



Episode 10: Team Topologies

Episode Transcript

Mik Kersten:

Hello, and welcome to the Mik + One podcast, where I sit down with industry leaders to discuss the project to Product movement, I'm Mik Kersten, founder and CEO of Tasktop, and bestselling author of Project to Product: How to Survive and Thrive in the Age of Digital Disruption with the Flow Framework®.

Mik Kersten:

Today, I'm joined by two special guests, Matthew Skelton and Manuel Pais, Authors of Team Topologies: Organizing Business and Technology Teams for Fast Flow. Matthew is Head of Consulting at Conflux and Manuel is an Independent IT and Organizational Design Consultant and trainer focused on team attractions, continuous delivery practices, and accelerating the flow of work.

Mik Kersten:

It was great catching up with them to record this episode where we focus on their new book in depth, which discusses the different team structures we can use to support product innovation and value stream flow. Note that this podcast was recorded at the London Seacon Conference at the end of last year, so you may also enjoy the background noise that will take you back to being at a physical conference. Enjoy.

Mik Kersten:

Hello, everyone. I'm here at the Seacon Conference in London. I've had the opportunity to bump into Manuel Pais and Matthew Skelton, and I would just love to hear from the two of you about what's been happening on Team Topologies. So great to be chatting.

Matthew:

It's great to be here. Thanks.

Mik Kersten:

And the book was ... there was a signing at the DevOps Enterprise Summit last week. There was a ton of interest in the book. I think a lot of organizations are out there and looking at trying to figure out how to really structure beyond just understanding what an agile team is, what a feature team is.

Mik Kersten:

There's bits and pieces of people trying to cookie cutter the Spotify model. And I think your book is providing us a new perspective and just more thorough and deeper thinking around this. So, if you could just give the listeners a summary of Team Topologies, and the core ideas, and really why you want them to get into the book.

Matthew:

Sure.

Manuel Pais:

Perhaps just to add, the initial problem around reorganizing. And like you said, companies trying to be Agile or be DevOps. And they're reorganizing massive transformation projects and they don't always see the benefits. So, one of the key ideas of Team Topologies book is to give the tools and the thinking around how do we become more of an organism, if you like, that is able to adapt to challenges. Rather than let's take the next big thing and do this massive transformation. Because it always takes a toll on people and motivation.

Matthew:



Episode 10: Team Topologies

Episode Transcription

Exactly. The book ends with a chapter that really talks about the evolution of the organization. So, the way in which can begin to listen to signals inside the organization, and outside the organization, and adapt itself. Adapt how teams interrelate. And how, if you like, teams are structured in order to meet new business challenges.

Matthew:

So Team Topologies is very much about setting things up for learning, for adaptation, with an emphasis on a flow of change. And we go through lots of different principles, and patterns, and things that lay the groundwork, if you like, for enabling an organization to much better listen to signals around it and inside it so that it can adapt.

Mik Kersten:

So, what have you seen as maybe the patterns and the anti-patterns of how teams are structured, how teams evolve, how restructurings are driven? Usually top down, I think. Just take me through some of what you've learned, some of the key concepts in the book.

Manuel Pais:

So, in terms of teams, what ends up happening is you see many organizations, we look at teams as if they're all the same, especially within engineering. And that's not really true if you want to have a real focus on what is the value stream that they're trying to address and bring value to. So, some of the patterns we talk about in the book is thinking about the types of teams because you have teams doing very different types of work.

Manuel Pais:

So we talk about streamline teams, which are your product team with end to end ownership of the life cycle. Because we know if we have teams that can build and operate their services, that's the fastest way to get feedback back into the product and really improve as quickly as possible. But then you need realistically other types of teams to help with that.

Manuel Pais:

Because what we talk about in terms of the team cognitive load, how much knowledge, skills, and tasks that they would need to do for this end to end ownership is too much for one team to grasp. So you need supporting types of teams. And that's where we talk about platform teams, enabling teams, sometimes complicated subsystem teams.

Matthew:

The starting point of the team and the cognitive load is a very different starting point to how most organizations think about this stuff. So some organizations are now doing this. There are some examples out there. For example, Twilio, the cloud telecoms provider, explicitly uses cognitive load as a way to assess different sizes of bits of its software system. The number of these organizations is quite rare. It's quite rare to find that at the moment.

