Rashmi:
This is ACM ByteCast a podcast series from the Association for Computing Machinery, the world's largest educational and scientific computing society. We talk to researchers, practitioners and innovators who are at the intersection of computing research and practice. They share their experiences, the lessons they've learned, and their own visions for the future of computing.

Rashmi:
Today's guests need very little introduction. Dave Patterson and John Hennessy are legends in the field of computer architecture. They recently won the ACM Turing award for their invaluable work impacting the microprocessor industry. Dave and John, welcome to ACM ByteCast.

John Hennessy:
Thank you. Glad to be here.

Dave Patterson:
Thanks for inviting me.

Rashmi:
Great, so I'd like to lead with a simple question that I ask all my guests. If you could please introduce yourself and talk about what you currently do, and also give us some insight into what drew you into this field of work to start with.

John Hennessy:
Okay. This is John. I'm back teaching at the university running a new scholarship program. And in my spare time I chair the Board of Directors of a small startup called Alphabet.

Dave Patterson:
This is Dave Patterson. I'm in the subsidiary of that small organization of Alphabet. I'm working right now, most of my time at Google. I was a professor at Berkeley for more than 40 years. I still work there part time, and I also work for the RISC-V Foundation. In terms of how I got into this, it goes back to ... I think we'll have to go back to college. I was a math major and a math class was canceled, so I took a computer course and like what Fred Brooks says in the Mythical Man once, I was just captured by the idea that the thoughts in your mind come alive and I was completely hooked. So that's how I got into it.

John Hennessy:

I started, I did a little bit of computing in a computer club when I was in high school and then got more and more interested as an undergraduate. And then microprocessors were just coming of age when I began my graduate work and it looked like a good opportunity to begin to think about how software systems would be developed for this new generation of computing technology.

Rashmi:
From that simple starting point, the way you're describing it, to all that you've achieved, your work has been a result of years of research and experimentation. I'm sure it's been a very iterative process. Do you have or recall any breakthrough or aha moments in your career? What was your product/market fit journey like?

Dave Patterson:
I don't know what that last phrase is. John's a person who's had a company, so I think the early aha moment was when we were doing the RISC technologies at Berkeley and Stanford. And RISC stands for Reduced Instruction Set Computers. It was back in the 80's, instead of building really complicated instruction sets, which are the vocabulary that software speaks to hardware, we built really simple ones, is we knew at the time the clock rate would be relatively high, but the question is how many more of these simple instructions would we execute?

Dave Patterson:
We actually took this toy program, the puzzle benchmark, and we actually calculated how many more instructions we had to execute than the more complicated vocabularies, at the time. And the aha moment back then, which must've been right around 1980, it wasn't that bad. Maybe we had to execute 30 or 40% more, even with our crummy compilers. But this could work, a high clock rate, not that many more instructions. This might be a good idea. So what I did is right after that I wrote this paper that's kind of an oped piece called A Case for Reduced Instruction Set Computers, because I thought, because I'd seen this result, this might really work.

John Hennessy:
I think Dave and I both also had exposure to the primary way in which many computers and even mainframes were designed then, using lots of microcode. I think we both looked at it and said, "Well this machine is doing a lot of things at runtime that could be done at compile time with less overhead and more efficiency. So why not just do it then, simplify the instruction set. Why not make the micro instructions the instruction set rather than add an extra level of interpretation. I think that was a great insight.

Rashmi:

Super. I mean, you speak about efficiency. At that time, when you saw the improvements that you did, was the concerns we seem to have today as we've grown in terms of the usage that we have, it seemed to be around energy efficiency in the world of parallel computing. I've heard the term performance per watt being bandied about quite a bit as a unit of measurement. What do you think of this unit? How have you seen the changes happen over a period of time? Is this what you try to optimize for when you are designing new systems today?

Dave Patterson:
I think the question you can ask with Moore's Law is how come you could keep doubling the transistors and the chip didn't burn up? And the answer was this thing called Dennard Scaling, where he made this observation that every time he got more transistors, you could lower the threshold voltage. So actually, the power stayed constant, which was kind of amazing. So back when we were doing this, microprocessors were, I think were five watts. And they stayed five watts for a long time. And then they started going up fast.

Dave Patterson:
So suddenly, they started using up so much power that it ended up that Intel built a processor intended for laptops that was so hot, you couldn't put it in a laptop. So they had to stop what they were doing. They had to cancel that project. So since the end of Dennard Scaling, which has been at least a decade ago, this has become a primary limit to what you could pull in a microprocessor. It's the power you use. How efficiently is your design using the power that you have available? So it's your power budget that limits you more than the transistors.

