

EPISODE 310**[INTRODUCTION]**

[0:00:00.3] JM: The impact of artificial intelligence on our everyday lives will be so profound that our modern institutions will change completely. Employment, government, romance, social norms, all of these things will be appended. To see the signs of this coming, you no longer have to read science-fiction. Every week, there are blog posts, news stories, and videos chronicling our strange, exciting time.

Rob May is an investor in artificial intelligence companies and is the CEO of the A.I. company Talla. Every week, he puts out the Technically Sentient newsletter, which is a compilation of the best pieces of information about A.I. over the past week. Each newsletter also contains a short essay by Rob in which he gives a big picture perspective on what he sees in the A.I. space and how that's developing from his point of view.

This was an illuminating conversation with Rob about the implications of artificial intelligence and the topics that he writes about in Technically Sentient. I recommend checking out the Technically Sentient newsletter, it's a great resource.

I first saw Rob speak at the LAUNCH Scale Festival, which is put on by Jason Calacanis. It's a free event for people who have started companies, and I am grateful for being able to have attended that event and see Rob's talk. After that, I subscribed to his newsletter, and now he's on the show. I hope you like this episode.

[SPONSOR MESSAGE]

[0:01:34.3] JM: Dice helps you easily manage your tech career by offering a wide range of job opportunities and tools to help you advance your career. Visit Dice and support Software Engineering Daily at dice.com/sedaily and check out the new Dice Careers mobile app. This user-friendly app gives you access to new opportunities in new ways. Not only can you browse thousands of tech jobs, but you can now discover what your skills are worth with the Dice Careers Market Value Calculator.

If you're wondering what's next, Dice's brand new Career Pathing tool helps you understand which roles you can transition to based on your job title, location, and skill set. Dice even identifies gaps in your experience and suggests the skills that you'll need to make a switch. Don't just look for a job; manage your tech career with Dice. Visit the app store and download the Dice Careers app on Android or IOS. To learn more and support Software Engineering Daily, go to dice.com/sedaily.

Thanks to Dice for being a sponsor of Software Engineering daily, we really appreciate it.

[INTERVIEW]

[0:02:47.3] JM: Rob May is the CEO of Talla and the author of the newsletter, Technically Sentient. Rob, welcome to Software Engineering Daily.

[0:02:58.2] RM: Hey Jeff, thanks for having me.

[0:03:00.5] JM: I want to start with a discussion of general artificial intelligence. How do you define that term, general artificial intelligence?

[0:03:10.9] RM: People define that sort of two different ways. I think the common way and probably the term that I would also use is it always seems to be whatever it is that computers can't do yet. That seems to be the case. I think there's a lot of times in the past people have said, "Oh, when the computer can play chess, when the computer can win at poker, or the computer can win at Go, we'll have achieved general A.I."

I think we've found the opposite. I think what we've found is that A.I. is in a state where we're really good at solving very specific problems with it, and we're not good at building generalized problems. I would say a generalized A.I. is one that can probably solve a broad range of problems the way that a human can. I don't know if the Turing Test is necessarily the best test for it, but probably something along those lines.

I think it's a difficult thing for us to engineer at this point in history, because I think we've become very used to, I mean, if you think about the software world, what have we always been told? "Solve one problem. Solve it well." All of our engineering practices and product practices and everything else are around doing one thing that's very specific. We're not very good at or used to building generalized products.

[0:04:17.5] JM: Humans like to think of ourselves as generally intelligent, but there are things that we're not very good at. In some ways, humans are that specific narrow sort of intelligence. For example, a human will never be able to sort and rank like a billion webpages, but a computer is quite good at that. Yet, this is a complex, multi-variant task. We might describe it as narrow A.I., but there are many things that go into the process of search ranking and serving search results that could arguably be categorized as general artificial intelligence.

When I think about this, it seems like even the difference between the notion of general A.I. — Beyond the notion of general A.I. is something that is whatever is out of reach of our current A.I. systems. If you look at general A.I. is whatever a human can do, because I think that is an alternate definition.

The difference between general A.I. and narrow A.I. seems to me to be a false dichotomy, because even humans in a certain light only have a narrow sense of intelligence. Does that seem plausible to you?

[0:05:41.4] RM: Yeah. I think that is true in a lot of ways, right? Another way to answer it might be to say that when we build something that can generalize from specific use cases, use inductive logic, and experience the world, and form generalizations from that, it will — maybe then will have a generalized A.I.

I think a lot of what you might be getting at if the human way of thinking is a process, and it's a kind of process that we haven't been able to put into code yet despite the fact that we can put some pretty smart things into software and hardware.

[0:06:19.3] JM: Now, some of the approaches I've heard for how we might get to this general A.I., or whatever is the A.I. dream. Name your A.I. dream, the approach to getting there, some

people say it might be we compose together a bunch of narrow A.I.'s and these narrow A.I.'s can be composed into something that is general enough.

The other approach that I've heard that is somewhat compelling is you write a deep learning system that optimizes for learning how to write a deep learning system. You have recursive effects. When you look at this from an engineering point of view, do you have any perspective for how we are likely to get to this general A.I.? The singularity level A.I.?

