

One Knight in Product - E169 - Luke Hohmann

📅 Thu, May 11, 2023 1:38PM ⌚ 54:27

SUMMARY KEYWORDS

book, safe, company, work, solution, consultants, software, profit, agile, certification, build, create, transactions, licence, product, customers, compliance, called, game, wrote

SPEAKERS

Luke Hohmann, Jason Knight



Jason Knight 00:00

As product people, we talk about value all the time, but what is value? How does it connect to what the company really cares about; cold hard cash? Tonight we're going to talk all about that with a little bit of scaled agile as an appetiser. Speaking of agile, whatever framework you're using, you're working with developers. And if you're non too technical might struggle to bridge the gap. If you're a non technical product manager or founder and want to build those bridges with your technical teams, why not check out Skiplevel? Skiplevel's an on-demand training programme that helps professionals and teams become more technical in just five weeks, or without learning how to code, you can learn the knowledge and skills you need to better communicate with devs and become more competent in your day to day role with the Skiplevel programme. So head over to <https://www.oneknightinproduct.com/skiplevel> to find out more, you can use the referral code OKIP to support this podcast. And make sure you check the show notes for more details. All right, so back to the episode and a wide ranging discussion about turning our products into metaphorical and indeed physical gold will talk about how to get agile to meet business reality and some of the things we need to consider in our whole solution if we're going to realise value and build something that sustains our customers and us that sounds good. Stick with us on One Knight in Product.



Jason Knight 01:21

So, my guest tonight is Luke Hohmann. Luke's an author, consultant and former aerobics instructor who's now trying to get companies to do enough star jumps and burpees to get themselves fit and ready for success. Luke used to work for insert ominous thunder sound effect here Scaled Agile incorporated as a former member of the SAFe framework team, SAFe principle contributor and says he hates it when people criticise SAFe without even learning about it. I'm sure that never happens. But he's also the co author of a new book, Software Profit Streams, which aims to help you create sustainably profitable software enabled solutions. Hi, Luke, how are you tonight?



Luke Hohmann 01:52



Luke Hohmann 01:52

Great. How are you? That was awesome as an intro.



Jason Knight 01:56

Well, you know, I just wanted to see if I get SAFe in there as many times as I could. I'm fine. Thank you very much. Glad to have you here. It's been a long time coming. We've been kind of back and forth about safe and other related issues for a while now. So be good to lay it all out on the table and see where we end up.



Luke Hohmann 02:11

Yeah, we'll start we'll start the meal with some good food.



Jason Knight 02:15

But before we do any of that, by day, you're the Chief Innovation Officer at Applied Frameworks. So what problem do or does Applied Frameworks solve for the world?



Luke Hohmann 02:25

Applied Frameworks is a boutique consulting firm, which is just fancy language for we're small, but we're punching above our weight. And we're better than most of the big firms in a lot of areas, we focus on helping companies build profitable software enabled solutions. So some companies will work in the government space, and they'll help the government build a better tax system or, you know, management system for managing permits or construction, whatever. And some companies will do specifically hardware or they'll specifically work on internal corporate IT initiatives. Our focus is if the offering has a price on it, and it's sold to a customer, you bring us in because we're going to help you create a more profitable solution. Sometimes that approach requires an improvement in your development processes. So we might bring in some form of agile practices. Sometimes it's going to include upskilling, your product managers, so we'll bring in our product management training, we have a product management academy, not unlike other product management courses, but our unique area and our unique skill set is in helping organisations with pricing licencing and packaging and creating a sustainable solution.



Jason Knight 03:50

But what does the Chief Innovation Officer do there? I mean, I'm assuming that doesn't mean that you're just sitting there in the bath waiting for eureka moments and running down the street. So how do you specifically fill your days when you're not writing books, or talking to people like me?



Luke Hohmann 04:03

LUKE HOHMANN 05:03

Well, I do delivery, because we're a boutique firm, which means everyone except for the operations and support staff, we all do delivery. So the Chief Innovation Officer gives me a little latitude on how I do delivery, partly because I'm often doing delivery in a way that is the tip of the spear that creates opportunity for the rest of the company or is maybe doing some experimental work. So I might work with some of our largest clients in ways that are experimental, that help them in ways that align with their innovation efforts. I often work with some of our tiniest companies, where there might be a startup of less than 15 people who is been just VC funded for the first time and they need to improve or adjust their pricing to hit the numbers that the VC fund is vested in. So my span of clients can be in the fortune 10, or in 10 people,



Jason Knight 05:07

But it's obviously, or it could be considered quite tricky to start to really go for profit when they're super early. Right? So I'm assuming that the principles still hold. But do you find that you're playing a different game? And that maybe you have to adapt the approach somewhat to get to these kinds of in some cases, even pre revenue companies? Or are you kind of getting them at the point where they start to make serious revenue?