Matthew:

The classic example, really, I suppose, is Amazon AWS, who back in 2003 with their two-pizza teams. Really famous description of the size of the team. Effectively, we're thinking about the cognitive load on the team by identifying a nice, clear team boundary and nice API around it. And so that clearly hasn't stopped AWS from scaling quite massively.

Matthew:



Episode 10: Team Topologies

Episode Transcript

But starting with this unit of the team and deciding to limit the cognitive load on the team in different ways, through limiting the size of the software system, or API, or product of the team build. Combined with these additional teams, like platform enabling and complicated subsystem as a way to make sure that this socio-technical system that we're working in, works effectively.

Matthew:

As I said, some organizations have had this in mind for quite a long time. Most organizations have not done that. And so, what we've tried to do is to pull together some of this thinking in a way which is easy for organizations to get to grips with some patterns in this kind of space.

Mik Kersten:

And I've heard this reflected around your work, around the book a lot, is that this, actually modeling and thinking through this in terms of the cognitive load on the team, and in the end, realizing that teams maybe do need some specialization. I think a really interesting approach.

Mik Kersten:

So one, what to me seems like anti-pattern that I've seen over the last couple of years talking to organizations is somehow I think a lot of leaders have taken this two-pizza team approach and this whole feature team approach to think that a team can deliver end to end an entire product. And you've got 100% cross-functional team with all the specialization on it.

Mik Kersten:

And then what I see in practice, it's too complex. That team's not going to support its own data backends, its own APIs, its own front ends. You actually end up with what Amazon's ended up with, which is dependencies between teams that are actually critical.

Mik Kersten:

So, I just wanted to take your view on this. Because one thing that in Project to Product, that I stated based on some of my experiences, is that we did an organizational construct that's bigger than the team as well. That the product value stream, and in some cases, a product volume stream can be one team. But oftentimes, you'll need a product value stream that's actually supported by multiple teams.

Mik Kersten:

And of course, some of those teams might actually be specialized. So, you'll actually have an API team within a product value stream as well as a front-end team. Or you might actually have a product value stream that's a whole bunch of platform and API teams that support a number of different teams and business customer facing products.

Mik Kersten:

So, what are your thoughts and what are your learnings in terms of team specialization? I think what's being recognized now, that you do talk about, is this need for platform teams and the like. What have you been learning about the dependencies between teams and how to manage those?

Matthew:

A very good question. I think the dependencies between teams that we see in lots of organizations now are accidental. That's worth saying in many cases. Lots of organizations have not designed their software architectural or software systems architecture for flow. So they're not optimized for a flow of change that is able to be handled by a single cross-functional team.



Episode 10: Team Topologies

Episode Transcription

Matthew:

So that's part of the problem, is actually if we want a rapid flow change, we actually have to rearchitect in an evolutionary way. But we have to put the investment in to enable that to happen. But you're right as well. We've got the concept of platforms. And effectively, we've got a kind of ... If you imagine onion, we've got a multilayered kind of platforms. Depending on who you're asking, their perception of what their platform is will be different.

Matthew:

So, for example, if you've got some people who are delivering a mobile app that allows someone to book something online, their perception of what their platform is, is different from the perception of some team that is working on a backend payment service, let's say. That backend famous service team will see their platform as consisting of something different, which is fine. The key thing is at multiple levels of the organization we have that platform concept defined and managed as a product.

Matthew:

So, we're thinking about user experience. In this case, our users are typically other software development teams. So thinking about developer experience. We're thinking about how easy is it to operate this stuff. We've got a timeline. We've got some good documentation. We understand when particular versions of things are going to be deprecated. All this stuff about product management is being proven. User experience, all this kind of focus.

Matthew:

But typically, a lot of people working on platforms traditionally were from of CIS admin or infrastructure background and didn't have the awareness or skills around product management, and modern agile techniques, and things. So, we'd expect to see a fractal or multilevel awareness of platforms, each one being managed as a product, if you like. So that the streamlined teams that depend on those platforms are able to be as effective as they can.

