The Answer to Organizational Complexity: Build in Lean Product and Process Capability

Guest: Lara Harrington, consultant; LPPD instructor, The Ohio State University; former chief engineer, Honda R&D Americas
Host: Lex Schroeder

LEX SCHROEDER 00:19

Lara, thank you so much for joining me today. For folks who may not know her, Lara Harrington is a consultant and teaches lean product and process development at The Ohio State University. She's also a former chief engineer at Honda R&D Americas and has 36 years of experience in the automotive and aerospace industries. Lara, welcome to The Design Brief. We're really excited to have you.

LARA HARRINGTON 00:36

Hi Lex, thanks for having me. I really appreciate being here.

LEX SCHROEDER 00:39

I want to start by asking you to share, in your own words, what lean product and process development is all about, and how it differs from the way most organizations are set up to do product development in the first place.

LARA HARRINGTON 00:49

I like to say, very simply, that LPPD is about building better products faster and more efficiently, all under the banner of strengthening your organizational capability. The way it differs is that organizations naturally, organically evolve to create silos. We unintentionally add complexity. When you're working within your silo and your own areas, the organization loses sight of the whole, the overall goal. LPPD takes a different approach by deliberately designing the system to optimize the whole rather than the parts.

There are three areas I think are important. The first is front-end loading: making sure all the requirements are set, and met to a certain extent, very early in development. We need to make sure we've explored the alternatives, invested early in understanding the customer, and built that knowledge before we actually start executing the solution.

I was the chief engineer on the 2019 Honda Passport, and we did a lot of great work up front to make sure we really understood the customer: who they were, how they were distinct from the Pilot customer, performance targets, customer targets, needs, wants, and priorities, all before we started executing the program. That went a long way toward aligning the entire organization, without people moving off into their

individual silos.

The second important area of LPPD is eliminating waste. You can think about waste easily in manufacturing, but in product development there's a lot of waste you don't really think about, like rework, delays, and handoffs, really anything that doesn't add value for the customer. You want to get rid of it, because all of that waste ends up affecting the customer, perhaps through cost, perhaps through quality degradation, perhaps through durability. So anything you can do to eliminate waste is a good thing.

Third, and this is like a stool with three legs, you have to have the right operating system: the right structure, the right management practices, the right routines, accountability, and so on. If you have those three legs of the stool, you are practicing lean product and process development. And if you're doing that, you are absolutely an organization that builds capability, keeps pace, and outpaces its competitors. You're going to go a long way toward creating value for your customer.

LEX SCHROEDER 03:35

You've spoken about this a little already, but can you talk about how, in your experience, lean product and process development helps companies not just build better products, which of course it does, but also helps leaders and teams reduce complexity and improve communication, and why those two things are critical? I ask because lean theory can be a big hurdle for some people. But I can't think of anyone who doesn't like the sound of reduced complexity or improved communication, or both.

LARA HARRINGTON 04:02

Any organization naturally builds complexity. It's unintentional and it's unavoidable. I keep coming back to local optimization, but people are very comfortable optimizing locally. When you do that, you make it harder to communicate and harder to work across barriers and boundaries. So working across them has to be intentionally built into the system to fight the complexity that naturally evolves.

As systems become more complex, communication obviously becomes more difficult. You find yourself spending more time in meetings, chasing information, resolving misunderstandings, and doing a lot more coordination across departments.

LPPD addresses both issues by changing how the operation works. As I mentioned, we create alignment around a common purpose, make problems visible, clarify responsibilities, and establish ways for people to work across these boundaries. That's why these things are so critical. In a nutshell, complexity consumes your capacity, poor communication slows everything down, and LPPD helps organizations recover that.

LEX SCHROEDER 05:25

Wow, well said, thank you. Can we break this down even more for people? Which specific LPPD practices reduce complexity or improve communication? Can you give a few examples that make this real?

LARA HARRINGTON 05:37

I'll give three examples that come to mind and that are synonymous with LPPD. The first is the concept paper, which is the process of getting everyone aligned up front. A concept paper is the written statement of what you're trying to do: your understanding of the customer, the problem you're solving, and how you're going about it.

