**JUDY ESTRIN:** (Voiceover) Right now, so much of what is going on in the industry is driven by the values of scale and speed. It is about maximizing growth. It is about maximizing connectivity. What about thinking about what we're losing in those values. There's nothing wrong with growth, but not at the expense of humanity.

**KEVIN SCOTT:** Hi, everyone. Welcome to Behind the Tech. I'm your host, Kevin Scott, Chief Technology Officer for Microsoft. In this podcast, we're going to get behind the tech. We'll talk with some of the people who have made our modern tech world possible and understand what motivated them to create what they did.

So, join me to maybe learn a little bit about the history of computing, and get a few behind-the-scenes insights into what's happening today. Stick around.

**[MUSIC]**

**KEVIN SCOTT:** Today, I'm joined by my colleague, Christina Warren. Christina is a senior cloud developer advocate at Microsoft. Welcome back, Christina.

**CHRISTINA WARREN:** Hey, Kevin. Great to be here again.

**KEVIN SCOTT:** So, we are going to talk with Judy Estrin today. And I have been super excited to get this interview on the books, because Judy is truly one of the most amazing individuals in tech who I've ever met.

**CHRISTINA WARREN:** Yeah. Just seeing how far back her association with the Internet goes, and all the different roles that she's had as an engineer and as an entrepreneur.

**KEVIN SCOTT:** Yeah, her story is incredible. Like the fact that it starts with amazing anecdotes like her dad working for John Von Neumann at the Institute for Advanced Study. And sort of goes all the way through her direct involvement in the creation of the Internet protocols, to illustrious career in the technology industry as an executive and board member. I mean, just incredible stuff.

**CHRISTINA WARREN:** I'm really looking forward to hearing what you two discuss.

**KEVIN SCOTT:** Awesome. Well, thanks for chatting, Christina. We'll catch up later in the show.

**CHRISTINA WARREN:** Talk soon.

**[MUSIC]**

**KEVIN SCOTT:** Coming up next, Judy Estrin. Internet pioneer, entrepreneur, tech executive, author of the book Closing the Innovation Gap, and one of the most incisive thinkers on the intersection of the Internet, democracy, and humanity. So, welcome, Judy.

**JUDY ESTRIN:** It's a pleasure to be here.

**KEVIN SCOTT:** Oh, awesome, awesome. I've been so looking forward to chatting with you. So, I wanted to start with your childhood because you had this amazing career in technology, and your father is a very well-known computer scientist. So, with you, it started especially early. Right?

**JUDY ESTRIN:** Yeah. I had the advantage of being a second-generation technologist when computers didn't really exist. But it wasn't just my father. it was my father and my mother. My father worked with Von Neumann on the part of the team at the Institute for Advanced Sciences on the first computer. He and my mom were then invited to go to Israel to help lead the team that built the first computer in the Middle East, the WEIZAC.

They then ultimately came back and came to California and my dad started the computer science department at UCLA, and was there for years. My mother also has a PhD in electrical engineering. And when she got her PhD from Wisconsin, there was one other woman in the country who got a PhD in electrical engineering that year.

**KEVIN SCOTT:** Wow.

**JUDY ESTRIN:** At least that was the story in our family. And so --

**KEVIN SCOTT:** And when was that?

**JUDY ESTRIN:** It was in the late '40s. An interesting story. When they came to UCLA, she could not work in the engineering department, where my dad was because of nepotism rules. And so she ended up starting a data processing lab in the Brain Research Institute at UCLA. So, she was one of the first biomedical engineers and really excelled and helped build that field. So, I had two unbelievable role models. And lots of positive aspects to that.

**KEVIN SCOTT:** And I'm guessing there's some intimidation, too, as well.

**JUDY ESTRIN:** Some intimidation. I can still remember in high school when my dad brought home some video tapes to learn Fortran. And my dad ended up leaving me notes in Fortran. So, he would leave me "if, then, else" statements of what I was supposed to do that day. (laugh) And so it was my way to bond with my father that I ended up probably going into computer science.

**KEVIN SCOTT:** Well, let's talk about your mother a little bit more. So, your dad is Gerald Estrin.

**JUDY ESTRIN:** Right.

**KEVIN SCOTT:** And I think I know less about your mother, but it sounds like her career path, in many ways, might be the most extraordinary one. Did she share any stories with you about what it was like to get that PhD in electrical engineering at that point in time? And, like, she must have had this beautiful vantage point to sort of see how the field unfolded over the past many decades.

