The Unicorn Project with Gene Kim

a transcription of the DevSecOps Podcast, recorded 16 October 2019

Mark Miller: Edwards Deming went to post-war Japan in the late 1940s to help with the census. While there, he built the relationships with some of the main manufacturers in the region helping them understand the value of building quality into a product as part of the production process, thus lowering time to market, eliminating rework, and saving company resources.

In his 1982 book, Out of the Crisis, Deming explained in detail why Japan was ahead of the American manufacturing industry and what to do about it. His 14 points on quality management helped revitalize American industry. Unknowingly, he laid the foundation for DevOps 40 years later. Eli Goldratt published The Goal in 1984 focusing on the theory of constraints, the idea that a process can only go as fast as its slowest part.

In fictionalized novel form, Goldratt was able to reach a wide audience who would utilize the theory to help find bottlenecks or constraints within production that were holding back the entire system. Once again, the theories espoused in the goal were a precursor to the DevOps movement 40 years later. In January 2013, 40 years after Deming and Goldratt reshaped the manufacturing processes in America, Gene Kim published The Phoenix Project.

He used the same format as Goldratt telling the story in a fictionalized novel format with characters who were easily identifiable within the software manufacturing process from a manager's point of view. The Phoenix Project is now one of the most important books in the industry and is used as a starting point for companies interested in participating in a DevOps transformation. It's now six years later, 2019. Gene's new book, The Unicorn Project, will be released at the upcoming DevOps Enterprise Summit in Las Vegas on October 28th.

This new book has an interesting premise. What was going on with the software development team in The Phoenix Project as the management team was flailing to get the project back on track? It's a novel approach to have parallel timelines in separate books looking at the same project. In this broadcast, Gene and I talk about how The Unicorn Project aligns with The Phoenix Project, the overlap in the storylines, and why he chose to speak for software developers in this iteration of the story. Do a quick review of The Phoenix Project, which is probably already on your bookshelf, and then listen in as we discuss using Deming, Goldratt, and Kim as the foundation for the principles of the DevOps movement.

Mark Miller: You and I were talking a little bit about your new book, The Unicorn Project. One of the things I liked about it and I think other people will appreciate is the parallel timeline you were able to establish between The Phoenix Project and The Unicorn Project.

Gene Kim: That narrative tool has been used for I think most famously in Intershadow, written from a different perspective as the protagonist of the Ender's Game, but I think the real motivation to write The Unicorn Project was this sense that these incredible problems still remain. In other words, someone could embrace fully all the principles and patterns espoused in The Phoenix Project. Lessons are being modeled by the DevOps enterprise community, but I think one of the problems is that there's still all these ... maybe a lack of recognition of all the invisible structures required to make developers productive.

That's the crushing weight of technical debt that in most large complex organizations has been building up for decades. I think there's also this other problem that I think even when leadership really wants to fully support it, what exact support is needed by leadership is still ambiguous. I think there's this third thing that for me was very eye-opening and something I wanted to explore in The Unicorn Project was just how DevOps identified this big problem of trying to get code to where it needed to be which is specifically in production so that customers are getting value.

There's this other parallel universe, this kind of orthogonal problem set, but it's not code. It's the data, so all this data about our customers that are locked up into these systems of record where it's just impossible to get where it needs to go which is in the hands of the frontline developers to use in their daily work. It's often trapped in data warehouses. It's trapped in these very precarious ETLs, so I wanted to help shine a light on that and just show how technologists can solve that problem just like we did for DevOps and code.

Mark Miller: When you said Maxine is the main character in the book, did you pattern her after anybody? Did you have a model in mind?

Gene Kim: In fact, the book is very much modeled after the DevOps enterprise community, all of those speakers that really show the heroic characteristics it takes to take on a very powerful, ancient order that can protect its own interests, whether it's a functional silo or whether it's dev or ops or project management or security, but also, it's modeled after some of the people who have influenced me most in the functional programming community.

You have people like Mike Nygard who just have a clear way to view things, so I think the book, The Unicorn Project, is very much aimed at the developer whose productivity, all of our hopes, dreams, and aspirations hinge upon, so yeah.