In my work with various companies, I've seen people spend a lot of time developing solutions before getting aligned. For example, I recently worked with a company that was really struggling to meet a goal. When we peeled apart the organization and what they were doing, engineering was trying to find a very elegant solution, the leadership team was trying to reduce cost, and manufacturing wanted to reduce headcount. None of these matched, so of course the solution was going to be suboptimized. The concept paper, as an intentional method of getting people aligned early, prevents a lot of unnecessary downstream work by making sure all parties are aligned.

Another example is the obeya, another term synonymous with LPPD. Obeya means "big room," and it's about bringing people from the various silos together on a regular cadence so the right conversations happen cross-functionally and across the organization, instead of everybody managing their own version of reality. Everyone understands where everyone else stands. As problems arise, the whole organization is in it together, either resolving the problem or at least understanding what's outstanding and when it will be resolved. As an organization, you're able to move forward together on a cadence. It's a very effective process.

The third is making sure roles and responsibilities are very clear. Toyota has a saying that I absolutely love: if everyone is in charge, nobody is in charge. It's absolutely true. We've all been part of a meeting or an activity where nobody is quite sure who owns a decision or who needs to be involved, so you keep meeting and re-meeting, the issue gets escalated, and you revisit it over and over. Through LPPD, we do a good job of clarifying that accountability, and through practices like the obeya, we encourage collaboration across functions.

I want to make sure I don't forget to say this: the goal is not to make people work harder or communicate more. What happens is that you create a system where less coordination is required, because people are already aligned. When people are already aligned, you're much better at letting them do what they do best, which is innovate. The less time they spend firefighting and in meetings, the more they're

able to innovate. That's what LPPD is all about.

LEX SCHROEDER 09:13

It reminds me of sports teams. Of course you'd want to be aligned, of course you'd want to know the play, of course you'd want to row in the same direction. As you say, it's not about working harder. It's about rowing in the same direction. I love that.

I understand you're writing a book on these themes: why organizational capability so often falls behind the complexity companies face, and what distinguishes teams that continuously strengthen their ability to adapt and perform. I love the idea that the real challenge for leaders today is ensuring that organizational capability grows faster than complexity. That's really compelling. Can you tell us more about what you mean, and what moved you to write the book?

LARA HARRINGTON 09:55

As I began my consulting work with different organizations, large and small, I noticed that as organizations became more complex, as they grew in size, output, and volume, they became less and less efficient. Output per head was actually decreasing as the company grew. I thought that was really interesting. At the same time, it was very common for an organization that had grown, say, ten times in size to still believe it should use the same operating system it used when it was much smaller.

That fascinated me, and I started reading about it. How can you get less efficient as your organization gets larger? It turns out there's a lot of research on this topic. There's a really good book from Boston Consulting Group called *Six Simple Rules*, with a lot of good studies in it. What they found was that as organizations add more things, be it people, databases, new methodologies, or training, it actually made the situation worse instead of better. So I thought, how do we develop systems that maintain capacity as the organization continues to grow in complexity?

I have that background at Honda, which is very good at LPPD principles, and I realized there's a huge overlap between this research and LPPD. So the book is really about understanding and accepting that complexity is inevitable, that it's part of human nature, and that building capability has to be very intentional, so that we build capability faster than complexity grows. That's what the whole book is about. It has a lot of good stories and practices, and many of them fall right in line with LPPD principles.

LEX SCHROEDER 12:11

I also like the idea that, of course, complexity is going to happen. That's the nature of things. I think that helps people relax a little: yes, it's really hard, we're working in complex systems, this is what it is. Then how do we move forward and build capability? That's a great premise for a book.

LARA HARRINGTON 12:28

Right. It's interesting to me how common it is to talk to an organization that assumes capability will keep pace as it adds headcount and systems. It's a rare company that understands building capability is a very intentional activity. Some companies do it really well. Amazon does it really well, Honda did it really well, Toyota does it really well. And any company listening to this podcast can do it really well too. It's a matter of applying LPPD principles in practice.