**JUDY ESTRIN:** Yeah, both of them were pioneers in their own right and in different ways. But one of the interesting things is that I remember when I was younger, and people assume that my father probably got my mother into engineering. It actually was somewhat reversed.

I just grew up in an environment where my parents were so equal in terms of their balance of career and passion. My mom was a very strong personality, very outspoken. And so, yes, we heard a lot about how hard it was, every step of the way, being a pioneer as a woman in the field.

She was very involved in the IEEE. She was always running for office in different things. She often talked about being a woman in a man's world. She went on to also, though be very, very involved in organizations that would help other women.

And so whether it was the Society for Women Engineers or Grace Hopper, started out as the Anita Borg Institute, she was a very strong advocate for exposing girls and women to engineering and the sciences and providing opportunities for them.

**KEVIN SCOTT:** Yeah. So, it sounds like some of your first exposure to programming were these Fortran tapes and your dad's "if, then, else" instructions --

**JUDY ESTRIN:** Right.

**KEVIN SCOTT:** -- about chores and whatnot. Do you remember the first program you wrote and what it did?

**JUDY ESTRIN:** Well, I don't know if I remember the first program. I vividly remember -- So, at UCLA -- now, let me back up a minute. For people who are exposed to the world today, most people who wrote their first program, you know, maybe it goes back to having an Apple or some kind of -- remember, there were no personal computers.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** So, in order to write a program, you needed access to a mainframe.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** So, there really wasn't an opportunity to write a program before being exposed to --

**KEVIN SCOTT:** Yeah, and so you had to get to a terminal room --

**JUDY ESTRIN:** Well --

**KEVIN SCOTT:** -- probably not even a terminal room.

**JUDY ESTRIN:** Right. Yes.

**KEVIN SCOTT:** You were doing punch cards.

**JUDY ESTRIN:** So, you were still doing punch cards and batch submitting. One of my favorite stories, and I ended up using this in the book, is that I can still remember that I was working on something. I kept submitting these cards and it kept coming back "abend" -- abnormal ending.\

So, you know, the equivalent of crashing your computer today. And I had been up all night doing it. I just came home in tears. I mean, I just was so frustrated. And my father sat me down and said, "Take a step back. When you have a problem that you just are overwhelmed by, try to figure out how to break it into pieces, and how can you solve the individual pieces? But keep in mind how all the pieces fit together."

That approach to problem solving stuck in my head through my whole life, and not just when it comes to writing a program. But how do you tackle really complex problems, not going to the extreme? And, frankly, we can talk about this if you want. One of my problems with Agile these days is the notion of break it into little pieces, but not keeping the architectural piece sometimes, of knowing how they fit together.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** That balance of addressing complex issues by being able to figure out pieces that you can solve and the interconnections, that system level of thinking was something that my dad, I think, it was so much about his approach to life and how he saw things.

**KEVIN SCOTT:** So, what you were just talking about, this amazing advice that your dad gave you about decomposing a problem is something that has come up somewhat frequently in the conversations I have with other computer scientists recently.

And there's this question of kids these days who are being trained as software engineers are dealing with such higher-level abstractions than the ones that we were dealing with when we were up and coming as programmers.

And the question that I hear people asking over and over again is whether or not they're missing something by not being exposed to just this wide differential between the top-level understanding of the problem and just these absolute nitty-gritty, atomic details of how to get a solution. And I don't know what the answer to the question is.

I know that I learned to code on Apple IIe's and a Radio Shack Color Computer 2, which hooked up to a little 13-inch television. And, subsequently, got the chance to muck around with manufacturing equipment that were controlled by PDP-8s and systems that had wire-wrapped backplane buses and whatnot. But I've never coded, you know, with punch cards.

**JUDY ESTRIN:** Right.

**KEVIN SCOTT:** And, you know, like, I just sort of wonder, like, what did I miss by not having that?

**JUDY ESTRIN:** So, I don't think it's the punch cards that you miss.

**KEVIN SCOTT:** Yeah. (laughs)

**JUDY ESTRIN:** Meaning, I do think you're really on to something here. The punch cards are part of it which has to do with a little bit of friction, a little bit of obstacle being put in. But let me start this off by saying, yes, we're missing something. But it isn't that everybody needs to go back to doing that, it's that we need to recognize that there's a certain type of training that is not happening, and figuring out how to be additive.

The underlying thing I would say is we've lost system thinking. In the days when computer science was about building systems, building infrastructure, understanding how compilers work, how you talk to a computer -- all of those things taught you a sense of learning of system thinking. And when I say system thinking, I mean two different things: One is how pieces are interconnected into a whole. So, there's an architectural sense there, as opposed to looking at things as just individual points. And then the second thing is thinking about the consequences of things.