[0:07:09.2] RM: I don't. I think there's still a bunch of problems that have to be solved. One is how do you learn from smaller datasets? Some of that is being explored via transfer learning and one shot learning and stuff like that. Newer techniques like probabilistic programming. I think there's — Look, there's still a question that goes on at the neurobiology level or neural science level about what base level capacities are preprogrammed into humans and what things do we learn. There's still a big debate about whether language is innate or not. Do we have an innate grammar? Is it very flexible? There's a lot of evidence on both sides.

One of the things — I wrote a post almost two years ago now about this idea that we still don't really understand the nature of intelligence very well, and it reminds me of taking a physics class and reading about the arguments between whether or not light is a wave or a particle. If you follow that debate, what happened was they basically settled in and said, "It's both. It has properties of a wave and properties of a particle."

I think it's going to take some kind of maybe weird thinking around intelligence to come to a similar conclusion about whether it's innate, whether it's not, how flexible is it? To take that back to your broader question about where it comes from — I don't have any evidence for this, but my gut feeling about all these is I think you're going to see changes in hardware. I think when you look at spiking neuron chips, like what IBM is doing with TrueNorth and some of these other neuromorphic chips that are out there.

I think when you look at a content address of all memories and some of the changes that are happening, I think we have to really make these breakthroughs. I think we're going to have to break the common computing mindset that we have of this separate memory and processing architecture. So much of our thinking is driven by the sort of legacy, X86 chip architecture that

was so popular when most of us who were in the hardware and software world sort of grew up and came of age.

I think some creative hardware breakthroughs and then starting to tool those up so that people can program on those and think in new ways will matter. The reason I say that is because when I went to school in the mid to late 90s, when I was a college, I came out, I was an FPGA programmer, and writing VHDL, or Verilog to describe hardware was actually very different than writing code at that time, because people didn't think of code in 1997 as state machines and being event driven and all of that. It was very sequential, and people thought about subroutines and stuff like that.

I realized early on, because I did both software and hardware, that the way you thought about designing hardware was very different, and there were chances to take some of the ideas from one domain to the other. There's no reason you couldn't build a webpage into an FPGA if you wanted to. Eventually, many of the hardware design ideas sort of made their way over to software if they were useful. I think you're going to see — I think it's going to be driven by changes in hardware that are going to open up new ways of thinking about designing programs and doing approximate inference and stuff like that. I think those things, those technologies as they evolve is what will lay the groundwork for what will eventually become generalized A.I.

[0:10:21.2] JM: I suspect you're right. It's funny how we often talk about, at the software level, how when something — well, I guess this is generally at the product level. When we get a new paradigm, like the smartphone for example, the first smartphone apps, the first smartphone web experiences, were formfitting the desktop web experience into the mobile experience and it didn't quite work right, and then it took us a while to get the mobile experience right for these apps and for the mobile web pages.

Similarly, the deep learning, what we're using for machine learning, for chips right now are GPUs. Basically, it's like, "Well, these GPUs seem to work well for this type of machine learning problem." Now, I think as you articulated, if you totally look at the problem from first principles, you may come to very different conclusions about, "Okay. Is a GPU really what we want to be using for machine learning?" It turns out the answer is probably no.

[0:11:32.6] RM: Right. It's interesting, because what GPUs really do really well is they actually map to neural networks really well, because the GPU is basically calculating, "Hey, for every pixel on your screen, at the next clock cycle update, what should it be?" It's a very simple calculation, but it's a calculation that's performed 10 million times in parallel, or whatever, compared to a CPU that will do 10 million calculations much faster sequentially, but can handle more complex calculations. That maps very well to how a neural network works.

There are other interesting technologies, some of the probabilistic programming languages, some of the ideas like hierarchical temporal memories that are coming out of places like Numenta, and the spiking neuron chips that I think are going to change the way that we think about some of these stuff. I think that stuff is going to be really interesting.

[SPONSOR BREAK]

[0:12:30.2] JM: You are building a data-intensive application. Maybe it involves data visualization, a recommendation engine, or multiple data sources. These applications often require data warehousing, glue code, lots of iteration, and lots of frustration. The Exaptive Studio is a rapid application development studio optimized for data projects. It minimizes the code required to build data-rich web applications and maximizes your time spent on your expertise.

Go to exaptive.com/sedaily to get a free account today. The Exaptive Studio provides a visual environment for using back end, algorithmic, and frontend component. Use the open source technologies you already use, but without having to modify the code, unless you want to, of course. Access a k-means clustering algorithm without knowing R, or use complex visualizations even if you don't know D3.

Spend your energy on the part that you know well and less time on the other stuff. Build faster and create better. Go to exaptive.com/sedaily for a free account. Thanks to Exaptive for being a new sponsor of Software Engineering Daily. It's a pleasure to have you onboard as a new sponsor.

[INTERVIEW CONTINUED]

[0:14:00.2] JM: Yeah, and we'll get back to the hardware discussion a little bit later. I want to ask you about Intel. Before that, let's talk some about Technically Sentient, which is the newsletter you're an author of. I love this newsletter. It's got a tone of links and aggregated content every week. I encourage people to subscribe. It's a once a week sort of thing.

