EPISODE 1244

[INTRODUCTION]

[00:00:00] JMeyerson: Volatility is the degree of fluctuation of something's price. Highly volatile assets may see rapid and large price changes while less volatile assets will maintain a steady price. This concept is important in decentralized finance because cryptocurrencies tend to be volatile assets. The company Synthetix provides assets called Synths that provide exposure to an asset without holding the underlying resource. For example, you can hold and trade Synths that attract the price of USD, synthetic gold and silver, and other currencies in commodities. Users use Synthetix to track the price equivalents of real-world assets on Ethereum. This lets them diversify their investment portfolios with less volatile assets while staying on the blockchain and executing trades against smart contracts.

In today's episode we talk with Justin Moses, CTO at Synthetix. Previously, Justin worked as a CTO at Havven and as a tech advisor at Blueshift. We discussed derivatives trading in DeFi, the liquidity and volatility of synthetic assets, and the rewards and features available from using Synthetix.

[INTERVIEW]

[00:01:05] JMeyerson: Justin, welcome to the show.

[00:01:06] JMoses: Thanks, Jeffrey. Good to be here.

[00:01:09] JMeyerson: Simple question, what is a synthetic asset?

[00:01:13] JMoses: A synthetic asset is something that derives its value from something else. I guess typically a synthetic is not the same as the real asset. Holding this synthetic is not like holding an asset but you're holding something that effectively matches the price of the asset.

[00:01:30] JMeyerson: Why are synthetic assets important in traditional markets?

[00:01:35] JMoses: I mean, synthetic assets are effectively – They're derivatives. And I'm sure a number of your viewers have heard about derivatives and what they are, but they basically allow you to derive extra value from something. They're useful in one thing, let's say, the price of – I don't know, the price of the Australian dollar to the US dollar for example, or say the price of Tesla on the Nasdaq or even say the price of Bitcoin. These are useful things, and they have use cases around the world. You also have things like mortgage-backed securities and stuff that obviously we've all heard what happened in the 2008 credit crisis, but it allows people to create more value and more financial instruments from a thing than is traditionally available.

[00:02:21] JMeyerson: What kinds of problems would synthetic assets solve in the blockchain space or the DeFi space?

[00:02:30] JMoses: Those are the problems that synthetic assets solve, is it gives people access to things that they may not be able to get otherwise. They may want access to the price action of, say, securities in some country they're not a registered investor in. They may want to have access to a currency saving in US dollar that they can't get access to based on their country. The rules that the government has applied on some of them. Or they may want to – Even, say, on a blockchain like Ethereum, they may want to hold something that tracks the value of an asset on another blockchain and they don't want to have to actually custody the asset on the other blockchain. If you're on Ethereum, you could potentially have – Excuse me. A synthetic Litecoin instrument that effectively tracks the price of Litecoin, but doesn't require you as an Ethereum user to actually go and exchange it and have a wallet and set up an account on the Litecoin uh blockchain for instance.

[00:03:26] JMeyerson: What about problems around liquidity and slippage? Are there problems around liquidity and slippage that could be addressed by synthetic assets?

[00:03:36] JMoses: Yeah. I mean, I think with our platform, I mean, the way it works in a nutshell is effectively that everyone who participates in creating the debt pool, we call I, everyone who puts up collateral in the form of our token into a contract and basically issues debt. Those people are obviously rewarded for their service by getting paid back periodically, every week. But what that does is – In Synthetix's perspective, it basically creates this big undifferentiated pool of debt. And that means that anyone can take one part of that debt pool that they can buy off any market, say, buying a synthetic USD, and they can re-price it into anything else that is offered as a synthetic asset.

If you were able to access let's say a thousand SUSD, you can then immediately turn that and trade into synthetic Bitcoin or synthetic Tesla or in a synthetic AUD or WI or what have you, and you don't get slippage, because what you're really doing is you're repricing a piece of a debt from one asset to another. The other side of that trade is actually the aggregate of synthetic stakes. It's sort of a novel way to sort of trade. It's instead of it being like an order book base where you would – There'd be someone who'd make a trade and someone could take the trade and you need to match it. Instead of that, what you have is this idea that there's this – As long as you get access to any synthetic asset, you can immediately convert it, shall we say, to any other synthetic asset without slippage. Slippage is typically when, as you say, there's not enough liquidity. There're enough people on the other side of the trade. So you have to accept a poorer price. In Synthetix we use the term Internet liquidity, because as long as you can get any synthetic asset, as much as you can get on off market of it, and right now I think there's about in order of – The debt pool is about 600 and something million in size. As long as you get any part of that however you can, you can convert 100% of it to any other synthetic asset. Without slippage, you just pay the exchange fee. An exchange fee which is anywhere from 30 BIPS, so 0.3 to 100 bips and gets paid and distributed proportionally to the people on the other side of the trade, which are the synthetic stakers who take the risk by basically having a piece of this debt pool. Having that sort of debt tied to them.