When you're building a system, you have inputs and then a complex set of interaction that may have different outputs, and poking it in different ways, that unless you're thinking about systems that are complicated, you don't think about consequences -- intended and unintended consequences. And so, those two things are really important.

And I think that whereas the abstraction, in some ways, is so beneficial because, you know, I remember writing assembly code for a PROM programmer. That wasn't necessarily a good use of how many hours it took to do it, or the punch cards. But it seems to me, we have three side effects: One is, do people know how to tackle hard problems? Or do they only look for things that can be easily solved? And do we turn away or avoid or actually try to simplify the problem to something we can solve and then there are world problems that get delayed? The second is this whole thing of pieces in a whole.

And as I mentioned before, I really worry about the extreme of everybody take a little piece and not enough thinking about the architecture, how they fit together, do you have a foundation that is not brittle? I worry about reactiveness. But there's a third thing which is, it is so easy to iterate on things. We all talk about iterating, iterating, iterating. So, what happens is that you don't take as much time to think about what you're doing. And that's one of the things, the batch cards, it wasn't so easy to do a turnaround and try again, and so, you thought a little bit more about how you were fixing it before you submitted the cards.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** When you have very rapid iteration, on one hand it is -- we all know this, it's wonderful. You can fix problems quickly. You can do A/B testing. It also makes you sloppy in your thinking at times or -- I shouldn't say "sloppy" -- lazy.

We don't have to put the same amount of thought into it, and if you're talking about failure that is not just inconvenient, but can be harmful, you can't just say, "Oops." Right? So, whether it is a self-driving car that hits somebody or a social media system whose implications is threatening democracy, we're not allowed to just say, "Oops." We have to be thinking about the consequences of the technologies that we are bringing to market, because we are now in the center of everything in our lives.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** And I just am concerned that as that happened, as we became more and more consequential to the world, we also have not been training ourselves or training new engineers and computer scientists to think about those systems and consequences in the same way.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** And, actually, the culture is in the other direction, which is move faster, get out a minimal viable product, get out your feature, and then we'll learn from our mistakes. I'm a big believer in learning from mistakes and learning from failure and taking risks, but we need to take a step back and say, "What are those risks?" "What are the consequences of the risks?"

**KEVIN SCOTT:** Yeah. You know, I couldn't agree with you more. I think there were a bunch of really powerful points you made there. Like, sometimes the struggle, itself, like having to sort of slow down and think about something holistically, really sharpens your thinking. And some of the most interesting breakthroughs happen that way.

I remember being irritated when I was a senior in high school taking my first real computer science class at my professor, who made us write the solutions to our programming assignments down on paper before we typed them into the computer. And he could tell if you typed the thing into the computer and done this iterative debug cycle to try to get it right. And I didn't appreciate at the time what slowing down was doing to help me be a better thinker about what it was that I was doing. And then your whole point about this sort of crossing of these two curves -- the importance of technology and its ubiquity in our day-to-day lives.

At the same time where we're sort of more fragmented in the way that we do software development. We've "atomized" things into a bunch of different pieces. Which, on the one hand, you've got to have some mechanism to deal with complexity, but just because you're trying to solve a complex problem doesn't absolve you of the responsibility of having to think very carefully about the problem and its second-order effects.

**JUDY ESTRIN:** So, I bring up these issues to say what we need to do is shift our values or remind ourselves of the values that drive that talent. Right now, so much of what is going on in the industry is driven by the values of scale and speed. It is about maximizing growth. It is about -- even in some ways, which it sounds like a good value, maximizing connectivity. What about thinking about what we're losing in those values.

There's nothing wrong with growth, but not at the expense of humanity. Not at the expense of society. It might be nice to think that maximum flat, borderless connectivity should be a goal, but if you actually look at the way humans act and understand a little bit more about people, you might say, "You know what? With connectivity needs to be some containment."

Look what happens when we build fast in Houston and don't think about how waters move and a hurricane comes and the floods were way worse because we built for scale and we didn't pay attention to containment. When you're fighting a wildfire, you think about containment. Well, misinformation spreads if you have maximum connectivity without thinking about where you need to contain, because bad things can happen.

So, our industry is filled with so many talented, wonderful people, but I think sometimes we as leaders are pointing those people in a direction and setting values which are not, in the end, changing the world for good.

**KEVIN SCOTT:** Yeah. So, I want to dig in more on this whole notion of values, but through the lens of some of your early experience. So, you were present at the literal creation of the modern Internet. So, tell us a little bit about that story, like how -- after you graduated UCLA.