For you, this must be a great excuse to consume vast quantities of information about A.I., and we're in this golden age of content about A.I. There's so much content about A.I. that is interesting, ranging from the technical to the philosophical. It's really just such a cool field to look at, and you're ingesting all of these information. You must be asymptoting towards some very unique viewpoints, because as anyone who goes into a scientific field knows, the deeper you study something, the more unique and well-defined your vision of that field becomes.

Give me some perspective of the unique beliefs that you have around A.I.

[0:15:12.2] RM: That's a great question. Let me put that in the context of some things that I've written in the newsletter over the years. The newsletter is about a year and a half old, and I think people know me well would tell you that I don't have a normal life. I don't pay attention to the things that most people pay attention to, but part of my skill set is consuming large amounts of information about all kinds of things. It's easy to do — I have a unique base to do this from, because I was a hardware engineer. I did a report on A.I. in graduate school. I did a partial master's degree in computer science, focused on A.I. I've been studying a lot of these stuff for a long time.

Some of the things I see that are maybe not that obvious to everybody else, in 2015 I wrote a lot about the A.I. industry and how this platform approach that was very popular was wrong. What happened was with the first real wave of A.I. funding that started in 2014, venture capitalists didn't really understand a lot about it, and so the main signal that they looked for in terms of what to invest in was, "Okay. Do you have three or four smart PhD's who have worked on A.I., or deep learning, or something like that? If so, great. Let's just fund you guys and hope for the best."

It turns out that a lot of times, those people, they don't know how to build companies, they don't know how to go into markets and figure out product needs. Some of them do, just not very many in general. What would happen is they would say, "We don't really have a product idea. Let's launch a platform so that other people can build their A.I. products on it without worrying about the A.I. part."

The problem is if you don't understand the A.I. part, really, then you don't understand how it might change what you might do at the product level. You had all these platforms that came out and they were very broad. The vast majority of them struggled. Some of them got acquired, and some of them pivoted into really good use cases. Clarify, in particular, I think did a really good job of moving into a market and finding customers and raising another round. They did that by harping on some end-user use cases in their content marketing.

I think when you start with new technologies like this, the early companies actually need to be full stack companies. If you look at what Salesforce did as an early cloud company, it wasn't like they launched something that was S3 like thing first and then Salesforce came after it. No. They had to pull off the entire cloud stack, from the infrastructure layer, from the bare metal layer all the way up to the application stack, the application layer.

I think this is what's going to happen with A.I. I think the early A.I. companies, there's going to be some opportunities to build some really big companies, but I think they're going to have to do a lot of stuff themselves. They're not going to be companies that are built on some of these A.I. platforms or NLP platforms. They're going to do most of it themselves, I think.

Then, as people understand the use cases, I think, then the next wave of A.I. will be to start to platform pieces like that out and make it more available and easier for other people to use. I don't think that was obvious to everybody at the time, but I think it started to play out that way, that the platforms weren't the place to start.

I think the second thing that's may be interesting to most people are missing right now is that there's this just love of deep learning, which is great when you have a whole lot of data. Most of us don't have enough data to do anything. I think people aren't paying enough attention to small

data A.I. How do you learn from smaller datasets, because there are many, many more small datasets in the world than there are large datasets?

That's something I see a few people working on my guess is if we did this interview a year from now, you're going to say, "It's really interesting. Deep learning seems to be slowing down and now some new technology, or method, is really picking up the slack."

[0:18:52.7] JM: Yeah, for sure. I agree with that. I also think that — would you say there is, yes it's disheartening this sense that, yes, maybe we don't have this large enough datasets to pull off deep learning. What is encouraging about that is we know how to make data. We know how to produce large volumes of data for specific domains.

If we are entering this world where we're no longer CPU bound, we're no longer looking at Moore's Law as the constraint on how fast technology goes, we're looking at how do we build the right pipe to shove data into and how do we get enough data to shove into that pipe? That's a very different question than, "Can we add more transistors to the same surface area, the chip, quickly enough?"

[0:19:44.7] RM: Yeah. It definitely is. I think a lot of what you're going to see is there's going to be this mismatch between investment opportunity and the future of A.I., because this last wave of software of the cloud has been going on since, I don't know, 2006, 2007. We've become used to a certain way of thinking about how you start a company? What that consists of. But these things are being broken down, and so it's really only going to be your sort of crazier VCs, the ones who are a little more out there who are going to catch the next wave of companies that are going to be really big, because that wave of companies is going to be bringing some of these different ideas and some of these unique ideas, new ways of doing things.

If you think about the things that you hear, "Oh! Starting a company is never been cheaper." No. For A.I. companies, you have to acquire datasets. Sometimes you have to build a product that will acquire datasets so that you can do something with that dataset to build the product you really want to build. Data scientists are expensive. You need data pipelines and data cleansing. They're much more expensive than starting, particularly, these sort of like consumer social

companies. You used to be able to do a lot on a million and a half dollars; you'd be lucky to get an A.I. prototype on that.