Luke Hohmann 05:27

The principles, as you suggest are the same. So when you're looking at the notion of how do I make profit it there's there's durable principles, you have to do a economic analysis of how you provide value, we call that a customer benefit analysis, we then have to have a mechanism of trading money for value, we call that a value exchange when then we have to have a pricing strategy. And what we'll work with on accompany, as we'll look at, well, where are you in the industry? lifecycle? Where are you in your solution lifecycle, I might start with a penetration strategy, when the product is first launched. And then as I move into a more sustainable position, I might raise my prices. And we see this consistently, we see people not raising prices, theoretically, in the Agile community, whether you're practising safer Scrum or just get it done, you're presumably accumulating value you're delivering value. In every Agile method says there's some form of a backlog and there's some form of value creation. So if you're delivering more value, presumably, there will be a moment when you're going to be able to charge more for it or to package it differently to further serve specific unique segments as the company evolves. So the principles are absolutely durable and universal. I do agree with you, Jason, when you say that, what about profit, in a really fast growing company profit is a choice. Yep. And I am taking in a bunch of revenue, and I'm accumulating revenue. This truly sustainable companies, though, there is a tipping point where on a per sale basis, or per unit basis, but typically think of it as per sale, my net income is starting to be positive, meaning I'm creating a generative process where I'm, we're somewhere along the line, if you're if your company is sustainable, you can't escape profit. Because on a per unit basis, you've now tipped and you're going to be successful. So with our startups, we do let them we sometimes will help them model like, Okay, if you slow down your investment, would you be profitable? And if they can't answer that question, then it's a more fundamental challenge. For very large companies who have margin targets or more complex structures, we might need to help them be profitable, much more quickly in that adoption curve.



Jason Knight 07:54

Oh, there you go. So there's something for everyone. But before we talk about your new book, yeah, I want to talk a little bit about SAFe.



Luke Hohmann 08:02

I'd love to!



Jason Knight 08:03

Of course, you would, and we talk about SAFe from time to time. And I'm always up for debate about these types of things. And there's obviously lots of say, and lots of talk about but you worked for Scaled Agile Inc for I think two and a half years.



Luke Hohmann 08:16

Not quite two and a half years, but...



Jason Knight 08:18

Nearly two and a half years.



Luke Hohmann 08:19

Yeah, about that long. I started as a consultant. And I helped evolve the framework and then Scaled Agile acquired my company conteneo. And we had built a software platform and some other things, and that augmented the safe offering. And so the software platform that we had created, called conteneo. Weave has evolved and continued to evolve. And now it's called SAFe collaborate. So SAFe Collaborate is the collaboration platform that is used in many safe implementations. It's used in training. It's also used in its originally intended purpose, which is collaborative problem solving and market research.



Jason Knight 09:00

So was that a SAFe-specific tool before you got acquired? Or was it something that they then kind of blended in?



Luke Hohmann 09:07

It was a generic platform, right. It was based on the work of innovation games, and it included some of the Game Storming games from David Grey. It also included other leadership frameworks and some very basic tools. So it had certain qualities to say a Miro or a Mural or

iObeya, but it also had some key differences in terms of how it was constructed. I would characterise some of those other tools as being better at like, pure open ended drawing of concepts and relationships. Our tool was better at specific tasks like scaling retrospectives, or doing market research at scale, or doing participatory budgeting at scale, which most other tools don't have the participatory budgeting capabilities that our tool has..



Jason Knight 09:57

Yeah, no, I've definitely had you mean don't get me wrong. I love tools like Miro, but I've definitely had moments where I'm like, I've got a copy this all out again now to actually do something with it. So yeah, no, I get what you mean. But what are you using safe before you got acquired by safe? Like how do you use safe before? Was that something you then had to kind of learning skill up in and kind of, you know, uplevel yourself?



Luke Hohmann 10:18

No, we were using concepts of SAFe that continue, I think of two things. There's the company building structures. And then there's the product development structures. So for the company building structures, we use Verne Harnish's work over at scale up or scaling up. And we think that that's a very comprehensive model for how to actually build a company. That's also informed with my friend Alex Osterwalder in his book, Business Model Generation and value proposition design. So we blended some of those concepts. From the SAFe side, you have to make choices at any company about how you're making investments, and how you're flowing work through your solution. So in terms of a formalism, we would be aligned with the portfolio configuration of SAFe, not the full SAFe and not large solution SAFe, because we were a small company when we were building a large solution. And we didn't need the full safe because we weren't big enough. And I wrote a really great article that people should reference at Scaled Agile, it's called Startup SAFe. And it's specifically how do you how do you tune save to a startup? And so you start with the portfolio? You chuck some of the things that you don't need, you add a couple of things that I think enhance the application? Because you know, SAFe's a framework, it's not a law.



Jason Knight 11:46

Right? Oh, don't tell some people!



Luke Hohmann 11:49

Well, then, I can tell people that. But looking at the notion of how I'm constructing my solution, for us SAFe really does align with our needs as a small company, because I know and I know, this is paradoxical people are like, wait SAFe is supposed to be for scaling agile? Well, yes, it is. But when you look at some of the other agile methods that are out there, you still have to make very important decisions, like I've got three big ideas to invest in. Now big is relative, right? Big for Cisco might be I've got 320 \$2 million dollar ideas to invest in big first startup might be I've got 360 \$1,000 ideas to invest in. So big is relative, but there's always big decisions to make.

How you make those decisions determines your degree of success in terms of collaborative behaviour, are you looking at a good process, etc, etc. And so for us startup SAFe in the portfolio safe tailored to startup is a really fantastic way to help companies succeed.