**JUDY ESTRIN:** Right. So, it goes back a little bit more in that, again, my dad was chairman for a time at the computer science department at UCLA. And actually, UCLA and Stanford compete for where the Internet began, because one of the initial ARPANET nodes was at UCLA.

Paul Baran, who's the father of packet switching, was one of my dad's PhD students. And so, again, by osmosis, I'm being exposed to this. The other person who was one of my dad's PhD students was Vint Cerf, who is known as the father of the Internet. So, I ended up coming to Stanford, doing my master's, and Vint Cerf was my advisor. And he was leading the team that was developing the initial TCP protocol spec. At that time, it wasn't TCP/IP yet, it was just TCP -- Carl Sunshine, Yogen Dalal.

There was no computer science department then. It was EE computer engineering. And I was one of three women in the master's program. They had already mapped out the initial spec for TCP, and it was my job to do the testing of the initial implementations. I can still remember the basement lab that I used to go sometimes at 3:00 in the morning because the two sites we tested was BBN in Boston and the University of College London.

So, we don't think of it today because one of the beauties of the Internet is that you have asynchronous communication. But we would send packets, and then we would have a teletype machine in real time to say, "Did you get it?" So, we had to be on the same time frame. So, it was a pretty exciting time in terms of being part of that.

I will say that it was also my first exposure of people not accepting -- they didn't really want a girl joining their group. Not all of them, not Yogen, but there were three unnamed people in that group that just made my life miserable for that year. And it was, I think, my first time of "Oh, this is what my mom's been talking about all of the time." It was a wonderful experience.

While I was at Stanford, I built a local area network out of these three circuit board suitcases with Z80s. That was the very beginning of Z80 microprocessors. And so I built this local area network. I ended up, when I left Stanford, my first job was at Zilog.

**KEVIN SCOTT:** And you were on the CPU design team for one of their microprocessors, right?

**JUDY ESTRIN:** I was on the design team of the Z8 and Z8000. I was part of the software group. And, you know, now it sounds obvious, but in those days, it was very advanced. My boss and the people at Zilog decided it would be good for me to be part of the group to look at the instruction sets from a software developer's perspective. Nobody had ever done that before. And actually, I suggested an instruction in the Z8000--

**KEVIN SCOTT:** Which one?

**JUDY ESTRIN:** Compare string. That nobody had ever put in a microprocessor. I was very lucky to be in an environment of -- that notion of it's not so much interdisciplinary, but the different perspectives and the strength of having software and hardware designers working together. This is one of the early examples of the abstraction of how do you abstract, and one of the benefits of doing it.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** Because it just made things like that so much easier.

**KEVIN SCOTT:** Yeah. And, you know, now it's just incredible to think about where the abstractions are.

**JUDY ESTRIN:** Right.

**KEVIN SCOTT:** You can, with a few lines of code, maybe no code at all, you can go to a cloud provider and click a few checkboxes and have petabyte-scale data storage system deployed planet wide that can do tens of millions of transactions per second, and build your application on top of this.

So, on the one hand, I think the abstractions are absolutely beneficial. You don't want everybody all the way back, you know, at the dawn of time having to write assembly language for their apps. But being able to punch past those abstraction boundaries when you need to and to be able to think holistically about these systems I think is still just as important as ever.

**JUDY ESTRIN:** We do realize that there's a huge amount of power in some of these systems that we are making -- enabling people to use who don't understand the power of it. There are some abstractions that maybe you shouldn't do, because if you understand how a system works, you understand how it can go wrong. And then you're a little bit more careful in terms of how you deploy it.

And we look at cars and how cars have evolved and where you can fix things, where you can't, and let's take self-driving cars out for a minute. When we let somebody get behind the wheel and yield the power of a machine that can do wonders, and do harm, we train them. They get a license. We could abstract brakes and putting your foot on the gas even more than we do, and we don't for a reason, because there are consequences if you don't understand a little bit about those trade-offs.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** And so, I worry, especially with cloud computing, that we can get to a place where that -- we unleash technology without people having to have an understanding of the consequences.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** And the stuff that's gone on about facial recognition. We really need to think about how we communicate those implications.

**KEVIN SCOTT:** Yeah. There's also this impetuousness to young engineers sometimes, where you'll just sort of disregard a bunch of learning that other people have done, especially if it's in the analog world, and just sort of assume that you can do better with software.

And, sometimes, like the control systems for cars, these things have evolved over the course of a century, where people have iterated and looked at the data and they understand that these things are safer when they're implemented in this particular way.

