EPISODE 533

[INTRODUCTION]

[0:00:00.3] JM: Joseph Bonneau is co-author of Bitcoin and cryptocurrency technologies, a popular textbook that you might have looked at if you are into cryptocurrencies. At NYU, he works as an assistant professor exploring cryptography and security and his YouTube lessons teaching Bitcoin have hundreds of thousands of views.

Joseph's material offers clear explanations for how Bitcoin works. Since Joe has a clear understanding of the objective facts about Bitcoin, about the engineering and the underpinnings of the technology, he's also the perfect person to discuss the more subjective topics, the common misunderstandings of Bitcoin and the myths of cryptocurrencies, the governance tradeoffs between Ethereum and Bitcoin, proof of work and proof of stake.

Joseph believes that the early mainstream cryptocurrency solutions will be largely centralized once we actually do have mainstream cryptocurrency solutions. He believes that we'll likely move beyond Bitcoin to more efficient currencies.

I enjoyed hearing his reasoning behind this perspective. It was really nice to talk to somebody who knows so much about Bitcoin and has spent a lot of time in the community, but is far from a Bitcoin maximalist. In fact, he's moving more and more into the Ethereum community and studying that more closely. I really enjoyed the conversation with him.

Meetups for Software Engineering Daily are being planned. You can go to softwareengineeringdaily.com/meetup if you want to register for an upcoming meetup. We've got March meetups scheduled at Datadog in New York and HubSpot in Boston. In April, I will be at TeleSign in LA. At these meetups, we'll have some speakers. I'll ask them some questions and it will be a lot of fun. I hope to see you there.

[SPONSOR MESSAGE]

[0:02:01.7] JM: Users have come to expect real-time. They crave alerts that their payment is received. They crave little cars zooming around on the map. They crave locking their doors at home when they're not at home. There is no need to reinvent the wheel when it comes to making your app real-time.

PubNub makes it simple, enabling you to build immersive and interactive experiences on the web, on mobile phones, embedded into hardware and any other device connected to the internet. With powerful APIs and a robust global infrastructure, you can stream geo-location data. You can send chat messages, you can turn on sprinklers, or you can rock your baby's crib when they start crying. PubNub literally powers IoT cribs. 70 SDKs for web, mobile, IoT and more means that you can start streaming data in real-time without a ton of compatibility headaches. No need to build your own SDKs from scratch.

Lastly, PubNub includes a ton of other real-time features beyond real-time messaging, like presence for online or offline detection and access manager to thwart trolls and hackers. Go to pubnub.com/sedaily to get started. They offer a generous Sandbox to you that's free forever until your app takes off that is. Pubnub.com/sedaily, that's P-U-B-N-U-B.com/sedaily.

Thank you PubNub for being a sponsor of Software Engineering Daily.

[INTERVIEW]

[0:03:45.3] JM: Joseph Bonneau is co-author of the textbook *Bitcoin and Cryptocurrency Technologies,* which is one of the most popular textbooks on the subject. He's also one of the instructors of a class that is available online on YouTube, which is I think it's the Princeton Bitcoin class, it's how it's commonly known. Joseph, welcome to Software Engineering Daily.

[0:04:08.3] JB: Thanks. Great to be here.

[0:04:09.7] JM: Your material has been tremendously useful to me in learning some of the basics of cryptocurrencies. Why did you decide to focus your effort on teaching cryptocurrencies?

[0:04:22.8] JB: I think it's been a fun topic to teach for a couple of reasons; one, well when I got to Princeton, which was three years ago now as a post doc, almost four years ago when we decided to move in this direction to teach class on it for the first time, probably it was experimental. Nobody else was really teaching on it, because it was so new. The first class, I think we realized the students were really engaged maybe seeing what was going on in industry and that fortunes were being made and the technology was really exciting.

We got students to maybe to show up initially because they've heard about Bitcoin and they've heard some buzz. The technology is just really interesting and I think that we ended up getting students to stay, because they were enjoying all the different concepts that were there. I think what I've thought a lot of other computer science academics who've been a little skeptical and they've said, "Is this really worth having an academic course on? Is it just a fad or a industry trend? Does this really make sense to teach?"

I've said, "Well, the great thing about it from a education standpoint, even if Bitcoin goes away, even if the area turns out to be a fad in retrospect, it's a really great way for students to learn a lot really fundamental computer science concepts they get exposed to." I mean, seriously they get exposed to crypto, they get exposed to some distributed systems. Now it's causing a lot of people to think more about verification and some really hardcore programming language concept, because they want to reason about smart contracts.

Then we even at one point in our class talk about – I think the book as well, talk about empty completeness. It's really like a fun tour of computer science shaped around this one hot application area. I think it's worked well for that reason.

[0:06:02.9] JM: Well, you think about computer science education when I think about my education over four years, I thought that I had learned how to compose data structures together in interesting ways and what taking a look at cryptocurrencies made me realize is that I wasn't even scratching the surface. You've got these really sophisticated data structures that are able to work together to accomplish just basically a revolutionary invention.

[0:06:32.2] JB: Yeah. There really are just so many interesting technical bits at every level from the consensus protocol and how it works thinking about the game theory, up to like you said, the

data structures are really interesting. It's just been a lot of fun and students really seem to get engaged with it. When they want to go further on a topic, like if they – sometimes they get really interested in some specific technical thing like zero-knowledge proofs and then there's a whole sub-world of people just looking at new ways to build things using zero knowledge proofs on top of blockchains.