[0:20:55.0] JM: My friend, Auren Hoffman, is working on a company called SafeGraph that is doing this dataset democratization thing. Initially, when he started to talk about this publicly, I didn't quite understand why is this dataset democratization, data sharing thing important. As I've heard from people who are deep into the industry, and now I'm hearing it from you as well, this dataset, getting access to these datasets is, oh, it's burdensome. If we could have mass adoption of datasets and democratization of datasets — A lot of these is rooted in the difficulty of anonymization, I think. If we can have anonymized datasets, that would be great. It's so easy to deanonymize a lot of data.

I know, it's a lot of interesting problems there, and what you're talking about with the investing and how you really need these savvy, gambly, perhaps, investors to come in and make these kinds of investments. It's an interesting field to be in. It must be interesting for you as an investor as well.

[0:21:55.9] RM: Yeah. I've done 17 angel investments now in the last year and a half and I would say, I think, maybe 12 or 13 of them are related to A.I. I've done pieces all up and down the stack. I did a low power voice neural network chip, a company called Isocline. I think they're an amazing company. They're going to do awesome. I've done a bunch of different pieces of software. I saw a company called reality.ai out of New York which is sort of like middleware, helps you — it makes it easy to design and deploy your own machine learning models for sensors that are tied to physical phenomena that you're trying to detect in sort of IoT devices, and then a bunch of application layer stuff. Like end-user A.I. for, lots for sales and marketing, some for legal and different stuff like that, some machine vision application.

I've loved it, and part of what — People ask sometimes, "Oh, how do you have time to do all of this?" If I like the team and I like the idea, then I'll do the investment. I don't do a whole bunch of diligence, competitive analysis, and everything else, because I do it as much, obviously, I'm hoping for good financial returns, but I'm also betting in a sort of rising tide, lifts all boats theory for A.I. and I'm looking to learn more about the industry and see the inside of how some of these companies work and see what I can learn from them and the stuff that they're doing.

[0:23:08.2] JM: Sure. It's also interesting, because maybe some of your downside risk that might otherwise be — Execution risk is not as big of a deal, because a lot of these A.I. companies just get acqui-hired, because there's such a struggle for talent, which might protect some of your downside risk.

I do want to talk more about company strategies and how A.I. looks at a technical level today. I want to ask you a little bit more about some far-flung societal questions, because you spend a lot of time thinking about these. I've looked at some of your thoughts on them. A.I. risk is heating up as a topic of conversation. For a while, this was something that part of the community was really saying, "This is just not something we need to worry about at all." Then, obviously, people like Elon Musk and Sam Harris have started to sound the alarm, Steven Hawking, Bill Gates.

I think, at this point, it's fair to say that A.I. risk can be categorized as something that is sort of like a tail risk from our current point of view, but it's something that we're going to have to tackle eventually. In some sense, it is an inevitability. Maybe it's an inevitable tail risk. That's another way you could look at it. Eventually, we are going to be dealing with a situation where there is some percentage chance of A.I. turning us all into staples, or whatever Nick Bostrom says.

[0:24:35.3] RM: Yeah. Paperclips.

[0:24:36.4] JM: Paperclips. That's what — I knew it was some office supply. What's your take on this? A.I. risk is a thing, but I guess, more, to what degree is it a thing from your point of view and what can we do about it?

[0:24:48.9] RM: Yeah. Boy, I struggle a lot with this question, because I'm not necessarily convinced that an A.I. will be nefarious if it happens, and I don't really have a strong prediction for if that's 10 years away, or 25 years away, or 50 years away. I do think it will happen at some point that we'll build machines smarter than ourselves.

I buy Sam Harris' argument, in general, that while we could augment ourselves, we probably don't augment ourselves faster than we create an A.I. One possibility is, some technology like a

neural lace, or some other neural technology takes off and we're able to keep humans on par with A.I. and it's not a big deal.

I think at this stage, it's difficult to assign probabilities to what might happen, and there's probably some things that we don't understand. I know in my own life — I look at my business career and I look at one of the biggest business development deals that we ever did in my last company. We had this meeting and we highlighted all, "Okay. We're about to close this deal. Let's go through the list of reasons that it might fail and risks what we might have." We've spent two hours and we drafted 20 of them and we felt like, "Okay. We've covered all the things."

Sure enough, the deal fell through for a reason that wasn't on the board, which was the company got their CEO replaced. There was a shakeup, a change in strategy. Everybody that we knew and worked with had left. There was no more support. That stuff happens. I think it's difficult for us to understand sometimes the unknowns unknowns of where these things might go. There's so many things that could happen, and I think the way it plays out depends on does generalized A.I. come from the military? Does it start in the United States? Does it start in a robot that has to have some grounding? Or is in a box that's just connected to the internet?

You could make the argument that, "Hey, before we have generalized A.I., we'll have pretty damn good military robots that won't be entirely smart, but it will be relatively autonomous." Those things could be programmed by somebody to wipe us out before we even get to GAI. So there's a lot to that. Actually, I spend time thinking about different ways it might play out, but I don't have a strong opinion for what it might be or what we could do at this point to prepare for it other than just keep having the discussion. I believe the saying that chance favors the prepared mind, so at least if we thought about it, have some good ideas maybe.