You know, I remember one of my bosses told me this story. So, he was a power systems engineer by training, and he was telling me about Three Mile Island, and that one of the reasons that they didn't notice the meltdown sooner was the monitoring systems were just too noisy.

**JUDY ESTRIN:** Right.

**KEVIN SCOTT:** The operator was staring at this wide, wide panel that had dials and gauges and indicators and flashing lights, and just sort of missed the one important one.

And I've always thought about that from and operations engineering perspective. You can monitor a large-scale, distributed system and literally have millions of instruments out there sort of looking at everything. But if you're not able to surface the most important thing to folks who are going to have to fix things, then it's all sort of pointless.

And we're having this conversation, I think, more urgently now around AI and data. You know, the facial recognition stuff and bias in data sets. I like the direction that the conversation is proceeding in, but I think there's still a lot more to be done.

**JUDY ESTRIN:** Right. So, I'm not just saying this to flatter you and because it's your podcast, but my conversations with you on this over the last year that we've been talking about this have given me some hope. Because I think there are forces in the industry, some that are really just so focused on progress at all costs, and others that are understanding of the technology, but are conscious of those trade-offs.

And then there are a lot of people who can throw stones at all of this, but if you're not in the middle of it, and you don't understand, look, I understand some, but I've been out of the sphere for enough time to know I don't know, at a visceral level, the details of everything. And so, again, not just saying this because it's your podcast, but the role you play, and being in a position to actually understand, make a difference, and having that conscious is just -- it's heartwarming.

**KEVIN SCOTT:** I appreciate you saying that. And I know that there are a bunch of other very thoughtful folks pushing against the problem as well at a bunch of companies. So, I'm hopeful.

**JUDY ESTRIN:** We're in your hands, so, please be. (laugh)

**KEVIN SCOTT:** Yeah. We all need to look at this with the gravity that it deserves.

**JUDY ESTRIN:** Right.

**KEVIN SCOTT:** It's just too important to be cavalier with.

**JUDY ESTRIN:** Yeah.

**KEVIN SCOTT:** So, you're at Zilog doing microprocessors, software code design. And then you become an entrepreneur.

**JUDY ESTRIN:** Well, I was at Zilog doing the microprocessor stuff. I was the project manager on something called Z-Net, which was the first commercial local area network. So, while at Zilog, I had the opportunity to be involved in the development and introduction of this local area network.

**KEVIN SCOTT:** And when is this?

**JUDY ESTRIN:** I think in '79 or '80.

**KEVIN SCOTT:** So, this is super early. Like the state of the art for high-performance computing then are like these big vector machines.

**JUDY ESTRIN:** Yeah. So, we built a system -- part of the reason -- this was a semiconductor company, a microprocessor company, why were we building a local area network? Zilog decided to go into the systems business around their microprocessors.

And so what we did was develop a local area network to interconnect these systems. And our computing system was called an MCZ system, which was an early PC, but it wasn't a PC. And we had an operating system that was called RIO -- Real I/O, which was a predecessor to the operating systems of PCs. But Zilog was a microprocessor company, not a systems company.

And so one of the things I learned was up until then, I was an engineering elitist. I thought marketing people weren't important. (laugh) But I learned a really important lesson, which is: If you don't look at the needs of the market and match the technology, the technology is for naught. And even if you meet those needs, if you don't understand how to market and sell it and you don't have distribution channels that match up, you don't get anywhere, you don't solve any problems.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** So, it's fine if what you're doing is research and discovery, and you're not trying to bring a product to market, but if you're trying to solve a problem, you need all these pieces. I very quickly, within those years at Zilog, moved from being an individual contributor to a first-level and then second-level. I moved into management pretty quickly.

**KEVIN SCOTT:** Did you enjoy the transition?

**JUDY ESTRIN:** Yes. I --

**KEVIN SCOTT:** And was it obvious to you that that was the right way to go?

**JUDY ESTRIN:** I never would have thought. I, as a kid, was not one of these people that thought I wanted lead, thought I wanted to be an entrepreneur. I didn't have a lemonade stand. I didn't do any of that stuff. And I found myself in this position, and what I realized very quickly is that I love people. I love collaborating. I love helping people and helping people develop and leading a team.

And, you know, the fact of the matter is, I wasn't a great software engineer. I think what I was best at was the architecture and the thinking about the kind of systems thinking. I never wrote the fastest or best code. And so, I ended up lucky that I had the opportunity. By going to Zilog, I also met the person who became my husband, now my ex-husband.

