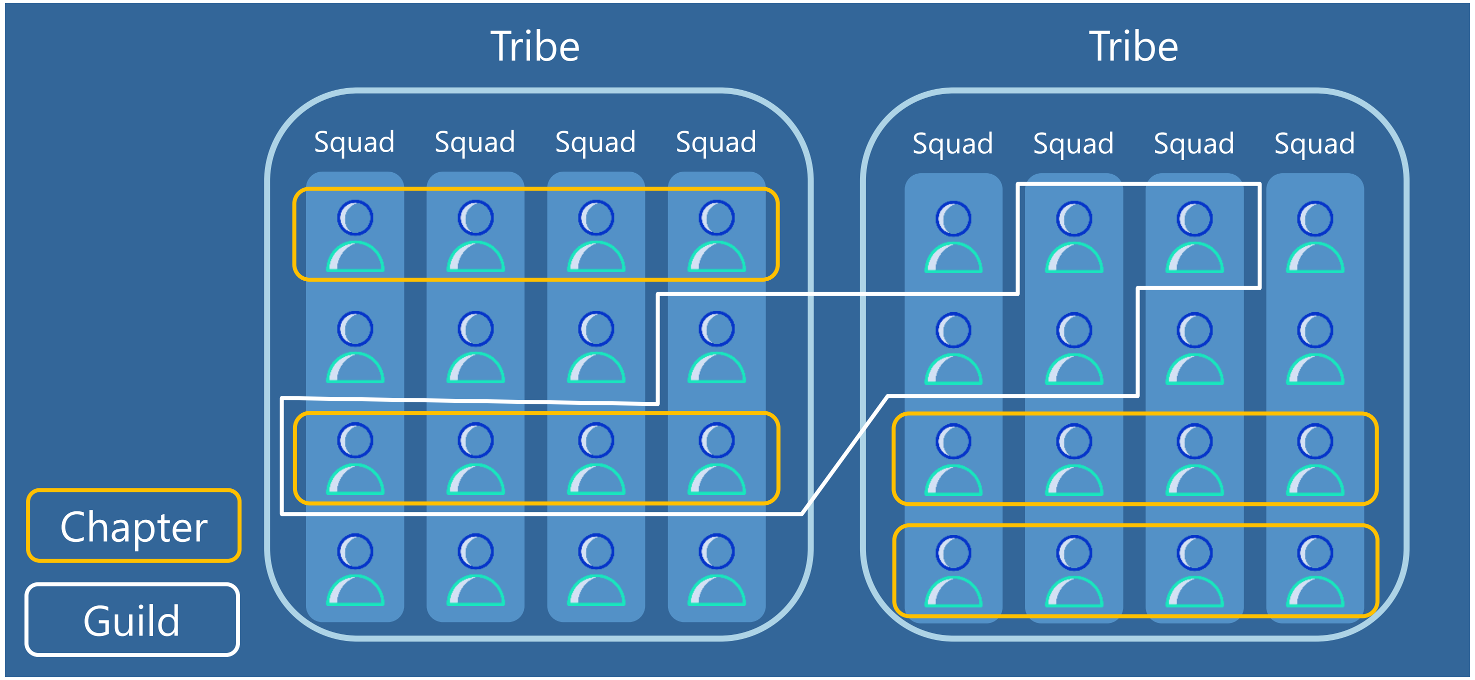
It is important that specialists in the Squads have alignment on the method to be followed and best practices used. That happens in **Chapters, a sort of competency group** that is there for each specialty and helps to use and **secure engineering standards in a discipline.** Chapters are usually led by a **senior technology lead,** who may also be the manager for the team members in that Chapter.



Source: PULZ, 2022

Structure: Guilds

A Guild is a kind of community that you can join **if you share a particular interest with colleagues.** While Chapters are always part of a particular Tribe, Guilds can include individuals from different Tribes. Anyone can join a Guild; membership and contribution is voluntary. **A Guild also has no formal leader,** but rather a volunteering coordinator who takes charge of the organization.