[00:06:06] JMeyerson: We’ve kind of skipped over the basics of what you’re building, of what Synthetix is. Can you explain what Synthetix is one sentence?

[00:06:16] JMoses: Yeah, it’s a decentralized derivatives platform that allows anyone to participate in the trading of synthetics assets.

[00:06:33] JMeyerson: And reiterate, what are the problems that Synthetix solves in the existing DeFi space?

[00:06:40] JMoses: I mean, Synthetix is one of the early sort of proponents on DeFi. It actually originally came from a project called Havven that was actually a decentralized stablecoin, a little similar to Maker and Dai and sort of evolved into this sort of derivatives platform. And we're really trying to solve people to access to all sorts of different assets that they couldn't otherwise acquire while they are still on Ethereum. We basically allow people to get access to any of those and even the stuff that might not be very well traded, so maybe something like – I was thinking of something like Dash, right? We have synthetic Dash. You can still get access to that and you can trade that without worrying about the fact that it's probably – There's very little people trading a synthetic dash on Ethereum, but you can still trade it for 100% of its value into any other synthetic asset. Basically giving people access to, I guess, the long tail of assets that they couldn't otherwise access and trade.

[00:07:43] JMeyerson: Walk me through what happens if I want to use Synthetix? What is a typical interaction that a user might have with Synthetix?

[00:07:53] JMoses: Well, there's actually two main cause of abuses. There are stakers and there will be traders. A trader, the one I've mostly been talking about, this would be a person who say they have a pile of Ether in their wallet and they would say, "I want to get access to a synthetic dash," as we were before, "or I want to get access," I don't know, "synthetic Eos all at once on Ethereum." They can basically take their Ether and they can use an aggregator like 1inch and they can convert their Ether into SUSD at whatever the market rate is. So that's using – 1inch is an aggregator that will look on-chain for all sorts of different smart contracts that can convert Ether to SUSD. They now get a pile of SUSD and now they're holding a synthetic asset, synthetic dollars, and then they can go on to adapt a decentralized app call Quenta and they can convert 100% of their SUSD into any other synthetic asset for the fee of, say, something on the order of 50, 100 bits depending on the asset. And 100% of the value minus the fee will be converted at the current market rate. And at any time in the future they can then convert that synthetic Eos, say, into other synth. They don't have to go back through synthetic US dollars. They can go directly to, say, synthetic Bitcoin for instance. They can go straight to it, again, with only paying that fee, exchange fee for whatever the asset they're going to is. That's the trader workflow.

Now, that's powered by the stake workflow. The staker workflow is a person who acquires SNX, or collateral tokens somehow. Potentially they would also take Ether and maybe they would go into 1inch or maybe even Uniswap or any of the many – Or Curve, or any of the many places to get assets and they would exchange their Ether for Synthetix, paying whatever rate those aggregators charge them or those swapping utilities charge them. And then they would take that SNX and they would take it to a dapp called Staking. They would stake it. Basically effectively locking it in place and they would issue for themselves some synthetic US dollar. And they then can trade that synthetic installer as someone would on Quenta, or they can put it into different places on Ethereum to get a yield so they can put it into Curve. Curve is another DeFi protocol. And they then get a piece of the curve. They get a liquidity pool, an LP token, pooled liquidity token and then they can stake it with us and get incentives back with Synthetix.

But regardless, the act of staking SNX and issuing debt means two things. It means, one, that the person who stakes is now a decentralized piece of the debt pool. They are contributing to basically this power that is anyone can move one synthetic asset into another. And for that they get paid on a weekly basis the percentage of the exchange fees. It's a little bit like miners on Ethereum or Bitcoin of basically being incentivized by the fees, the transaction fees that happen in other block they mine.

Now we also – The protocol also gives inflation. There's a set inflation schedule, and every week these get divvied out in proportion to the SNX stakers based on how much debt they hold. And that's a little like block rewards, I mean, mining block rewards. The idea is that this is a bit of a bootstrapping mechanism. A bit of a bootstrapping mechanism, a little bit like Ethereum and Bitcoin, the idea is that those block rewards overtime will die out and the transaction fee should be sufficient. Now, obviously Ethereum is moving to proof of stake and stuff, but the idea of proof of work in Bitcoin and Ethone's version are basically that you don't need to get block awards eventually. The network will power itself.

