EPISODE 1208

[INTRODUCTION]

[00:00:00] JM: Prediction markets provide an exchange for trading based on the outcome of events. Most prediction markets are centralized. They operate like a casino where betting takes place under the supervision of one central governing organization. This makes the market less efficient than it otherwise might be. The central organization is a business and makes money by extracting value from the trades the customers make. Augur is a prediction market built on the Ethereum blockchain. A trading network built on a blockchain can have a decentralized permissionless transaction record without a centralized governing body. Augur's network is built to be transparent, low-cost and free from interference.

Joey Krug is the founder of Augur and joins the show to discuss what it takes to build a trustworthy decentralized market. How Augur is solving challenges such as the Oracle problem and why blockchain may be a key to democratizing financial problems like prediction markets.

[INTERVIEW]

[00:00:53] JM: Joey, welcome to the show.

[00:00:54] JK: Thanks for having me.

[00:00:55] JM: You are the founder of Augur, which is a decentralized prediction market. What is the function of a prediction market?

[00:01:04] JK: Yeah. So a prediction market is basically a platform that lets people bet on or against real-world events, and by taking the odds of those bets you can basically get a forecast for how likely the event is to actually happen.

[00:01:21] JM: And what practical functionality does that serve?

[00:01:25] JK: Yeah. So it does a couple things. One is that for the people who are actually betting on the market, it lets them actually just make a straight bet if they're betting on something like a sporting event. But it also lets people hedge risks. So as an example, which is like a fairly real example, say you are a person who wanted to hedge against the risk that taxes increase with a new democratic congress. Well, you could basically bet on the democrats, and if they win you've hedged a little bit of whatever the tax change is. They would basically make more money if they win. And if they lose, you don't get the tax hike, but you also lose your bet. The other use is somebody who has just an opinion. They're not looking to hedge any risk. They're looking to take risk on because they believe they know something about the world that someone else doesn't. And prediction markets have a sort of wide range of things you could bet on. It could be a sporting event, politics, whether SpaceX launches their next rocket successfully, really anything.

[00:02:26] JM: Now in order to actually make those bets substantive and to make – Like if I wanted to make a significant hedge against the incoming tax policy makers, I need a lot of liquidity to be able to place that bet against. What are the challenges in a prediction market aggregating enough liquidity on either side of a trade?

[00:02:50] JK: Yeah. So there's a bunch of challenges with it. One is that prior to being blockchain-based prediction markets, most prediction markets were run and owned by a central company, which means they tended to fragment the liquidity. You might have a prediction market which focused on, say, New Zealand and you might have another prediction market, which focused on Spain and a separate one for the U.K. and so on and so forth. And so one problem is you have this liquidity fragmentation. The second issue is that just forming liquidity for new events is hard. So if you look at something like the stock market, it has these huge systems and firms and entities that exist solely for providing liquidity and there's a ton of different ways that they can hedge their risk. But with a prediction market, you're betting over real-world events. And not only that, but if you're wrong and if you hold the risk in the wrong way, you can actually lose all your money. The example of this is like take a presidential election. If you bet on Trump and Trump loses, you lose all the money that you bet on him. And so for a market maker, that's very risky.

If you look at, say, like market making Apple stock, the worst case scenario is that Apple moves a little quickly and you end up holding too many or too few Apple shares and you maybe lose in a big move a number of percents. But with prediction markets, something can swing from a ten percent chance to a zero percent chance or a hundred percent chance in a matter of seconds or minutes. And so that's what makes market making and getting liquidity very difficult.

[00:04:25] JM: Before you started Augur, there were a number of centralized prediction markets. What are the problems with centralized prediction markets?

[00:04:34] JK: Yeah. So if you look at the problems of the centralized prediction markets, there's a couple. There's one problem that even when I got into the space I thought it was just a problem in theory. I didn't think it would actually be a problem in practice, but it turns out it is a problem in practice sometimes as well. And that problem is because these things are real-world events, they have a resolution date at some point where the event has to be paid out. You need to declare a winner. And actually sometimes there's controversy over how a winner gets declared. So that's one problem with a centralized one that in a decentralized system you could have a more community-driven process for resolving a market and paying it out that sort of ensures more people are happy with the result. So it's less arbitrary in that sense.