There's a very large word cloud of people that Maxine, the hero, is modeled after for sure.

Mark Miller: When you're looking at this to-

Gene Kim: Oh, and by the way, I mentioned the Mike Nygard. I'm also thinking about Cornelia Davis, Scott Havens, all these people who have taught me a better way to think when it comes to programming.

Mark Miller: It’s finding those type of people that resonate with youthat makes it work, right?

Gene Kim: Absolutely.

Mark Miller: When we're talking about the characters now, for people that read The Phoenix Project, they were used to thinking of the analogy of the manufacturing plant, as was set up with The Ggoal. You've actually changed that now that it's the developer mindset, and you really actually get into the weeds on the developer side of it.

Gene Kim: You say in the weeds, and it's funny you use those words because it's something I did a lot of intense, I would say, soul searching or a lot of thinking about who is this book for. I think The Phoenix Project, really the audience was the technology leader, and maybe if we did a really good job, that book would be passed on to that person's boss, whether it's the CFO or the business leader.

As much as I love that audience in The Unicorn Project, especially even in the late stages of editing, it was like, "Who is this book for? Is it for the developer or is it for that technology leader?” When I say technology leader, in the extreme, it's the person ultimately accountable for all the technology functions, whether it's the CIO or CTO, whatever. Two and a half weeks before the deadline when the book had to be sent to print, I finally decided, all right, it is really for the developer. Again, just on the hopes that if the book is good enough, it will get passed to the technology leaders and maybe even to the technology leader's boss, but that freedom really did afford me to double down on functional programming, which I think is truly a better way to think and talk about architectures.

When we talk about the invisible structures that make developers either incredibly productive or incredibly unproductive, really, that's architecture. It's the invisible structures that allow developers to do their daily work and to be able to talk in more detail about what are those things that enable fast feedback, flow, and joy. What does architecture look like in the small and the very large?

It was actually a decision that I was very reticent to fully commit to, but one that was ultimately ... I think allowed me to jump more fully into, as you say, the weeds, but something I think is very, very important that ultimately every technology leaders needs to know about.

Mark Miller: It is. I agree with you there. One of the unifying themes between The Phoenix and The Unicorn Projects is still using the theory of constraints and bottlenecks as one of the showstoppers for the process.

Gene Kim: For sure. In fact, I'm a big fan of the Kurt Vonnegut shape of stories. There's only a finite number of shapes of stories. You have the hero's journey where something terrible happens and then the hero has to climb out and seek enlightenment and wisdom. There's a tragic ending. There's a certain number of shapes, and someone in the review process, a bunch of people noted that, wow, a lot of answers come very easy to our hero, Maxine, in The Unicorn Project, whereas in The Phoenix Project, Bill, our hero, had to solve every puzzle.

I think it's because it is a different shape of story. I think the role of Maxine is to be able to see the difference between awesome, like transcendently awesome, and the horrible, bad awfulness, so her job is to witness that and be able to appreciate the good and recoil from the bad. She doesn't need any puzzling out to figure that out, so certain things I think become very obvious.

Gene Kim: One of them is the notion of the constraint moving. She has a very intuitive feel that, hey, it starts often with deployment and then testing and then architecture, and then ultimately it is about how many developers and good ideas can we come up with to achieve all the goals and aspirations of the organizations that we serve. So yes, you're right. Theory of Constraint definitely guides the story along, but I think unlike Bill, who had to really search for where the constraint was, I think Maxine brings with her a decade of experience that just helps her come into it where it is.

Mark Miller: I just wrote down that note about Maxine as you were talking, that she has the intuitive sense because she has the experience to back it up.

Gene Kim: Yeah. Yeah, yeah. Modeling the book in my head for almost three years now, it's always been the story of rebellion just like the DevOps enterprise community, these people taking on the ancient powerful order. But there's a couple other things in my head that I wanted to model the book after, so it's the rebellion in Star Wars. It's red shirts from Star Trek. It's people doing the work, not about the bridge crew, but people doing the work. Hogan's Heroes, and also the movie, Brazil, where the number one fugitive in the movie, Brazil, is the rogue air conditioner repairman who breaks into people's apartments and fixes their air condition because central services won't do it for them.