A stakers role, as I said, is obviously to stake and to issue these stablecoins and then distribute them however they distribute them and then they get paid for doing that. Now they obviously have the risk, as I sort of mentioned before, and the risk to them is that the debt that they hold isn't just the SUSD that they were issued. The debt they hold is actually variable. They hold a percentage of the debt pool. And the debt pool is denominated in all the different synthetic assets that we mentioned. For a staker, there is actually some risk there because if the price of the all the assets in cyber debt pool go up, then the debt pool goes up. If you ever want to reclaim your SNX that you have locked away as a staker, you need to pay that debt off. You'll need to acquire more SUSD somehow in order to pay off your debt. Get your SNX back and then you can transfer it away and so on or what have you.

[00:12:51] JMeyerson: So if I'm a participant in Synthetix, I need to stake. I need to put up collateral. And that's going to be either SNX tokens or Ether, right?

[00:13:04] JMoses: As I said before, there's two ways to participate. Like if you want to be part of the protocol in terms of giving people access to synthetic assets, then yes, you would stake. If you want to just use the protocol in order to acquire synthetic assets, or even to not even acquire them but just to use them for their liquidity, which I can get into if you want, then you don't need to stake at all. You just need to acquire synthetic. If you acquire as USD via 1inch or Uniswap, you don't have any debt risk. There's very much two different classes of users. But yes, if you are the one who wants to stake and sort of get these rewards periodically, then yes, you need to stake SNX, or we actually have other ways as you mentioned. You can take out a loan using Ether as well. I think we have support for renBTC as well, which is one of the BTC assets on Ethereum.

[00:14:02] JMeyerson: Again, if I want to mint a new SNX token, how does that SNX token get minted? From a system perspective, why and how do SNX tokens get minted?

[00:14:18] JMoses: You as such wouldn't really effectively mint that token. The protocol has is programmed with one of our contracts called supply schedule on a predetermined schedule to emit an amount of SNX every week or mint more SNX. Now, one of the problems when you're engineering them on blockchain is how do you do scheduled tasks? And the way you typically do this is you actually incentivize people, third-party people. You just refer to them as keepers to call the function for you. Every week at a certain time the SNX supply can be minted by any user effectively. They can call function the contract and they can pay the gas cost in Ether and deterministic amount of SNX gets issued and it gets distributed to another contract that we call our reward escrow contract. And then eventually every week when people then claim their rewards they get a piece of that SNX taken to them. I guess you could say one user every week does mint more SNX supply, but anyone could do that. It's really just a scheduled task if you will that's programmed into the contracts, or at least it's a scheduled task that can be performed every week once after a certain time threshold has elapsed.

[00:15:34] JMeyerson: Are there a pre-approved list of synthetic asset types or can anybody mint any time in any type of synthetic asset?

[00:15:43] JMoses: No. Because the risk to the debt pool is so high, the debt pool is shared. There's nothing theoretically stopping everyone who has any synthetic asset from moving into, say, SE, synthetic ETH, and sETH could obviously go up and then everyone's debt would go up. It'd be too risky to allow any type of asset to be minted. Synthetic assets are chosen through a number of methodologies. Effectively what happens is people in the community will propose a new synthetic asset. One of them recently being proposed is Curve. I think I saw another one maybe for the DeFi pulse syntax. I think I've seen some other ones for more traditional equities like the SPY, the SNP index. People said thanks socks. But then what has to happen is there needs to be some research from sort of a third party to say, "Is there enough liquidity in this particular asset outside of Synthetix?"

For example, let's say if we bring in an asset that isn't very liquid. So maybe there's only a million dollars of volume a day and we say, "Well, we're going to have a synthetic version of it." Well, then somebody could manipulate the actual market, the real-world market, with a million dollars or more a day and then basically capitalize on it synthetically. Because the synthetic market, when somebody trades on Synthetix, it doesn't affect the external spot market.

For example, what happened at one point Synthetix supported Maker, the MKR token, the governance token for MakerDAO, and what happened is somebody ended up having – Somebody had some I don't know however many Maker token, a lot of them, thousands of them. And this person was actually manipulating the price of Maker because I think the liquidity was only – The volume, it was only about five million a day, and they were able to sell these Maker tokens. And right before that they were able to go into our inverse synthetic Maker, which basically moves in the opposite direction, and they were able to manipulate the price of Maker because it was quite easy for one person to do that and make the profit on Synthetix.

Long story short, it's important for the protocol to do due diligence on any synthetic asset before it gets listed or gets – Basically, before it gets sort of permissioned into Synthetix, and once it's permissioned, anyone can move any synth into that new synthetic asset. So it's a process whereby people in the community submit what we call SIPs, a synthetic improvement proposal. The decentralized council, the Spartan Council, the seven members who are elected every three months will basically vote on that whether or not it's worthwhile, and we'll have a study will go off a by third-party to figure out is there enough liquidity for it. And assuming all that flies, then the synth will actually be implemented and added into the system.