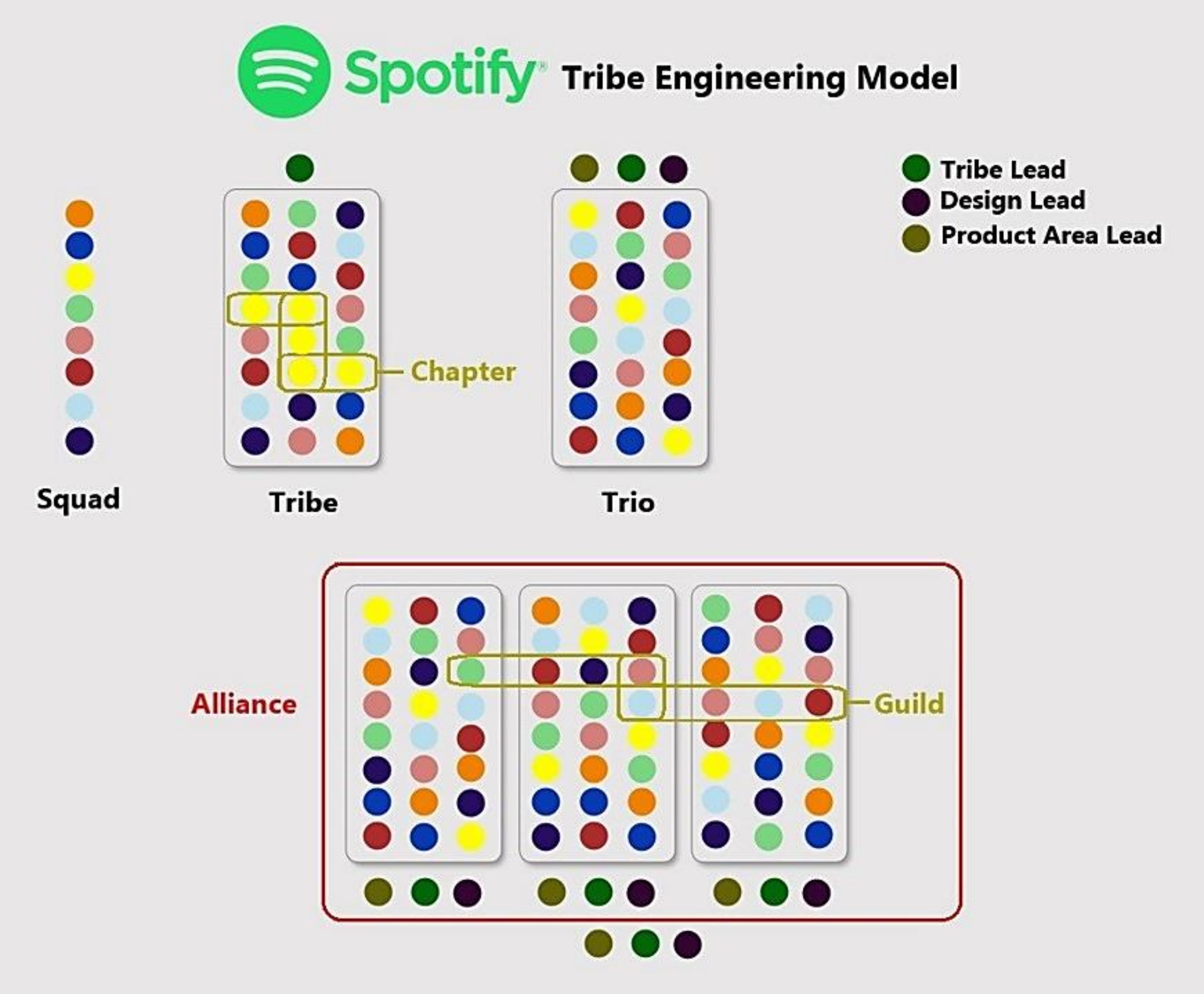


Source: PULZ, 2022

Structure: TPD Trio and Alliances

Each Tribe has a (TPD) **Trio** to ensure that there is **continuous alignment between the three axes Tribe, Product and Design** when working on functional areas. The TPD Trio consists of the Tribe lead, Product lead and Design lead.

