[THEME MUSIC "TELEBOT"]

JORDAN GREENBERG: Welcome to season 7 of the Prodcast, Google's podcast about site reliability engineering and production software. This season, we are exploring what it takes to ship software and what that even means to different folks. Join us as we discuss how to ship it as SREs. And, of course, the most important part is the friends we made along the way. Happy listening, and may all your incidents be novel.

JAVI BELTRAN: (SINGING) You missed a page from Telebot

DAVID BLANK-EDELMAN: Hello, hello. Well, welcome. I'm David Blank-Edelman, your guest host today on the Prodcast. And today, we're going to be talking about, even though the people on this call probably don't want to cop to it, the seminal work in our field, the SRE book, the thing that helped start it all out into the world, that did a good job of bringing SRE to the people. But there are some exciting news. And so as to not bury the lead, we're also going to be talking especially about the second edition, which is nigh. And so I was hoping that we could talk about various things that came up as part of both editions, and that we could dig into what we learned about SRE through the SRE book and the process of that. And so what I want to do is start off with the beginning. And Niall, so my first question is for you, which is, what were you thinking? And just in case you're concerned, I'm going to actually ask that question again in a moment for the second edition.

NIALL RICHARD MURPHY: Very good. So first edition actually arose quite a few number of years before it actually A, turned into a project and B, got published. So it's sometime around 2012 or 2013. And I think Cian Synnott and myself are sitting in some kind of meeting room where we have gone through some extremely complicated meeting where a lot of things about SRE are being discussed. And in my memory, I kind of turned to him and said, this stuff is complex enough that we should write a book. [LAUGHS] And in the very short term after that, I did a bit of poking. And it turns out that that idea, as many ideas were, was not, in fact, new. That a few people in Mountain View headquarters at Google at the time and still had been kicking around the idea of doing something like that, mostly under the heading of 97 Things That Every SRE Should Know, which then actually later, about four years ago, got published separately by people who had nothing to do with Google.

But the basic idea of encapsulating this body of knowledge and this way of working and in some metaphorical way, this way of seeing the world actually happened a few years beforehand. But it was people, some on this call, some not, who helped to turn that into a reality. And it was about 1 and 1/2, 2 years worth of effort to turn that into something real, ending up being published in early 2016.

DAVID BLANK-EDELMAN: So I know, Betsy, that you were involved in that quite a bit. What was it like to be in the middle of a maelstrom of lots and lots of people with opinions about what should and should not be in the book?

BETSY BEYER: Yeah, I mean, my experience in 1e and 2e is quite different, because 1e, I was the overall editor for the book. And for 2e, I'm managing the project as a program manager. So as an editor, there was a lot of smoothing between disparate voices, right? You kind of had to designate the technical experts on the core team to make those judgment calls when there was bikeshedding or lack of convergence.

In 2e, my role was nagging people to write for us, which is always fun. [LAUGHS] I would like to say we learned so many lessons from 1e, and there were never disparate views among authors, but of course,

that would be a lie. Again, the technical judgment of the core team was really key in that diplomacy. Yeah, we got there in the end. I will say for 2e, there were no chapters we started out with that we had to just chop because we couldn't reach consensus, which was the case for 1e. So perhaps we did learn some lessons.

DAVID BLANK-EDELMAN: Interesting. Well, it occurs to me that just for people who are not in the book world, 1e means first edition. 2e means second edition. That's what we're talking about when we say 1e and 2e. And for anybody who is confused about that, I apologize for not saying it at the front. So what happened after this book got out into the world? What sort of reaction did it have? Did you get any feedback? Anybody? Niall?

NIALL RICHARD MURPHY: Yeah, so I have a slightly funny, slightly sad story, which is that we had obviously gone to a lot of effort to try and get the original out into the world with 560-something pages, whatever it is. And I remember this so clearly the first time it had been released, and there was a day or two of it being available on e-book or paper or whatever. I was standing at the side of a soccer, what we call here football, match with my then 11-year-old or whatever he was.