Right now we have a SIP. I believe it's 113, SIP113, for adding KRW, Korean won, and I believe that's been approved. Or when I talked to the council last I think they were interested in going ahead with it and we're pretty confident that Korean won has enough external liquidity that no one person could manipulate the Korean won market. Not easily enough anyway.

[00:19:07] JMeyerson: Why did you choose to build Synthetix on the Ethereum blockchain instead of a side chain or a new chain? Like I want to start to get into the implementation details and the smart contracts, but let's just give an overview of the basic implementation choices.

[00:19:27] JMoses: Yeah. I mean, I think the protocol came from, it evolved out of a thing called the Haven Protocol, H-A-V-V-E-N. And as I said before, it was designed as a sort of decentralized stablecoin. And at the time in 2017 Ethereum was very much the “bees-knees” of smart contracts. Software, smart contract infrastructure, and we’re big believers in it. And the protocol ICO’s in early 2018 on Ethereum, which honestly at the time it was the most evolved platform, had the most support and it was the easiest at least get going on and it felt like the most robust.

I know the time late into 20 – As we got into 2018 and obviously we went into slowly starting going to the crypto winter, lots of other ETH killers, blockchain 3.0 as they call them like EOS and stuff, it became very prevalent. IND indeed actually ended up spending about a month personally looking into porting some of our contracts over onto EOS, and EOS uses C++ for their smart contract language. And the thing that really struck me, and this is anecdotal of course, but the thing that really, really struck me was how little, how little support there was in the developer community over there. It really felt like there was a lot of money. There was a lot of big block producers trying to push this whole new EOS ecosystem and everyone's like, “Oh! The technology and the throughput and TPS, transactions per second.” But it was really just crickets out there. When I was writing open source software even, I was basically taking some of the stuff that I would consider table stakes, which is I want smart contracts to be tested and I want to be and I want them tested in CI, CircleCI, and to write a bunch of stuff in Docker and connect all the dots and got it all working and, again, still crickets.

There's one other engineer in Brazil that I was actually talking to out of the thousands on this Telegram channel, and this was just such a difference to Ethereum. I come from the JavaScript node world, particularly the node world, and it felt to me like when I first got into Ethereum there was definitely a lot of synergies with JavaScript and Solidity. Now, obviously they're different in many ways, but there just felt like there was some overlap there and I felt like lots of great people and great ideas were coming up and being sort of blossoming in the Ethereum ecosystem. And on EOS I was just like, “This is a wasteland.” There it wasn't the support for it. It just felt like lip service to what a Blockchain is. It was like, “Okay, everything–” They're hitting all the right notes in terms of, “Okay. This is what the blockchain technology is and this is what it does,” but they just didn't have that groundswell of developers.

And I think very much that this anecdote illustrates that Ethereum has an incredible ecosystem of smart contract engineers. Now, Solidity, is obviously quite mature. It seems like it's kind of really eclipsed even Viper, which is like another language that's a Python-like language that compiles into the EVM code. But I think that the value of something like Solidity, it's a procedural. It feels somewhat like JavaScript, C-based language. It feels like that plus all the infrastructure now that’s come up has really given an incredible lead.

For us, while we do look around here and there for what else is happening, it would take a lot to I guess convince the core contributors of Synthetix to like, “Hey, we need to support another chain or another protocol,” because perhaps we got a bit lucky that the no ETH killer came along and dethroned Ethereum, but now we're at a point where even though gas is incredibly, incredibly expensive and very prohibitive for a lot of people, we still have an incredible amount of innovation in the space, which is hats off I think to the powers that they and particularly like in the EF who I think really do – As hard as it is to do what the EF are doing, the Ethereum Foundation, I think they really lead first in terms of not prescribing, allowing different groups to come up and implement sometimes even competing solutions to the benefit of the Ethereum community.

[00:23:42] JMeyerson: Walk me through the architecture of Synthetix. What are the contract types and what do they do?

[00:23:51] JMoses: Okay. Well, it's probably one of the most complex smart contract suites on Ethereum. I might get through all of it. But effectively what we have is we have an ERC20, which is a standard Ethereum token for our Synthetix token, SNX. And then we have an ERC20 token for all of the synthetic assets, synthetic Ether, synthetic Bbitcoin, synthetic USD, and these are all on their own tokens, right? And so they live inside your wallets and you can transfer them as expected and they conform to the ERC20 interface, which is balance of, approve, transfer, etc.., etc. Then we have proxies wrapping around on various contracts like Synthetix and all the synths so that we're able to sort of change the brains of them as it were. We kind of have somewhat of the model view controller idea. The controller is kind of the guts of our contracts, the brains of our contracts. The view is kind of the proxy. The models the state contracts, and we sort of split them out that way. So we sort of have three contracts for a lot of the main things. You'll see that for all of our tokens. We'll have a proxy – Sorry. For all of our synths, we'll have like a proxy. We'll have the synth, which is like the brains, and then we'll have the token state contract. Because, as I'm sure, a number of your viewers know that like a contract has logic, but it also has state, right? And when you need to upgrade logic, the question is do you migrate the state across or do you sort of reach out to another sort of singleton contract to handle the state for you?