It's really worked well I think at bringing a lot of people into computer science from outside, or from related fields. I guess, like you said people who've been working with computers for a long time, discovering a lot of cool new stuff.

[0:07:13.2] JM: I did some shows around deep learning a while ago. What I got the sense from deep learning was that here is a field that is also so dense and rich that you can imagine people who need to learn some aspects of deep learning, they need to learn some of the technical aspects, but they may never need to write a line of code.

They could learn to be very sophisticated contributors to technical projects, but just maybe they never write code and I can really imagine the same kind of things happening with cryptocurrencies. I just was just thinking about that when you said it's really bringing in people from outside of the computer science industry.

It seems like people who don't even write code, they have a capacity to understand data structures. They do have a capacity to understand some of the technological things. Is that your interpretation as well? Are you seeing the same thing?

[0:08:08.3] JB: Yeah, definitely. I would highlight crypto there and say it's really gotten a lot of people to think more about cryptography and learn what a digital signature is for the first time, learn what a hash function is, learn that cryptography is more than just keeping data secret or encrypting data.

I really think that this cryptocurrency topic has exposed more people to cryptography than anything. Even things like Snowden revelations where that got some public attention on cryptography, but not on the same technical way where I think a lot of people really want to

figure out how these things work and what they mean. That's been pretty exciting for someone who has worked in cryptography for 10 years now.

[0:08:48.8] JM: When people are trying to learn about cryptocurrencies, what are the concepts that confuse them the most frequently?
[0:08:56.1] JB: It's a good question. I think really understanding the consensus layer is pretty hard. There's a lot that we still don't really know as academics. We don't really know if it's stable that there is – it's frequently claimed that Bitcoin is incentive compatible and that everything is okay, but it's really not. We know there are a lot of different strategies minors could potentially take that would undermine the consensus process.

We really don't know how those change in different variations. If fees are lower, if you don't have a constant block reward. Not to mention proof of stake, proof of space, all these different other kinds of variations.

We really don't have a good fundamental understanding of how those change the trust model, if they're likely to be stable in the long run. That's surprisingly, students are usually quite confused, so they often just assumed that minors have to behave the way the protocols says they do.

That's the first state that students get to, and then once they scratch a little bit further and they say, "Well, nothing is really policing minor behavior, they can do whatever they want," then they don't understand why it's secure it at all and they don't understand why no one is doing the selfish mining attack, nobody is greedily trying to horde transactions with high fees, all these other potential things that look like they make a profit on paper, but we don't actually see on practice. I think, really understanding how minors behave and the incentives they're facing is a challenge for a lot of students. It's a challenge for me honestly.

[0:10:29.3] JM: When you look at proof of work versus proof of stake versus something like Stellar, I don't know Stellar's consensus protocol too well, but it seems like it's a little new, it's a new twist. I mean, there's lots of new twists on consensus mechanisms. Many times, there are not thorough proofs around them, or there are proofs that seem I guess speculative. How do

you evaluate these different consensus mechanisms when it's incredibly difficult to have a thorough vetting of them in an academic fashion?

[0:11:03.9] JB: Yeah, it's a great question. I guess, well one thing I could say about Stellar, the good thing about Stellar is that it has a very clearly defined model and its properties actually are proven pretty rigorously. Stellar is relatively easy to reason about. The problem is just that people don't like the model as much. It's not an open participation model like Bitcoin where anybody can just show up.

It's a little bit complicated to explain, but basically different people have a notion of who's trusted and as long as people generally agree on who the trusted nodes are, then the system works out. The nice thing about Stellar and also some related systems that are more semi-centralized is that you can reason about them more. They just don't provide the full Utopian decentralized world that people initially fell in love with with that with Bitcoin.

The problem is that in that fully decentralized world, we're not really sure anything works or why anything is stable. It has been a real problem on the research side too. I mean, I do a lot of work in peer review reviewing research papers for academic conferences and journals and there are a lot of submissions, a lot of people over in papers proposing new consensus models that makes the modifications to good points design.

It's been difficult to provide good feedback to them. Oftentimes, the papers get rejected from publication with the critique that they haven't really proved that their solution works, or proven that any of the claim properties really hold, which is usually true, but the problem is that Bitcoin can't really prove anything either. It's very difficult for scientific authors to prove that their system is really better or worse than Bitcoin.

[0:12:49.4] JM: In deep learning we see this trend towards, "Well, my model works empirically. I don't know why it works, but it seems to work. I trained it. It can identify cats quite effectively and who cares why it works." Is it okay if we go in that direction with cryptocurrencies?

[0:13:09.5] JB: Well, the problem with that with cryptocurrencies is that you really can't say that for a new proposal that only lives on paper, because you can make that claim for Bitcoin

somewhat credibly. You can say it's worth a few hundred billion dollars depending on how you count and it's generally the blockchain has behaved pretty consistently, there haven't been any major attacks on Bitcoin at the consensus layer.

You can make the claim empirically that Bitcoin's consensus seems to work. If I come along and write a paper and I say, "Oh, it would be better if we change this aspect." Well, it's hard for me to – it's a pretty high bar to – if you say, "Well, to prove your idea you have to go make a new old coin. Wait until it's worth billions of dollars and then I'll believe you that it works." We like to be able to reason about these things on paper, in a laboratory before there is real money invested in them.

[0:14:03.8] JM: Indeed. You wrote an article recently Five Myths of Bitcoin. Let's talk about these. One myth is that there is a finite supply of Bitcoin, the 21 million amount that we are gradually asymptoting towards and that this is an immutable truth. As you say, there is no guarantee that the supply of Bitcoin will not change, because the majority of the network could vote to change that limitation.