Jason Knight 12:50

Well, I'll make sure to link the article in just so people can have a little look at that themselves. But I do want to say for the record that, whilst I love to make fun of SAFe, I also love to make fun of all of the other frameworks out there, and indeed make fun of myself half the time and product managers and the very concept of entrepreneurship. I think everyone should be fair game. So I do believe that in many ways, I'm kind of Switzerland in these matters. I mean, I will also say that I've never used safe myself, I've never worked in safe. I do know many people that have and my experience is limited basically to talking to those people, or reading the book SAFe distilled. But as a somewhat more of a believer in why SAFe is awesome, than maybe I am. Why is SAFe, awesome. And what does it really enable?



Luke Hohmann 13:35

Powerful question, and it speaks to the problem that we're solving. If you're a consultant who seeks to get a lot of money to help a company build a method, then SAFe is awful. Because SAFe is going to have so many answers for the client that you're working with that you're going to twiddle your thumbs and go, What do I do? I don't I mean, I want someone to pay me to build them the process that they should use to build roadmaps. If I go with SAFe, there's already a roadmapping process built into SAFe which actually was built on my pattern language for strategic product roadmapping from my book Beyond Software Architecture. So there's a pattern language for building roadmaps and SAFe. Oh, as a consultant, I'm missing revenue. Now our position would be when you look at the complexities in the world and what people need to build. We used to believe and let me give you a different analogy. Jason, let me give you a really different analogy to kind of frame the point. Okay, when SAP first came out with SAP R 3, it was a breakthrough in enterprise resource planning. And you would have SAP consultants descend and they would tune SAP R 3 to your manufacturing process. And we would code like it was semi source code available, and you would tune it to in tune it. And then what we found what we found was that didn't work. The reason it didn't work is because it said that what the company was doing was always the best practice. And when SAP evolved its solution, you had to go back and you had to you had to work again, you had to work again, you had to work again. And eventually people are like, Wait a minute. SAP sees manufacturing processes and practices across 1000s of companies. And their job is to distil those best practices into a solution that helps you adopt them, maybe, maybe, just maybe the right way to do this is to adopt the software, and let the software change my organisation and change my manufacturing practices. So I'm getting truly the next best practice, even if it's hard for me to make that transformation. And I think of SAFe in very much the same way. It's not a law, and we joke that, but it's got amazingly good, durable practices. And, sure, SAFe didn't invent every practice that it has in it, but who cares? Like, literally, who cares? Well, the consultants who make money from inventing practices care, I'm more interested in going to my clients and saying, Your problem is you're dealing with a complex competitive environment. Why should you be software methodologists? Like, that's a hard job.





Jason Knight 16:29

It is, but I'm gonna have to point out one thing, which is gonna be the obvious rebuttal, which is that SAFe itself has its own bunch of certifications, and consultants. And, in fact, I've got SAFe distilled up on my screen now. And it's talking about getting your SPC certified, SAFe implementer to come in. So like, I'm not saying that you're wrong at all, like those things are obviously true. And anyone that's selling some other framework, and even me that sells no frameworks, but still obviously wants people to hire me. Like we've all got a vested interest in whatever it is that we're selling. But that could also be squarely aimed at SAFe certified implementers as well, right.



Luke Hohmann 17:04

Sure. So let's tackle certifications. And I wrote a very powerful article with my friend James Bach in 1999. Called certification is discrimination. And I'll happily add that and we can edit into the show notes. The question becomes, who is certifying what against whom? And how, and society's love certifications? Would you go to a doctor who wasn't certified?



Jason Knight 17:29

No



Luke Hohmann 17:30

Why not?



Jason Knight 17:32

Because I wouldn't believe that they necessarily had any knowledge about my body.



Luke Hohmann 17:36

Right? And would you trust the outcome of a doctor who wasn't certified?



Jason Knight 17:40

I would, probably not.



Luke Hohmann 17:42

And when you look at certification, you start to also realise that there's a there's a, there's a there's a related concept called licensure. So, now, a doctor is both certified and licenced. Are you certified to drive your car?



Jason Knight 17:59

Yeah, absolutely.



Luke Hohmann 18:00

Who certified you? You have a licence to drive your car, but are you certified?



Jason Knight 18:05

Oh, no, sorry. Okay. If you're using that definition, then I'm licenced but not certified.



Luke Hohmann 18:09

So now let's follow it out, right? Societies create licences and certifications to create a couple of societal needs are fulfilled, fulfil a couple of societal needs. One is the attribution of liability when something goes wrong to is the realisation that we as a society want to protect people who don't know how to protect themselves, meaning, I don't know how to select the doctor. So I rely on the fact that doctors are both certified and licenced by a body of typically a government agency that I can trust or some agency that I can trust. So now let's look at certifications in our industry that people say they like because people will say, I don't like certifications, you start to really talk to them. And they're like, Well, you know, a Cisco Certified network engineer, that's not so bad. Well, wait, I thought you were against certifications, or a Certified Information Security Services professional CISSP. Oh, wait, what? Yeah, look, that certification is okay, too. So I like to point out that interest goes where money flows. And people who don't like a certain certification are you might ask yourself, well, how do you make money or not make money on a certification or not a certification? Because it's really curious to me when people say I don't like certifications. So you'll let me operate on you, right? You'll or you're you're selling your home and you're going to let anyone sell your home? Are you going to let a licenced and certified realtor, sell your home? Help me out here. Help me understand your perspective of certifications. Now, what is a more interesting conversation is people will say, Well, you know, this certification was granted on a two day class, and you can't possibly be skillful in a two day class. Fair enough, like, fair enough. When I built the continuous certification model, there were three levels. One was certification based on training. The second level was certification based on demonstrated facilitation experience. And the third level was certification based on producing large events that had multiple facilitators with postprocessing. It was a bit more complex than needed. And what I learned is, all of the people who complained about certifications are also saying, the people who use in leverage certifications are complete idiots and don't understand what they're doing or buying. I don't quite buy into that. I don't think that HR people are absolutely daft. And don't understand that taking a class based certification is the start of a journey towards commonality. Let me give another example from a technical side. In the development community, there was a breakthrough book and it's in the concept of patterns is one of the precursors to agility, right? Because patterns let us compress knowledge and not have to wait longer MRDs and things like that. Right? In one of the breakthrough books was designed patterns. Well, why would I teach someone design patterns? Because it's an efficient way to communicate shared bodies of