[0:27:09.7] JM: Agreed. On that matter of chance favoring the prepared mind, we are hurdling towards this highly automated world. Obviously, the A.I. conversation quickly shifts to the basic income conversation here. You have written about a variation on this universal *Basic Income*, universal basic robots, which is basically this idea that instead of just giving people a stipend, you give people, perhaps, a general robot that maybe can till the land and give them farmland and stuff like that. It can clean their house.

My response to this was the same response I had to the first time I read *Basic Income*, which was like, “This is preposterous. Why would this ever work?” Of course, my opinion has completely shifted on *Basic Income*. Now, I see this like, “This is something that’s very plausible.” Explain universal basic robots.

[0:28:02.6] RM: It came from a discussion that I was having with some various friends of mine and the real challenge here with universal basic income is I think it actually — I think it’s hard to pull off, because what you’re seeing as we transition to more automation is you’re seeing a change the shift of income that goes to capital and the shift of income that goes to labor. You’re seeing more and more of the income go to capital instead of to labor.

Universal basic income doesn’t fix that underlying structural problem. All it does is say, “Hey, you capital owners, you’re getting too much. We’re going to tax you and give it to labor.” You’re still putting — Owning economic creation and production power is more than just the cash that it generates. You have the ability to strike. You have the ability to say withdraw your assets from the market and do all these kinds of things.

I think if that’s the approach, A; all the capital is going to be concentrated in a handful of people who own the robots and all of the A.I. and then we’re just going to give everybody else a UVI. I think you’re still going to struggle with all kinds of problems, because the same people are still going to have — Power is going to be hugely concentrated.

The problem that you really want to solve is you want to say, “How do we distribute capital to more people?” There’s a couple of ways to do that. One would be to say, “Okay. Look, if you’re an A.I. company, you have to put 10% of your stock in a common pool that goes to the government, or it goes to some group that then gets paid out. You have to pay out 10% of your income in dividends, and that goes to people,” or something which, I know sounds a little bit the same, but it actually starts to change the underlying structure.

As I started running the numbers on some of those things, they create a lot of their own problems as well anyway. That’s when I just sort of have this thought, “Look, why not just — If the smart robots are going to do everything, why don’t just give everybody a robot instead?” The government, rather than paying you \$10,000 or \$20,000 a year, or whatever it will be, we just

give you a robot one time and it seems like it would be more politically palatable, right? It's fair to the left in the sense, "Hey, everybody gets the same thing. When you turn 18, you get your robot." It's fair to the right, because what you do with your robot is your choice. If you say, "I want my robot to basically grow pot so I could smoke it up all day." That's one thing. If you say, "I want my robot to build these things and then charge other people to use them and then use that money to acquire more robots and do these other things," then you can still amass wealth if that's what you're interested in.

If you want to just move off the grid and live in rural Montana and have your robot be your farmer, and hunter, and defender of your land, then that works for you as well. These, hopefully, will be pretty powerful intelligent robots and they don't require much power and whatever. I got a lot of really good responses to that, some funny, like, "Oh, great. You want the government to build robots. They're definitely not going to work. They'll be the worst of the worst and they'll be overpriced and everything else."

Then, I got a lot of criticisms about why it wouldn't work, and I got a lot of sort of affirmations about things people liked about it. It was one of my popular, I think, commentaries in terms of the amount of feedback that it solicited. That's the general idea. The problem I'm really trying to solve is how do you more equitably distribute capital rather than income?

[0:31:18.5] JM: The other government things that you've written about, the intersection of government and artificial intelligence, and I think you wrote recently about this legislation that when it applies in the U.K., I think about how the artificial intelligence companies — Basically, trying to legislate against A.I. stuff is going to be really difficult, especially with the technical sophistication level of our current legislators. This is worldwide, our current legislators.

You did point out that the legislation of liability, like if your A.I. screws up, if your self-driving car crashes into a guard rail, you're liable. The person who wrote the A.I, if we can track it, is liable. It's interesting first step towards A.I. policy. I really hope that if in the U.S. we survive this presidency, it does seem like the technical side of our industry has woken up and is saying, "Okay. We've had enough of these politicians, let's step in and start to do things politically the way that we do them on the tech side, which is more well-reasoned.

I think it's too early to tell how that assuming we make it over that four-year, eight-year cliff and survive. I think it's too early to tell where government is going to go, but one question I do think is interesting to examine at this point is what is the public perception of A.I.? When you think about not just Silicon Valley, Boston, New York, these technically elite cities, does the public have an idea for how fast A.I. is moving?

I talked to my parents, for example, or some of my less technical friends, and they seem like they don't get it all. They don't understand that this is like a really incredible time of how fast the pace is moving. Does A.I. even enter into the concept of the public perception? What's your sense of that?

[0:33:27.0] RM: Yeah, that's a great question and there's a lot in there. I don't think your average person thinks very much about A.I. or realizes where we are. I think they probably even still feel like self-driving cars are a long way off, and they probably think I wouldn't let a car drive me. I think it's problematic. I think the fact that so many tech jobs have clustered in maybe three major hubs and another four, five minor hubs around the country, is a problem for a few reasons.

One is that, sort of as you mentioned, there's a small number of people who are seeing and creating the future and then everybody else has to deal with those. I think a lot of what you've seen in this election is that there are whole slots of the country that has a different set of problems that tech is not tackling necessarily.