Hypothetically, what would be the sequence of events that would lead to a change in the cap on the number of Bitcoins? Why would that occur?

[0:14:43.3] JB: Well, I think the most likely scenario would be that somebody would propose a fork similar to the Bitcoin cash fork and they would come up with some brand name and call it Bitcoin unlimited coins or whatever, and that didn't have this 21 million restriction, that was going to keep creating new units. That that fork would overtake the classic view of Bitcoin and become more important.

I think that's the most likely way. It's possible everybody would get together and let's say, let's change the definition of Bitcoin and there would be enough support that there would actually be a hard fork in what we call Bitcoin would actually change its currency cap. That's a little bit less plausible, because it has always been a core value of Bitcoin that 21 million will be the cap and the number.

I mean, I should about that calling that a myth, the myth part of it is that there's some mathematical law that says 21 million is the number. The same way there are reasons, the chemical reasons why it's difficult to produce gold. It's not a policy decision that we have a finite supply of gold. It's just a property of our world and of physics.

With Bitcoin, the 21 million is a policy decision that the community agrees with right now, but the community could change its mind. You don't have to undermine any mathematical assumption for that to change.

[0:16:02.8] JM: Right. If one of the bearish statements I hear most frequently about Bitcoin is that we have never had a successful currency that is deflationary and that is what Bitcoin is going to be. Therefore, how dare you say that Bitcoin is ever going to be a successful currency? We tried this before. What do you say to those kinds of arguments?

[0:16:29.8] JB: To me they're plausible. That makes me a little bit of an outsider maybe in the Bitcoin community. I mean, there's probably other people who agree, but I'm not an economist. I have talked to a lot of economists and at least there is always this caveat of mainstream economist, because you can find somebody who considers himself an economist who says anything.

Mainstream respective economists, they pretty much all agree that you want your currency to be inflationary. The debate is how inflationary you want it to be? Do you want 1% inflation annually? Do you want 5%? This is true even among fairly conservative economists like Milton Friedman, who pretty much all agree that you want some slight constant inflation to your currency.

It's a fairly well established belief in the economics community. It's possible that they are wrong and Bitcoin will challenge their assumptions and everything will change. I am more willing to stake my bets that 100 plus years of economists studying this has discovered some important knowledge there, so that's where I come down on it.

[0:17:37.8] JM: What about the chance that we could just increase the divisibility of Bitcoin, so that as it was growing in value through its deflationary properties, we could just increase the

divisibility at a rate that outpaced the deflation? Because I mean, we've never had a currency where you could do that.

[0:17:55.8] JB: That's the separate problem. The fact that Bitcoin has limit on divisibility, it does seem like a problem when you just run the numbers and you think about how many Satoshis there are for every human on earth. It's not very many. I mean, I think a lot of subsequent designs have had more atomic units of currency.

Yeah, if you were going to redo Bitcoin again, you'd probably multiply the total number of Satoshis by, I don't know, at least a million or a billion and then the numbers start to look more reasonable. I mean, even if you fix the divisibility problem, the major problem that deflation causes or is believed to cause based on experience is still there, which is that nobody will want to spend or circulate the currency. Everybody will want to horde, because it's becoming more valuable over time. That prevents it from being an effective means of exchange, because people don't want to actually spend that they only want to hold onto it.

[0:18:52.5] JM: Yeah. You don't believe that there could be some kind of function where the divisibility outpaces that property though? Or it's just not something you've studied, not something you spent much time on?

[0:19:00.7] JB: No, I'm saying the increasing the – that is a problem, the lack of divisibility. Deflation is a separate problem. Even if you fixed the divisibility problem, even if you make every Satoshi divisible into a million micro-Satoshis, that wouldn't fix the core problem that deflation causes for currency.

[0:19:20.8] JM: Are you saying that because deflation can be an unbounded property and it's hard to make unbounded divisibility?

[0:19:28.3] JB: No, no, no. Even if you had unbounded divisibility, deflation would still be a problem. The problem is just that people wouldn't – people want to hold the currency if it's becoming – if they know it's becoming more valuable over time, they don't want to spend it. When nobody wants to spend and everybody wants to hold the economy tends to grind to a halt. The thing no longer functions as a currency is a means of exchange.

[SPONSOR MESSAGE]

[0:19:58.0] JM: Apps today are built on a wide range of back ends, from traditional databases like PostgreSQL to MongoDB and Elasticsearch, to file systems like S3. When it comes to analytics, the diversity and scale of these formats makes delivering data science and BI workloads very challenging. Building data pipelines seems like a never-ending job, as each new analytical tool requires designing from scratch.

There's a new open source project called Dremio that is designed to simplify analytics on all these sources. It's also designed to handle some of the hard work, like scaling performance of analytical jobs. Dremio is the team behind Apache Arrow, a new standard for end-memory columnar data analytics.

Arrow has been adapted across dozens of projects, like Pandas, to improve the performance of analytical workloads on CPUs and GPUs. It's free and open source. It's designed for everyone from your laptop, to clusters of over 1,000 nodes.

Check out Dremio today at dremio.com/sedaily. Dremio solved hard engineering problems to build their platform, and you can hear about how it works under the hood by checking out our interviews with Dremio's CTO Jacques Nadeau, as well as the CEO Tomer Shiran. At dremio.com/sedaily you can find all the necessary resources to get started with Dremio for free.