It's the way the book was shaped. Maxine has experience and intuition, and it isn't until she meets the rebellion led by Kurt where that's the magical combination it takes to make these things happen. You have the person who knows how, and then you have the person who is politically savvy enough to figure out how to either make the right friends or just simply hide the effort.

Mark Miller: Oh, there has to be both sides to that or else it can't work.

Gene Kim: Yeah, exactly. I love that sort of duality, right?

Mark Miller: Mm-hmm (affirmative).

Gene Kim: Oh, and for your amusement, I think Maxine is the lawful good, and Kurt is certainly the chaotic good, right, willing to bend the rules-

Mark Miller: Yeah.

Gene Kim: ... to make help make the right things happen.

Mark Miller: I think it's interesting, too, that you placed a woman as the central character of the book knowing the problems that we're having in the tech industry in general. You have several strong women characters in here.

Gene Kim: I don't know when I decided that, but I think it was around 2015 where there was this figure that happened in The Phoenix Project timeline, the person who allegedly caused the payroll outage. It was a software developer named Max. I think it was only one or two sentences in it that was alluded to. They're trying to figure out what caused the payroll outage. I just thought it would be great to have the main character be Maxine and actually have it be a well-intended change. Essentially, Maxine's being set up as the fall guy, the person who's being-

Mark Miller: I was just going to be bring that up. The opening of the book is she gets her head handed to her on a plate!

Gene Kim: That line, it actually came from Dr. Steven Spear. He said that. I put it right into Maxine's mouth. He was saying this was caused by a technology problem and a human error. “Are you saying that I'm the human error?”, when it was actually the co-creation of Chris, the VP of engineering, and Maxine.

Mark Miller: I actually got angry for her when I was reading that.

Gene Kim: Right. In fact, so did a lot of the reviews. They were like, "Why isn't she immediately quitting or shopping her resume around?" Some people had significant mental dissonance, so I had to actually add that just to give people a

way to better relate to Maxine. In The Unicorn Project, there were these, we're calling it, the five ideals. Just like The Phoenix Project had the three types of work, I'm sorry, the three ways, the four types of work, the five ideals are really meant to be the construct to elevate these things, and one of them is psychological safety.

The fourth ideal is psychological safety and really just showing the contrast. Again, everything is clear when we contrast it, too. We know what DevOps cultures look like. They're generative. We seek ideas. We seek novelty. We train messengers to tell bad news, and essentially that opening scene in the book is like the opposite that. It's how everyone's afraid to tell bad news. Everyone is afraid of being blamed and even the evolving into the people getting fired for things going wrong. Who wants to propose a new idea when you can get called out and essentially be blamed?

Mark Miller: Or when you're on vacation.

Gene Kim: Yeah, so hopefully it does evoke that sense of feeling of being genuinely unjust. I'm hoping that for people who are in environments like that, that maybe this book can be a way to, in a disarming way or in a safe way, to be able to say, "Man, isn't that terrible?" and huh, maybe an internal book club can be able to say, "Is there anything about the book that resonates with our daily experience?"

Mark Miller: Your Dwayne character who is on part of the project, as a joke says, "Hey, let's call this The Unicorn Project." Was it a joke when you first started? Is that where that came from?

Gene Kim: That scene was also alluded to in The Phoenix Project where I think Bill was despairing at the fact that developers were so whimsical and the project names were so different than what I think most people in corporate IT are used to. It was fun to have as Maggie, essentially the business leader with a vision to sponsor and fully protect this initiative, say, "We need a name for this project." I love the whole sort of whimsical names that come out of like Narwhal and Unikitty, Unicorn, and so forth.

By the way, you mentioned there are a lot of strong female characters, and I think that also is my attempt to pay tribute to the DevOps enterprise community. Take a look at these people leading these transformations that are ... I think it's impossible to overstate the value that they create and the amount of personal jeopardy they put themselves in. A lot of them are led by women, almost half. I think that was really inspired by just seeing all the experience reports coming out of the DevOps enterprise community, but the number one villain is still Sarah.