Manuel Pais:

In terms of this discussion between specialists, and generalists, and the cross-functional teams, I think, again, in a way we're taking the wrong starting point about who should be in the team and these kinds of questions. Because cross-functionality per se is not providing us value, I would say, if that's all we're focusing on. And instead, if we start with the team and try to understand this team which is working for this value stream, what are the skills that they need, the capabilities that they need.

Manuel Pais:

So, we start with that and understand, "Okay, what's the gap? And what do they need to achieve?" And then after that, we can think about it in which ways we can get the team up to speed, if you like. Is it because they need to hire an expert on some specific technology or automation? Or is it that the platform should provide them this capability so they can go faster? You have to actually have the cognitive load of understanding some technologies, some logging, some monitoring. What are the ways in which we can enable this value stream aligned team to deliver the work in the best way?

Manuel Pais:

And typically, that's going to end up with some cross-functionality and some T-shaped people rather than very specialized. But that's not the starting point. The starting point is, "What do they need to achieve as a team? What are the gaps?" And I think where we help, Team Topologies also helps by having these clear types of teams and interaction was having a 10,000 feet view of which kind of teams do we have, how are they interacting, and what is the gap to where we need to be next?



Episode 10: Team Topologies

Episode Transcription

Mik Kersten:

Okay. Excellent. Tell me, what do you think in terms of ... or what have you learned organizing teams of teams?

Matthew:

It's a good question. What we've definitely seen is many organizations, most organizations probably, are not thinking in terms of multiple different size levels inside the organization. We've got historical examples from companies like W.L. Gore who make the Gore-Tex water fabric.

Matthew:

For many years, when one of their offices exceeds 150 people in size, they will open a new office instead of scaling it beyond that. Because this is Dunbar's number. Whether they knew about exactly that number or whether they were aware of that trust boundary before, the key thing is that it's clearly successful.

Matthew:

They're explicitly considering the human social trust boundaries and the effects of that has on how well people work together. And I think what we've seen from books, like Team of Teams by-

Mik Kersten:

Yeah. Love that one.

Matthew:

...Stanley McChrystal is a great book. We're looking forward to interviewing someone who's got that experience in the British Royal Marines commando unit. So, we'll be speaking to him soon and we will have a blog post on that. So, he's got direct military experience of this kind of stuff, which is great.

Matthew:

But a lot of organizations are not thinking about this multilevel scaling and I think that's a real, real problem. We were just talking to a friend of mine who is an engineering manager for a relatively new organization selling stuff online. They grew from 30 to 60 people in the last year. And guess what? They're having challenges. Because the dynamics and trust, it worked fine before with about 30 people, no longer work at all. It's really slowing down.

Matthew:

And so I think we need some patterns for how organizations work as team of teams, if you like. What that looks like, what kind of boundaries in terms of the numbers of people are we expecting to see in different organization units or different streams of value. And at what point do we have to start planning to identify the need to split that value stream? For example, if it gets too big, too successful.

Mik Kersten:

This has been a fascinating topic for me because as I've been helping organizations deploy the flow framework, the focus is on, again, creating that operating level above that agile team layer because it's meant to be aligned to the business. And then there's all this consternation as well. How big should the value streams be? How many people? And so on.

Mik Kersten:

So, I had this long conversation with Dave West, who is now the CEO of scrum.org. It was actually prior to writing Project to Product. And I said, "Well, Dave, how big do you think Scrum should scale to?" Dave



Episode 10: Team Topologies

Episode Transcription

said, "Well, look. SpaceX puts rockets into space and they're well under 100 people. I think you can solve every problem with less than 100 people." And he was saying it flippantly. But we got to talking more and realizing that we, just anecdotally, did not have many proof points of effective software product value streams over Dunbar's number.

Matthew:
Interesting.

Mik Kersten:
Right? All of Google search.

Matthew:
Interesting.

Mik Kersten:
Just like W.L. Gore. They're stuck in one building that won't take many more people than actually thought maxes out before Dunbar's number from another story that they recounted to me. So, the Scrum of Scrum's level, whatever we call it, what I've actually been saying is if you're over 10 teams on a product value streams, that could be a problem. So, what's your reflection on that? Do you think that's right? Do you think that's wrong?

Matthew:
Based on what I actually [crosstalk 00:14:37].

Manuel Pais:
[crosstalk 00:14:37].