I'm really excited about Dremio. The shows we did about it were really technical and really interesting. If you like those episodes, or you like Dremio itself, be sure to tweet @dremiohq and let them know you heard about it from Software Engineering Daily.

Thanks again to Dremio and check it out at dremio.com/sedaily to learn more.

[INTERVIEW CONTINUED]

[0:21:59.3] JM: Another myth that you dispelled in your article is that Bitcoin wastes energy. One problem with this statement is that it suggests that we have some direct substitute for

Bitcoin that is lower energy that we are aware of. Why is this mythological, the idea that Bitcoin waste energy?

[0:22:19.7] JB: Well, the analogy I've used a couple times is that saying something waste energy means, like you said you could do the exact same thing for lower energy using a different technology. Incandescent light bulbs waste energy, because LED lightbulbs essentially direct substitutes use a lot less energy, so anybody using an incandescent lightbulb, that's a way so you should switch to this other technology and save energy.

With Bitcoin, we can't really say that. There are things like proof of stake, or more centralized models like Stellar that do use less energy. They use a lot less energy, but they don't have the same trust model that Bitcoin does.
You're using less energy, but you're also changing the trust model. Whether or not that matters, I mean maybe it's perfectly legitimate to say you think Bitcoin's trust model is too strong and is not useful and that things would work just as well on a slightly more centralized model or in a proof of stake model, so we should switch to that.

That's fine, when the fact that they use less energy is definitely a plus, but they're not direct substitute. These are direct substitutes, that's why I don't see the energy as a waste. I see it as the only way we know to get to Bitcoin's trust model for decentralized consensus.

[0:23:32.8] JM: You mentioned that you felt like an outlier in terms of the deflation discussion. What are the other areas where if you go to a Bitcoin conference, you feel heretical when you're sitting around a lunch table?

[0:23:49.6] JB: Well, I suppose I'm relatively bearish on Bitcoin as a whole. I don't think that Bitcoin itself will be around in 10 or 20 years. I think it's going to be supplanted by other currencies. I think the biggest advantage they have is that they've gotten to learn from Bitcoin's design in fixing the stakes.

I also think probably the other big area where I'm a little bit of a heretic, I do think Bitcoin's trust model is too strong, or maybe not really what we need in practice. I think the place we're all headed is a more semi-centralized world. Be technologically maybe a little bit dull, but I think

we're likely to end up in a world where the most important blockchains, where a lot of everyday commerce happens are actually just centralized blockchains, where three out of five parties sign it and that's it, that's a consensus protocol. There's no fancy decentralized thing.

I often get pushed back when people say that's so boring and what if those parties are corrupt? I don't know. To me, it's still an improvement over the world we live in today. I think that for most applications, I think people are willing to live with that model and it's going to be simpler and faster and not waste energy and a whole bunch of other nice properties.

I think if there's a core fallacy in the Bitcoin community, it's the classic technological trampolism that the most elegant and beautiful technological solution will win. Like that's really true. I think something simple and kind of boring will probably win out in the end, even if it's much more centralized than a lot of people, especially early people on the Bitcoin community wanted.

[0:25:33.8] JM: When did you start focusing on Ethereum?

[0:25:35.8] JB: Ethereum, let's see. I remember, probably two or three years ago when it was coming out. I remember hearing the idea at first and thinking it was crazy that you could write arbitrary programs and have them run on a blockchain. I think once I sat down for an hour and saw the model and how it worked with guests, I said, "Oh, okay. This is cool. This is definitely a quantum leap forward over Bitcoin and I think this will be a much more exciting in the long run."

I do think it is a lot more exciting. I mean, I think if you look at the last year the Bitcoin community has argued without end about the block size and relatively small tweaks that aren't all that interesting. In Ethereum land, there's been so much exciting stuff, I mean good and bad with smart contracts being developed, all these new applications being explored, a lot of interesting thinking around how to make the whole thing scale. To me it's – I don't' want to be giving investment advice, because I don't know more than anybody else, but from a technology standpoint it's been much, much more interesting recently.

[0:26:40.4] JM: What are the examples that you give when people say, "Okay, well Ethereum is all about smart contracts." I don't see any smart contracts that are deployed, that are useful today, so why are smart contracts useful?

[0:26:55.2] JB: Yeah. I think the safest answer is we don't know what is actually going to be useful in the end. CryptoKitties is not a very good sales pitch if that's the most successful application. I think on paper, I usually start – when I'm teaching, I usually start by saying Alice and Bob want to play a game of chess. They don't know each other and they don't trust each other, but they want to play chess and they want to bet on the outcome. Just because chess is pretty universal and people know the game and they can understand that scenario and they can see why there isn't really a good solution if two people want to play chess for money remotely.

Usually that is the gateway to start describing smart contracts. Then I found auctions are one of the best examples you can transition to and say, "Well, once you can do this chess example, let's talk about options," which I guess it requires one or two more technical concepts, because usually you want to use commitments to commit to bids and then reveal them. That's a good example to me of something that people understand doing an auction without a trusted auctioneer remotely unanimously.
It's actually real and that's one of the real examples that I think will probably start happening on the blockchain. I think it's probably five or 10 years away, but I think that that will be one of the first.

[0:28:14.4] JM: Five or 10 years away from –

[0:28:15.9] JB: I think yeah, five or 10 years from somebody selling the Da Vinci painting that went for a couple 100 million this year before someone says, "Well, let's actually do this big art auction on using a blockchain."