Mark Miller: I was going there right now. Sarah is still in the mix.

Gene Kim: Yeah, so Jeffery Snover, a technical developer at Microsoft, he gave me this line that's like, "Yeah, a chainsaw's great at cutting down trees and also potentially cutting off limbs." That is Sarah. Very strong woman.

Mark Miller: I actually made a note to myself, Sarah is still scowling and she is back in spades.

Gene Kim: Right, with powerful allies. Yeah, I love that one of reviewer's comments, it was actually one of my favorites, was, "Just when I thought I couldn't despise Sarah more." I just thought that was one of the best compliments given to me during the review process. One of the things I'm really excited to do is actually work with Elisabeth Hendrickson, who's VP of engineering at Pivotal. She brought up this fascinating notion of describing in more detail how we communicate with people like Sarah and what did technology do so bad to her that made her the way she is.

We just have this wonderful discussion just trying to imagine what her bookshelf looked like and what was her career arc and where did she get burned so badly, and what causes her to behave the way she does.

Mark Miller: Wouldn't it be fascinating to have a third book from Sarah's point of view?

Gene Kim: You're not the first person that mentioned that, and I'm actually hoping to do a podcast just to fully explore that. Actually, one of the promises I did make was to actually write a scene from Sarah's perspective, which I'm not proud of this at all, but Elisabeth Hendrickson, one of my mentors and people I idolize, she said, "Tell me more about Sarah's background," and I laughed because I've written the background and resume of almost every one of the primary characters in The Phoenix Project and The Unicorn Project, but I never made one for Sarah. I was thunderstruck. I was gobsmacked, and it just showed to me she was just a very convenient villain based on someone I know.

Mark Miller: You had to think that through in order to develop that character. What is the background?

Gene Kim: Thanks to Elisabeth's goading, I think she's a brilliant merchandiser. I think she knows intuitively what's wrong with the Parts Unlimited business. She knows what is going wrong in terms of the presentation of goods within the store. Intuitively, I think she understand how the strategy needs to change, but I think she must've been burned by a lot of technology initiatives before. I also think she spent a lot of times in mergers and acquisitions which I think encourages a certain mindset and behavior and a belief that people are truly fungible and replaceable.

We even went to brainstorm about what's on her bookshelf, and I think the number one book is Who Moved My Cheese? She likes strategy books, but I wouldn't call her a good people leader as we think of them now.

She is ruthless and great at accountability. She wants to hold people accountable.

Mark Miller: I was still happy to see Brent there, even though he's a minimal part now.

I actually use that analogy from The Phoenix Project a lot. Who is your Brent? Everybody knows what I'm talking about. It's that one old timer who knows everything. He knows where all the bodies are buried. Everybody goes to him.

Gene Kim: In fact, my reaction to “is Brent a minimal character?”, I don't think so. He's one of the core members of the rebellion, and it's so clear that the rebellion is actually trying to protect Brent and try to give Brent some semblance of his life back. For me, it was really fun to be able to speculate and explore what are Brent's goals, dreams, ambitions, and aspirations. That was actually really fun to play out and it was also fun to paint more these battle scenes with Brent front and center.

One of my favorite scenes is in the Phoenix rollout, the beginning of the release that was absolutely the catastrophe that it was, having not only the database issue and migration go wrong, but also having all the prices disappear from the e-commerce site and the mobile app due to a bad upload.

Mark Miller: The visual thing that you did with Brent in that scene, though, was to have him staring off into space knowing that his mind is just going a million miles an hour and nobody knows what he's thinking.

Gene Kim: Exactly right. I think something I would've liked to explore more, but we just ran into word count problems, was what Maxine sees in that. I think what she's most offended by in that mini-episode was the fact that this bad data upload problem has happened before. It's happened many times and I think just deeply offends her sensibilities, and I think she says, "We're going to make sure that this doesn't happen again. We're going to catch malformed CSV files before they get into our production database. We're going to catch when the wrong number of rows are there." Just so you know, that scene is actually modeled after something Dr. Nicole Forsgren did to me when she handed me this CVS file with a byte order mark.