The other kind of big issue with centralized ones is that they tend to be located in one or two jurisdictions or a handful of jurisdictions. Nobody's really created a global centralized prediction market mostly for the reason that the compliance cost of getting licensed in 180 countries would just be too much overhead and too much burden for any one entity to do it.

[00:05:45] JM: So are you saying that the main value of a decentralized prediction market is compliance?

[00:05:53] JK: I would say it's mostly like the element of if you look at these protocols or these decentralized systems they sort of push compliance to the end points, I would say. So if you look at like –I'll walk through Bitcoin as just a quick example. You look at Bitcoin, Bitcoin miners and Bitcoin nodes themselves aren't regulated. But the endpoints where you get your dollars in and out, Coinbase, Kraken, Gemini, et cetera, those endpoints are regulated. So if you look at prediction markets, there's not really any sort of formal approach towards regulating decentralized prediction markets yet. I think that's probably years away and there needs to be a lot more usage before that actually becomes a reality. But you could envision different types of regulating it. One way to think about it would be just, “Well, you just regulate the fiat on ramps just like you do with Bitcoin. Another approach might be to say, “Well, you could regulate the people who create the market and set the terms for the market.” That's one approach you could envision regulators taking at some point. A third approach might be to say, “If somebody is hosting a UI on top, it's up to that person to do things like block certain customers depending on what jurisdiction they're in and that sort of thing.” There's a bunch of different ways you can envision it, but the most important thing is that when you look at these decentralized prediction markets, besides Ethereum nodes, nobody is actually holding the money and processing the bets. And those tend to be the biggest regulated activities. Anytime you touch money, whether it's for a prediction market or anything else, you start to fall under a lot of regulatory issues, and that's kind of the innovation behind Bitcoin, the Bitcoin nodes. It's a bunch of nodes that touch the money and there's not like one entity processing all the transactions. It's kind of a similar story for decentralized prediction markets.

[00:07:42] JM: So are decentralized prediction markets legal today? Are they legal around the world?

[00:07:48] JK: Yeah, that's a good question. I think it depends what you're doing like anything else. One way to think about that question is to sort of think about it's kind of like asking is Bittorrent legal? And the answer is, well, it really depends what you use it for. It depends on what country you're in and where you're based and it depends on a bunch of different factors. I think if you look at the history of betting in the United States, one thing that's generally been fairly protected is the role of individual bettors. So assuming you're not market making, assuming you're not taking other people's bets and assuming you're just an individual better, there haven't really been any successful prosecutions of individual bettors for using betting sites regardless of where the sites are located. Obviously a decentralized version is kind of different because it's like where is it located? Well, it's located on Ethereum. But I think some of the principles from those, that case precedent would apply.

The other element is the people who create the market itself. So the person who proposes the question, who sets the terms for the market and who takes a fee on that market is a really important element. In that part, it depends on – Every country has a different way of regulating it is the short version. Some countries, it's not regulated at all. Some countries, you get a license and it's pretty straightforward. In other countries it's outright illegal. And in some countries there's a licensing process, but it's more involved. The U.K. would be an example of that. So there's not really a short or simple answer. I wish there was. I think these markets are really useful and provide a lot of useful real-world information, but they've never really gotten popular enough yet for there to be kind of a global regulatory framework for it. Instead it's fairly fragmented.

[00:09:35] JM: We can set aside the legal distinctions for a moment. Let's talk about engineering. So you set up or you conceived of Augur on the Ethereum blockchain. And I'd just love to get a high-level overview of how you envisioned Augur working.

[00:09:57] JK: Yeah. So if you go way back to kind of late 2015 when we were really making a lot of the core design decisions, actually late 2014, the main thing we were thinking about is, "Okay, How can we even build this in the first place?" We initially started looking at Bitcoin and trying to make some modifications to Bitcoin but decided that would be far too complicated. And what we would end up with wouldn't really be Bitcoin anymore. It would just be this really modified piece of software. And then Ethereum came out. It was about to launch. And we decided to try to build what we've been building on Bitcoin, but on Ethereum. And it became clear that that was going to be the winning architecture. It was far easier to build on. The kind of idea of smart contracts is basically perfect for what you're doing with these prediction markets. Prediction markets are basically codified financial contracts or codified agreements that move money around, and so smart contracts were a perfect fit for that.