John Hennessy:
Yeah, Rashmi, I think you actually said it right. It's about efficiency. When Dave and I started, the number of transistors you had available on a microprocessor was in the tens of thousands rather than the tens of millions or billions. So you had to efficiently use those transistors. Today, power is what you have to use efficiently. But also with the slowdown in Moore's Law, making sure you use transistors efficiently is going to become more important in the future.

Rashmi:
Got it. And it's interesting that you say that. We were talking about laptops, but in the last 10 to 12 years, as the form factor of our devices have shrunk significantly, you've had to think about smaller devices as well. So what you're saying is very applicable in that space too. But in the similar vein, for smaller devices, we've also had to think of memory architecture differently. I mean, we are suddenly storing so much data, much more than we ever have before. That puts pressure on our storage needs too. Could you elaborate a little bit on that and tell us where that field of research is headed? Are there new viable options that we should be considering?

Dave Patterson:
I think over John's and my career, there've been a bunch of claims of new exciting memory technologies and they usually don't end up being commercially viable. An exception to that, over my career, when John and I started, I think memory was not made out of semiconductors. It was a magnetic memory. This was what's called core memory. You can still hear people refer to memory as core. That's like a Baby Boomers or something, old people. But it's been semiconductor for a long time, but the backing storage that would store all the information in the world would be magnetic discs. They would still use magnetism to store information that they would spin.

Dave Patterson:
Well, a new technology that really made it was flash memory, which is a semiconductor technology. So virtually, these laptops today don't have discs. They used to have discs. They used to make noise and spin. Now it's a semiconductor memory. So that technology has made tremendous advances. There are people who think there are even better technologies out there based on more exotic memory technologies that'd be much better than flash or are kind of the standard bearer that we execute programs of called DRAM. It's hard to know based on our career whether these technologies are going to be commercially viable or not, but there's a lot of enthusiasm around them in the research community.

John Hennessy:
Yeah, and just as flash has changed what we can do and the amount we can store and really was the one technology that beat magnetic discs out, the question of whether these other technologies will be successful, I think a key insight here is density becomes the key limit. And as DRAMs begin to see their end of life and their slow down, whether or not we'll have an alternative memory technology is going to be a critical question going forward.

Rashmi:
I mean, are there popular names that get thrown around? I know I did a little bit of research and I read a little bit about 3D NAND. Any thought about that?

John Hennessy:
Yeah. Well flash, one of the reasons flash has been so successful is it's been able to go 3D, and I think that certainly helps. There's been discussions about various kinds of magnetoresistive or other approaches that could allow you to go to 3D. The problem is conventional DRAM technology can't go there.

John Hennessy:
So whether or not one of these technologies makes it, I think the thing to understand from a computing viewpoint is the time between when a person who's a technology person builds one in the lab and the time that technology is competitive in the market as a replacement technology is probably close to a decade. It just takes a long time to go from building one to building millions and billions of them at prices that are competitive. The place to look right now is in the laboratories and see what people are doing.

Rashmi:
But along that same vein, John, the way we're talking about it, just the process of doing this research and investing in something like this for 10 years to me sounds fairly expensive. So where is the innovation in this field coming from? Is it restricted to large organizations that have deep pockets? Where would you look for that innovation?

John Hennessy:
Yeah, it is very expensive. I think traditionally semiconductor companies have devoted a lot of money to it. I think the big change that's occurred in the industry over the last 5 to 10 years is that you now have a number of big vertically-oriented companies, Google, Facebook, Amazon, Microsoft who are also playing in the basic technology arena, with quantum and other fundamental breakthrough technologies. They're doing that because they see that as critical to their longterm success and their ability to innovate.

Rashmi:
Do startups have a chance at all in this? How could they make this a viable option for them without the large amount of funding that they might need?

John Hennessy:
I think they have a chance at it. If you can find an interesting technology, which might well be a university spin-out initially, universities are still doing lots of very fundamental technology work. You'll need several hundred million dollars even to get to the point where you can actually demonstrate a first product. So it's challenging, but Moore's Law and the importance of the integrated circuits industry to the entire information technology industry is so critical that if you really had a viable technology to either replace DRAM, to get back to something like Dennard Scaling, as Dave mentioned, you could find the money for it.