knowledge? So if I am one of my clients was be when party. Okay, I'd be when party I make over a billion in gaming revenue. I've got over 140 agile teams, do they really want every agile team to learn a different agile framework? Seriously? Like, how many understand that how does that benefit the client? I get how it benefits the consultants pay me when party have hired five consultants and they've all told me something different. Really? Wow, how is that customer centric?



Jason Knight 21:52

Well, one of the interesting things that maybe an agile fundamentalist would say they're like an XP fundamentalist or something would be like, Well, every single team should be able to self organise and self manage and just choose whatever it is that they want. I'm assuming that you don't buy into that.



Luke Hohmann 22:07

Well, I actually believe in governance because I've, I build adult software in adult contexts. And I built I built mission critical stuff. Most of my career, I started my career at electronic data systems and the mainframe world, right? We built things like claims processing system for health care, I build credit card processing systems, I built stuff that kind of needed to work. So no, it really doesn't help me when I've got to agile teams, like just like, Hey, Luke, we want to have a date format. And one person wants to use ISO standard A and another person wants to use ISO standard B for the dates. Are you kidding me? You know, Jason, one of the I have this conversation all the time, like people ask me, Why do Silicon Valley companies who are large moves so fast, and one of the secret powers is there's really strict governance. When I was at Conteneo, our tech stack we were the one of the things we were proud of is we were the first production system written in Scala. And we funded David Pollock to create the Lift framework. He's a dear friend, but we needed Lift. So we've we literally funded Lift and then made it open source. Now here's the governance. I hired developer developer says, Hey, I'd really like to use Erling or Java and we're like, Great, take a different job. We use Scala, like, here's our governance and what people in the Agile community I think often miss is that governance is designed to create innovation and opportunity in a way that provides scale and safety. And I'll give you another analogy, which is really powerful. Humans have lived in cities of more than 100,000 people. For more than 5000 years, Babylon had over 100,000 people. Now, I'm in my city, you're in your city. We all connect to pipes, plumbing power, using the standard at the time, our house was built or remodelled, and those standards evolve. Now I drive down my street, I see houses with different sightings, I'd see houses with different architecture, different paintings, but I know that all of them are connecting into the water and sewage and power system safely because they were built according to the standard and they were inspected. That's governance. And the agile company is like, Oh, well, you know, we can't do that. Look, we want to let every agile team make their own choices. Are you effing kidding me? Like we let everyone in society make their own choices of how they connect into the power grid? Sure. How's that gonna work?




Jason Knight 24:45

I'm sure there's some guys in sheds in the mountains with shotguns that would absolutely love the sound of that. But I think it's also fair to say though, that I mean, obviously yeah, you do


have a bunch of agile fundamentalist detractors of SAFe and they'll I come out with ...

 Luke Hohmann 25:00

Many of them are my friends and I love them to bits. I respect them, and I love them. I just have a different point of view.

 Jason Knight 25:08

But some of those criticisms will be, you know, it's to top down, it's just micromanagement under a different name. It's a fake, agile beard that makes enterprise leaders feel happy that they're transforming without really transforming. Are these all things that people say, or again, as we said earlier, it's all about certifications and consultants, that it promotes processes over making people do their best work. It promotes feature factory mindsets, like these are all things that I looked up. And these are things that people are saying. So why on a high level, because I know we want to talk about the book in a minute to have these people wrong about that?


 Luke Hohmann 25:41

But they're not wrong. There are isolated instances where all of that has occurred. And some of that has been occurring in the context of safe and some of that has been occurring in the context of Scrum. And some of that has occurred in the context of LeSS, or Spotify or whatever. So to claim that human frailty is such that we can adopt a method and never make a mistake is kind of missing the point. I mean, really, look, I'm not perfect as my kids, my wife, right? As the people who work with me, I am absolutely not perfect, I have many flaws. That doesn't mean I strive every day to be better. And for me, regardless of the methods safer. Otherwise, people often ask me, What's the biggest root cause to failure in any math, you know, Agile transformation. And my response is to, it's always the same. It's impatience and believing that the word transformation is actually appropriate. Because transformation first in patients, it takes time for people to change. Like, it takes time for big companies to transform. And impatience on anyone's agenda hurts the company doesn't help them. And the second thing that I point out about this root cause of failure is this kind of elitism that often goes into consultants, by definition consultants who are brought in to quote unquote, fix a problem see problems. And they assume that just because the client brings them in, there's problems, and so they have to look for problems. One of the nice things about applied frameworks is, because we're helping build profit, we are not finding these dire, awful implementations of whatever. What I tell people is we're typically brought in when we're dealing with elite people who want to be better Tiger Woods changed his swing to become a better golfer, Tony Romo, the football quarterback got drafted out of college and then retrained, designed to be a better quarterback. People bring in our team when they're already operating at a pretty decent level, and we can help make them better. And so this notion of top down or heavyweight, or whatever, sure, that happens, that happens in other methods too.