[0:28:27.0] JM: The smart contracts are built with a scripting language and Bitcoin has a scripting language as well. Bitcoin script is mostly used for checking the correct signatures, but the Bitcoin scripting language is rich enough to do things like multi-signature transactions. What kind of smart contract like functionality could you get out of Bitcoin? Could we be writing more sophisticated smart contracts in Bitcoin? What is it that precludes us from doing that?

[0:28:58.6] JB: Yes. It's important to realize this is where – this is a great example of using cryptocurrency to teach computer science theory, that Bitcoin scripting language from a theoretical computer science point of view is strictly weaker than Ethereum. Ethereum is in

some sense turning complete, which means you can basically express any program that you can think of using it. Bitcoin is not, which means there's a lot of things you can't do.

In particular in Bitcoin scripting language, there is no loops, which is was a design feature from the beginning because that means contracts can't – the scripts and Bitcoin transactions can't run forever. You can look at a Bitcoin script and basically its runtime is limited to the length of the script.

Whereas, for Ethereum there is loops so you can write an infinite loop, you can have something that runs forever. That's a really famous celebrated result in computer science, that once you have that general purpose programming language, it's impossible to prove that your programs will actually hold, that they won't run forever.

This is a designed goal of Bitcoin's originally. That means it's fairly limited. You can basically only use it for a handful of things that were in mind when the language is designed. Like you said, you can do multi-sig. You can do some time-dependent things. You can have different conditions for spending money at different points in time. You can require a hash premise to be revealed to spend money.

There is a handful of things and it's just powerful enough to build up these lightning network payment channels, these off-chain channels, which is a pretty exciting application. You can do a couple of other things. You can do escrow transactions using Bitcoin, but if you're giving a presentation, you can list on one PowerPoint slide, these are the four or five things you can do with Bitcoin script. As far as we know, that's pretty much it.

Unless, you get into this very cutting-edge, zero knowledge proof stuff where you can expand it to more things. Whereas, Ethereum you can basically in some sense program any application you can dream up. There is a very large difference.

[0:31:03.5] JM: Back before Ethereum, you worked on a prediction market coin called Futurecoin. What were the lessons from that experience and how did development of – I would say, that maybe fell into Bitcoin 1.5 or 1.2 timeline type of development. How did that

experience, the development experience compare to say the development of somebody developing an ERC20 token today?

[0:31:30.8] JB: Yeah. I think that Futurecoin paper is actually a really good – it makes a good sales pitch for why Ethereum exists as a project today. I mean, at the time when we wrote that there were other projects, Namecoin is a famous example where people said, "If we could just tweak Bitcoin, add a little bit of functionality, then we could do this other cool thing."

At least there isn't an obvious way to do prediction markets on top of Bitcoin. There wasn't an obvious way to do a name registry on top of Bitcoin, so people were proposing all these alternatives saying we're going to make a new kind of application-specific currency that we'll tweak Bitcoin, add a couple of things and then we can do something really cool.

Of course, the inevitable results of that of a lot of people proposing these application-specific coins, which is never going to be a model that work was the Ethereum project and a couple others that are forgotten about now. The Ethereum project is saying, "Well, can we build one blockchain that could actually host all of these different applications and be programmable in the future for new applications people will think at?"

It makes a nice story, because you can look at how we proposed retrofitting prediction markets onto Bitcoin by changing it, versus how you would do it today, which is probably you'd write a smart contract away, I guess [inaudible 0:32:46.6] and some other folks doing prediction markets on Ethereum have done and say, "Yeah, obviously this model makes more sense. It's more straight forward to just program on top of an existing platform rather than have to make an entirely new blockchain hardcoded for your application."

[0:33:05.0] JM: In different fields that we've covered on Software Engineering Daily, we've seen the importance of developer ergonomics. In Kubernetes for example, you see a better API for building distributed systems. It's easier and simpler than perhaps something like OpenStack. In the JavaScript area you see this with ReactJS, or ViewJS, where it's much easier to build UI components than it was in backbone or perhaps the first version of angular.

You see this with Tensorflow and that Tensorflow maybe didn't exactly do anything new in terms of building deep learning models, but the APIs are so much better that it allows us to leap forward. Is that a useful way of looking at Ethereum that this is effectively a better API. You could spin up your own prediction market coin, but why would you do that when you have this beautiful set of APIs within Ethereum?

[0:34:01.8] JB: Yeah. I mean, I'm trying to think of a right analogy to web development. It would almost be like if – imagine before JavaScript existed, you couldn't do a lot of things we like today. You couldn't do Gmail in the browser implemented as a web application. People said, "Well, okay in addition to your web browser, you'll have an e-mail client."

You couldn't do online games in the browser, because there was no JavaScript, so people said, "Okay, we'll download and distribute software for an online game." Eventually people said, "Well, why don't we just add JavaScript and make the browser programmable and then most of these client site software is going to go away and people will just access things as web applications, which is –" There's still client site software of course, but most people just use the programmable browsers a way to deliver functionality. There's a couple of leaps there, but that's a rough analogy to how making a blockchain programmable, the result on something like Ethereum.

[0:34:58.8] JM: For my point of view, it seems like Bitcoin has an advantage of network affects in terms of more people using it for an actual currency. Ethereum has the advantage of network effect of more developers onboard. When you're looking at Ethereum and Bitcoin, you're saying you're a little bit more bullish on Ethereum, or significantly more bullish, is that because you're betting on the strength of the developer network effects over the current usage network effects of Bitcoin?