Dave Patterson:
Yeah. One of the changes, we used to bemoan the fact that software people get startups, but hardware people can't. That's no longer the case. This excitement around machine learning

accelerators, I think there's a hundred startup companies that are building their own version of that. So I think conventional wisdom has changed, that there are opportunities for new companies around hardware. So far it's been in this accelerator space. But I think if these are successful, I can imagine continued investment around more innovative memory technologies too.

Rashmi:
Got it. I want to continue along that path, Dave. As I was preparing for this session, I listened to one of your lectures where you spoke about the proliferation of ML papers being the only count that follows Moore's Law today.

Dave Patterson:
Yeah.

Rashmi:
I thought that was funny. But indeed, the world is all about ML and AI, from software job opportunities to startup ideas. You had spoken about hardware-enabled AI. Would you care to elaborate a little bit on that?

Dave Patterson:
Well, I'm recording this right now at Google in the Brain Group, which is one of the leading groups in the world in this area. These are the type of colleagues you want if you're a computer designer. These are thirsty people. They want as many completed cycles as we can build. The ironic thing about it is it's right as Moore's Law is slowing down, where you can't just get the next processor from Intel and get this doubling or tripling performance by just giving them cash. Exactly when we need an explosion in computing to explore this exciting field of machine learning, computers are slowing down. So this presents a tremendous opportunity for people who are the age when John and I did our RISC work to invent brand new fields of how to build hardware that can execute machine learning well.

Dave Patterson:
One of the exciting things about it is that not only do we need new hardware ideas and compiler ideas, the field of machine learning is still changing really fast. So it's like skeet shooting, except the skeets have rockets associated with them. It's really difficult to know exactly what the right thing to do. But like you said, if you read the newspapers, it's really important thing to do. So these are really fun times.

John Hennessy:

Yeah, and I think it's somewhat reminiscent of an earlier time with the sort of explosion of micro computer companies, companies that were taking microprocessors and building computers for a whole range of uses, before the desktop became the primary motion. Or even earlier than that, the mini computer revolution. There were 16 mini computer companies in the US at one point. Now there are none, but for a long time there were two or three big dominant players. I think we'll see the same thing here. Lots of creativity and a few of these companies in the machine learning space will become successful companies going forward.

Rashmi:
I'm definitely looking forward to that. But as we are talking about this, one of the areas that we haven't spoken about is really about security. When we have so much computing that's happening at the edge, we have so much data that we're collecting. Security is a huge consideration, especially in the case of, when I think of security, I think about somebody hacking into my bank account to steal money. But now we're talking about the world of automotives, advancement in that technology, which is creating new risks. You could take over my car and cause catastrophic damage. So could you speak a little bit to that in terms of what's happening in the world of security enabled by hardware?

Dave Patterson:
Yeah. How much time have you got?

Rashmi:
We've got all the time in the world. Go for it.

Dave Patterson:
I think, I'll speak for John, let's let him say it too. But I would say we've been in this field for four decades, right? I think security is one of the embarrassing parts of our field. There's so many things we can be proud of and how it's saved lives and spread education and enabled people. But security's just embarrassingly ... how bad it is, that we're enabling criminals all over the world to hold stores ransom. It's really embarrassing how bad it is. So I mentioned RISC-V, which is this open architecture, which is different. That means everybody in the world can work on it.

Dave Patterson:
In the past, attempts at security because the instruction sets have been proprietary, have been limited kind of to those companies. They'll announce some new feature, kind of slowly roll it out. With RISC-V, everybody can work on it and with field-programmable gate arrays, you can have implementations that runs slowly, but fast enough you can connect it to the internet.

Dave Patterson:
So I'm actually giving a keynote talk here in Silicon Valley to something called HOST, which is the Hardware for Security and Trust, I think it's called. My keynote address is going to be, it's time for the security community to start creating things that are supposed to be secure rather than just complaining about other people's hardware as being insecure. So I think that community needs to shift to be doing synthesis, rather than it's largely been a criticism. I'm hoping that if we can innovate in the hardware and the software, we'll make some progress on this terrible problem we have.

John Hennessy:
Yeah, this is a really difficult area where I think we don't even use all the technology that's been developed in the research labs around the university, around the world. I think we're still fighting to find that balance between building a system that is sufficiently secure, but which preserves ease of use, right?

John Hennessy:
How many of us use the same password for many different accounts? And if we change it, if we use different password for every account, we inevitably forget some of them. Then we have this crazy way to get the password back, which isn't particularly secure, because somebody can probably figure out how to fish it in many cases. So there's a lot of work to be done here.