I think tech is sometimes removed from some of the problems that we create via automation and even things like social media, maybe, addiction, or some of that stuff. Then, you have this — Tech has typically been in this unregulated, "Hey, we don't care about government. Leave us alone," sort of mindset. As a result, you don't have a lot of strong tech understanding amongst politicians or government. Not a lot of tech people have become politicians.

In fact, we haven't really participated much in the political system other than just voting. Gone to Washington — Google and Facebook and some of those companies do some lobbying now, but that's all relatively new the last four or five years. I would say until the sort of Snowden Revelations that really sort of shook the foundations of the tech community, I think tech was

almost entirely out of it in terms of what they thought or cared about policy. That's been one problem.

The second problem has just been the fact that, I mean, I grew up in Kentucky, so I understand the mindset of someone who lives in a very rural place where —

[0:35:27.7] JM: The Hillbilly Elegy.

[0:35:29.9] RM: Yeah, your nearest neighbor might be a mile and a half away. You might actually hunt some of your food, which I know sounds crazy to my friends here in Boston. You think about the world very differently when you lived down in Cambridge, Massachusetts and you're like, "Look, there's no place to park. We're all jammed in here together. It's tight. We've got to find ways to just get along and make the best of this sort of crowdedness and be nice to each other." Versus, "Hey, I farm and sometimes I don't see anybody outside of my family for a couple of days, and I'm incredibly self-reliant. Yeah, I don't know what I'd do if I didn't own a gun."

Not having those people live next door to each other and have to interact every day, even if you disagree with somebody, like if you have to go work with them every day, or if they're part of your family members, you typically, you disagree, but you're a little nicer about it. You become a little more tolerant. I think the tech divide and the geographic divide that's been driven by technology job creation has also enhanced some of the viciousness on both sides of the isle.

To get back to the sort of your start of your question, yeah, there are two interesting trends in Europe. One is this law that they're debating that says if an A.I. algorithm makes a decision about you, it has to be able to explain why that decision was made. "Oh, you were denied a credit score, or you were denied a credit card. Why?" Some of these models aren't very introspectable, and we can't really explain. We can't say, "Oh, because this node has a .3 weight." That doesn't mean anything.

That's going to do one or two things. It's going to drive people to solve the introspection problem of neural networks, and there are some interesting at MIT and other places being done on that. It's going to lean people away from some of these A.I. solutions to other solutions, like

probabilistic models, Bayesian models are a little easier to understand and introspect. Some regulation like that might drive a different type of innovation.

The second point is this sort of other stuff that you were talking about, which again is stuff happening in Europe where they're saying, "Okay, well how liable is the creator, versus the user? If one company, company A makes the robot, I buy the robot, I rent the robot to you and the robot does something bad. How much are we all liable if the robot is autonomous and learns on its own?" They're sort of saying, "Well, you're liable to the percent that you had something to do with how the robot got to this state. How much you programmed it, or taught it, or whatever." That's a very interesting, but I think decent approach for where we are in the lifecycle of this stuff.

[SPONSOR BREAK]

[0:37:57.7] JM: Simplify continuous delivery with GoCD, the on-premise, open source, continuous delivery tool by ThoughtWorks. With GoCD, you can easily model complex deployment workflows using pipelines, and you can visualize them end to end with its value stream map. You get complete visibility into and control of your company's deployments.

At gocd.io/sedaily, you can find out how to bring continuous delivery to your teams. Say goodbye to deployment panic and hello to consistent predictable deliveries. Visit gocd.io/sedaily to learn more about GoCD. Commercial support and enterprise add-ons, including disaster recovery, are available. Thank you to GoCD and thank you to ThoughtWorks. I'm a huge fan of ThoughtWorks and their products including GoCD, and we're fans of continuous delivery. Check out gocd.io/sedaily.

[INTERVIEW CONTINUED]

[0:39:10.7] JM: Let's talk at a more substantive level. It's very fun to talk about the future, but thinking about the present, especially from a business context, it might be helpful to people. You are working on Talla, which is an A.I. intelligent assistant. It works with chat interfaces like Slack, and HipChat, and Microsoft Teams. Since you're building this chatbot, you are closely — You

think very deeply, I can tell, about the pace at which this, the chatbot interface is moving. Obviously, the voice interfaces are a top of mind for you as well.

Talking, I guess, more generally about all of the companies in this space, and you write about all these different companies. I see kind of like, at least, three — For the big companies, I see three trends. I see there's these — There's a Google. Google type companies that are offering the building blocks of A.I. to developers. There are the hardware companies, like Intel, that are making specific chips for machine learning, like we discussed. Perhaps, you will have a cloud offering where you can run your deep learning jobs against Intel's cloud, with their specific chips.

Then, there are companies like Salesforce that are building A.I. into their products. Then, there's all of the smaller companies. We've seen lots of innovation in A.I. from smaller companies. Many of these companies are getting acqui-hired. How should a developer look at this space? Whether they're a developer at a smaller company, or a bigger company? Perhaps, in your work at Talla, how does the work of a software engineer change in this space, and how does it depend on the side of the company, the type of company that they're working in?