 Jason Knight 27:54

Well, yeah, I was gonna say, there's plenty of people that would complain just as much about Scrum as well. So I think one of the things that's reflected on with regards to your legacy Scrum


Scrum as well. So I think one of the things I've reflected on with regards to, you know, Scrum, and, and SAFe and some of these other methodologies that do have some kind of process or framework around them is that it's really easy to complain about the worst implementations of them that you've seen. And I'm not going to sit there and make excuses for any framework, because again, I'm completely neutral in all of this. But I do think it is interesting, because one of the thinking points that I've had over in this case, Scrum, is whether it's actually the responsibility of the people that make the framework or the methodology or whatever it's called, whether it's their responsibility to make it harder for people to misuse it, or whether that's just something that's going to happen within companies. And I guess I could ask you the same question about SAFe like, obviously, you've worked for Scaled Agile, there's the whole organisation that promotes and sponsors the framework, like is it on them to try and help make safe more, I guess, to make it better applied? Or is that very much in the hands of the companies themselves?

 Luke Hohmann 29:00

I think that the Scaled Agile Framework team, and I was a member of it, I'm no longer a member of it. But I know, the Scaled Agile Framework team, I think they take that responsibility very, very seriously. The question is, how do we construct the framework to be both evolving with the needs of our customers? And how do we help prevent misuse, if you will. So one of the more important elements of safe is that it does evolve, right? You can see SAFe go from two to three to 4, 4.5, 5.0, 5.1. It's going to continue to evolve. And next week, there's going to be a new announcement of of the evolution and some of the improvements in safe that are coming. So I think that those evolutions include an awareness of, oh, people are starting to make this kind of mistake, what kind of guardrail or what kind of guidance or what kind of recommendation can we adjust? How do we change our language? How do we how do we improve like, oh, people are misinterpreting And our meaning behind this, how do we clarify that? I was the principal author of the SAFe problem and APM and lpm courses. And I took a lot of time trying to like, what's in what's out? How do you frame things as best you can? How do you create activities and exercises? In? Absolutely. I don't think the framework people can be held responsible for how it's absolutely applied. But I do agree with you. And I think that the SAFe leadership really does very carefully consider, how do we construct something so that it's more likely to be used as intended?

 Jason Knight 30:37

Oh, there you go. Well, I do believe we should have another deep dive into SAFe and go into some of the specifics at some point

 Luke Hohmann 30:44

I love that

 Jason Knight 30:46

I'm going to take advantage of this quick segue to remind you about skip level. If you're a non technical product manager or founder and want to build bridges with your technical teams, why

not check out skip level and on Demand training programme that helps professionals and teams become more technical in just five weeks, all without learning how to code, check the show notes for more details. Okay, back to the episode.



Jason Knight 31:06

I do want to give you a chance to talk about your book. So your new book, Software Profit Streams, which I've been privileged enough to have a chance to have an early review of and gave my comments and hopefully helped out a little bit. But for the people that haven't had the chance to do that. I guess the obvious question is, well, what's a software profit stream? And why have you co written a book about it?



Luke Hohmann 31:26

A software profit stream is the necessary evolution of a value stream to something that executives and leaders can actually understand. The Agile community has latched on to value stream and value stream mapping in a way that is sidestepping in many cases, what executives understand I'm producing value. Well, what does that actually mean? I'm actually running a public company, I have to produce a profit. I have shareholders like so. So I'm not...



Jason Knight 31:55

Unless you're Elon obviously.



Luke Hohmann 31:57

Yeah, right. Exactly. Unless you're Elon, right? Well, you know, there's a joke in Silicon Valley with the best way to make a good bit of money, start with an enormous amount of money and invest it. In Elon Musk is a great example of how do I make a good bit of money while he had he was the world's richest man, and then he pissed it away. You see that with with the founders who sell their companies, right? Hey, I sold and I made these millions. And now I'm not worth much. Well, what do you do I invest in a lot of other startups. Okay, so let's go back to the book software profit streams. So yes, when you look at the value streams, what we're saying is we want a continuous flow of value to our customers. Absolutely. If you're internal IT, if you're a government, it, whatever, fantastic. But if you're a company who needs to make a profit, that value stream has to actually show that it's making a profit. To make a profit, I have to go through the black and dark art of pricing and licencing. And what we have is an entire collection of books about pricing and licencing that were written for boomers, by boomers, and they talk about pricing pencils, and they're dense, and they're text heavy, and they're frankly, boring. And they talk about pencils, like it's software, but it's not software is not a pencil. It's intellectual property. So it comes with compliance. It comes with licence agreements, there's not a single software solution that can be built without an end licence of technology. When people say, Look, I'm not licencing anything really what you write it in? Well, I wrote it in Java. Well, Java comes with a licence, buckle up, buttercup, it's common. And so So what what we did, and what we know to be true, Jason is that there's a couple of really important precepts. The first is pricing is not a number, it is a system and the system has to work together to