I think one thing that we underestimated is how complicated the user interface layer would be for this stuff, because it's pretty easy to write a smart contract and to publish data to Ethereum and everything, but getting the data off the blockchain, reading the data from the chain in a way that's efficient, converting basically a relatively inefficient data structure of blocks into something that can live in a user interface on the user's own machine in a way that actually loads quickly turned out to be like a multi-year, really challenging problem that we've only really just sort of figured out.

[00:11:34] JM: So Augur has a token associated with it. What is the purpose of the Augur token?

[00:11:43] JK: Yeah. So if you look at the token, it's called rap or reputation. And the idea for it came about basically by trying to answer the question of how do we actually pay out an Augur market? So say it's a market on will the rocket launch successfully or not? And the resolution criteria are is the launch sequence completed and did get off the launch pad? That's sort of all we care about in this example. So you need somebody to tell Ethereum to tell the blockchain what actually happened in the real-world. And the first reaction that anybody who comes to is, “Okay, you just appoint somebody and have them do it.” Now the problem with that is it centralizes the system. That person can lie. They can be bribed and they can just hold the system. There's a ton of problems with that. So that doesn't really work.

The next kind of thing people think of is like, “Well, can't you just like plug into an API?” And Ethereum nodes don't – Within the EVM, within the Ethereum virtual machine kind of execution environment, they don't actually have access to the internet. So they can't make API calls. If they did that would actually be a pretty solid solution. And so the solution we came upon is the idea of basically having real people, put something at stake. If you're familiar with proof of stake, it's kind of a similar concept to that. And you basically stake on what the actual reality is. And a market can have multiple possible answers. It could be for a simple yes or no market, it could be yes, no, or it could be invalid. An invalid market might be something like – It would be like, “Is today a good day?” is probably the best example of that. So what we basically came up with is you have this token. People have to stake it on what they believe the true result is. If you're with the consensus, you basically earn fees. And if you're against it, you actually lose a portion of your stake. And so that's sort of how the rap token came about.

[00:13:40] JM: Can you clarify in a little more detail why do you need to use a rap token? Why couldn't you use eth as the mechanism of settlement?

[00:13:50] JK: Yeah, for sure. So the reason why you need a separate token is, basically, if you use a token like eth, you run into this problem where the solution you end up creating becomes a Keynesian beauty contest. So what ends up happening is whoever sort of has the most money wins no matter what. And there's this other problem called the P + epsilon problem. I’m probably going to ruin the explanation of it here, but Vitalik Buterin has a great blog post on it. But basically the idea in a nutshell is if you have any sort of like staking system, people can basically say they can effectively bribe you for far less money than you would originally think that it would cost because they can basically, via another smart contract, promise to pay you a small amount of money if you vote the way that they want you to. Otherwise you basically agree to cover their losses. And the like game theoretic equilibrium here is that people are incentivized because it's contractual and you can prove it on chain. They're incentivized to go through the manipulative route. And because of that, the consensus result ends up being manipulated. Nobody loses any stake because you manipulated the result and the majority of people went along with it. And you also get this extra small return on top.

And so the way to combat that problem that Vitalik proposes in the blog post is you add what's called forking. So the network can basically split into two networks where it now has kind of two parallel universes. And so like in theory you could use eth as the currency to resolve these markets, but you would require Ethereum to fork for it to actually be secure. And so I think it's pretty unrealistic to assume that Ethereum would fork around any one application even like Augur. And so we basically created a new currency so that we could solve this attack vector.

[00:15:46] JM: So what properties did you have to add to rap to prevent these attacks?

[00:15:53] JK: Yeah. So the property that you add to it basically is when a fork happens, it needs to be able to split into two new tokens. And so an example would be like if you had a market that was like will the weather be good? And the answer is yes or no or invalid. Say a bunch of people said that the weather was indeed good, which is like completely arbitrary. The network in theory should eventually fork into two where you have people who think the weather was good and people who think the whole question was just invalid. It's not a question you could actually ask Augur. What would happen on the token side is there're basically now two tokens. There's a token that represents the yes universe and there's a token that represents the invalid universe. In each of the new universes, people get minted tokens that reflect the losses from the other side.

