

DEVELOPMENT LEADERS

NORTH WEST



OLLIE WILDMAN

CONTRACT TEAM LEAD

ollie.wildman@maxwellbond.co.uk



maxwellbond

Introduction

Exploring Product Onboarding, Team Topologies and Scrum Teams

Following an insightful roundtable at the end of 2020, with local Development Leaders, we have pulled together the key facts, pieces of advice, and solutions for Development Leaders to utilise as they go forwards into 2021. Our attending leaders explored issues around how to effectively onboard new developers in complex business structures and multi-project companies, the effectiveness of team topologies, and scrum teams without product owners.

This is the first in a series of Development Leaders: North West roundtables, so if you're interested in attending future sessions, please contact Ollie Wildman.



Contents

Page 2: Product Onboarding for Devs

Page 3: Team Topologies

Page 4: Scrum Team Variations

Page 5: Contact Details

Product Onboarding

Product Onboarding for Developers

Product functions can vary from business to business in terms of complexity. For example, some businesses have more than one core product, as well as lots of other auxiliary products on the side. The scope and scale of each product, and the technology used to support it, can be significantly different. This introduces quite a lot of complexity into maintaining those products, but also onboarding new developers because there are so many layers and ideas. The challenge is to successfully onboard developers so that they have a full understanding of all products, and how they all work differently, within the same product function.

Some businesses have found success implementing the following strategies to ensure a thorough and informative onboarding process:

1. Using Azure DevOps to build a knowledge bank and updating, creating, and merging documents into a dynamic “Wiki” which can be searched more easily. Essentially businesses should document everything and make sure that this documentation is accessible to everyone. Documents should be standardised and up to date.
2. Put the links to documents and processes in the comments next to the code. So, if someone picks up a feature and needs some information about it, they can easily find the documentation linked to that.
3. Create videos demonstrating the full product lifecycle and then build it into the induction process and program. Make these videos a mandatory part of the onboarding process. This should give new starters a fuller idea of functionality and the what the product does.
4. Use surveys to ask engineers questions about what methods of communication, information and tooling they prefer. Use this feedback to refine the process and ensure information is conveyed in the best ways.

Each product should have its own set of documentation that covers the following strands:

- Customers: who the customers are, what their role is, and what problem they have that your product solves
- Product: introduce the high-level product and how to interact with it
- Code: introduce the high-level code, core repositories, and core API layer

This helps to build a living knowledge base around the product, audience, and support escalation.

Team Topologies

Are they effective?

Studies suggest there are four main team topologies:

- Stream-aligned team: aligned to a flow of work from (usually) a segment of the business domain
- Enabling team: helps a Stream-aligned team to overcome obstacles. Also detects missing capabilities.
- Complicated Subsystem team: where significant mathematics/calculation/technical expertise is needed.
- Platform team: a grouping of other team types that provide a compelling internal product to accelerate delivery by Stream-aligned teams

Whilst in theory, team topographies seem like they could help concisely define team dynamics and purposes, there are issues related to implementing this. This is especially difficult when you are rapidly growing teams. During rapid growth it becomes hard to know and define who is dependent on what team and in what way. So, although topologies allow complimentary collaboration, it can be difficult to maintain in an environment that requires flexibility, agility, and speed.

When you have more than one stream, it can also become more complex and issues around competing requirements and priority conflicts arise. This leaves us with the question: "Are team topologies fully executable in teams?"



Scrum Team Variations

Can you have Scrum without Product Owners?

Some companies have vigorous testing regimes which means that testers work slower than developers, resulting in potentially valuable pieces of software being stuck on the shelf for months. Then, by the time these were tested, the business had moved on, requirements had changed, and therefore the product was now irrelevant. Companies often move towards an Agile way of working to mitigate this issue and to streamline and clarify their working processes. However, some businesses are reluctant to fully invest in this, meaning that scrum teams are sometimes built without enough Product Owners to oversee all the teams.

This leads to issues around prioritising updates and requests that have been logged, and this decision is then left to the developers themselves or delivery. It often leads to internal priority conflict which naturally slows processes down and creates a process that is not streamlined, defined, or effective.

In companies where resources are less abundant, it becomes fundamental to define roles and responsibilities really early on. This way, everybody knows processes, escalation points, and decision-making points. It also encourages people to take full ownership and responsibility of their own tasks. Providing different people with set responsibilities and areas of ownership, and also enabling developers to be a part of the decision-making process, allows them to gain a fuller understanding of why decisions are made, why certain features are prioritised, and they then become more invested in the company.

Once developers are invested and fully understand the business goals and align themselves with these, there should be less priority conflicts, as they should be able to determine which requests really are priority. In some teams, they also delegate prioritising the backlog to one senior person, who in this respect, acts as a Product Owner. That said, the most successful scrum teams and delivery functions tend to operate best when Product Owners are present, as structurally this works most effectively. Therefore, companies that invest in Product Owners have an advantage over those that do not.



Contact Us

Build high performing teams through exceptional contract and perm staffing solutions and business advice with Maxwell Bond, the recruitment partner of choice across the UK and Germany.



Ollie Wildman
Contract Team Leader
ollie.wildman@maxwellbond.co.uk
0161 359 3280

Find out more at www.maxwellbond.co.uk