accomplish the objective. Number two, if you are providing value over the solution lifecycle and your solution is evolving, then your pricing and packaging also need to evolve. Number three, pricing is a team sport. Product Managers are the centre of the sun because they represent the universe, right that product, but they're working with legal, they're working with architecture. They're saying, Oh, our licence agreement says that it's a term based licence. It's enterprise software. It's in the cloud. It's a term based licence. What happens at the end of the year? Well, I don't know my licence agreement, my technology, architecture, and my business model all have to align for that term completion, to be correctly implemented in the country in which I'm being governed. GDPR has different rules than America for how the data ends when the licence is over. So when we wrote this book. And there are many books out there like I love I've mentioned earlier, my friend, Alex Osterwalder. They're the great strategize your books. But even in the great strategize your books, there's there's not a statement about how to price and licence your software. And that's endemic to or it's required to get to product market fit. So we we decided, there's no me there's lots of great, there's a lot of great product management books out there, right, like, Melissa Perri has a great book. And there Marty Cagan has a useful book. And there's lots of great stuff out there. None of them cover the actual like, how do I price and licence my solution?



Jason Knight 35:39

That's interesting, though, because you talk about the strategies of books. And of course, those books are very visual. And of course, your book is also very visual. So is that kind of a homage to the strategize er books? Or was there another reason that you decided to kind of, I mean, you talked about dense books, but was it? Was it kind of a mixture of homage and not wanting to be dense? or was there some other reason you decided to go kind of visual and a bit more creative?



Luke Hohmann 36:05

Yeah, I absolutely believe that it's undeniable that we were inspired by Alex Osterwalder and the strategize er book series. I know those people, but there's other books that are similar, like the design thinking playbook is similar in that structure, where it's very visual. And I've written for other books, right. And so I wanted to also challenge my like, my first book is the epitome of dense text, right? Journey of the software professionals very dense, very academic, very, in a sense, academically written, and therefore it's a little awkward. What we did in software, profit streams that I wanted to push myself, and so did, Jason. So we literally wrote the book by hand. We it was caveman writing. And we wrote it by hand with coloured pencils. And what I found was, when I wrote something down by hand, and I didn't like it, and I had to erase it, I wrote fewer words, because I didn't have the the natural flow that you get. So one of the virtues of the book is even though it's 400 pages, everyone that I'm getting feedback on it saying it's unbelievably readable, because every sentence matters, and those pictures really help.



Jason Knight 37:21

Okay, but that's really interesting about the readability because the book does. I mean, you say it's 400 pages, which of course it is, but those pages cover a lot. They cover everything from Agile principles to design thinking to market sizing to pricing and packaging, financials,

compliance, legal, a lot more besides that, as well. So I guess I do have to ask, is this designed to be read through easy or not in one sitting? Or is it more like an encyclopaedia that you kind of dip into when you have a specific problem that you want to solve?

L

Luke Hohmann 37:49

It's a little bit of both. The way that we are starting to see people use the book and practice in the way that we're employing it in both our training and in our consulting, is that depending on where the client is in their solution lifecycle, they're going to use the book a little differently. So if in your if you're a new product development, and you're kind of instantiating, the first business model and establishing the first price of the thing that you're working on, the book tends to be needed to be read all the way through, but there is a guide. So there's an Applications section of the book. And Jason, you mentioned the the homage to the Business Model Generation or the business model canvas, we have a canvas to as the profit stream canvas, oh, yes, right, we did it for the same reasons, which is we're trying to create a compact representational form of the interactions of the system elements. And what we find is that in the new product development, you have to have all of the canvas blocks, but a lot of times you have kind of provisional choices that relate to the other blocks. And so what you're looking at is, if I do this, what's the impact on this block? I like that choice for my market, let's solidify it. Once you're in the market, the use of the canvas changes to look at what we call triggers, then there are a few triggering events that motivate the application of the canvas to change things. So one trigger event might be your own improvements to your software. I keep putting more features in I'm going up the adoption curve. When should I repackage? Or when should I raise my pricing though, that's a that's an internal trigger, because it's something that you did. And we have a guide on how to apply the canvas and internal triggers. There's also external triggers when GDPR came out, and it's an easy one to talk about, because it was so profound. But when GDPR came out, it was an external trigger that caused us to make significant changes to our compliance, and in some cases did affect our pricing and licencing.



Jason Knight 39:59

Yeah, well, you Yeah, I was hearing. Well, we were in Europe at the time when GDPR came in. And it was definitely an interesting one. But you talk about the profit stream canvas. And that's obviously, as you say, kind of a framework for the book, it's there to help teams design profitable solutions. And it's also freely available on your website, Creative Commons, and all that. So obviously, you're giving something back to the community, as well as the book that they can buy. And there are a lot of sections in there. So I didn't want to go through them all. But I did want to touch on some of the overarching themes and things that you might want to think about. And those themes, I believe, are customer solution, monetization and compliance. And there's obviously lots of other sub themes and other bits and pieces then that those encompass. Now in pure product management terms, we might consider customers to be almost like our sweet spot, right? Like, we're always they're talking about the jobs to be done the five why's so uncovering their pain points, all of the stuff that all of those product management books you talked about earlier, are talking about all the time. But what are some of the things that we need to think about with regards those customers when we're talking about a profit stream context, like things that we might want to consider that we weren't considering before?