[0:40:57.4] RM: Yeah, a lot of great questions there. I think we think a lot at Talla, we think we have to build full stack, and the reason that we are building most of things in-house — well, there are two reasons. Number one; if you're building for the enterprise, you ultimately want to be able to offer an SLA. We've looked at tools like api.ai which is a great tool, but they are a startup too, and if we want to offer an SLA someday and they can't offer us one, then that makes it difficult for us to give 49's to our customers or uptime, or whatever they need. You don't want to be relying on a third party.

That applies for other types of technologies. We looked a company called message.ao, which is another great company that will help port your bots to lots of platforms. Again, we felt like we wanted to own those connections as well. I also think for enterprise use cases where data security matters and you're not using public datasets, you're using stuff that the company has. I think there's going to be a lot of chances to tweak your solution and be very — To really get some gains by controlling more pieces of the stack yourself. You can do some optimization across the type of data that your customers have.

Which, when you're building a horizontal platform, you have to be very generic. You have to support a lot of different use cases. You take something like Amazon DynamoDB. It's probably fine for most type of scalable, sort of NoSQL database use cases you need. For some subset of use cases, it's probably still pretty high, like 20% to 30%. You're probably like, "You know what? It would be better to run our own clusters of Mongo, or Cassandra, or whatever you want to do, and tune them to our own needs. We'd probably get better performance and at scale, maybe a cheaper price." It's more capital to get started and everything else. Again, when I talk about some of these companies aren't as cheap as they used to be, that's part of the reason.

We have a lot of these discussions about what to build and what not to build, and I think if you're building, particular for enterprise A.I. right now, I think you probably want to build more of the stack yourself. Probably in three years that's not the case, but that's sort of what I think it is today. I think if you're an engineer, I think you're just looking at — I think it depends on the kind of company that you want to go work for and how deep in the machine learning stack you want to be with respect to the tools you should learn.

[0:43:07.1] JM: Okay. This is really fascinating, because the build versus buy question — Before I start doing a lot of shows about A.I., when most of the earlier shows I did were sort of, you're building a web app, you're like an Airbnb company, or you are Slack, or something. Well I guess, maybe Slack is arguably an A.I. company. These companies that are just like a web app, basically. The argument would always be, "You should always buy. Don't build. Buy where you can, because it's going to save you time."

But I think what I'm hearing from you is that in A.I., perhaps you want to build, because the performance is so paramount. Just the other question, and you can choose whether to discuss this or not, but Google's hosted machine learning, for example. They take care of something that's very unique, which is like the scalability aspects of running your machine learning jobs. I've heard very good things about the Google cloud machine learning framework, but perhaps that's only useful for a very specific domains. Yeah, I guess, explore the build versus buy question when it comes to A.I. companies.

[0:44:09.6] RM: Yeah. If I was building most sort of general web apps today, I would agree with you right. I would buy almost everything. The only you want to build is this small specific piece that's relevant to you. Today in A.I. there's a couple of reasons that you want to build, and one of them is just the maturity issue. It's back to the SLA stuff that I mentioned before, which is the ecosystem is immature, so you can't really rely on the other pieces of ecosystem to be as good as you need at this point.

You don't want to get in a situation where you pick a platform and you're one of only 30 or 40 early stage customers they have. You don't know how their product roadmap is going to develop and if it will stay useful to you, compared to when they have 10,000 customers and they're curing similar signals from everybody and they're announcing their product roadmaps in a little bit in advance and all that.

You introduce a lot of risk by doing that. Plus, you don't know yet which are these startups are going to be standalone and who's going to buy who, and so you have this risk that if Google buys somebody, do they shut them down, do they change their roadmap? Do they —

[0:45:11.2] JM: The parse risk.

[0:45:12.8] RM: Yeah, exactly. Yeah, the parse risks. You have a bunch of risks like that. You also have this risk where I think sometimes — I'll give you an example. If you're working on NLP, I'm not sure that you know yet what the most valuable piece of your stack is and which pieces are going to be easily available via open source and everything else. You can make some guesses like syntax parsing is not going to be where you're going to make your stand, because that's a pretty well-solved problem.

I think it's something that you got to spend a lot of time thinking about. I think at this point, it still makes sense to build a lot, because the battle scars that you get and the things that you learn — If nothing else, if you decide to go buy at some point in the future, you've really learned a lot about product you need. It's different, because if you need to go choose a NoSQL database, there's a lot of people that understand those, that used them in the early days and can tell you, help you make a decision, and all that's relatively well understood, and a lot of the A.I. stuff is not.

I think you want the opportunity to be able to tweak things for your own needs to be able to, if there's some brand new idea that's coming out of academia or something, try that out. When we started Talla, for the chat piece, we tried IBM Watson, we tried wit.ai, we tried api.ai, and they were all fine tools and depending on what you're doing they may work. None of them really worked for our use case.

So we ended up rolling — Yeah, they were just too early. We ended up rolling a lot of our own NLP. A lot of what you want to do, honestly, just frankly, in the A.I. world and the NLP world at this point is you don't need to solve every problem. Sometimes you just need to put UX and UI rails around some of the use cases and drop little hints for people about how they should speak, and talk, and communicate with your bots so that you don't have to handle every use case at this point. You have a better ability to do that if you control your own product stack.