LEX SCHROEDER 13:05

You're a former chief engineer, Lara, so I have to ask you how important this role is for companies truly committed to learning LPPD. Jim Morgan, a senior advisor at LEI, writes about the role of the chief engineer and its importance, and I know you've written about it yourself. LEI just released a book, *The Work of a Chief Engineer*, by Naoto Kitagawa. What do you think business leaders who are new to LPPD most often miss about the chief engineer role and what it requires?

LARA HARRINGTON 13:32

First, I want to congratulate you and the LEI team on launching this book. It's so good, and I learned so much. Mr. Kitagawa is from Toyota, where he was chief engineer on some really significant programs. Even having been a chief engineer myself, I learned a lot from it. What I really liked is that it has a lot of really good stories and charts. And it's not just about the chief engineer; it's about the fundamentals of LPPD. It's an easy read for people who don't like to read training books. Anybody listening, I would highly recommend it.

If you're looking for a definition, I'd say the chief engineer is the person who is ultimately responsible for the integrity of the entire product, not just the sum of its pieces. In organizations that don't have a chief engineer, someone is responsible for cost, someone else for engines, someone else for manufacturing. And, back to the silo principle, every group is trained to optimize its own area.

So what comes out is the sum of all those individual decisions. But the whole may not be as good as the sum of the parts, because everybody has optimized locally. What was originally envisioned may look nothing like what comes out the other end, and honestly, from what I've seen, it very often doesn't. You end up with a lot of compromises. Some areas get compromised. Maybe another group is especially strong, so you see a lot of strength there. But what absolutely happens is that the customer gets lost in the shuffle.

Adding the role of chief engineer changes that. Their role is to satisfy the customer: what the customer values and what the product needs to deliver. As the design takes shape, they constantly evaluate trade-offs to make sure the product delivers what was originally intended.

Here’s a simple example from my own experience. As you mentioned, I was chief engineer on the 2019 Honda Passport. We had a great team that did a lot of LPPD-type work. We developed a strong concept up front, clearly understood the customer, did our homework, and established good product requirements. Early in development, during front-end loading, manufacturing identified an issue with the spare tire installation. They proposed a few different solutions. Some cost more, some cost less, and there were a lot of trade-offs involved.

In an organization without a chief engineer, that issue would almost certainly have been snowplowed downstream, because nobody would have really owned it. You would have ended up going into mass production with significant last-minute cost increases, because manufacturing would have had to change the line layout right before launch.

But because we had a chief engineer, and I was lucky to also have a really good manufacturing lead, together we could look at the bigger picture early, during concept development. We asked right away: will the customer see any value from this potential added cost? The answer was absolutely not. So together we challenged manufacturing and engineering to come up with a solution with zero impact on the customer, because the customer did not value that cost. To manufacturing’s credit, they came up with an elegant solution that ended up costing nothing.

It’s a simple example, but the point is that without a chief engineer, the product is just the sum of its parts. With a strong chief engineer, every trade-off is managed deliberately and with the customer in mind. That doesn’t happen without a chief engineer.

People often ask me what kind of person makes a good chief engineer. I think there are two requirements: one about the person and one about the organization. First, the person. In automotive, it needs to be someone who is both technically and organizationally capable, and who has strong leadership skills. LPPD isn’t just for automotive, of course. It applies to software, medicine, insurance, teaching, and many other areas, so the specific skills will vary. But you should have strong cross-functional skills, and those need to be developed intentionally. People need to want the job, and they need to spend a lot of time shaping their roles within the company toward it. I worked at Honda for 30 years, and from my first day I wanted to be a chief engineer. It took me 25 years to get there, but every job I had in those 25 years was a step toward it, and I finally got there.