And I remember the coach of the side yelling, clear it out! at the top of his voice as he was trying to encourage the 11-year-olds to behave as if they were a team and not 11 individuals, which is a thing that I thought kind of resonated with me given my most immediate experience. But I happened to come across a review of the book. I won't say who it was. I think they've deleted it since, et cetera.

But the review of the book was very bad. And it was saying, this is a terrible thing. Google is coming into this bit of the world with all of its prejudices, and it's just overriding all of this good work, lambasting it. And I kind of finished it on my phone and went, oh, well, damn it, there goes the past 2 and 1/2 years or whatever it is. It's going to be a terrible failure. And then it turned out not to be a terrible failure extremely rapidly.

So it went into bestseller status. Admittedly, Amazon counts this in a very weird way, but there's a relatively objective target on the publisher end of things, and just continued to sell and sell and sell and sell a bit more and sell some more. And O'Reilly were ecstatically happy. They said it was like book publishing used to be, in the sense that first of all, there were a lot of sales. But second of all, there was an unusual number of paper copies sold.

And we were trying to figure out the reason for this. And I think the closest we got was people who encountered the work, who were in a leadership position, would often buy paper copies of it for everyone in their org in order to transmit this vision of how things should or could be done. And so there was an unusually high ratio of paper copies. But it just became a kind of a smash hit, a bestseller success. Runaway success. Continues to sell to this day, actually, even though there are online copies of it provided for free by Google. And I'm sure that you could find a PDF copy of it if you went looking hard enough.

DAVID BLANK-EDELMAN: So my question for you, Christof, is why? Why do you think that everybody really dug the first edition? Like, what was in it as far as you're concerned, that gave it value to so many people?

CHRISTOF LENG: I think there was just like the time when a lot of companies went so much bigger in their digital side of things. It became instead of this thing that happens in the basement, critical to the business. And people realized that when the systems are down, that is actually impacting the business. And they needed to figure out, how can we do that? How can we do this in with the new scale of

systems? How can we do this with the complexity of all of the systems interacting? And how can we do this sustainably? And this was a very concrete approach on how to do this. So people tried it out, and it seemed it worked for quite a few companies.

DAVID BLANK-EDELMAN: And you think people saw themselves not just Google in that book, presumably, right? That's been a tricky question, right? Like, this was an explanation of how Google succeeded, but somehow Google had found a recipe or something that might be further useful outside in the world. At that time, did any of you folks know that this would be applicable beyond Google? Presumably you wrote it with that in mind, right? It wasn't a historical document saying, hey, cool, here's what Google did. Please applaud us. How did other people be able to take it into their org? Like, did you hear stories from other people that said, oh, we sat down and discussed it inside, and now we're doing everything exactly the way Google does it? Or what was your sense of how people took it into their own org, just like you're talking, Christof?

CHRISTOF LENG: I think that there are parts in the first edition that are very Google-specific on like, I don't know, how to implement your own load balancer. Very few companies do that. But I think people, kind of like having a complete view into how a company like Google does that, and Google seems to do that relatively successfully.

And I think the people that were most successful with adopting SRE in their space made it their own thing, instead of trying to copy Google exactly, because every organization is individual. Every organization has its own culture, its own needs. And no organization is like Google. And that's good. And I think a lot of the principles in the book were general enough that people can apply them to their context.

DAVID BLANK-EDELMAN: Cool. I see a lot of hands up. So we're going to do this in alphabetical order, starting with or at least the order in my screen. David, do you want to go ahead and respond?

DAVID HUSKA: Sure. It's also my introduction and reason for being a part of this book. When the first edition was released, I was working in an area of cloud SRE that was helping our largest customers come on to Google Cloud, onto GCP. It wasn't a new thing at the time, but it was definitely gaining in popularity. And we learned the hard way that if customers are shifting their very large, very old systems and bringing with them reliability issues, those don't disappear automatically when they come into the cloud.

And this was right at the time that the first edition was becoming extremely popular. And so it kept coming up over and over again. People and companies were trying to use it as a playbook. It's like, this is the way Google does it. This is the way we're going to do it. And that didn't work for a lot of reasons, like Christof pointed out. And that's one of the things that we address in the second book.