So as an example, say you basically put your rap on yes. In the yes universe, you'll get rap from the people who staked on invalid. In the invalid universe, you get the rap that the people lost from the rap side. So now there's two parallel versions of the rap token where the token has effectively been redistributed based on which side you took in the fork. And the idea is that the actual market, like people trading the rap token, should start to value one of these more highly than the other because it's the one that actually reflects reality. And then the idea is that people would sell off the one that doesn't reflect reality.

[00:17:22] JM: So these problems were being solved over the course of the last five years or did you solve these before Augur even launched?

[00:17:28] JK: Yeah. So I would say the P + epsilon problem we solved before it launched. Vitalik was actually the person who came up with the solution for that. And let's see, other problems though – Like there's a bunch of other problems we didn't solve until after launch. Like as an example, there was this problem when Augur v1 came out where people started creating markets that were clearly invalid to try to trick users into losing money on bets that looked obvious. It would be like someone would let you bet on the price of eth. Will it be above a hundred dollars? And eth was trading at 500. And they'd let you buy that really cheaply and it looked like you were getting a great deal, but in reality they're market resolved according to what their personal twitter account said. So there were like incentive issues and stuff that we fixed along the way to mitigate issues like that. But the P + epsilon like forking problem, that's something we fixed prior to launch.

[00:18:24] JM: Interesting. Can you explain that example with the Twitter resolution a little more? I guess that gets into Oracle's, right?

[00:18:31] JK: Yeah. Yeah, that's an interesting problem. So if you look at Augur, you could create a market and you could really specify one of two things. You can say, “Hey, reporters, I want you to resolve this market according to like general knowledge. Like what you would be able to find out pretty quickly on your own.” That works for some things. Like it works for like a sporting event. It probably works for a presidential election. What it doesn't work for is it doesn't work for things like what's the price of ether on a certain date. And the reason it doesn't is for kind of an obvious one, which is there's a ton of exchanges and people may be looking at a different exchange, different time zone, whatever. And so you really need to say what is the price of ether on this date according to some source? And so the problem that happened there is people realize that most users don't actually read what the source is. They just kind of glance over it and decide to place a bet based on what they think. And so people started making markets where the source was something that was just a scam, like their Twitter account or etherpricefeeds.com or something, a random site they made up. And they would take a bet and position accordingly and then the reporters would be forced to resolve the market according to that source. But it wouldn't actually be the true outcome that you and I sitting here would actually know to be the truth. And so that's an incentive problem. That's like one. There're a ton of interesting incentive problems that pop up when you're designing something like Augur. The short version of that is – The short fix for how we fix that is in v2 we decided to let people actually bet on whether a market is invalid. And when you do this, the free markets actually pretty good at weeding out markets that it thinks is invalid. And then you can basically kind of, by default, those don't show up in the UI unless you choose like an expert mode where you can start toying around with those.

[00:20:31] JM: Gotcha. So how do you feel about the Oracle system as it stands today?

[00:20:36] JK: Yeah, as it stands today, I think it's extremely robust. It works really well at this point. Most of the major kinks have been worked out, but there is one kind of major deficiency still that I see with it, which is that it's slow. And v1 of Augur, it took markets months to resolve because we were really conservative with the parameters that the system was launched with. And v2, most markets resolve in a few days. Some markets may take a couple weeks. But I still think that's too long. The average user, the average trader or better is looking to get their market paid out. They'd ideally prefer same day. The kind of worst case, next day would maybe be acceptable. And that's like a challenge for how do you make an Oracle system that's fast, but also secure? We're kind of trying to think through how will we do that for Augur v3. That's like the next improvement that needs to happen.

[00:21:34] JM: Tell me about the state of Augur today. What is usage like and what are the most common use cases?

[00:21:43] JK: Yeah. So I'd say the most common use cases tend to be political markets and some sport markets and then some crypto markets. Those tend to be the three categories. There's a project built on top called Catnip, which actually has the majority of usage even more than the default UI. And what their approach has been is pretty clever. They basically said, “Hey, it's pretty tough for people to market using an order book. It requires a lot of specialized knowledge. What if we use you know automated market makers like the same tech that Uniswap uses for these prediction markets?” And so they actually built that using a system called Balancer. And they've actually been doing pretty decent volume. There was about 20 or so million dollars bet on the election. Most of it through the Catnip UI, which is a UI that uses the Augur protocol underneath.