[0:35:33.7] JB: Well, I mean I guess I should say I'm bullish, but I think it's technologically interesting. I don't know if I'm bullish on the price. I generally believe in efficient markets, so I don't think I have a better insight on the price than the market does.

I do from a technical sense think that Ethereum is more interesting. It doesn't necessarily mean the Ethereum project or Ethereum that specific blockchain will be where all the action ends up

down the road. I just think that the model of smart contracts and a fully programmable blockchain is a lot more interesting technically.

[0:36:04.7] JM: Is this a bet on developer network effects over usage network effects?

[0:36:10.9] JB: I suppose so. I mean, that is Bitcoin's core value proposition at this point that it's the biggest and the oldest and the most stable. People feel like it's the – I don't know, I mean to a first approximation if you don't have any other information about which cryptocurrency is the best place to park your money. Another thing you care about is this project going to survive? Is it going to be valuable 10, 20 a 100 years from now?

It does make sense to say, well Bitcoin has been around for 10 years and Ethereum has been around for less than five years, so that's some signal that Bitcoin might last for a longer time. I mean, people do this reasoning all the time. Like if I asked you which bank you wanted to store your life savings between bank A and bank B, and you didn't know anything about the banks except that bank A has been in business for a 100 years and bank B has been in business for five years, you'd probably pick the one with the long track record, right?

[0:37:08.1] JM: Indeed.

[0:37:09.3] JB: It makes sense that Bitcoin has – it has more and more market cap, the price is higher, the branding is much better. I think the name is actually simpler and better and clearer to the end user and that thing matters. I think when I talk to friends who don't work in tech who was going to ask what I'm up to, most of them have heard of Bitcoin, not that many have heard of Ethereum. If they have, they're not even sure if Ethereum is a coin. They're not sure if they can own an Ethereum. Some of them think it's just a thing that runs on top of Bitcoin. I do think Ethereum's at a big branding disadvantage with the general public. It's obviously a surmountable problem, but it is a problem.

[SPONSOR MESSAGE]

[0:37:57.9] JM: Software Engineering Daily is brought to you by ConsenSys. Do you think blockchain technology is only used for cryptocurrency? Think again. ConsenSys develops tools

and infrastructure to enable a decentralized future built on Ethereum, the most advanced blockchain development platform.

ConsenSys has hundreds of web3 developers that are building decentralized applications, focusing on world-changing ideas, like creating a system for self-sovereign identity, managing supply chains, developing a more efficient electricity provider and much more.

Listeners, why continue to build the internet of today when you can build the internet of the future on the blockchain? ConsenSys is actively hiring talented software developers to help build the decentralized web.

Learn more about consensus projects and open source jobs at consensys.net/sedaily. That's C-O-N-S-E-N-S-Y-S.net/sedaily. Consensys.net/sedaily. Thanks again, ConsenSys.

[INTERVIEW CONTINUED]

[0:39:14.6] JM: We've done some shows recently, where we've explored the basics of Ethereum Dapp development, solidity. I'd like to ask you some slightly more advanced questions. On the EVM, the Ethereum Virtual Machine, the storage and memory space is tremendous. You have I think 2 to the 256 of storage and memory space. I guess, that's 2 to the 256 addressable places, let's say I don't know how long each of those address places is.

[0:39:50.0] JB: It reaches 256 bit words actually.

[0:39:52.7] JM: 256 bit words, okay. With that kind of compute space, you would imagine you could do some really interesting distributed system stuff. Can people run giant MapReduce jobs on Ethereum?

[0:40:08.2] JB: The short answer is no. The system is never really designed for that and is unlikely to scale to – the goal of Ethereum is not to replace something like EC2, which is a service where you just pay for computation.

With the memory specifically, it's important to realize that even though you have all these addressable memory in theory, in practice you can never use it. Obviously, that much storage doesn't exist in the entire universe. Clearly, you can't actually store 2 to the 256 words.

Then particularly, the way that's enforced in Ethereum is that rights are pretty expensive. That's one of the most expensive operations gas wise is writing to storage. I don't have the numbers off the top of my head, but it's a couple writes to storage will usually cost around a penny US. Even if you want to write a megabyte to storage, you're talking about dollars of gas just to write that megabyte out, which is so counter-intuitive in the modern world where memory is – storage is so cheap. On Ethereum, it's definitely not cheap.

The computation is not close to free either. Yeah, I think it's – I've heard this before. People say like, "Oh, this is decentralized." People sometimes first conceptualize Ethereum as a decentralized EC2. It's really not that. I don't think it's ever going to be that. There's a lot of work to make it more scalable and have it be cheaper to do a computation. The goal is always to just verify that contracts are executed correctly, not to actually do heavyweight computation.

Usually, if you do want to do heavyweight computation you try to architecture your application in such a way that the expensive computation happens off the blockchain and you just verify it on chain. There are some really nice examples where that's straightforward.
For example, if you want to compute a square root, which can be an expensive operation, the way you ideally like to design this is that you compute the square root off chain and then you just feed the answer to your smart contract, which can then do the verification, which is squaring which is usually much cheaper. Then you have the effect of having computed that square root in a trustless manner by just verifying the results and sort of offloading the computation elsewhere.

To the extent possible, that's how – that's like when I teach smart contracts that that's really the paradigm is that you do the expensive computation elsewhere and the smart contracts is just for verification.

[0:42:41.1] JM: Imagine we've got access to a set of like the full decentralized stack as it has been hypothesized today in various forms. Let's say we've got Gollum for our computation and

we've got Filecoin/IPFS for our storage and we've built a decentralized Facebook, except it's a more sophisticated version of Facebook where people can issue queries.