The next few years were a lot of really interesting lessons with these large companies behind closed doors. Part technical consulting, part grief counseling, part, why isn't this working for us? But it was also very validating. We saw that the problems that we described there just sort of laws of physics. You're bumping up against the same issues, and they are having them as well. And they're really interested in how Google solved them. And Google has held up often, especially in the area of reliability, as an exemplar. And they wanted to follow these patterns.

Many of them worked and still are working. But what we learned hearing directly from those customers, we fed into the second edition. And you see that a lot. We'll probably get to some of those topics, but that really did shape what that second edition looked like, based on exactly what we were seeing. It says pretty clear that first edition was how Google runs production. But getting in there and seeing how that looked when applied to other customers was really informative. These are really smart engineers who

know their business well. And the fit wasn't always perfect. And we looked at that and said, well, what can we change?

DAVID BLANK-EDELMAN: Niall, and then I want to follow up on something you said, David.

NIALL RICHARD MURPHY: Yeah, super quickly. The first moment when I realized like this might be more generally applicable was I was sitting in the Dublin office some months after the thing had been published, and I got a notification that there was a visitor at the front desk for me. And I'm like, oh, I wasn't expecting anyone. So I go down, and it turns out that it's six people from a consulting firm.

And I bring them into a meeting room, and they all get their folders out. And they have lots of documents with bullet points, and they ask me extremely detailed questions for this hour or so. And at the end of the hour, I'm less-- well, right now, I'm less naive than I was then 10 years ago. But it suddenly dawns on me that they are there to interrogate me about the nuances of the book, so that they can then go and sell that to their consulting clients.

And when they did that, I figured out that, oh, they must be going to do this to a much broader range of companies across a whole bunch of sectors and with different orientations towards technology, and so on and so forth. And I think then a bit after that, I realize, OK, this is going mainstream. So there is, of course, a long discussion to be had about the size of your organization, the number of resources it has, I mean, actual money and CPU, and so on, rather than people.

All of these kinds of variations with respect to Google itself that you might think make it inapplicable, but it's actually when the consulting companies go, no, there's a ton of these that are immediately reusable in your organization. I think the title of the Forrester article is something like, 19 takeaways from the SRE book that are immediately usable in your org, or something like that. I'm like, OK, this makes sense for other folks. Now, I think there is a case to be made that startups are a kind of inherently different organization, which I understand, and almost even partially believe.

But I think in terms of applicability to general companies and the jobs they're doing, and so on and so forth, there's a lot that's recyclable. And actually, that was part of the motivation for 1e that I was sitting there steeped in the old-school kind of development and operations hard split. And I had sat for my career like 75% in operations, 25% in development. And I went, it doesn't have to be this way. And one of the things that Google was so, so good at was rethinking things from first principles and coming up with a new way of addressing it. So anyway, that's my spiel.

DAVID BLANK-EDELMAN: David, I wanted to follow up on something you said as a way working our way towards the second edition. I am curious now that you have a second edition on the way, and obviously, there was a lot of thought that went into it, what did the field or the authors get wrong, or suboptimal, or didn't quite get right in the first edition that you would think that you're hoping to put into the second edition?

I mean, presumably we're all-- I don't know, so I'm personally a flawed human, but let's just assume that other people have similar-- nope, nope, not you. No. But my question is, we SREs are really big about learning from failure. So what did we learn? What did we learn? What's in the first edition if you could wave your wand and go back that you think, oh, the second edition is attempting to fill that gap or to correct something on the record or anything like that?

DAVID HUSKA: There aren't too many things I would say were outright wrong. There are some things that were accurate at the time or the viewpoint that we were writing from, which was how Google runs production systems, that today have moved on, are dated, and there's plenty of things that fit in that

category. We also made-- because the audience has changed a lot in this second edition. So the first one was really heavily oriented towards technical practitioners, mostly SREs or people who are doing the equivalent of that in their organization.