I would say the other state of things right now is we're working on basically a much more simplified UI. I think in the past we kind of had this attitude that we needed to build something that was useful to the average better and solve a zillion kind of really complex problems. And I think at this point our viewpoint is more that crypto is actually growing quite fast. If we can build a product that's easy to use and understand for people within crypto, that's probably a better spot to start. And so that's kind of what we're working on right now and that's ready to go live in a few weeks.

[00:23:08] JM: Could you take one of those examples like the political election example and just walk through? Let's start at a high-level and then we can dive a little bit deeper into what the user experience is like for wagering on a political outcome.

[00:23:25] JK: Yeah. So the way it works is you would go to one of the Auger UIs. The cool thing about something like Augur is that anybody can launch a UI on top, and there're multiple variants and they all kind of have different flavors. It's kind of like if you think about the different user environments on Linux. That's kind of like that. And so say you went to the Catnip UI. Their model is basically showing you a few top markets that have a good amount of liquidity in all of them. And so you might go there. You're going to click in the UI. You'll see a handful of markets listed and you might click on one of them. Say it's the market on the Super Bowl. And the first thing you're going to see is you're going to see this window and it basically has an input box for how much you want to bet. And yeah, maybe you put in 50. The other thing you'll know is that it's actually not in dollars per se. It's in a currency called Dai. Dai is pegged to the dollar. It's a decentralized stable coin basically.

And if you look at it from the user perspective – So if you're on the site, one of the things you're going to have to do is you're going to have to acquire Dai. And so this is one of the like UX challenges and things that using this stuff tough. And so you can get Dai on Coinbase though. So you go to Coinbase. You send them 50 bucks. You get your Dai and you withdraw it. And where you're going to withdraw it is you're going to withdraw it to your Ethereum wallet. And most people are using a piece of software called Metamask. It's a Chrome browser extension and basically you would send your Dai to Metamask and then you'd go back to Catnip and your Dai is now in your wallet. And you're going to choose what you want to bet on. So you could say if we look at the Super Bowl, it looks like – I actually know like nothing about the NFL. I'm more into horse racing. But it looks like the two teams for the Super Bowl this year are the Buccaneers and the Chiefs. So say you click on the Buccaneers. It's going to tell you how many shares of Buccaneers you're going to get and what the price is.

So right now $50 of Dai gets you 121 Buccaneers shares. That means you're paying 41 cents per share. So what that means is if the Buccaneers win, you're going to make about 59 cents in profit per share. It'll run the calculation for you and tell you that your max profit on the $50 bet is going to be $71.60 cents. And then so the next step if you wanted to move forward with it as a user, you'll click confirm. Then what's going to happen is Metamask is going to pop up. It's going to have you sign a transaction. And what this transaction is going to do, I think first you have to sign a transaction that enables betting of your Dai. So you're going to approve a smart contract to be able to spend the Dai or the dollars that you have in your wallet and then it's going to have a second transaction. In the second transaction is basically going to be an intent to trade. You're basically calling a function on this decentralized exchange that plugs into Augur and you're basically saying, "Hey, I'm willing to spend 50 Dai to get Buccaneer shares. And then that basically gets broadcast to Ethereum. It gets put into a queue by miners and transactions get processed according to gas price priority. So if you pay more for your transaction, it'll get processed faster. If you pay less, if you're not too time sensitive on getting your bet placed, you can have a lot cheaper transaction fee and it'll get processed slower. And that's basically it.

The next phase of this, what happens next, is the Super Bowl happens, right? And then the payout would occur. And we already kind of walked through how the reporting system worked at a high-level and I'm happy to get into any more detail on that if you like. But basically on the user standpoint, once the market's resolved, they would then go to the UI and basically be able to claim their winnings. And that's just another Ethereum transaction where it basically exchanges your Buccaneer shares for underlying Dai. And then hopefully you won, you made $71 and you could choose to keep betting. You could choose to use another DeFi app where you could if you wanted. Withdraw it back to Coinbase and cash out for regular dollars.

[00:27:40] JM: So now that we've given a high level example, let's go a little bit deeper. Tell me about the interface between Augur and Ethereum and basically like what is the set of smart contracts that are built on top of Ethereum that are powering this?