We have all these tokens. We have all these synths in Synthetix. And then we have a number of sort of contracts to help us do things. One of the difficulties in programming on Ethereum is you're limited in certain ways. You're limited obviously by gas usage, which can be somewhat frustrating. Sometimes you realize you shouldn't be doing loops. You shouldn't be calling externally. You should limit how much storage you use. That can be quite difficult to work around. But another difficulty is actually the size of the contracts. They can only be a certain size. So you hit this 24 kilobyte limit of the size of contracts, which is I believe EIP173. A few years ago it was required that the contracts be a certain maximum size. And for us in Synthetix, a lot of our stuff is intertwined, because as I mentioned before we have this idea of a shared debt pool, and the debt pool is basically the sum of the the USD value of all of the synthetic assets out in the wild. And so to figure that out you have to basically iterate over all these different synthetic assets and say, "What's the supply of these and what's the current market rate of them?” And add them all together, which is, A, very gas intensive because of looping. Any loop, as you can imagine, is going to require more gas. But also it requires reaching out to all these different contracts to kind of figure out, “Hey –” You asked this contract, “Hey, what supplies you?” You ask another contract, “Hey, what supply you?” Every time you ask, it's called an external call. That also costs gas. We often find ourselves architecting and trying to find ways to reduce gas, but also except that we have this 24 kilobyte limit per contract.

In Synthetix, when you try to do something like mint, which our old dapp was called Minter. Our new dapp is called staking. We use the word mint in issue and stake kind of all fairly interchangeably. Whenever you do that action of minting or staking or issuing, you're effectively calling a function inside the Synthetix contract, which then call out to another contract, which is called the issue a contract, to do all the heavy lifting, because there's just too much code to fit inside the Synthetix contract. And then if you want to claim every week your rewards as an SNX staker, you'll reach into the fee pool contract, the basic claim rewards for the free pool. And if you want to exchange one synth for another, you'll reach through Synthetix into the exchanger contract that allows you to exchange your synth for any other synth in the system.

And there’s kind of the – The other kind of the crux contracts. There're other ones like a supply schedule, which is how often Synthetix can be minted, as we talked about before. There's an escrow contract which holds their rewards that SNX stakers get every week Holds them in escrow for 12 months. Yeah, there's a contract for liquidations. If somebody has issued too much debt for the amount of SNX they have, they can be liquidated by someone. And there's a contract that handles all of the rates, the exchange rates. Basically it reaches out to all the chain link aggregated feeds that are used in Synthetix to figure out what are the rates of each synthetic that is supported in the protocol. There are quite a lot of contracts and there's still a bunch more that I haven't really talked about, but they're the crux.

[00:28:53] JMeyerson: Most of the shows I do are about I guess normal software engineering or web apps and e-commerce stuff. This is very different. A lot of the companies I cover do – They have their architecture in Kubernetes and deployed to cloud infrastructure, and the tooling is pretty good. Is the tooling in the Ethereum space just like super primitive and hard to work with?

[00:29:22] JMoses: Relative to where I'm sure a number of you listeners are coming from. Yes, I would say it's very primitive. But it has definitely had a real shot in the arm in the last year or so. There're been a couple of teams, one in particular, the group behind Hardhat that have really taken – They've really leaned into better developer tooling. It's their passion. And the Ethereum Foundation have been giving them a lot of grants. There's a library called Ethers that has really been pushing the boundaries – Pushing the envelope I should say. But when I was getting my head around it in 2017 and then deeper into 2018, I was pretty shocked at how bad things were. Again, coming from Node and just being like expecting all this tooling and just being flabbergasted in how little there is there. Like even to the point where my stuff, some of the stuff that I've written is still very prominent and used, because there's just not a lot of people writing develop a tool in the space.

I mean, there're different parts of it. I think for a listener who doesn't really understand about blockchain engineering, I think the best analogy I have I think about is that effectively your smart contracts are basically your application to you and they're also your database because they have storage, right? And the blockchain is effectively your backend. So you effectively write this code, deploy it, push it out to Ethereum, and then thankfully you never really have to deal with availability or uptime, because your backend is now the blockchain. And so everyone around the world should have access to the blockchain. Obviously, one of the values is no one country can easily censor it. Anyone can get access to this contract.