And, of course, **different Tribes often need to work together to deliver a single system or goal.** This is done in an **Alliance,** a combination of Tribes that serves to ensure synchronization.



Benefits of the Spotify Model

- **Less formal processes** and ceremonial departmental/business meetings.
- **More autonomy,** less top-down management.
- There is **alignment** (of practices, collaboration between squads and tribes) despite autonomy.
- There is **governance based on trust** rather than control.
- The system architecture of the Spotify product is designed in such a way that there are **minimized interfaces** between functions and releases (something that in itself is unrelated to the Spotify Model and cannot simply be adopted by other organizations).
- Productivity is promoted primarily by **management through motivation** and **allowing employees to excel in their areas of expertise:**

Productivity = Effort x Competence x Environment x Motivation

- Specific **(coaching) resources** are made available to help teams grow in the way they work.
- Emphasis is put on **employee satisfaction** and putting the value of their work **before business interests.**
- **Making mistakes is accepted** because it is seen as a necessary element of **Innovation.**

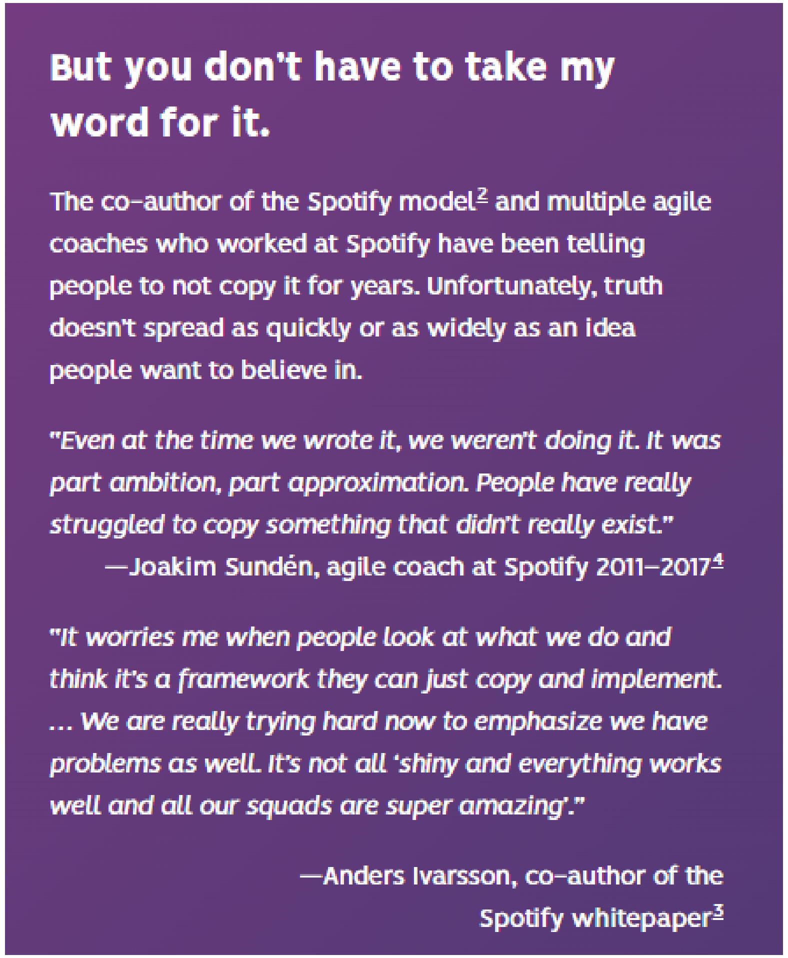
Source: Atlassian, 2022

Does the Spotify model have any drawbacks?