[00:27:55] JK: Yeah. So there're a few things that happen. So on the smart contract side there's really I'd say a few main components. One of them is what's called the market contract. That's a smart contract that basically contains a description of the market. It has information about how many outcomes the market has, information about the names of those outcomes. And it also has the source, that the market as a resolution source. And the next set of smart contracts is these contracts called the complete set contracts. And what those are, this kind of gets into how it actually works. So if you think of any Augur market, any Augur, yes, no invalid market, those are the three potential outcomes. If you buy a complete set, what that means is you put up one Dai, which is worth $1. So you put up $1 and you get one yes share, one no share, and one in valid share. One of those things has to happen, and one of those will be worth $1 when the market resolves. Ahead of time you obviously don't know which one it's going to be.

This contract is pretty important. It's kind of the crux of the system. This contract also lets you sell complete sets, and there's actually a small fee on selling complete sets that goes to the market creator and goes to people who report on the market to resolve it. And selling works the same way. You give the system – Say you want to sell 100 complete sets. You would give it 100 yes shares, 100 no shares and 100 invalid shares and you would exchange them back. Then with that component you can do a few different things. The smart contracts themselves have a trading engine built in that lets you use actual limit orders. And so that contract, the trading contract, lets you basically pass it an array of limit orders you want to try to fill and it will start trying to fill them based on which one has the most favorable price to you. And the way limit orders work is pretty interesting. Obviously it would be a pain if you had to store every order on chain. So instead what happens is you actually sign a message with your Ethereum wallet. Every wallet has this functionality where it can sign a message. And most of the wallets have like an API where you can just request this on behalf of the user and then the user just has to confirm it or deny it so that the user doesn't have to do anything complex here.

And the message you would sign would like have basically hex encoded. The market ID, the contract address for the market you want to participate in. Which of these outcomes you want to buy or sell? Whether you're on the buy side or the sell side, what price, what your limit price is that you want to do it at and what your quantity is. And then if somebody else – So you can then broadcast that and that gets broadcast over what's called 0X mesh. So 0X is like the set of – It's this decentralized exchange that basically lets you like broadcast these orders over a mesh network that uses a lot of the IPFS gossip kind of stuff in browser, gossip orders to other peers. And other peers would just be other actual users of augur.

And so on the UI side, you're entering a limit order. In the backend what's happening is you're assigning this message. It gets broadcast over this IPFS gossip sub thing. And on the other side a user may say, “I want to trade.” And when they do that, the system basically selects a set of orders that fill their criteria and then broadcast them to the trade function on chain and they end up getting their fill. And just to be like completely extensive, that's one way trading works. The second way that trading works, which is the one that actually sees more usage, is the way Catnip does it. And so the cool part about these systems being open source and fairly modular in the fact that anyone can access them is Catnip basically said, “We don't really like that trading model.We don't actually think it's the best way to do things,” and they actually ended up being right on this. And they said, “We're going to hook into the complete set system,” the one that I walked through earlier, but when we use those complete sets, we're going to hook up to a different trading backend. Our trading backend is not going to use an order book. It's going to use an automated market maker. And so on their end they're basically buying complete sets through the same contract, but then they're depositing them into another set of contracts, which is this protocol called Balancer. And the Balancer smart contracts are basically a set of contracts where you deposit in a number of tokens and it basically automatically quotes a price based on a math function. So there're no actual explicit orders in the book. It instead just quotes a price based on how many each of these tokens are left in the smart contract. And they've come up with some really clever math that ends up making it so that being a market maker is actually decently profitable as long as you don't have risk when the event resolves.

So like right before the market resolves, you'd want to pull your money out, because at that point the market can resolve and you have your capital at risk. So you wouldn't want to do that. That's most of everything. The very last thing that kind of fully understand how Augur works on a contract level is the reporting system. And what I walked through earlier is pretty much maps to how the actual contracts work. The one thing I didn't address, which I'll just address quickly here, is the disputing process. And so for reporting, when a market goes into reporting to actually resolve it and pay out the outcome, the way it works is the person who created the market says what they think happened. And if that person's honest and if things are going fine, nothing's going to happen. 24 hours passes and the market pays out. However, if that person is dishonest, somebody can challenge it and they can basically post collateral and say, "Hey, this person said the Buccaneers won. Actually the Buccaneers didn't win. The Chiefs won." And that person is basically posting collateral in rap. And then it goes into these dispute rounds where people can basically keep disputing it back and forth. And eventually if enough disputes happen, the system reaches that fork threshold that I mentioned earlier where it kind of splits into two and redistributes rap based on which side you were on.