And so we really spoke to them how to implement many of these things. In fact, that's right off the bat in the second edition, we talk a lot about platform engineering. At the time 10 years ago, we talked a lot, or it was SRE culture, more or less teams, developers built their own infrastructure and ran it. And we've shifted much of that from left to down into the platform. And so that's one example. We didn't get it wrong. That's how we ran things. It was very successful at Google. But we also noticed that many organizations started to copy us because of that book.

So there's probably maybe the way I would answer your question is, what trends did we spawned that I wish we could take back and have some of the people outside of Google do differently? Because it worked well at Google, but maybe not well outside. There's a few things that fall in that category. Well, it was a simpler time. We had plenty of like N+1 idle compute sitting around just to fall failover to.

We had a monoculture of VMs which, again, pioneered things like, well, Borg, Kubernetes, et cetera. So I'm glad we did it. It just has moved on somewhat from that. Meantime to resolution was a nice workable metric following a nice bell curve. [LAUGHS] We followed it. Not as relevant today. There are a lot of things that we could go into.

Probably the biggest thing, though, that I'm glad this edition addresses is that we've really seen that other organizations do these things very well and have come up with very good solutions. And we've tried hard to incorporate them into this book from actually having them write what they've done, but also showcasing it internally, how we've adopted and how we've changed, as well as writing to different audiences that weren't present in the first book.

DAVID BLANK-EDELMAN: Cool. Betsy?

BETSY BEYER: One of the things we realized was pretty much missing in the first book was Google was operating under this assumption that, oh yeah, we have Google culture. These are the foundations for what you need in place with respect to your people and management and blamelessness and all this kind of stuff that you need as a background in order to have a successful SRE implementation. So you will see in 2e, we do spend a lot more time on sociotechnical details. And those are going to be a lot more evergreen than what details about table or something like that.

DAVID BLANK-EDELMAN: OK, well, that's another similar question I have. In my experience, when one does a second edition, one doesn't get an extra book worth of pages, right? And so some stuff has to leave, and some stuff has to join. What sort of stuff-- I now understand-- thanks, David-- that platform engineering, and from you, Betsy, I hear about the technical stuff. What sort of stuff leaves to make room in the second edition for that new material?

DAVID HUSKA: There was a crap ton of, very interesting for the time, technical descriptions, load balancing, distributed cron, yada, yada, yada, most of which has been commoditized now. Or if it's not commoditized, then it's certainly much cheaper to do via LLMs or similar generation than it was previously. And as a result, we debated the value of keeping some of that in, but ultimately decided that actually the things which were mostly missing in 1e were the cultural descriptions of how humans interact in teams, individually, how you can justify the value of the work you're doing, and so on and so forth. Which were the water we swam in, to reuse, I think it's David Foster Wallace's phrase, and documenting that which we had partially left out of 1e because we couldn't see it. But in some cases, we could see it.