This was many years ago in the State of DevOps Report where none of the fields looked right because the first field had this, whatever the byte order mark character is, so that took four hours for me to figure out why are the column markers not lining up as they should be, so based on a real story.

Mark Miller: Well, the real story there, too, is wasn't it the configuration file was missing the correct database connection?

Gene Kim: Oh, that's another one. That's one of the things I really want to highlight. We have all these initiatives whether this is around machine learning, big data, whatever, and I'm not sure if people fully understand that these are inherently and intrinsically and inseparably software initiatives and you need world class software practices. It frightens me that some of these are $50 million programs where some things are even in version control, and that terrifies me.

Gene Kim: One of my aspirations for the book is that people realize that these data initiatives do need every one of the practices that we've explored in the State of DevOps Report because it's software.

Mark Miller: One of the things that Josh Corman and I used to talk about quite a bit is the horizon lines. You brought that up, horizon one, two, and three here, and it's a critical part of the book. Can you explain horizon lines to people that haven't thought about it before?

Gene Kim: The three horizons, that was actually I think first put into the literature by McKinsey, but was brought to the forefront by Geoffrey Moore in his book, Zones to Win. This is embodied in the fifth ideal of customer focus. Very briefly, horizon one is if you're a multibillion dollar organization, the main lines of business are the horizon one businesses, and this is where you only talk about the characteristic of each, and then horizon two are the businesses that hope to be horizon businesses, and then horizon three are the new exploratory innovative parts where we don't even know if there's market fit, technology fit.

We don't know what the risks and the market risks really are, so most startups are really horizon three efforts. Most startups would die to be horizon two where you're generating 10 to $100 million of revenue. The hallmarks of that is that you need greatness in daily operations, everything from supply chain to development to product operations to sales and marketing.

For me, the amazing contrast is that what we need for horizon one is so different than horizon three. Horizon three is all about learning, and horizon one tends to be all about compliance. So much of the culture misfit that we see in the DevOps enterprise community is exactly that clash is horizon one and two battling horizon three. In the book, Sarah is defending horizon one and Maxine and Kurt, they're trying to champion horizons two and three.

Mark Miller: As I was reading through the book, one of the things I started to do mentally is think about how does each character align with a horizon line, and could we define a character by a horizon line?

Gene Kim: That's a great question. In fact, before we go there, as I mentioned, these things called five ideals, I want to enumerate them so we can have it in one place. The first ideal is about locality and simplicity, so it's primarily about how decisions are made and how the code is organized. The second ideal is focus, flow, and joy. Done right, work should allow us to be focused and have a sense of flow and have this sense of transcendent joy of doing work. I think anyone who's done coding knows what that feels like and we actively seek it.

The third ideal is improvement of daily work. In other words, prioritizing improvement of daily work even over daily work itself. Fourth ideal was psychological safety, and the fifth ideal is customer focus. Customer focus, I think this really came into focus for me with Chris O'Malley, the CEO of Compuware, when he said, "If a customer isn't willing to pay for it, then it isn't core. It's really context." That's another Geoffrey Moore model.

Core is what creates durable lasting business advantage. Context is everything else, so it might be mission-critical, but it's still context. I think to your question, how do the characters align there? I think one of the things in successive drafts in The Unicorn Project, Chris, the VP of R&D, became increasingly weaker and weaker. I like that because he's essentially trying to defend status quo, defend silos, defend the annual planning process and allowed to Kurt and Maxine to rest more control from him. I think Chris embodies the way we've organized our work over the last 20, 30 years.

Maxine and Kurt, they really are the ones pioneering new ways of working that represent how do the startups organize themselves just naturally in horizon three and how should they be organizing themselves in large complex organizations. I think ultimately those things will take over horizon one. It's clearly as John Smart from Deloitte, formerly of Barclays, says, "This is a way to deliver better value safer, faster, and happier." I think it's DevOps. This is inexorable. This is a constellation of characters. This is what it takes to get us from here to there.