[00:34:24] JM: How did you test Augur as you were developing it and rolling it out to production over time? Do you have a consistent means of making sure it functions as intended?

[00:34:36] JK: Yeah, it's a great question. So these systems are really challenging to test because they involve real money and economic incentives. And so the main way we've tested things has been one is having auditors review the code and review the incentive design before we do releases. They tend to actually catch a lot of good issues that would have been bad if they weren't caught having an economist look at the system. The other kind of way that's a pretty good way of testing is the Augur community is actually pretty helpful with this. There's a public discord where kind of all the development talk happens and there's actually a game theory chat in there. Any kind of new idea or design for the system really gets put through the ringer in that chat. There're a couple hundred people and sometimes even people who like haven't been active for six months or a year when they see an important conversation happening in there. They'll kind of resurface. It's pretty interesting. And so we've been able to prevent a lot of design issues based off that. It's this kind of process where people get pretty adversarial. They'll try to really break any idea that you propose and make sure it's resistant to attack. But there're some things that have slipped through that process. Like one thing that was just kind of an oversight is the disputing process. So the very first dispute is actually quite cheap to do. It costs like 10 to 20 dollars in U.S. dollars.

And so there's a problem there, which is somebody could just dispute a market just to delay the resolution just kind of to troll or out of spite or for any reason. And there was this mechanism where people could basically pre-stake. So the market creator could say, “Hey, I’m so confident in this market resolving the way I think it's going to resolve that I’m willing to put some capital at risk up front.” And most people thought that that meant that it would cost more to challenge it, but instead kind of due to like an incentive design oversight it actually still only cost $15 to do the first challenge. After the first challenge, it gets expensive again. And that's just like one example of something like we're going to have to fix or iterate through in the next release that didn't get caught. So not everything gets caught, but it's tough designing these open systems where anyone in the world can can access it to try to do whatever they want.

[00:37:02] JM: We're now in a time when DeFi is becoming a word that's really commonly associated with the crypto markets. And I wonder if you have any perspective on the intersection between prediction markets and DeFi. For example, I could imagine some sort of tokenized prediction market position that you might be able to leverage or borrow against or do something like that. Are we like way too early for that or do you think that's a possibility?

[00:37:38] JK: Yeah, that's a good question. I think it's like I could see that happening in a few years. I think right now with prediction markets, the challenge with them is just if you look at DeFi, they can get a lot of usage because the dollar sums are very high. And so an example of this is there's people out there who own 100,000 plus or even a million plus dollars’ worth of Ether. But even those people who happen to be very wealthy, they may not be comfortable betting more than 500 on some event even if they're otherwise very wealthy. And then the people who are comfortable betting huge sums of money on events, there tends to be problems with Ethereum that make it tough to use, that make them not want to use something like Augur quite yet. And so prediction markers are kind of in this weird in between like uncanny valley stage where like their Ethereum needs to scale more so that it's cheap enough to use so that Ethereum folks will use them. And then it's still too difficult to use for average bettors to use them. And so I think what's going to happen first is Ethereum will scale if you have these layer two systems, which will then make prediction markets easy to use for Ethereum folks. And then after that, we'll figure out the usability piece.

[00:38:59] JM: Going a little bit deeper on the DeFi area, given how much you have worked on building an engineering stack around Ethereum, do you have any perspective on the state of building blocks, DeFi infrastructure building blocks? And can you just give me a pulse check on what it's like to build decentralized infrastructure today on top of Ethereum?

[00:39:27] JK: Yeah. I think – Man! The space is advanced so much since I first got involved in it. At this point, it's fairly quick to launch some new sort of DeFi protocol. There're so many different primitives and things you can build on top of. There's decentralized lending protocols like Aave and Compound. There's like 45, I think, different decentralized exchanges that have all different kind of design variants within them that you can interact with and build on top of. And then there's even like decentralized insurance protocols now. And so all these kind of underlying primitives exist. I think it's easier than ever to make a new decentralized finance protocol just because the dev stack has improved so much. And then it's also getting easier and easier to build user interfaces on top. It's still fairly challenging to do it in a way if you're trying to maintain decentralization even at the UI level. I would say that's still five years away from being easy. But I guess the way I would describe it is back when I started building on Ethereum back in 2014, it felt sort of like somewhere between assembly and MS-DOS, and then now like it's kind of like Windows 95 or something. It's like the infrastructure like AWS and all that stuff just doesn't exist yet really. It's not that easy, but it's much, much better than it used to be.