We have an API setup, where we've got to setup smart contracts where people can request information, like maybe I can request a MapReduce on the names of all of my friends, or all of the interesting comments that my friends have made and maybe that contract would reach out to on my IPFS storage and the compute would be handled by Gollum. Is that fanciful, or does this compute model also follow what you think is going to happen with the financial system, where there will be a good deal of centralization in the traditional ways.

Like maybe the storage is still handled by S3, the compute is still handled by EC2 and that – it's just that the smart contract is handled by – is just handled by Ethereum. Have you started to envision these types of things, or is it just way too early to even start to hypothesize and we've got better things to think about?

[0:44:11.1] JB: No. I think it's a great thing to think about and I've been thinking about of some. I think the way you described it there, that's actually – that's how Dapp development should work. You're not doing the heavyweight storage and computation on the Ethereum network. You're offloading it to these other things and then verifying it.

Like you said, you could be offloading your storage onto something like Filecoin. You could be offloading it to EC2. Either way you'd want to try to commit to what you're storing on – actually on the blockchain, on your smart contracts so that you can verify that the storage was done correctly. I think that makes sense. Yeah, it's not clear to me if these decentralized projects are going to win out in the end. I'm doubtful that they're going to be able to compete on cost with Amazon and Microsoft and Google in the long run on providing computing power, if they were able to compute on cost.

I feel like those big players will either come down on the cost, or start to make services available through this. There's nothing preventing – it was really popular – if Filecoin becomes really popular, there is nothing preventing Amazon from just being the largest Filecoin storage provider. Whereas, some other would be tightened tech company in the future that corners that market.

I think that fundamentally, there are big returns to scale for storage and computation so they're likely to be somewhat centralized. The same way Bitcoin mining is fairly centralized, because it turns out there are big returns to scale to running a big mining operation. I think that's the most likely is that it's either today's existing big players, or some future big players that we don't know about yet. I could be wrong. We'll see how it develops.

[0:45:54.5] JM: Lightning Networks as a means of scaling blockchain technology, give the bull and the bear case for Lightning Networks.

[0:46:05.8] JB: Okay. Well, I can give the bull case first because it's probably – that's the one I tend to believe. It's another good example of this paradigm of not recording every transaction on the blockchain trying to do most of the work offchain and most of the work in this case means if you engage in a lot of financial transactions with the same party, you are willing to just have one summary transaction go on the blockchain and not the whole history of everyone back and forth.

I think it's a really elegant design and I think it's exactly what this paradigm of minimizing how much you need to use the blockchain. I think in the future that's both for payments, in the Bitcoin case with Lightning, but also with a lot of smart contract development, there's a very clear analogy of state channels where you can try to do your smart contract operations offchain and then verify.

The example would be, again you can go back to the chess example of Alice and Bob are playing chess. You can get away with not having every single move in the game be written to the blockchain and invoke the cost of having a transaction. The only thing you really want to verify at the end of the game is that one party actually won.

You can play the whole game offline with two parties just sending their moves back and forth and signing them, and then at the very end you just send the state of the game at the end and it's signed by both parties. If everybody behaves honestly, the whole thing is great. If not, you have some fallback procedure the smart contract will implement. That's the bull case. I think this is clearly the way it should be designed for efficiency.

The bear case in the case of payment specifically is that Lightning Network, it looks like it will increase centralization in a couple of different ways and that there will be some central payment hubs that have to process a lot of transactions. It's possible that will collapse it to the way today's credit card processing networks work, where you have a handful of banks that are – issuing banks and a handful of acquiring banks. That's how most transactions are processed.

I think the opposition to the Lightning Network model is mostly that it will increase – I'm being a little pleasant here saying it will increase the amount of centralization, because we don't really have a great way of quantifying that. It is a slight step away from again this utopian ideal of a fully decentralized world.

[0:48:35.8] JM: Are the bottlenecks to getting Lightning Network deployed, are they implementation bottlenecks, or do we still have a lot of theoretical work to overcome?

[0:48:46.4] JB: No, I think it's mostly implementation and I think the technology is basically there and it works. To some extent, I think it's just a matter of demand in the market not quite existing yet. It's hard to build up – anytime you're trying to build up a payment system, you need to have both the population of people who wanted to pay and a population of people willing to receive the payments.

It took credit cards a long time to get off the ground for that reason. When credit cards were first proposed, well a very few stores accepted them. Very few customers have them and the customers said, "Well, it's because no stores accept them." It's just difficult to grow. I think for Lightning Networks to take off, the biggest thing that's needed is a killer application where the people really want to use them for.

It might be something like paying for decentralized storage by the byte, or decentralized anonymous browsing tutor by the packet or something like that where people really want to make anonymous cryptocurrency payments that are repeated, rather than I guess the killer examples of things people actually want to pay with for Bitcoin that have been around so far are mostly one big payment where you don't necessarily need Lightning Networks.

[0:50:05.2] JM: There is a debate around the proof of work versus proof of stake models. Why is this such an important discussion? Is this about scalability? Is it about reducing the transaction cost? Why is proof of work versus proof of stake a debate worth having?

[0:50:22.1] JB: Well, I think the number one reason in what originally got people thinking about proof of stake is reducing the energy cost. Proof of stake to get rid of almost all of the energy that's consumed by Bitcoin mining. That was the initial appeal was that we can do this in a low energy way.