And we had written it down, but it had to be removed for length management and other reasons. So we took out a lot of the tech stuff and put in a lot more of the organizational stuff.
DAVID BLANK-EDELMAN: OK. Christof, I actually have a question for you also, but I see your hands up. So why don't you answer the direct question? And then I have a follow-up for you, and we can make sure to get Betsy in as well. Go for it.
CHRISTOF LENG: Yeah, I think 1e described very true to the subtitle how Google runs production, but it didn't really explicitly say that this is just one way of doing it, and there are many others. So in the second edition, we tried to bring in many more diverse perspectives from other companies, like Spotify, Microsoft, Etsy, Disney, Waymo, that have their own approach to SRE, their own journey to SRE and can tell stories on how they adopted it, what went well for them, what didn't. And I think that it's very useful when you want to start your own SRE journey, instead of just looking at what Google already has and thus not explaining how we got there.
DAVID BLANK-EDELMAN: Betsy, you had your hand up. Do you still want to speak?
BETSY BEYER: Yes, and I do want to point out one thing about 2e, which is really important to get on the table that in terms of what we kept in, left out, updated, 2e is a complete rewriting for which we did not revisit 1e and lift and shift to 2e. So, for example, 1e has a chapter on simplicity. Yes, that is a basic tenet of SRE that we still stand by. And that chapter is still very much valid. So we really see 2e as a companion to 1e versus a replacement.
So no one needs to go out and garbage their 1e. I think it's still very valuable to the point where in the beginning of the book, for those people who actually read the preface, it does give a reader guide for different audiences. One of those audiences is the aspiring SRE, and we say, start with 1e.
DAVID BLANK-EDELMAN: OK. Well, I was going to ask, do you do anything heretical? Do you say, yeah, these SLIs and SLOs, they were cool once upon a time, but not so much? Or reliability is cool, but let me tell you about resilience? Or is there any heresy to be found in the 2e that we should be looking for that if the 1e people were talking to the 2e people, they'd be like, what? David?
DAVID HUSKA: [LAUGHS] Heresy, dogma, that comes up a lot around SLOs, error budgets. That was probably the runaway success when the first edition. And working with cloud customers, it was probably the one we heard about the most. We use the error budgets, SLOs, et cetera as a cooling mechanism. If you're using them, it's probably because things are not going well. And so it keeps you plan when to freeze, when to focus on reliability, and when to move forward with feature releases, et cetera.
And it was a good plan to follow if everyone agreed to it in advance so that when things aren't going well, you already have a plan in place. And those are still valid and true. It still remains a good guidance principle. And a lot of people who were using it felt that it removed agency from them to make decisions about the risk trade-offs within their own organizations and say, well, OK, we can use this, but we don't have to adopt it wholesale.
And I think what a lot of customers converged on, a lot of readers, not just GCP customers, is that with transparency within the organization and knowing their own risk trade-offs, they are able to make very good decisions about what to do about SLOs and what to do about error budgets and manage their own business accordingly. The biggest value I think that remains from the first edition of the book, and Niall did a really good job of highlighting it in this book-- I see now, smiling. Probably we had a lot-- SLOs brings a lot of opinions.

So we had a lot of people offer opinions on SLOs for this book. And Niall was tasked with unifying that into one voice, and did an excellent job of that. The thing that we still would hammer on with SLOs is as a mechanism for surfacing what your users actually experience and what they care about. And I don't think that's changed at all, to Betsy's point, about things that just are invariant. What you do with that, error budgets, freezes, et cetera, yeah, we start getting into dogma and heresy. And we've changed our own views on that as well. And actually, that's thanks to feedback and seeing how other people adopt them.
DAVID BLANK-EDELMAN: Appreciate that. Niall, you have the floor. And then I have two remaining things in our call.
NIALL RICHARD MURPHY: Yeah, super quickly, I think the SLOs chapter in 2e is unique in that it actually contains a section saying weaknesses of SLOs, or a subsection saying weaknesses of SLOs. But more broadly, I would say 2e is aware, however we put that, of a wider trade-off space in many of the chapters. And so it's a bit more nuanced, more detailed, et cetera, about describing how you would adopt various things.
I will also say, which might derail from the things you're about to just say, but I also feel compelled to mention that you're talking about heresy. But also controversy is another kind of emotional stimulant. And of course, we must talk about our favorite two vowels in the course of this book and--
DAVID BLANK-EDELMAN: Oh, and that's the second thing I want to talk about. So, yeah, I'm ready for you.
NIALL RICHARD MURPHY: Well, I'm just going to say that we have a chapter on AI for SRE, and we've a chapter on SRE for AI. And those are very different topics. I'll leave it there. [LAUGHS]
DAVID BLANK-EDELMAN: Fabulous. Well, OK, so we will definitely come to that as the last future conversation. But I would like to do a quick lightning round starting off with Christof. I would love to hear what your favorite chapter is that you did not write in the book that you recommend people read, and maybe a sentence or two about why.
CHRISTOF LENG: I think personally, for me, it's the cultural context of SRE, because it provides a lot of background on how you can be successful in the SRE space and how you can fail spectacularly without even touching the technology yet. So this is, I think, often overlooked. And if you completely ignore this space, then you run into a lot of trouble and might not even understand why that is. So I think it's important that we discussed this in depth and put it very early in the book.
DAVID BLANK-EDELMAN: Fabulous. Betsy. Hard problem, because you have your finger in all the chapters. Got at least a recommendation if not a favorite?
BETSY BEYER: I do-- one that's near and dear to my heart is automation and tooling. I think it does a really good job between giving a very progressive view for anyone. Like, start with A, then go to B, then go to C. So there's a jumping endpoint for any of our audiences. I think that's valuable. And then it also has a really lengthy description and discussion of toil, which I think is one of those subjects near and dear to an SRE's heart. And that chapter does a particularly good job of getting a pretty nuanced dive into that.
DAVID BLANK-EDELMAN: Fabulous. David, what would you say?
DAVID HUSKA: I think I probably already answered it. Favorite that I did not work on, I'll probably say platform engineering, because it really demonstrates that evolution of what we recommended in the first edition, and then how we address it internally at Google and recommend externally. It bridges software engineering and site reliability engineering in the way it was always intended and really demonstrates that