We started Bridge Communications, which was one of the early local area network providers. And so the name Bridge was about interconnecting networks. And it was interesting, our business plan, when we started, was going to be about interconnecting networks. And very quickly, we realized there are not enough networks to interconnect. And so we started off selling communications servers, which were connecting devices into networks, and then also interconnected those networks.

**KEVIN SCOTT:** That's awesome. So, presumably, you ran engineering and technology at the startup.

**JUDY ESTRIN:** Right.

**KEVIN SCOTT:** So, you went from being a second-level manager to the buck stops with you. So, how was that transition?

**JUDY ESTRIN:** It was -- So, Bill was the CEO. I was the Executive Vice President in charge of engineering for the first couple of years. But, you know, early on, it felt very natural. And because Bill and I were partners, we kind of shared a lot of the responsibility. But the thing I really had to learn was how to make decisions. And not how to make decisions, but how to make decisions in that kind of environment. And as an engineer who loves solving problems, I wanted to get it right.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** And sometimes you have to make a decision with the data you have, and you can't know that it's right.

**KEVIN SCOTT:** Yeah.

**JUDY ESTRIN:** And it was a really interesting challenge for me. Bill had a very different style than me. He sometimes would say, "Go and fire that person," or "Go pound on the table." And there are people who are very effective with an intimidating style. That isn't me. And if I had tried to be that, I wouldn't have been, I don't think, as successful, because I wouldn't have been authentic to who I am.

**KEVIN SCOTT:** Yeah. I think that's really interesting. One of the things that I struggled with when I first became a manager, I struggled with different flavors of this for a very, very long time, until I was managing hundreds of people. I just didn't understand that in leadership, many, many, many times, maybe more often than not, there isn't a right and a wrong. There's the best you can do at any particular point in time. That's particularly true around people.

**JUDY ESTRIN:** Right.

**KEVIN SCOTT:** For a super long time, this is maybe the last big management lesson that I learned is that I would keep people on the team who were toxic for the culture that I was trying to create just because by the numbers, they were doing their job well.

**JUDY ESTRIN:** Right.

**KEVIN SCOTT:** And giving myself permission to say, "Okay, this is my team, there's no right and wrong of it. This is the group of people that I want to surround myself with, and this is the culture that we want to have in order to go solve these problems." That's sort of okay.

**JUDY ESTRIN:** But you just used an interesting example, which is there is no right or wrong decision, often, but internally, you're guided by what is right or wrong.

**KEVIN SCOTT:** Yes.

**JUDY ESTRIN:** So, you used an example of not tolerating behavior because of performance based on you're driven by what -- your values in terms of what is right or wrong. I've seen leaders who use the excuse of, "There is no right or wrong decision," to actually go in the opposite direction.

And so I think that learning how to be able to make those decisions with a combination of your gut and your intellect, with a combination of data and compassion. It's the balance. It's why we're not machine learning algorithms. Right? You know, we bring something different to it because there's a lot of nuance.

**KEVIN SCOTT:** I love what you just said. This whole notion of data plus compassion, I think actually does lead to the best decisions.

**JUDY ESTRIN:** Right.

**KEVIN SCOTT:** And that's a hard thing to get pounded through the head of a computer scientist.

**JUDY ESTRIN:** Right. I like to say that you can have data-driven management, but you need human-driven leadership. Leadership is not data driven. Managing is data driven. And so, you somehow have to combine the two of those.

**KEVIN SCOTT:** So, at some point, like, your career -- you built this successful business. You're this hugely successful tech executive, and one of the things that you and I have chatted about you were CTO of Cisco for a while. There are very few people who have been CTOs of big tech companies.

And I've gotten a bunch of good advice from you about how to do my job. So, what was the transition like for you, going from entrepreneur, you got the mission of the company, you're building the team, you know everybody, to, "Holy crap, I'm CTO of Cisco."

**JUDY ESTRIN:** Yeah, fascinating. So, Cisco bought Precept, and I became Cisco's Chief Technology Officer. And that was in 1998. So, I was there from '98 to 2000. The company went from 18,000 to 36,000 people in that timeframe. I was CTO. I had legal, M&A also reporting to me. It was fascinating to go from being an entrepreneur who was always fighting, the scrappy company fighting against big companies to all of a sudden be at this company where everybody returned your phone call. Everybody wanted to see you. And I loved learning a different level of issues. It was also intensely frustrating for me, because I was not building, I was the CTO.

**KEVIN SCOTT:** But you must have learned something in that role about managing at a distance and via influence, because you had these incredibly successful and long board runs at Disney and at FedEx.