John Hennessy:
I think the big change that's occurred is 20 years ago you sent out spam or you sent out an unacceptable message, you became a social outcast. Now we get more spam than we get legitimate emails. And people are actually out there trying to steal information, trying to steal accounts, trying to steal money. So I think we just have to double down and really find ways to get better security but still make it easy to do the right thing to have a secure system.

Rashmi:
Got it. Yeah. I think both of you touched upon one piece of information that I wanted to sort of dig deeper into, which is really about collaboration. Whether this is about collaboration between the security experts and the hardware experts, who are actually building these systems, or between universities and academia and industry, how do we foster that? Do you have any ideas? I know we've tried in many ways in the past. In some cases there's been a lot of success. In some cases those conversations are still difficult to have.

Dave Patterson:

I'm very excited about the ... open source software is now mainstream. 20 years ago it was kind of a backwater thing that maybe universities, funny people at universities used rather than ... companies around the world collaborate to build software that we all use every day. It's a thing I brag about. This is a positive of our community. It sounds Pollyanna-ish, but this is the real world. They found, there was a person who used to be at HP, and I think he said to run the HP operating system, he spent $300 million a year. But if he ran the LINUX part, cost him only $3 million a year to maintain it. So companies spread that development cost across lots of companies, and then they share the software and work on it together. The RISC-V community's talking about bringing that same ethos to hardware.

Dave Patterson:
So I see this next decade, plenty of opportunity for companies and researchers and people all over the world to collaborate around the open source movement, which is a main line technology development. It already makes economic sense to collaborate with competitors and with universities. So I'm hopeful this opportunity will lead to tremendous collaboration.

John Hennessy:
But I think the point about collaboration across various boundaries, from hardware to design to software is the key one. After all, a system is only as secure as the weakest link. So you have to think about security as an end-to-end function, which means you're talking not only to hardware designers and architects, but also to operating system people, to people who are building metaware, and through other levels. We've seen the break ins occur because they exploit one little difficulty in this large software stack that's become extremely complex. We need to think about how we get that, increase the security of those systems.

Dave Patterson:
Yeah, I agree with John. I just, like I said, I'm hopeful that the people who care about security, which are in all of these disciplines, will start trying to create secure systems. It's easy to get depressed to think it's unsolvable. If you go to these meetings, it's like, "Well this problem will never get solved given the legacy around a piece of hardware, piece of software." So I'm hopeful that they'll start getting into the ideas of creating things that are secure, using ideas like formal verification of Kernels and formal verification of hardware and more exotic ideas like I think there were some people at Michigan that changed the instruction set encoding every 20 milliseconds and stuff like that. But just starting to experiment, put something out there, offer rewards for attacks, and see if it's arise. Otherwise it's like global warming, right? It'll be like, "Well this is a problem we ought to solve. It doesn't seem like we solve it, and it will just keep getting worse.

Rashmi:
Oh, absolutely. And we're talking about complex problems, we're talking about technology. I'd like to switch gears a little bit and also talk about technical careers, related to what we were just talking about. Both of you come from academia and are now deeply entrenched in industry and you make that transition seem [crosstalk 00:21:26].

Dave Patterson:
I don't think that's right about John.

John Hennessy:
I have a foot in both camps.

Rashmi:
All right. But we are also talking about complex systems and development that's going to get more specialized. I'm wondering if you have any advice for practitioners who are our listeners, who are trying to stay abreast of the latest changes in their industry and are also delivering against a business goal. How does one sort of future proof their career and job?

Dave Patterson:
I'll let John take that one.

John Hennessy:
Well, I think you hit the nail on the head. I mean, this is a field that continues to move quickly. Any of us, if we went to school and didn't keep up with what the changes in the field that were occurring, would fall behind. The good news is increasingly, there's lots of ways to learn online now. Lots of course material and lots of opportunities to learn different things. If I look at the Google course on machine learning, boy, I went through that course and it had great things to help you teach how to ... a crash course in machine learning. So I think there are things like that that we ought to be doing. And it's hard. I mean, people work hard and then you tell them, "Well, in addition to that, you've got to keep your skills up." But I think that's the reality of what it's like in this field right now.

Dave Patterson:
I do remember when I was a graduate student, my research contract had ended. So I started working at an aerospace company to support myself and my family while I finished my PhD. I was struck how the people at that aerospace company just coasted on their knowledge. It was very different from the university. I ended up working almost full time there for a year or so. And then I went back to the university, it was like my mind was opened again.