these principles don't go away. It's just how we deal with them. And that's a running theme through everything.

Like, the principles are still there from when it's just the tools we have at our disposal, the scale we're dealing with, and the business environment has changed. And so how we apply these has necessarily changed as well. If I'm allowed to also get in a recommendation, I heard you try and let out of Betsy, that'd be one that I did work on. That would be valuable reliability. And it's an example of something, a new approach we took in this book, but it's good for us to be reminded as SREs that at the end of the day, SRE is a business function.

It's very easy to consider it purely an engineering discipline, but it is there to help organizations scale sublinearly to costs so that they can remain in business and grow. And value reliability covers that, and it explains why that needs to be there. If this is just trendy, if it's something that people are excited about, but it's still viewed as an operational cost center rather than a competitive advantage, then the next time this comes up for debate, whether that's a budget cycle or whatever it is, your SRE program is at risk. But if everyone's on board and understanding why it's there, then you can start from that common place and understanding. So that highlights something new in the book as well that I like a lot.

DAVID BLANK-EDELMAN: Cool. Niall, what would you say?

NIALL RICHARD MURPHY: You must understand I cannot pick a favorite child, David, and all such efforts are doomed to failure. But I do want to particularly highlight Christof's work on generating a successor to the Dickerson hierarchy, which turns out in various ways not to be-- It has some value to it, but it's not as useful as it could be these days with respect to understanding the relative prioritization of things that you might do in production and how they relate to each other, and so on. And I think Christof did a stellar job of coming up with a new hierarchy, which is one of the things that I most regard with favor.

DAVID BLANK-EDELMAN: I appreciate you saying that. OK, well, let's end our conversation by talking a little bit about our new robot overlords. I am curious about a lot of things in the land of AI. I'm very curious about how you all were able to write about AI when in my experience, writing about AI is like writing in disappearing ink. Because as soon as you've written it down in just a few moments, that which is true is gone, right? And we've moved on in some way, shape, or fashion. So I know that every SRE is probably thinking about AI in some ways. What could you possibly put down on a dead tree that you thought would hold for at least the duration of the readers reading it? How did you all take on this question?

BETSY BEYER: I will say, I think we did a very good job of teeing up and easing into the topic with the tipping point chapter that really addresses the elephant in the room, which is, how is this going to affect the SRE role? And there's a lot of anxiety and uncertainty around that. I mean, I can't tell you that we definitively answer what it's going to do for SRE, but I think we make some pretty darn good guesses.

DAVID BLANK-EDELMAN: OK. David.

DAVID HUSKA: There were things we were able to say with certainty. We had already, and we are now, running AI as SREs. We've been doing this for a while with machine learning, so there were some things we could talk about in principal terms. So fortunately, that gave us a bit of a rock to stand on. You did hit, I think for me, the greatest source of angst in this book, which is just like the last book was around for 10 years. So what can we say on the subject that 10 years from now won't seem ridiculous when week by week everything is changing? And we really did go back and forth on that a lot.