[0:46:59.5] JM: It's funny, because — so chatbot investments really heated up about a year — Was it a year ago? A year and a half ago? Where people were just like chatbots, or everything, and then much like investor trends do. Like Peter, "Okay. Chatbots, it's not fashionable anymore."

It was just funny, because like this whole time — Throughout the whole time, I was like, "It's just an interface. It's an interface to advance A.I, and the A.I. is fundamentally getting better." It reminds me of Bitcoin where there was the fashion of Bitcoin for a while, and then the investor said, "Okay. Bitcoin is not fashionable anymore. What are you saying? Bitcoin, it's still fundamental advancement in computer science. Why would you start to say, "No, it's not fashionable anymore? There's no killer app for Bitcoin." It's a fundamental breakthrough."

We've had fundamental breakthroughs in A.I, chatbots are mere interface into that A.I. I guess where are we in the — I think part of it is like when do we get the bridge between the conversational Amazon Echo type of interface, the voice interface, and the chatbots that are effectively the same thing. Does it take an AR product to get us to where voice becomes more prominent in our world? Voice is still — We're still in such early days of voice. I guess, what is it going to take for people to realize that these like voice and chatbots are not — They're not

fashionable. These are a completely new platform that is just really going to be a paradigm shift as big as mobile.

[0:48:37.8] RM: Yeah, I think there's three things that have held these things back, right? I believe voice is going to be big, but I believe it will not big at work, because since most people have open office floor plans, you can imagine having 20 people in a room yelling at their devices, trying to talk over one another.

I don't know if you've heard people try to argue with an Alexa at the same time. It can confuses Alexa, right? If somebody says, "Hey Alexa, play —" and then somebody else chips in and goes, "Alexa, play this other song," and they try to say it faster, and Alexa gets confused. I think it will be really weird for us to sit and talk to our devices right now in open office spaces.

The way this will evolve is I think one of two ways, either as voice becomes prevalent and we get used to talking to devices in private, or you're comfortable talking to them in public without feeling weird, or you may see a return to more closed off private office spaces. There's a lot of arguments for that. Then you might see voice rise in the workplace again. That's one point.

In terms of chatbots specifically, I think they're primarily going to be text driven through Slack, Hipchat, Microsoft Teams, maybe Cisco Spark and some others that are coming up. I think there's two things holding them back. One is it turns out a lot of things are really hard to do just through conversation. If you look at Talla and some of the things Talla can do; help schedule recruiting interviews, help with employee onboarding, help answer basic HR questions.

What we found was while having a chatbot interface is nice and is a better experience than e-mail, trying to do your configuration work and your setup work and all of that to a chat interface by explaining to the bot what you need, is miserable. You can think about this in human relations. You and I could be having a discussion and I might say, "Actually, here Jeff, let me draw it for you," and jump up to the whiteboard, and that happens a lot.

I think what you've seen with all the popular bots that have been out for a while that are focused on enterprise, they're all going to a web interface, plus a bot. This is what I think you're going to see. I think the trend is that every enterprise application category is going to be rewritten to be

intelligent and conversational. The web piece is not going away, it's just going to be built in a way that makes it highly conversational so that you can do a lot of it through a bot interface.

The second thing that I think we're waiting on is the NLP is going to get better as we get more chat datasets. I would love to see Slack their datasets, scrub them off certain sentences that contain proper nouns, scrub them of identifiable information and release them so the bot developers could use them to train models or for Slack to provide your general, you know, some level of NLP model at some point. I think these things are going to happen and are going to drive things forward, and I think the next couple of years are going to be really big for that.

[0:51:19.4] JM: Okay. Rob, it's been really interesting talking to you. I'm a fan of your newsletter. I also saw you talk at the LAUNCH Scale Festival, I think. You gave a very interesting presentation. Is that on YouTube?

[0:51:31.7] RM: You know, it's a good question. I don't know.

[0:51:33.8] JM: Well, I don't know.

[0:51:34.4] RM: We'll have to figure that out. Jason is a good friend of mine, so he's invited me to speak at those things a couple of times. They're a lot of fun.

[0:51:40.6] JM: Yeah, great stuff. Okay. Cool. Thanks for coming on Software Engineering Daily, and I'll link to Technically Sentient, and Talla, and everything else in the show notes.

[0:51:47.9] RM: All right. Thanks for having me, Jeff. It was fun.

[0:51:49.6] JM: Okay, great.

[END OF INTERVIEW]

[0:51:54.3] JM: Listeners have asked how they can support Software Engineering Daily. Please write us a review on iTunes, or share your favorite episodes on Twitter and Facebook. Follow us on Twitter @software_daily, or on our Facebook group, called Software Engineering Daily.

Please join our Slack channel and subscribe to our newsletter at softwareengineeringdaily.com, and you can always e-mail me, jeff@softwareengineeringdaily.com if you're a listener and you want to give your feedback, or ideas for shows, or your criticism. I'd love to hear from you.

Of course, if you're interested in sponsoring Software Engineering Daily, please reach out. You can e-mail me at jeff@softwareengineeringdaily.com. Thanks again for listening to this show.

[END]