Matthew:
If it feels right. If feels about right to me. This kind of thinking that we definitely need to start to bring to bear in organizations. The evidence suggests that 150, that's pretty limited, on a certain kind of trust. Do we want push that? Probably not, because we'll start to see some decay before then. We should expect. Once we get to about a 100, 110 people, that's probably about the limit, and that's a useful constraint.

Manuel Pais:
Beyond that limit, it immediately leads to lack of ownership as an individual within an organization feeling like, "What I'm deciding is or what I'm doing is now localized." And then that leads to local optimization as well. It's us versus them mentality.

Matthew:
The advantage of what we ended up pulling together into the Team Topologies book is that you might get to a point where your value stream has, let's say, a 100 people in it. We're getting a little bit close to Dunbar's number. Well, actually, can we identify some stuff which we can push down into a platform? Can we identify some opportunity for some people to work in an enabling team or potentially a complicated subsystem team?

Matthew:
Again, to reduce the cognitive load on the stream aligned teams and therefore potentially free up some capacity to be able to go faster and get more stuff out the door. It's not just about splitting the value stream at that point. It's potentially about doing some other stuff that improves the flow of the stream level



Episode 10: Team Topologies

Episode Transcription

because we've taken some cognitive load away from those people. They've got more capacity to focus on that domain, business relevant things. There's not immediately specific. But at some point, yes, you might have to get to the point where you decide to split.

Mik Kersten:

Yeah. And Matthew, I think earlier you mentioned the interaction of the software architecture and then the team structure as well. In all of my experiences, and whether it's the Amazon stories, or open source projects, or start-ups I've worked with, just amazing things happen when the team structure, the software architecture, and the end, the product value stream is what you're delivering to your customers are aligned. And obviously Amazon is one of the more interesting examples of that.

Mik Kersten:

As you said, you have to incrementally change these. But I actually love this approach of actually tailoring it out to cognitive load. You can't expect this one team to be expert in every part of your technology stack, and then to bring in the new technologies that you're going to need, and so on. So that cognitive load, that expertise that the team builds up over time, that I think in the right kind of organization and code base is actually evolving that code base to their expertise.

Mik Kersten:

It's a feedback system with a software architecture. That's where I think you get that amazing flow at the team level. Not just at the individual level. I've done some visualizations of past open source projects where you can actually see the software architecture evolve to match the team structure, which is this fascinating thing.

Mik Kersten:

So, I guess, what guidance would you give for organizations to get to that state where they're taking your lessons learned on structuring teams and then the division of labor? Really, the dependencies between teams based on cognitive load, which I think is a great approach. But then making sure that the architecture is matching up. Because so often in these larger enterprise organizations, there's all this frustration where you've got these new teams trying to work agile, or the software architecture is ... it's a massive monolithic thing that 18 teams need to work on and can't get anything done.

Manuel Pais:

So, there's this relationship between software architecture and the team structure called Conway's law. So that there's a mirroring effect between the two, like you were talking about in your experience. If you have different people deciding on two different things that are closely related, you might end up with HR or managers who are not technically having influence on the software architecture indirectly by deciding the teams without this technical concerns.

Mik Kersten:

Does that happen?

Matthew:

Well, it effectively does.

Mik Kersten:

No, I'm kidding. I'm concerned that's not the fact, the [crosstalk 00:18:26].

Manuel Pais:



Episode 10: Team Topologies

Episode Transcription

That's the [crosstalk 00:18:26].

Matthew:

This is a crazy thing. So, if you put it in these terms, HR and non-technical managers are designing your systems architecture. Do you really want that? No. No way.

Mik Kersten:

It actually sounds even crazier when you put it that way than I was already very concerned about. So, yeah.

Manuel Pais:

There's also cases ... and that's why today's talk we tried to focus on cognitive load and not on the architecture to start with. Because there are other dimensions. Even if you have a great microservices architecture, it's all the rage and everything, we still come across many examples where, "Okay, we have microservices, we have teams aligned to microservices, and we still cannot deliver."

Manuel Pais:

Because actually, the part that you said in beginning there, this has to deliver value to the customers was not really well thought out. So actually, the value is aligned to user personas or user journeys. And then anytime we need to deliver a value on that, we need to have dependencies between 5, 6, 10 teams, each responsible for their microservice. Then you still don't have a software architecture that's really powering value for the business.