I think the end result, and I agree with Betsy, under the circumstances, we did a great job, was to again, return to first principles, things we know are true and the things that we can talk about, and how those

remain true even in the context of AI. So I guess that's a bit of a teaser for the book. You'll have to get it to see how we went about that. And generally, I think it was tricky. But I'm happy also with what we came up with.

DAVID BLANK-EDELMAN: So I'm following the order of hand raises that I've seen. Niall first and then Christof, please.

NIALL RICHARD MURPHY: Yeah, I'm going to send a note of caution [LAUGHS] slash other things here. I am somewhat more deeply day to day in the AI revolution than perhaps some of the other folks on this call. And as a result, I look at, for example, this famous Meta matter graph, which is used as a proxy for the complexity of the tasks that AI systems can complete, and that is going literally like that. I denote that what we say is going to be relevant like today, never mind data publication, never mind when people are going to read it.

But I can say that we went through a lot of iterations of this in the team deciding what precise angles to hit on. And we did try a little bit like 1e tried, was to emphasize principles and things which might well have more continuity or more of an extended life than talking about HTML5 for dummies or whatever the hugely specific term or version of things that people are working with. That wasn't our intention.

I will also say that one of the ways that we struggle with correctly capturing the contribution of AI to any particular vertical SRE or software or whatever, is, of course, the rate of change. But it also is this underpinning fact of the LLM architecture, which is that it hallucinates, invents things, can get distracted. It's in some ways a superhuman cognitive partner, and in some ways just an entirely inadequate kind of cognitive partner. And I think trying to figure out how to deal with that, particularly in the context of production stuff, where it actually matters quite a lot whether you are 99.9% accurate at doing this or 99.5% accurate.

And again, we have tried to write down some guidelines here. I will say here is a prediction for you. I don't think that we exit this year without having to have every SRE in the world have an informed view of this, if not necessarily a engagement with AI tools. I think we don't exit the decade without knowing how to control this stuff.

DAVID BLANK-EDELMAN: Christof, you get to go after that very-- I could hear the low brass thrumming in the background. But why don't you go forth with what you wanted to say about this question?

CHRISTOF LENG: AI is changing our industry very fast, but the tech industry is always changing. That is what makes it so exciting. So for no topic in that space, it's a good time to publish a book at any point of time. But you still need to talk about these things. Like, writing a book, and talking to people that have read the book and learning together, I think is really the important part here.

And the things that we have focused on in the book is trying to find core things that we already know that are true. And everybody is right now trying to find out how we use AI in production. And nobody knows what this will look like in 2, 5, or 10 years. But a part of the book talks about how you can safely adopt AI in production, how you do this step by step in a way that you build confidence, that you minimize the risks. And I think that is a very SRE way to do it.

DAVID BLANK-EDELMAN: Well, I'm hoping in our second conversation, we can get into further topics about this. I'm personally interested in, for example, for all the years that I've been in the operations business, we've sort of tried to work with largely deterministic things to get a deterministic outcomes. And now all of a sudden, we have a tool that doesn't really favor determinism. And so it will be very interesting to continue our conversation.

And so I'd like to thank you all for the first part of a two-part conversation. We'll have some folks returning and some new folks joining us. Thank you all for talking a little bit about this upcoming fabulous thing called the second edition.
BETSY BEYER: Thanks, David.
DAVID HUSKA: Thanks, David. Appreciate it.
CHRISTOF LENG: Thank you, David.
JORDAN GREENBERG: You've been listening to the Prodcast, Google's podcast, on site reliability engineering and production software. Visit us on the web at sre.google, where you can find books, papers, workshops, videos, and more about SRE. This season is brought to you by hosts Jordan Greenberg, Steve McGhee, and Matt Siegler. The podcast is produced by Paul Guglielmino and Salim Virji, with contributions from many SREs behind the scenes. The Prodcast theme is Telebot by Javi Beltran and Jordan Greenberg.
JAVI BELTRAN: (SINGING) You missed a page from Telebot