> When I attended the 2017 Software-Centric Systems Conference in Eindhoven in regard to giving a talk, I listened with interest to a Spotify keynote. In it, they told with great enthusiasm about the Spotify Model and how it was applied. *Autonomy had no limits in that story, because people were not to be constrained in an innovative organization.*

Two things surprised me. First, I thought "as nice as it may sound, actually all the challenges and balancing acts of a traditional Matrix organization in terms of alignment between teams, competencies and product standardization, just remain in the Spotify Model.” Besides, so much focus on autonomy will require above-average leadership from team members. All those self-chosen ways of working and use of coaches will also make the organization rather expensive. Finally, I was shocked by the surely naive impulse of people in the audience. "That's really something for us, we should implement that model too" sounded from many voices.

There are wonderful (and logical) best practices in the Spotify Model. However, to blindly copy is actually planning to fail, something the founding fathers have been trying to make clear for years (often against the force of hype)…



But you don't have to take my word for it.

The co-author of the Spotify model[2] and multiple agile coaches who worked at Spotify have been telling people to not copy it for years. Unfortunately, truth doesn't spread as quickly or as widely as an idea people want to believe in.

"Even at the time we wrote it, we weren't doing it. It was part ambition, part approximation. People have really struggled to copy something that didn't really exist."

—Joakim Sundén, agile coach at Spotify 2011–2017[4]

"It worries me when people look at what we do and think it's a framework they can just copy and implement. … We are really trying hard now to emphasize we have problems as well. It's not all 'shiny and everything works well and all our squads are super amazing'."

—Anders Ivarsson, co-author of the Spotify whitepaper[3]

Source: PULZ, 2022

Does the Spotify model have any drawbacks?

Brad Swanson, in a 2020 blog at www.agility11.com, lists the following reasons why Spotify itself has also had to change the Spotify Model:

- **Re-labeling the Matrix:** a Squad was simply a cross-functional team, a Chapter a functional group (testing, programming, designing) and a Tribe a classic department.
  In other words, *old wine in new bottles according to Brad Swanson.*
- **Matrix:** each Squad had to deal with multiple functional managers (one per Chapter). This led to the necessary conflicts that regularly had to be resolved by the Tribe Lead.
  So pretty classic and conflicting with the focus on autonomy.
- **Team autonomy:** Squads had maximum autonomy. That works nicely for small start-ups, but not anymore for scale-ups.
  I personally think that, without any guidance, this leads to many "cauliflower" effects in designs and too little standardization with all its consequences (many resources needed, high risk and costs).
- **Agile competence:** giving team members autonomy over the organization of their own work is one thing, but also over the work processes and approach requires a lot from them. *My experience is that this requires a lot of guidance, which is often underestimated. Even the application of standard Scrum is already demanding for many organizations…*

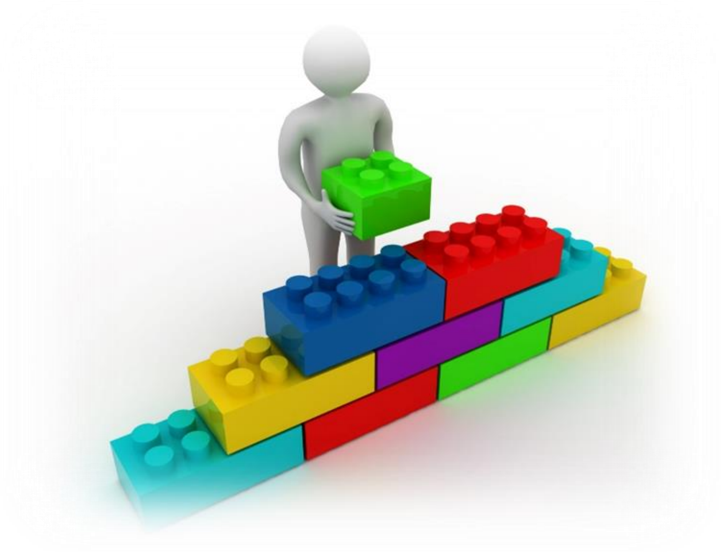
> I've spent a few years working with Spotify and published a few things that have gained a surprizing amount of attention – especially the scaling agile article and spotify engineering culture video. This has come to be known as the "Spotify Model" in the agile world, although it wasn't actually intended to be a generic framework or "model" at all. it's just an example of how one company works. The reason why I shared this material is because my Spotify colleagues encouraged me to, and because, well, that's what I do – help companies improve, by learning stuff and spreading knowledge.
>
> Since then, many companies have been "copying" the Spotify model, which I found rather scary at first. But now I've met quite a number of those companies (and heard back from even more), and so far I have yet to see a case where a company ended up in a worse position than where they were. In fact, some have seen some huge improvements! Pretty surprising actually. So apparently, yes, companies can copy models from each other, and it can sometimes be helpful – mainly because it's almost always valuable to look at your own organization and process with a critical eye, and take inspiration from others. As long as you adapt to your local context (which most do, eventually).
>
> Henrik Kniberg, 2015