**JUDY ESTRIN:** So, I was just about to say that I think it was the inverse in some ways. I had been on the board of FedEx -- I went on the board of FedEx in 1989. So, I had been on the FedEx board for a long time, which gave me exposure to the issues of a big company, a different type of leadership.

I went on the board of Disney the same year I became CTO at Cisco. And I had been on the board of Sun Microsystems for a while. So, the board work gave me a sense of scale and innovation and the issues of that size company. It also gave me a sense of how to make an impact without direct-line responsibility.

**KEVIN SCOTT:** And I think you had some really interesting moments in your board tenure.

**JUDY ESTRIN:** Yeah. That is true. But I've got to tell you, the opportunity to serve on the FedEx and Disney boards is just phenomenal in terms of helping to build my understanding of a bunch of different things of innovation at scale, what it's like to use technology, as opposed to being the developer of the technology.

The media business, what I learned at Disney about the media business, and now that there's been a confluence between the technology and the media business, it's really important part of my sets of experiences.

**KEVIN SCOTT:** Awesome. Well, so, one last thing before we let you go. I want to talk about some of the stuff that you've been doing more recently. So, and you and I have been having conversations about where both technology and technologists, potentially can have impact, both positive and negative, on the public good. You're doing some really interesting thinking and collaborations there. So, tell us a little bit about that.

**JUDY ESTRIN:** So, I'm spending a a certain amount of time thinking and writing and collaborating in the area of the impacts of today's technology on democracy, on humanity, on media, and trying to look at some of the unintended and intended consequences of today's business model, of the disruption of disinformation, of addiction to technology. There's a whole set of interrelated issues.

My concern comes from, often, we want to, again, just look at a piece. Is it data privacy? Is it just election hacking? It's a lot of things. And so I've been working with a number of people both in terms of writing and collaborating, but also some specific things of, okay, what are some of the things we can do about this? And I really do believe this is similar to big oil or big pharma or big food, where you have an industry that is very successfully focused in an area, but there are consequences.

And then what happens is there's opportunity in developing alternative energy. There's opportunity in developing healthy food. And the existing legacy companies have a choice. They can either choose to understand the consequences of their action, and embrace that opportunity and work with new companies and, ultimately, open up to provide all the wonderful things they provide without some of the harm. Or they can deny climate and go in their hole and, ultimately, they'll get regulated, disrupted -- hopefully, in the climate case. But I just think we need to be more aware of all of those consequences. So, I'm spending a certain amount of time on that.

**KEVIN SCOTT:** Which is fantastic. And I'm really, really glad that you are engaged here, because one of the real difficulties that I see is that these issues are, I think, unprecedented in terms of the complexity. There's just nothing in human experience that would let you develop an intuitive feel for what's happening under the hood of some of this technology.

And it's not all sinister, right? I would put forth that most of this stuff, like the vast majority, especially built by tech companies, is like well-intentioned technology, where we haven't thought about the second-order consequences of what people are going to do with them, and a variety of different things.

But getting the public to be informed enough about how these systems work so that they can have a degree of agency, they understand what's going on both for themselves, and for how they are pushing their own advocacy out in the world for how they would like things regulated, and how they would like companies to better serve them. I think is really, really important. And it's tough because of the complexity.

**JUDY ESTRIN:** Yeah, it is tough. And I think that we just need to keep in mind -- and you said it, it's not that people are bad, but we look at the incentives of organizations. When we were talking about making decisions, all of these decisions are made by who are you serving?

And so it is easy in a capitalist, for-profit environment that you serve shareholders who are pushing you for short-term returns, to want to maximize growth. It's not like your trying to do harm, but you're measuring the things. We manage what we measure. And if we pick those metrics around things that are focused on short-term growth, short-term profits, it's not like we want to do harm, but we're not measuring the harm. And if you're not measuring the harm, then the people in the organization don't rally against it. And I am fortunate at this point in my life to have the understanding of technology at a broad level, but no longer being in the middle of an organization or set of organizations.

So, I see if I can bridge that gap without an agenda, so having an understanding of the for-profit world, having an understanding of those incentives, understanding technology, but now being independent enough to actually be able to look at it and say, "You know what? When I was doing that, I had that problem also, but now I can see that we need to think about how to change that, because it's doing harm." And so I'm just really fortunate to be at a point where I can at least try to add my voice to the debate.

**KEVIN SCOTT:** We're so glad that this is how you're choosing to spend your time. So, thank you so much for being on the podcast today.