Matthew:

So, it speaks to what you said, it's only in your keynote. You've got to invest time, and expertise, and effort in understanding what your value streams are, really. Because if you get this stuff wrong, you'll be fighting against this huge force. And it's different in different organizations.

Matthew:

The same kind of organization could have very different value stream structure to one that looks superficially similar. The streams can be around business domains and a DDD domain design type [inaudible 00:00:19:55]. But yeah, it might be around user experience, it might be around kind a checkout flow for a user, or something like this. It depends.

Matthew:

It depends what you want to optimize for. You optimize for that particular kind of flow, other kinds of flow aren't going to work so well. So you might need to split off part of the system if you've got another part of the system which has a very different set of flow needs, if you like. So you might end up with two different software systems. That's fine.

Matthew:

But you've done the model. You've done the understanding with the domain experts to enable you to align teams to the flow. So it's great. We've got things like DDD, and Flow Framework, and things now to help us do that, which weren't a few years ago. So we're in a good position with this stuff. But it's still very unfamiliar philosophy to think about it in these terms.

Mik Kersten:



Episode 10: Team Topologies

Episode Transcript

Absolutely. I could not agree more. So just a final question. You've just written a great and impactful book. How do you find your own flow? So if each of you could tell me some of the key things that you've adopted in terms of your own practices.

Manuel Pais:
It's a good question.

Matthew:
Yeah. Right?

Mik Kersten:
Manuel, you go first.

Manuel Pais:
So you mean in terms of consulting?

Mik Kersten:
Your own work, your own flow, your own happiness, and productivity.

Manuel Pais:
Well, it's like I say, it's always a struggle when you try to apply it to yourself, especially with kids and traveling.

Mik Kersten:
That's why I'm asking.

Manuel Pais:
Well, first of all, clarity of my own business vision, if you like. Like what are the really important things to me. Not just work, but also family, and try to set the right priorities. So that helps a lot to align the work. And we were just talking, we have a Trello board with hundreds of cards around things we could do around Team Topologies. But obviously we need to focus on what's really the highest priority.

Manuel Pais:
So I think in terms of the book and what we're doing ... So, we're right now with giving talks and first day of training around fundamentals around Team Topologies to raise awareness. And like Matthew was saying, tell people, "Here are some tools to think about Conway's law, think about Dunbar's number. Start thinking of teams as the means of delivery rather than individuals aligned to those streams."

Manuel Pais:
So I think at this stage, we're focusing on raising awareness. And then from next year, we'll focus more on how we can help organizations with some kind of tooling. I don't mean necessarily software, but any help to actually apply this concept and have more deep-dive training as well. That's how I see my flow or my roadmap at the moment.

Mik Kersten:
All right. Matthew?

Matthew:



Episode 10: Team Topologies

Episode Transcript

From my consulting business, effectively, I've got four value streams. Currently, three of them reflected on my website consulting. So, I work hands on and alongside head of engineering. I'm this kind of role inside organizations, helping them to deliver stuff myself. I deliver training. But also, work with other people to deliver training as well.

Matthew:

And my company also publishes several books. I'm the author of one or two. But also, Manuel was actually the editor for some of those books, too. So that's a value stream. They're complementary, but somewhat separate. It's been interesting to think about the relationship between those, and how things are flowing through the different streams, and so on.

Mik Kersten:

Makes sense to me. You're tracking value streams to optimize the flows.

Matthew:

Yeah.

Mik Kersten:

All right. Excellent. Thank you so much. Check out Team Topologies, and great to see you and chat with you.

Matthew:

Perfect. Cheers, mate.

Mik Kersten:

Thank you to both Matthew and Manuel for taking the time to chat with me. I hope you'll agree that this was a highly insightful conversation. I hope it urges you to buy their book, Team Topologies. If you want to reach out to them, you can find both on Twitter. You can find me on Twitter and LinkedIn as well, or using the hashtag #MikPlusOne or #ProjectToProduct. You can also search for Project to Product to get the book. And remember that all of the proceeds go to supporting women and minorities in technology. Thanks, stay safe, and until next time.