L


Luke Hohmann 41:03

Yeah, I think there's a couple of things that we add in that body of knowledge. And in fact, one of the sections of the book that we wrote that we'll be publishing on the website, because it didn't make the limit of 400 pages was a relationship to, to our customer benefit analysis and safe jobs to be done. And there's a couple of things that I think that we think about that I haven't seen as well represented in the jobs to be done literature, because we have this very, very economic approach. But one thing is that in a complex, especially in the business to business environment, but also in high end, business to consumer, like B to C value doesn't occur in isolation value is a set of relationships that then impact each other. So for example, let's say I'm in the business world, and I'm in the trucking industry, if I'm building a solution, in which I'm helping improve fuel economy, in diesel powered trucks, that's a positive thing. And we know that the trucking company wants that. But if it has a negative effect on driver satisfaction, because the driver has to alter their driving habits, well, then the solution is intrinsically less valuable. Because nodes in the value map, we call it a solution value map, there are nodes that are related. And if the maximisation of one node creates a deleterious effect on a different node, your overall solution may not be successful. And that's one thing that we do see missing in a lot of analysis. The other thing that we see missing is, sometimes we get into situations where we can create more quality that people care about or are willing to pay for. And a lot of times consultants will walk into organisations where their quality is low. And their job is to kind of raise the quality but because applied frameworks tends to deal with successful companies, we've actually had a couple of our clients who were overshooting the quality bar. And then we had to point out like your customers aren't paying for this quality, they don't care about this level of quality, you need to shift your focus over here, which is an underrepresented thing. So we have a way of visualising that in the book. And in our work with clients, like there's a point at which the client kind of doesn't care. And if you keep maximising it, you're not being helpful. And then there's ways that you can get out that with quinoa analysis and other things. Now that whether the other element of about jobs to be done, which is a fantastic tool, but I think it's somewhat insufficient, because like I said, it doesn't have the causal network relationships and some of the jobs to be done work that I've seen. And the second thing is I don't think the jobs to be done has enough relationship to the software technicalities, because jobs to be done is generic. And our book is focused on software enabled solutions. So one of the things that we did, which we've never seen in any other book is we took all of the software LEDs that we talk about as product managers maintainability, reliability, scalability, and we map them into specific statements of value, both for the current solution and for the future solution. And that deals with roadmapping. And it deals with some other things that we've talked about in the past. But that notion of tying those things together is really part of the value and it's something that we feel uniquely contributing into the body of knowledge.



Jason Knight 44:22

No, absolutely. I think you've actually kind of somewhat answered my next question, actually, about the solution design as well, like how you bake profit into the solution design. And I think, a lot of what you've just touched on annuities and all of that, all of that thinking kind of answer that question, but I do think it's also fair to say that a lot of traditional thinking would be like, well, customers don't care about our profit.



L

Luke Hohmann 44:44

Well, your customer doesn't care about your profit per se, but they do care that you're able to sustain them over time. And so I love it when people say well, you know, Luke, like here's an example, mother nature would take care of itself if humans left Mother Nature alone. Well that's kind of true, right? You mother nature would take care of itself. But then the question is, but you can't leave mother nature alone completely, right? Because what if I turn off the sun? What if I turn off the sun all life nice, because sun provides the energy that starts everything for us. And so the analogy is in a business, I have to be creative, my customer doesn't care about my profit. But they do care that if I bought version three of the software, I can get version four. Or if I really love and one of my kids is a gamer, and he loves the Witcher series. He loves the books, and he loves the game. And he's like, Dad, Witcher four is coming out, and they're going to redo Witcher one, and I'm really looking forward to that I'm excited about that, it's going to be cool. Well, they can't do that if they don't have enough money to invest in their own ongoing development and solution. And I want that I kind of want the next thing.



Jason Knight 45:56

So they implicitly care about the profitability, even if they don't really care. And I'm sure that there are some companies that they would much rather prefer didn't make any profits and went bankrupt as well. But you know, that depends on the company, I guess. Yeah. But solutions lead to monetization, which is obviously closely linked to profit, or at least the mechanism through which you're going to make profit, you got to monetize your offerings. And that's something that you touched on earlier as well. He's talking about all of the elements of profit working in harmony. So what are some of those elements of profit elements of monetization that you need to think about? And how can we get them to work in harmony?

L

Luke Hohmann 46:30

Well, the first thing that we want you to focus on is the core element of the like, the most essential thing is how you trade money for value. And that's what we call the value exchange. The good news is this is that there's as near as we can tell, there's not an infinite number of value exchange, there's a finite number of value exchanges, that product managers, once they understand they can make choices about so I'll give you a couple of the most obvious ones. One is the and the most common one is what we call time based access. And that's where the software is licenced to the customer for a period of time. Now, whether or not the customer uses the software as a separate thing, right? Yep, I can buy a licence to Spotify for a month. Whether or not I listen to Spotify, in some ways, doesn't matter. Now, of course, it actually does, because I want to repeat the transaction. So I do want people listening otherwise, they won't get the value. But for time based access, it simply says you give me the money, I give you the right to use it, whether or not you do, which is different. And I'm just I'll just do a couple I can do more with you if you want me to. But to illustrate it to the listeners, couples goods, a transaction transactions are a defined and measurable unit of work. And examples of transactions would be a strike transaction, or a PayPal transaction, or a Visa transaction or an ASEAN like payment processors are easy ways to understand transactions with your Google ads. Google doesn't make money until the ad is shown or clicked on. And so transactions are a very different kind of value exchange, which leads to very different business models. If my value exchange is time based access, then my licence agreement is going to talk about time