Dave Patterson:
These days, it's not like the IBMs of the world, well, you can stay there for your whole career and they'll assure you a job. That's long gone. So it's up to you to be on top of the latest technology to be marketable. So you have to factor that in. The good news is like John said, there's tremendous resources online. So these massive open online courses, where great universities all over the world make this stuff available pretty inexpensively, intended to be able to pick up in small doses at small time. It's on you to stay on top of technologies. If you're at a university, there's less excuse not to stay on top of technology because everybody's trying to teach. But if you're going to have a long career, you need to learn about the latest things that are going on.

Rashmi:
Yeah, absolutely. But do you feel like some of the online courses that are available are more sort of geared towards software professionals? Or is it just my lens that I've only been looking for those?

Dave Patterson:
Probably true, but I think the massive, that's probably the MOOCs, so-called massive online courses. MOOCs are on just about every topic, I would think. I know there are architecture ones.

John Hennessy:
Yeah. There are a lot of beginning courses, but there are also more advanced courses as well that are in new fields, whether it be security or machine learning. Machine learning, there's a lot of them in machine learning. Some of them have quite good material and lots of open exercises, because of course in this field, you just look at the lecture, the online lecture, you will not learn the material. You have to actually have some practice with it. Nobody learns programming by just reading a book about how to program or watching a lecture on how to program. It's the same thing I think in these other areas.

Rashmi:
Got it. Yeah. I wanted to talk about one more thing as we're coming to the close of our conversation. Your contributions to the world of computing has been immense and I'm sure some of that is attributed to the power of your partnership. I mean, you guys finish each other's sentences. You know sort of how you ... I mean this interview has been so simple because of how well you work together. How does one go about seeking a partner to work with? What should you be looking for? How much of it is serendipity?

Dave Patterson:
Well, I can say that I was just telling my wife how fortunate I was to find John and be working with him all these years. I think we both grew up on the opposite coast. I was kind of a surfer. He grew up in New York. Both families are large families, they're religious. We both went to public schools, did our PhDs at not the most prestigious schools in the United States. So we're kind of similar. I think we both have pretty good common sense. I think we make good decisions. So even though there would be no particular reason why we'd be pals, we had the same worldview or something and I was just really fortunate to find him to work with. So I would, I guess, look for people who you can get along with, who seem to be smart people with interesting thoughts, and are nice people, I say. I certainly discourage people, if somebody thinks they're the smartest person in the room, my advice is to run away, because that's not going to be somebody you want to work with.

John Hennessy:
I think we've had a great partnership. I think some of it was cemented early on when we were both working on the early ideas. They did not gain an audience readily in industry. In fact, quite the opposite. Many people in industry were quite antagonistic. So we found ourselves on the same side of the line. And then when we to begin writing the book, I think we, in our initial meetings, we found that we agreed on so many things. I think one of the really magic things, besides our personal relationship, we're able to help each other do better. That really means that we can work as a team and get something done that we couldn't do as individuals. I think that's a remarkable opportunity and it's been a great experience.

Dave Patterson:
Yeah, it's made my career. I would say, I think both of us are naturally collaborative. I don't think I've written very many sole-authored papers.

John Hennessy:
Right.

Dave Patterson:
I don't think you have either, right?

John Hennessy:
No.

Dave Patterson:

I mean, most of the stuff we do collaboratively. This is a team sport as far as I can tell. I think early in the computing industry, one person could take on a problem and do it all by themselves. I think that's less true today. It helps to be able to work well with other people, to be able to pull this off. We couldn't have written this book by ourselves. I don't think.

John Hennessy:
No. There's no way I could have possibly finished it. And it wouldn't be as good.

Dave Patterson:
Yeah. It wouldn't be as good. Yeah. Because the process was we took time off to work on it, and we'd meet three times, twice, three times a week?

John Hennessy:
Yeah.

Dave Patterson:
And we'd say, "Hey I got this idea. And what do you think of that?" And we'd read through it and give feedback to each other on it and make it a lot better. The ideas evolved quickly and without somebody to ... you would feel guilty if you didn't get the work done, because you're going to see him the next day, and then get really great suggestions how to make it better.

Dave Patterson:
I just read an article about this somewhere. I mean, they're talking about Jeff Dean and Sanjay Ghemawat. It was in the New Yorker and in that article, there was a kind of a theory about how two people can be much more powerful than two times one, and that if you find partnerships like that, they can do extraordinary things, that two people seemed like the knee of the curve of the most effectiveness for innovation. I can't remember. I think that's the right reference, but I don't remember exactly what their argument was. But they had a lot of examples of that. So if you can find that, that'd be great.