[00:40:52] JM: And this is pulling out of your exact expertise in Augur, but I'd love to know your perspective on the intersection between crypto markets and "normal markets" and kind of what that's going to look like. I mean, when does DeFi start to provide functionality that is useful to somebody who is, I don't know, getting a mortgage, for example? Or is that the wrong way of looking at it?

[00:41:24] JK: Yeah. I mean, I think the way I kind of described the progression is two years ago, for the most part, DeFi wasn't really useful to anybody. Even within crypto it was kind of a toy. At this point DeFi is incredibly useful to people within crypto. I use it on a regular basis. For instance, if you're trying to trade one cryptocurrency for another, if it's an Ethereum based token, these decentralized exchanges tend to offer actually depending on how much size you're trading. Better rates than the centralized ones. And so I think it's actually adding a lot of value whether it's that or using these lending protocols today for people within crypto.

The next phase is going to be adding value for people outside of crypto, people who get into crypto because there's a DeFi app that adds some real-world value for them. I think that's going to be a much longer process. It's going to take 5-10 years. We might see some early use cases. I have heard like the MakerDAO protocol talk about like this idea of tokenizing mortgages and putting them on chain to offer people slightly better rates than they could get elsewhere. I don't know how practical that is or if it's going to happen anytime soon. I think those use cases are further out than probably most people think, but I think they will happen. And if you look at kind of the real world, people are starting to kind of run into the issues that DeFi solves. Like if you look at kind of – it's a humorous example, but if you look at the stuff that's been happening with GameStop recently Robinhood and a bunch of other brokers stopped letting people buy the stock. The cynical view on that is, well, they did it because hedge funds told them to or they just didn't want to deal with it. The more practical view that I have is I think they actually did it because they are running into capital requirements issues. There's a bunch of complex capital requirement rules for firms like them where they have to post a bunch of margin and stuff. And I think if they would have kept letting it go on, they would have been underwater on that, is what I think actually happened.

And if you look at DeFi, it just doesn't have these problems. I mean, first, someone can't decide to unilaterally prevent you from trading a given DeFi asset. And then second, all the collateral stuff is calculated on-chain. There is no like three day settlement period where you need to post a collateral but you're waiting for a wire to settle and then somebody at the DTCC, which clears all the securities in the U.S., has to do something. And there are all these complex layers. All the stuff we discussed about Ethereum and Augur, it sounds incredibly complicated. But Augur’s like smart contracts, they're probably a few thousand lines of code at most versus like Wall Street is just an insane amount of lines of code that happens from the second you place an order on Robinhood to when it gets executed, versus something like Uniswap is thousands of times simpler. So I do think it is the direction that things are going. It's just going to take time.

[00:44:23] JM: Well, as we begin to draw to a close, how do you expect to see prediction markets change within the next five to ten years?

[00:44:32] JK: Yeah. So I think within the next five to ten years, I think prediction markets will be much more popular. I think most of them will be more global than they were in the past. I think it'll still be a mix of centralized ones and decentralized ones. I think the decentralized ones will have worked through the major kind of user experience issues. And so they'll actually be quite easy to use. And because I think in that time frame, acquiring crypto will be much, much easier. You'll be able to just input your debit card or credit card right on the site and there'll be a crypto exchange that processes it in the background. All that stuff is going to be way built out by that time period. And so I think the user experience will actually be in many scenarios better than the centralized version. And so I think that's sort of where I see it going. I think very long term you could start to see these things being viewed as forecast. If you look at how on CNBC they're always talking about the likelihood that the fed raises rates. And they just get that from the markets. There're markets where you can basically get the pricing of that risk. And I think people will start to use other markets to price different types of risk. So it wouldn't surprise me if in five to ten years on CNBC when they're talking about maybe we're in another financial crisis by that point and they're talking about another stimulus package. It wouldn't surprise me if they quote the odds from prediction markets. What are the odds this bill passes as written? Stuff like that.

[00:46:01] JM: Cool. Well, Joey, thank you so much for coming on the show. It's been a real pleasure talking to you.

[00:46:04] JK: Yeah, thank you for having me.

[END]