and it's going to talk about terms. And if I'm selling enterprise software, I'm going to talk about multi term. If I'm selling subscriptions to Microsoft Game Pass, I'm going to talk about one month, but now I want to make it easy to buy more and provide more value, which is different than transactions. If I'm stripe, that might want you to have more frequent larger transactions, because that's how I make money. So I'm going to start layering in other aspects to my overall solution to help you create more frequent larger transactions. I'm also going to look at fraud, I'm going to look at other elements of support and service. If I have a transaction value exchange model, my support and my licence agreements need to reflect transactions. Whereas if it's in a term, I might be sending out an email to someone saying, Hey, your annual term is up for renewal. Are you ready to have another credit card swipe? Which wouldn't be the same with Stripe if you will?



Jason Knight 49:20

No, absolutely. And there's obviously plenty more different models in the book as well, which hopefully people can go and buy and read. But the last one you talk about is compliance, which is an interesting one, because obviously our products have to comply with laws and regulations. And you talk a lot in the book about contracting and terms and stuff. And you've kind of mentioned that tonight as well. But why have you specifically called out as part of your matrix like a kind of a top tier member? Because is compliance strictly related to profit? Or is it just kind of overhead that you just have to deal with?



Luke Hohmann 49:48

It can be both. And for the listeners when we talk about compliance we mean all sorts of elements of compliance compliance with your technology in licences having your customers As remain in compliance with your licence, so if I, Jason if I licence, and I used to work for an Israeli security firm Aladdin knowledge systems, and if you remember those dongles that would prevent software from being stolen. I was the VP of engineering and product management of building those things. And so we learned that for certain software, you do want copy protection, because when you're running on prem, etc, etc, people will steal from you. Right. So part of compliance is not just compliance with laws and regulations. But it's looking at your solution and saying, What are the terms? How do people steal? How might they steal? And how do I put in place the technologies and the business practices, so that they, in fact, do remain in compliance with my licence agreement? Because if people are stealing from you not pretty sure that you're lowering your profits? Pretty sure about that one!



Jason Knight 50:57

Yeah, no, I think you've probably got me there. All right. So you filled out the matrix, you've gone through all the different boxes you've used, whatever it is that you need in the circumstances, or the situation that you're in? What's next? Like? Is it just guaranteed success or all over? But the dancing? Or is this where the work really begins?



Luke Hohmann 51:14

Well I think the most important thing that would be next is that you would probably send me a

Well, I think the most important thing that would be next is that you would probably send me a nice bottle of Cabernet Sauvignon, or tequila, I am a tequila fan. So I think as people realise the power of this book to help them succeed in their careers, just hit me up, I'll let you know when I'm drinking. And you can send me the champagne celebration, etc. And after that, I write about this in my book innovation games. Then, because it's a game based book, I talked in that game, I didn't talk about it in this book. But in the innovation games book I talked about, are you playing a finite game or an infinite game based on the work of James cars? Yep. And a product release is a finite game. The relationship that I have with my customer is an infinite game. Now, there's a little subtlety there. Jane McGonigal, in her brilliant book, she points out that we play games until the game is boring, meaning you don't play Snakes and Ladders with your kids anymore, because it's a boring game. But you might play Scrabble because you still find it interesting. And so businesses in game theory, do the same thing. We play the business game until that business game is boring, we serve those customers until serving those customers are boring to the business. And so then we sell the division or stop the product line. So if you are playing the game well, and it's an infinite game, like marriage, if I play the marriage game, well, today, I don't, I don't have a guarantee, I get the right to play again. all I get is a ticket to play again. So you're going to do all of this stuff in the book, you're going to write up the solution lifecycle curve, and you get a chance to do it again.



Jason Knight 52:57

Well, there you go. Because it'd be boring to have to stop.



Luke Hohmann 52:59

Right!



Jason Knight 53:00

Well, excellent advice. And obviously, hopefully, people can pick up the book and find out a little bit more. When is the book coming out? Specifically, April 4, April 4, so not too long after we're recording this. So hopefully, we'll get a few people were looking at your pictures. But where can people find you after this if they want to argue with you about safe or find out more about software value streams, or maybe even see if they can get some aerobics advice?



Luke Hohmann 53:24

Sure. Hit me up on LinkedIn or just type in Luke Hohmann. And I'm easy to find. And who knows, I might tap you on the shoulder myself.



Jason Knight 53:34

Yeah, well, there you go. The threat is always there. Well, I made sure to link that all into the show notes. And hopefully I'll get a few people heading in your direction to find out a lot more. Well, that's been a fantastic chat. So obviously really appreciate you taking the time to speak

about some interesting and thought provoking topics. Obviously, we'll try and get around to to go deep on safe at some point. But yeah, that's for now. Thanks for taking the time.



Luke Hohmann 53:56

Thank you so much, Jason.



Jason Knight 54:00

As always, thanks for listening. I hope you found the episode inspiring and insightful. If you did again, I can only encourage you to hop over to <https://www.oneknightinproduct.com>, check out some of my other fantastic guests, sign up to the mailing list or subscribe on your favourite podcast app and make sure you share your friends so you and they can never miss another episode again. I'll be back soon with another inspiring guest but as for now, thanks and good night.