I think that's important. I think everybody is concerned about energy consumption and it's been something that constantly gets brought up when people are critical of Bitcoin is that it uses so much energy.

Since then, a couple of other facets of the debate between the two models have come up. Some people think that the proof of stake model is actually more secure and trustworthy because it relies on stakeholders, or the currency not misbehaving instead of miners not misbehaving.

I'm not sure this is such a huge difference. Again, we don't have a great way to quantify it or to really technically say what the difference is. Proof of stake has a nice property that decision makers are those who hold the most currency. They should have a strong interest in seeing that the currency thrives and does well. I think in practice, Bitcoin miners have the same interest as they've invested so much in Bitcoin mining hardware and their future is very tied to the future of Bitcoin.

They also generally want to pay it well and see the Bitcoin system do well. That is a debate which system is more stable, more likely to – less likely to fork, less likely to see attacks and misbehavior.

Then the debate really comes down to will any of these proof of stake systems work in practice when there's lots and lots of money behind them? Like I said earlier, it's very hard to reason about these things on paper. We just don't have the scientific tools to do it yet. The idea of

switching even a system of Ethereum size, which is in the many, many billions of dollars switching it over to possible but unproven technology, it gives me a lot of pause I'd say.

It would be great if proof of stake system could slowly grow and prove that it works as it scales up to be worth a lot of money, rather than switching an already valuable system over to proof of stake.

[0:52:38.7] JM: Indeed. You've got a lot of different places you could be spending your time. You could be teaching, you could be writing, you could be doing research, you could be launching an ICO of some sort. What are your priorities? How are you spending your time these days and are you starting to think about how you're going to map out your impact in the cryptocurrency community over the next five or 10 years? Are you really keeping your optionality open and just like thinking for the next couple of years and then you'll make a big bet?

[0:53:11.0] JB: Well, yeah. It's a great question. I've been trying to focus my impact on teaching. I'm not note where we are hoping to come out with a second edition of our textbook that will have a lot more coverage of more recent developments and smart contracts in particular. Working on a follow-up [inaudible 0:53:29.5] course that will be about smart contract development

I guess, teaching is what I like the most, one that I'm most passionate about and I guess where I think I can have the most impact personally. I'm trying to preserve that which has been maybe a reason I have not been interested in launching an ICO specifically. Also tend to think most ICOs are a little bit scammy, so it's a space I hadn't wanted to – I don't know, I guess I have some value in trying to preserve myself as an independent voice and not be tied to any one project. Although I am involved as an adviser with a couple of different projects in the space, which is hopefully a good way for me to get some exposure to the industry side of things.

I have noticed for a lot of academics, it's easy to look around in the space and see a lot of people making a lot of money, who frankly a lot of academics look at and think these people are idiots. If they are making millions of dollars off of an ICO, why am I toiling around teaching for a professor salary when I could just launch an ICO and raise all these money regardless of how good my idea is? I guess, I've tried to resist that temptation and focus on teaching and research

and trying to research more deep and fundamental questions that aren't necessarily money makers.

[0:54:51.0] JM: Well, you got to imagine that if those ICOs do not end in flames and lawsuits and litigation and SCC investigations and jail time, that must be an indication that this market as a whole is just flourishing and burgeoning in the next five years. In which case, I think you'll be sufficiently hedged for that outcome. I'm sure there will be some way that if by some insane circumstances those ICOs do not end in flames, that must be a rising tide of such velocity that it will lift your boat as well.

[0:55:30.0] JB: Well, I hope so. I mean, a lot of them are going to end in flames technically. Not in such a way that the founders aren't making off of a lot of money. Although, we'll see. I mean, I guess it's not worth going to jail and probably a few people are going to go to jail. Yeah, I don't know. I'm happy to try to be a teacher and a writer and an educator first, to the extent that I get involved in the industry side of things, it's a bonus.

[0:55:59.8] JM: Well, you're doing a fantastic job. Your material is incredible. It's technical dense and yet entertaining. I just can't sing enough positive praise. I can't encourage people enough to check out your textbook, which is available for free online, unless you want the hard copy book. Your YouTube videos are amazing, so I really think your stuff is tremendous. Thanks for being such a powerful educator.

[0:56:24.2] JB: Thanks so much for all the kind words and thanks for having me on the podcast. This is great.

[END OF INTERVIEW]

[0:56:31.1] JM: Your enterprise produces lots of data, but you aren't capturing as much as you would like. You aren't storing it in the right place and you don't have the proper tools to run complex queries against your data.

MapR is a converged data platform that runs across any cloud. MapR provides storage, analytics and machine learning engines. Use the MapR operational database and event

streams to capture your data. Use the MapR analytics and machine learning engines to analyze your data in batch, or interactively across any cloud, on premise, or at the edge.

MapR's technology is trusted by major industries like Audi, which uses MapR to accelerate deep learning in autonomous driving applications. MapR also powers Aadhaar, the world's largest biometric database, which adds 20 million biometrics per day.

To learn more about how MapR can solve problems for your enterprise, go to softwareengineeringdaily.com/mapr to find white papers, videos and e-books. MapR can leverage the high volumes of data produced within your company. Whether you're an oil company like Anadarko, or a major FinTech provider like Kabbage, who uses MapR to automate loan risk, and has 3 billion dollars of automated loans to date.

Go to softwareengineeringdaily.com/mapr to find out how MapR can help your business take full advantage of its data. Thanks to MapR for being a sponsor of Software Engineering Daily.

[END]