And the other cool thing about most blockchains, Ethereum for example, anyone can read that data whatever they want. It's free for anyone to read anything at any point. All sorts of cool tools exist out there for just reading data of all this state. Now, you couldn't try and encrypt it if you want, but it's very difficult actually to encrypt anything. Truly, there are ways to do it, but it's very difficult. And it creates this very different system where people I think are used to like, "Okay, I'll write my infrastructure," as you said, Kubernetes, whatever, and deploy all my infrastructure and my database up here, and blah-blah-blah. But the blockchain effectively will do all that for you, but the costs are that obviously it can be slow to deploy. It can be difficult. It can cost you a lot of money to deploy in terms of gas. And then you have to deal with these limitations that I mentioned. Like you have to fit your code into contracts. Your users have to pay every time they want to do anything like everything they want to transact. It's a very different paradigm I think. But the thing that I've loved and the things I love the most about is I'm very much a tinkerer. I've enjoyed that there hasn't been a ton of tooling because I’ve liked writing some of it. It's been fun to write some of the tools. And it feels very much like version territory. It feels a bit like when a lot of Ruby people went over to Node in the early days and they were just writing – because Ruby was – The gem world was so rich with all these gems in Rails, gems, and this, that. And then with Node, everyone started writing all this you know cool stuff. And then it came with React. I mean, we had a backbone for a little bit, but then obviously with React. Maybe even a bit with Angular. There was all this this innovation that happened, and I feel like that's that's happened somewhat to Ethereum, but in some ways it hasn't because it hasn't just been JavaScript people. We've had people from all sorts of different backgrounds come in. Some people moved in Python and what have you, and it hasn't ever chosen a developer tooling language. I think TypeScript's becoming a little bit of the de facto choice. But still fairly nascent I think in terms of developer tooling in this space.

[00:33:07] JMeyerson: So how are people actually using Ssynthetix? How is it getting used in practice?

[00:33:15] JMoses: In practice, I think lots of people are using it to get access to something that it isn't on Ethereum. Lots of people are using it to get access to synthetic Bitcoin for example. If you have access to – The three kind of main contenders of holding Bitcoin on Ethereum would be Rap Bitcoin, Red BTC and Synthetic Bitcoin. And so I think a lot of people are using it as a way to hold their Bitcoin. They can just get synthetic Bitcoin, rather than having to hold these other ones. Now Rap Bitcoin is obviously centralized if you’re aware. There is actually a team that are actually custodying the Bitcoin. And if you are a decentralization maximalist you would not want to carry something like that. Whereas RenBTC is it’s much further decentralized, but then you could argue synthetic Bitcoin is even more decentralized. That's definitely one big use case.

And there are a lot of things that – We have a synthetic DeFi index. We basically are an index of all the different DeFi protocols, and people have found that to be a very useful instrument to hold because it gives them exposure to a number of different DeFi protocols. We've had inverse synths that I kind of touched on before, which are synths that are basically like shorts, kind of like shorts, but not truly. They've been somewhat problematic for us because they require a lot of manual intervention from the core contributors to maintain them, because the way they work is they require a moment in time when they're created. This is sort of way that these sort of financial instruments work and that the math that works on them, it only works when the the price is within some sort of bounds when it was created.

But people do like holding those inverse synths because they're an easy way to get kind of inverse price movement. You hold an inverse, the iETH synth, and as the price of Ether goes down, price of your iETH goes up for example. This has been sort of one of the big main use cases, but another one that's sort of come up not too long ago that's been worked on by the team at Curve has been cross asset swaps. We talk about Synthetix as being this big debt pool where you can move any one synth to any other synth. And so what sort of happened or the insight that happened for Curve and from us was that people could step through synthetics as a step through to go from one asset to another.

One of the big problems on Ethereum right now is the amount of slippage you get if you try to move from Ether to a stablecoin. If you hold Ether and you want to move – Like I'm talking significant amounts. Like so we're probably like in the millions of dollars' worth. But once you start trying to move like say a million dollars or more of Ether into say USDC or USDT, you'll see a lot of slippage just because the various AMMs, the auto market makers like Uniswap and what have you, only have so much liquidity. And once you start draining too much liquidity in one trade, you'll get a lot of slippage.

And so what Curve has done has created a way to go from something like Ether or Rap Bitcoin to a stablecoin, and one of the steps is they hop through synthetics. Let's say you have Rap Bitcoin. Let's say Ether just for the sake of argument. Let's say you held Ether and you wanted to maybe a thousand Ether let's say for instance. Let's say you had a thousand Ether and you wanted to move that through to stablecoin. What curve has done is allowed you to do – You can take that 1000 Ether. You can swap it in Curve for a thousand synthetic Ether that are ready to go there for you, and then in the contract it will take – It'll reach out and it'll reach to Synthetix and say, "Hey, I want to convert this thousand synthetic Ether to synthetic US dollars." It does that with the 30 bit fee, right? And then the last step is it needs to convert that synthetic USD to the USDC stablecoin, which you can do after a few minutes of what we call our waiting period just so the price is – Just to make sure there's no price volatility. So that's a really interesting use case of using synthetics as a protocol to achieve an aim that actually wasn't really necessarily what Synthetix was created for initially, but it's a nifty use of the protocol to get that zero slippage as sort of a stepping stone between different assets.