Mark Miller: You've been mentioning a lot of nice names, and one of the reasons you can talk with the type of people that you do is because of the DevOps Enterprise Summit. Just a short backstory on that, I talk about DOES and I talk about The Phoenix Project in just about every presentation I do because it seems to me that The Phoenix Project is what led to DOES if I can follow the timeline.

Gene Kim: That's absolutely right. To fully describe that narrative, the reason DevOps Enterprise came about was this desire to have a conference not for the unicorns, but for the horses, large complex organizations that weren't Facebook, Amazon, Netflix, Google, Microsoft. I actually tried to get this started inside of the O'Reilly Velocity Conference or any conference and wasn't able to do that, but we got close, but it was really ...

Gene Kim: I was actually going back through my notes researching the books and looking at the tweets from the 2014 DevOps Enterprise Conference, and it was ... Several people said, "Well, I'll remember the conference because there was just nothing like it." It was the first time that we could see some of the most well-known brands across every industry vertical just talking about the same problem, and the unexpected side effect for me was that it was this incredible way to, in my mind, meet the pioneers, the people who were courageous enough to even try doing this in organizations that mostly didn't want to.

Gene Kim: These are organizations, thousands of engineers, sometimes over 10,000 engineers who wanted to have nothing to do with it, so I just think it's that community of our people with a phenomenal sense of judgment around both technology and leadership and organizational politics and architecture. I say architecture not in the small, but in the large. How do we need to organize our systems to enable developers to get their work done? I think that's really what motivates them, which is we have goals. How do we best organize ourselves and design our systems so that people can actually get what they need done done.

Mark Miller: I'm still hearing the argument in the field that “we're too big to do DevOps”, that the idea that an enterprise the size of Nike or Walt Disney, they're almost unicorns by being able to do it. That's the value I get when I go to DOES each year, to be able to talk to people like Courtney, like Paula Thrasher, like Rosalyn that are really actually doing it at scale. That's the value that I perceived for what I get there.

Gene Kim: It's interesting to me. If you take a look at ... I think it's Google 2014. I think they had 15,000 developers, right? That was some years ago, but that's about the size of some of these organizations that we're talking about. Capital One is about 15,000 developers. It's obvious to me, and I can't say that I can conclusively prove this, but just intuitively, the same management structures and the same architectural patterns that the technology giants created, whether it's eBay, LinkedIn, Facebook, Amazon, Netflix, Google, Spotify, whatever, it was forged out of the need to be able to coordinate the activities of thousands and thousands developers or tens of thousands of developers.

That was primarily the domain of the tech giants, but now it's a problem being faced every day by every large organization on the planet. Ao to me, there's an equivalence there to try to distinguish which ones can only be done by the tech giants and which ones can be done by the largest brands in every industry vertical. It's just obvious to me that those two are one in the same.

Mark Miller: It's interesting you talk about that because I think that the patterns that you're describing are consistent. 80% of it is consistent no matter what company you're dealing with. It's the 20% that ... that's not what we're dealing with here. What are the patterns that are going to allow a company to do this?

Gene Kim: That's a great point, and I think there actually is a very big difference between the tech giants that were born in the late '80s and '90s and in the 2000s is the amount of technical debt they carry.

Mark Miller: It's horrendous.

Gene Kim: Right. One of the things that actually was Dr. Mik Kersten who brought this front and center for me was that every tech giant has almost been killed by technical debt. Every one of those companies essentially did a stand down, a feature freeze, for sometimes a year. Like Microsoft, the security stand down because they knew that they got in the position where it got so dangerous to change things that they just couldn't change things any longer and they had to refactor, re-architect, re-platform, etc.

The ones that survived were the ones that did that. The ones that didn't are the ones like Nokia. There's a phenomenal book by Risto Siilasmaa, the chairman of Nokia, that's featured in the book, where the chairman of Nokia, or who became the chairman of Nokia, he said in 2010 when he learned that the Symbian OS build time was 48 hours, he knew in a moment that all that the platform of which near-term profitability and longterm viability hinged upon was a mirage.