Rashmi:
Absolutely. I mean the power of collaboration and the power of feedback, I think very valuable inputs. I will definitely look for that article and put it into the show notes for our listeners so that they can actually go up and look at that.

Dave Patterson:
It's called The Friendship that Made Google Huge.

Rashmi:
Awesome. Thank you so much for that. So one other thing that I wanted to ask you is would you share with us anything that you may not have shared in another interview? I know you do a lot of interviews, but who are John and Dave when they're not computer scientists?

Dave Patterson:
Okay. Well, I love sports. I actually wrestled in high school and college. And still, I think what makes me happy in a week, versus brilliant idea or playing well at soccer, it's a tough call, really enjoy that. I'm a big sports fan. I'm sitting here recording this a few days before the Super Bowl. I'm sitting wearing my 49ers jersey. If you can see, so I'm a big sports fan as well. Both of us are very family oriented. I'm fortunate to live near my kids and grandkids and spend a lot of time with them. I've been married to my wife for 52 years. John's been married for a little while.

John Hennessy:
Yeah.

Dave Patterson:
Not a real number like [inaudible 00:30:06].

John Hennessy:
I think Dave's love of sports is one axis. Mine is probably culture and travel and the arts. My wife and I are big arts fans, so we [crosstalk 00:30:23].

Dave Patterson:
You read [inaudible 00:30:22] books, right?

John Hennessy:
And books and I'm a voracious reader. I'm a voracious reader. That's been my other passion. But as Dave said, we're both fortunate enough to be grandparents and it's the one redeeming thing about getting older.

Dave Patterson:
Well, you should mention John's book too, about Leadership Matters. Is that the right title?

John Hennessy:
Leading Matters.

Dave Patterson:

Yeah. But in there you can find about John's history, and he talks about how he's been such a successful leader. And to me what I picked up was he learned a lot about how to avoid mistakes. He was President of Stanford for 16 years without making a single misstep, and he learned that from reading the books of great people, which I find [inaudible 00:31:06] really interesting.

John Hennessy:
Yeah, agreed. It's much easier to learn from other people's mistakes than it is to have to make them yourself.

Rashmi:
Thank you so much for sharing that insight into who you are as people. It's very interesting to me and I'm sure to all of our listeners. For our final byte, I'd like both of you to answer this question please, which is what is it that you're most excited about in the field of technology maybe over the next 5 or 10 years?

Dave Patterson:
I think when people answer this question, if you think this is such a great idea, why aren't you working on it would be my followup question. So I'm working on the thing I'm very excited about, which is both open instruction set architecture, is bringing the innovations of open source software to open source hardware around the RISC-V effort, and this how do we accelerate machine learning? It feels like machine learning is a revolutionary technology as big as the internet and the microprocessor. You can only tell later, or the worldwide web. I think it's one of those things and we need to figure out how to build the hardware-software systems that accelerate it. So I'm pretty sure for the next five years for sure that's going to be an exciting topic to work on. And then I believe it so much I'm working on it.

John Hennessy:
Yeah, I agree with Dave and I think it's important to understand that this machine learning revolution certainly required people who did the fundamental work on algorithms and back propagation and training methods and other inference methods. But it also required a tremendous amount of computing power to be harnessed. I think the key to going forward is going to be to make sure that we not only have new algorithms and innovation in terms of the methods, but we also have the computing power that's going to be necessary to enable those. With that, I think we're going to continue to be surprised by the rate of progress in this field. I think every now and then, and the fascinating thing about computer science is discipline, is every now and then some part of the discipline just explodes in terms of the rate of progress. I think that's what Dave and I saw in the 80s with the RISC revolution and with what happened in

microprocessors and now it's happening in AI machine learning. That's really exciting to see as a field.

Rashmi:
Perfect. Thank you so much. I mean, so much to look forward to. This has been an absolutely fascinating conversation. Thank you John and Dave for taking the time to speak with us at ACM ByteCast.

John Hennessy:
Thank you.

Dave Patterson:
Thanks for inviting us.

Rashmi:
ACM ByteCast is a production of the Association for Computing Machinery's Practitioners Board. To learn more about ACM and its activities, visit acm.org. For more information about this and other episodes, please visit our website at acm.org/bytecast. That's acm doT org slash B-Y-T-E-C-A-S-T.