**JUDY ESTRIN:** It was my pleasure. Thank you for inviting me.

**[MUSIC]**

**KEVIN SCOTT:** Well, thanks for joining us for Behind the Tech. What a great conversation we just had with Judy Estrin.

**CHRISTINA WARREN:** Yeah, it was really great hearing the two of you talk. And hearing about Judy's career and all the things that she's done and how she's literally been involved with the Internet from basically, its inception through the present is so interesting.

**KEVIN SCOTT:** Yeah. Now, Judy has been an incredible mentor to me. Her story is inspiring. She has so much wisdom. She's such a good technologist. I've learned a ton from her over the years.

**CHRISTINA WARREN:** I was really struck by one of the things that she said in your conversation where she was talking about growth without the expense of humanity. And how that seems easy to say, but seems in practice, actually, really difficult.

**KEVIN SCOTT:** I think there are a bunch of things that we can do. She was sort of dead on with this notion of we don't don't fix what we don't measure. And part of what I think we're learning right now with some of the things that are going on in the tech industry is how to measure some of the things where people are using the technology that we've built in these sort of unanticipated, bad ways.

And I think we will learn very quickly as an industry, how to get better on this stuff. But, I think it really is going to require us to start thinking about our role as computer scientists and engineers and technologists a little bit differently. And for us to start when we are educating folks to make sure that that human part of the job is just as well emphasized as the technological part. I went to a liberal arts school to get my computer science degree. And even there, at a liberal arts school, things like taking an ethics class weren't mandatory. I wound up taking a philosophy class. We've sort of developed over the many centuries this whole notion of a liberal arts education because it's important. What all of us have to like really understand, especially those of us who are in fields where we're building technology that has such a pervasive impact.

**CHRISTINA WARREN:** Do you think having that background in humanities, do you think that helped you as you became a technologist, and as you've transitioned into becoming a manager and now an executive?

**KEVIN SCOTT:** Yeah, I think it has in interesting unanticipated ways. If you'd asked me the question years ago, probably I would have said, "Not so much." The obvious things I think that it helped with were with just writing, for instance. And being an effective communicator. The thing that I think is really useful, part of this is -- part of this, I think you sort of learn more from being a manager than you do from a liberal arts education, is just having compassion for people.

Once put yourself in the mindset that, you have to take care of people, it really does change how you make decisions. And, I think if anything, what we need to have a consistent understanding of in the technology industry is, like, that's our job. We have to take care of the people who are using our products.

**CHRISTINA WARREN:** So, Judy talked about her transition from being an engineer and into management and entrepreneurship. Did you relate to any of that at all? Did any of that kind of resonate with you in your transition from being an individual contributor to leading hundreds and thousands and more people?

**KEVIN SCOTT:** Yeah, no, totally. I think it's really interesting to see what a consistent experience that is for leaders. I think a lot of what Judy went through, I went through as well. It's really challenging as an engineer to go from this mindset of, like, "Okay, I'm solving problems that have a right and a wrong." To, like, "Okay, now I'm solving problems that don't have a right and wrong."

And a bunch of constituencies who are asserting their right to be heard, and you have a bunch stakeholders involved in the outcome of the decision, you gotta just sort of basically, make calls on imperfect and incomplete data. And I think that struggle is something that every leader goes through at some point. It is an interesting transition. And it makes you, on many days, wish that you just stayed an engineer. (laughter)

**CHRISTINA WARREN:** That's really -- that's amazing. But I think what you're just talking about, again, goes back to kind of what I picked up as the thesis of you and Judy's discussion, which is all about thinking about and being aware of consequences as you're building things.

**KEVIN SCOTT:** Yeah. And given the complexity of the things that you're building that is easier said than done, but increasingly necessary, nonetheless.

**CHRISTINA WARREN:** Well, I'm glad that people like Judy are working to help other businesses and other people think about those things. I'm glad that you're aware of those things and are doing that as well, because we definitely need all the help we can get.

**KEVIN SCOTT:** Yeah. I'm super grateful for the folks like Judy, who are choosing to use their time in this way to like try to benefit everyone. Great conversation, Christina. Looking forward to chatting with you again on the next episode.

**CHRISTINA WARREN:** I can’t wait.

**[MUSIC]**

**KEVIN SCOTT:** Next time on Behind the Tech, we'll talk with Danielle Feinberg. Danielle is director of photography at Pixar Studios. Her love of combining computers and art began when she was eight years old. This eventually led her to a BA in computer science from Harvard. Today, besides making films for Pixar, she mentors teenage girls, encouraging them to pursue code, math, and science. So, be sure to tell your friends about our podcast, Behind the Tech, hope to see you next time.