I mean, there are actually another – I think another – There's an interesting other protocol that's come out of Australia called dHEDGE, and their kind of idea is that it's like this decentralized hedge fund managers. And they use things like synthetic assets to very quickly and easily compile a portfolio. And you can basically go in there and say, "I want to follow somebody's portfolio. I want to elect that person. I want to buy into what their portfolio is and let this person be like a decentralized asset manager." That's another really interesting thing that's popped up that uses the synthetic protocol that could be – That a lot of people found very, very useful.

[00:38:17] JMeyerson: What is the current state of the company built around Synthetix? How does the company get rewarded and what are you working on?

[00:38:32] JMoses: There actually is no company as such anymore. It's just a DAO. So there was a company at one point when there was money raised, but that has been dissolved into various DAOs, decentralized autonomous organizations. And these DAOs are effectively contracts that live on Ethereum that hold capital in the form of Ether or stablecoins. The company itself, there is no company. And so the people that that do get paid by the protocol effectively would just submit expenses to the protocol that would then pay for – Send them crypto for their services over time, which can present challenges. It means that if anyone ever wants to get paid by a protocol, they need to be – They're an independent contractor, and they have to do stuff that way for example.

It's definitely not that easy to be fully decentralized in that way, but that's something that the powers that be and the people that – People are involved in the protocol really care about is being decentralized first. And your second question is the future? What is the future of the company or the protocol?

[00:39:40] JMeyerson: I didn't ask about the future quite yet, but we should go there. Let's start to talk about the future. I guess first, yeah, the future of Synthetix and then the future of DeFi in general.

[00:39:53] JMoses: I mean, the future of Synthetix, a big part of it is we're already on a layer two. Optimistic Ethereum are the team that used to be plasma team, and they're very close to the EF and they’re working very heavily on this idea of optimistic roll-ups, and it's a way of basically providing faster and cheaper transactions in Ethereum. And so we've been working with them to get our stuff onto this layer too, and it's different from a side chain, and that effectively it is like this – Sorry. It's kind of like this miniaturized version. Basically it takes a lot of the heavy lifting of Ethereum. It does it on its own layer and then just rolls up the summaries of the transactions later on. It uses the main Ethereum chain to secure its value. And so for us we've already started putting some of our infrastructure on there. We actually support that right now where you can take your synthetics on Ethereum and you can actually deposit it onto this layer 2 and actually start issuing SUSD on layer 2.

And so the future for us is very much about supporting more functionality on layer 2. You'll be able to actually exchange synths on layer two and effectively do it a lot cheaper. And that unfortunately does create a lot more complexity. I don't know if any of your listeners can imagine, but from our perspective now we have this different dimension of contracts that live not just on layer one, but now we have a different dimension of layer two and we're trying to constantly – We're maintaining the same code base, but that requires a little bit of polymorphic font. So we have to write different kind of – Implement slightly different implementations for layer two just based on some certain requirements or things that the layer 2 engine can do based on the fact that it's quite a sophisticated piece of technology. But for us we very much want to keep delivering and moving on to layer 2 because it's cheaper and we see it a lot more interesting composability benefits.

And we didn't talk about composability at all, but I think that one thing that's very exciting in the space of DeFi for engineers is this idea of composability, that anyone can write a contract, and that contract is a first-class citizen Ethereum just like it can hold tokens. It can interact with anyone else. And there's a lot of power in that, because then you can write a contract that just does a number of things for you. You can write it to do a number of your tasks, but you can also deterministically say, “This contract is going to do this if someone gives it money.” People like to call it programmable money and stuff like that, but it is a very, very powerful piece of technology because it allows these network effects. As more developers come in and they start to understand it, they can sort of build these building blocks, “Oh, I can take this thing from Curve and then I can move it through Synthetix, and then I can swap it through Uniswap, and then I can do this, and then I can do that."

For us, as layer 2 becomes more and more prevalent, we expect to see a lot of interesting composability that happen on layer 2 that won't happen on layer 1 right now because layer 1 is just too expensive in gas to do anything for most normal innovative projects. And it's going to cost you somewhere between – In current day prices between $200 to $500 per transaction to do stuff. You're just not going to get innovation. For us we're really excited to keep working with Optimism and see where they're going and get them to their main net, which is coming out pretty soon. And the other big thing for us is it's basically futures.