The second piece is super important: the organization has to respect the role. The chief engineer sits outside the traditional structure. You have your functional divisions, the silos, and the chief engineer sits off to the side of all that. At Honda, people respected the role, so the chief engineer was on equal footing with the functional divisions. I’d even say that a strong chief engineer who really understood the customer was on a higher footing. But if your organization doesn’t respect the

chief engineer, and every functional leader can override them, the system doesn't work. So if you want to develop chief engineers, one, start them early, and two, make sure your organization allows them to lead.

LEX SCHROEDER 19:53
A related question: how do you carve out a role for a chief engineer if you lead an organization that doesn't have one?

LARA HARRINGTON 20:01
I'll tell you what I did. I started out as a body-in-white steel stamping design engineer, designing body panels, roofs, side panels, and closures. Then I took over leadership of what we called the body group, which includes everything from doors to roof racks. The body is an interesting area because everything has to mount to it, so right away I developed close relationships with interior, chassis, engine, and exterior engineers. I was lucky to have a role that gave me a broad view within my first ten years at the company.

Then I moved into more leadership roles. I took a position at Honda known as assistant chief engineer, where you control the schedule, the cadence, the budget, and a lot of things a chief engineer needs to know. I also spent a few years in the fundamental research division to get that experience and expertise under my belt, because when you're developing a platform at the front end, fundamental research is quite critical. After 20 to 25 years, I definitely had the skill set, and I was fortunate to be offered the opportunity. It was a unique, exciting, and rewarding challenge.

LEX SCHROEDER 21:31
And what about companies that don't have a slot for a chief engineer? What would you recommend to leadership teams who earnestly want to do lean product and process development and have learned about the role? How would they move their current structure to one that includes a chief engineer who, as you said, isn't constrained by that structure?

LARA HARRINGTON 21:52
That's a good question, and different people have different viewpoints. Mine is: just go for it. It needs to be someone with good people skills, technical skills, and a strong interest in and desire to do the job. Go for it, see how it goes, and learn from the experiment. It's not one-and-done, and it's not an overnight process. Everybody is going to grow together, including the whole organization around the chief engineer. So I would say just go for it.

LEX SCHROEDER 22:21

Thank you so much, Lara. To close, I want to spend a few minutes on an idea of yours that I love. You talk about the importance of building true operating systems rather than "throwing people at problems." This idea really resonates, and I suspect it will for listeners. Why do you think it's tempting to throw people at problems? And what does it take to be more intentional and, as you've written, build systems that continuously develop people and strengthen capabilities, particularly with so much change happening in the world?

LARA HARRINGTON 22:48

That's a really good question. I think it's tempting to add people because it's almost always the lowest-hanging fruit and the most obvious solution. When I go into companies, some of the first people I'm introduced to are the ones who rally. They say, "This person is our hero. This person stays until midnight. This person rolls up their sleeves and really gets it done when something needs fixing."

Culturally, we actually like that. We like to think that when projects fall behind, we have the capability to just get it done. So when projects fall behind, we add engineers and managers. If problems keep occurring, we add processes, meetings, and oversight. Again, that's just how organizations naturally evolve.

I'm not saying those things are never necessary. But here's what's important: every time we add these things, these people, these heroes, we almost never look at how those people are compensating for a weak operating system. Instead of adding more heroes, we need to ask: what needs to change in the work? What needs to change in the way we do things? Honestly, when I hear about all these heroes and people who rally, it sounds to me like a failure of leadership.

You need to ask, "It's great that this person rallied, but what should we be doing differently in our operating system so people don't have to rally next time?" That's what LPPD is all about: designing an operating system that aligns people around a common purpose, makes problems visible, and encourages learning, so people don't have to rally. Then we're not always trying to fix yesterday's problems, and we can focus on how we innovate for tomorrow.

LEX SCHROEDER 24:57

Thank you so much, Lara. I learned a lot from this conversation, and I'm excited about your book. You've made lean product and process development real for people who are new to it, and you've helped people already in that space learn more. Thank you for taking the time.

LARA HARRINGTON 25:09

Absolutely, Lex. I really appreciate it, and I'll let you know when the book comes out.