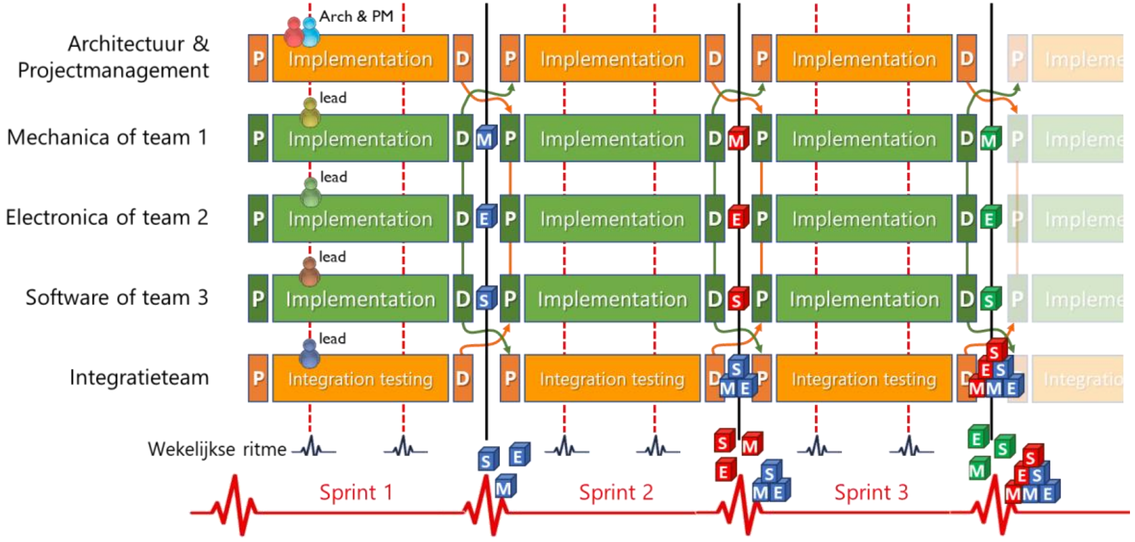
Source: PULZ, 2022

Conclusion

So it is obvious and actually quite understandable that **you cannot copy the Spotify Model 1:1 without modifications.** First, because **it is not meant to be a generically applicable model.** At least as important, the way an organization organizes itself must align with its goals and should grow with the maturity level of those involved.

It is therefore wise to make **incremental improvements,** learn from them and then take the next step. The **building blocks** of the Spotify Model can certainly help with this, as can the **essences** we cover in The Complete Project Manager and in this Master Class with in-depth blocks. In this sense, the multidisciplinary heartbeat as covered in Module 1 is also a composition of basic blocks.





Source: PULZ, 2022

PULZ

Project Management &
Leadership Development

This publication was produced with the utmost care and attention. Nevertheless, the text may contain errors. PULZ Project Management & Leadership Development B.V. and the author are not liable for any errors and/or inaccuracies in this text.

www.roelwessels.nl