There is no way that if it takes 48 hours for a developer to know whether it worked or didn't work, could that be viable. That's what actually led them to go all in on Windows mobile just because they knew that Symbian OS couldn't take it to them. Ultimately, that led to the 98, 93% decimation of the market cap of Nokia.

This is my long-winded getting to the point where essentially all of these firms declared technical bankruptcy. I think that's a decision that has been deferred or postponed in large complex organizations. What we're seeing in the DevOps enterprise community is the need to be able to say, "All right. It's not acceptable that we run on an SAP instance that is 20 years old." We have 45 different warehouse management systems, that we're carrying around decades of this tundra of technical debt that is just slowing us down. That's something I'm hoping over the next couple years to really inject directly into the center of the DevOps enterprise programming, and that really starts in two weeks where we're going to see some direct discussions around that.

Mark Miller: I hope so because that is the big elephant in the room that hasn't had the correct focus yet. Everybody talked about technical debt and talks about legacy systems.

Gene Kim: Yeah, and legacy isn't bad. It's just that can we have an unflinching conversation of when the way something's been built or the platform it runs on is directly

impeding the achievement of the most important goals of the company. Often, that it's an architecture that was great 20, 30 years ago, but is supremely unsuited for the conditions that the business is challenged by today.

Mark Miller: Do you have a focus on the technical debt in this year's DOES?

Gene Kim: Yes. In fact, there's at least two of them that are going to be in the preliminary stage that I think are spectacular examples of this. One of them is Scott Prugh. He and Erik had a lot of ideas of what they wanted to present. I'm like, "Nope. This time, I want you to talk about the engineering you did," and I think that the business frame couldn't be simpler, that there's been a tenfold increase in transactions at CSG if you were to overlay the cost of those transactions.

If that were to increase by 10x, the company would not be making money. What are the incredible things they did to re-engineer the platforms that run the most critical business processes and the revenue generation processes at CSG so that they could actually scale to what the business needed. These are some of the most heroic achievements that I've ever seen.

Scott Havens, who was director of engineering at Walmart, will be describing what he did to move the entire supply chain management systems at Walmart, the largest company in the world, to this more functional style architecture to dramatically reduce the cost of it, make it more reliable, safer. I think it's just magnificent and it's going to be even better than the presentation he gave in London … and the talk from Adidas from Fernando Cornago-

Mark Miller: Oh, Fernando's coming? Great.

Gene Kim: All of these are talks about architecture that fundamentally change the way how teams have to work together to create value. I'm really, really excited about that.

Mark Miller: As we're thinking about DOES, what else should people be aware of that's going to happen? What other ideas did you run with this year?

Gene Kim: We have two talks. We have a talk where all of the big four auditors will be on a panel with me, talking about busting the myths whether DevOps can be done in an auditable, compliant, secure way. That's a watershed moment for the industry to be able to have every auditor just say unequivocally it can. I don't care who tells you otherwise. Take it from us.

There's another focus around organizational learnings. Dr. Andre Martin, chief learning officer from Google, will be presenting. Previously, he was at Target, and then before that, Nike, and then Dr. Dave Almeda from ... he's a chief

people officer from Kronos, will be presenting on Kronos' amazing journey to, for five plus years, become one of the best places to work.

Mark Miller: One of the big things too at the conference, one of the biggest responses last year, was the book giveaways that you put together. That was packed for two hours.

Gene Kim: Yeah. The Unicorn Project comes out on November 26th, but we did a special print run to basically make sure that every attendee will get a copy.

Mark Miller: Oh, wow.

Gene Kim: That'll be great fun, which I just find very delightful just because I would say The Unicorn Project is really inspired by and dedicated to the achievements of the DevOps enterprise community, and my largest aspirations are that people read it and are dazzled by it and then realize, oh, this isn't some hallucination or a wild ... This isn't some bad dream. This is actually what is already happening maybe even in their organizations already, and they just need to find those pioneers and support them.