What we have right now is effectively these synths that we think of a spot synths that they basically just track the price of the spot market. But the idea of futures is that we'd actually have a balanced portfolio. If any of you ever traded futures, you sort of get this idea that you have longs and shorts and that you have this idea of these perpetual features that are constantly being managed every eight hours and you basically look back and say, "These people have these positions open. Who should be liquidated? Who shouldn't? Who pays interest to who?" It's a very sophisticated way of handling derivatives and it's something that we see as a huge part of our future. But it's very difficult to do stuff like this on a blockchain because of that problem I mentioned before. How do you do scheduled tasks? How do you get stuff to happen regularly? You have to pay the gas to do it. You have to write a bot that constantly does the thing for you. And if that's mission critical to you, that can be a real problem, because what happens if the bot breaks down? Runs out of ETH, whatever? It's its own realm of problems.

For us, getting deeper onto L2, supporting futures. And the other main thing is us basically doing a massive rewrite of our architecture. We're calling it V3. And for us a big part of that theme besides a lot of functionality is trying to be better Ethereum citizens. I think we've been a little bit – We should have done things our own way a bunch, and I think that's hurt us in terms of integrating with other protocols, and I think it's something that that we're very keen to remediate.

For example, a lot of protocols like Compound, Aave, Uniswap, when you do a thing, like when you give away a token like in Aave. If you were to say, "I want to deposit some SUSD. They will give you a wrapped token called an ASUSD token back, which is like your claim on it. They don't have to do that. And in Ethereum they could just leave it in their contract. They don't have to create a token. But by tokenizing it, gives the user something tangible. And this unfortunately isn't something that we've been doing so far, but something we'd like to do. For us like just being better Ethereum citizens I think. Conforming more to the standards that have sort of evolved around us are pretty critical. That's the sort of the third prong of this is that we very much want to – We really want to rewrite a lot of the architecture. Clean up a lot of the legacy code that we have, which unfortunately we have quite a lot due to the nature of working on a blockchain and state being stored in contracts. And part of this we're going to actually get support with all the other protocols on Ethereum.

[00:45:53] JMeyerson: Is Ethereum scalability at all a problem for the day-to-day functionality of Synthetix?

[00:46:01] JMoses: It's an interesting question. I mean, it's a problem in that it's prohibitive to many people to use. The cost of gas and the way that we've architected the code, it's at a point where it costs – It can cost in somewhere between 50 to 100 something dollars for someone to issue in terms of Ether, the cost of Ether right now, to issue SUSD or to claim every week. And once you have a certain amount of SNX or once your rewards are high enough, that is probably okay for you. But if you're not collecting enough rewards, you're not going to do that. It is a problem and that we find that basically we've got to this – Sadly, this kind of elitism where we call them the whales, right? The whales are able to do whatever they want. They can pay the cost of a hundred, two hundred dollars, but the rest of us are stuck sort of saying, "Well, we can't do this thing. We can't afford it."

Scalability is a massive problem, and things like optimistic roll-ups that we work with Optimism are going to help a lot. And there are going to be other layer two solutions, like ZK-Rollups or another one that are just going to help it this way. Because we're probably going to get to the point where a lot of the real interesting innovation will move off the main chain, because effectively people just can't afford to be paying this, and people kind of – The old sort of trope of the artists move in when a place is affordable, and they're the ones who have the ability to innovate and experiment. And we need to create that place again. And unfortunately the main chain Ethereum isn't that. Maybe the prices go down once a lot of the projects move on to layer two, but I don't think so. I think it's just too much – like USDT, like Tether, for example, just puts way too much transaction load through the main chain of Ethereum and I don't see that moving to layer two anytime soon.

[00:48:00] JMeyerson: Well, as we wind down, what closing thoughts would you give to the listeners? What would you want them to take away? What lessons would you want them to learn from the world of synthetic assets and synthetix?

[00:48:13] JMoses: I mean, I think I actually would rather just say from Ethereum rather than necessarily Synthetix and DeFi. I would say that like for a lot of your listeners who aren't in the space, like it's fascinating. It's challenging for sure, as I said before, like writing code that is going to be executed in a decentralized manner and having the backend be your blockchain. And anyone can read data at any moment, like it's such a different paradigm shift. And it took me a while. I guess I pride myself on picking things up pretty quickly, but it took me months to really get my head around how this stuff works. And even now I’m obviously still always learning more. But I really think it behooves people in the space. If you get your architecture, you get infrastructure, you get databases. Try this out for size. Like it's pretty wild, but like it's pretty incredible. Like this idea that anyone can get access and program money, deterministically program money, and anyone in the world can have access to it. It just starts to break down the barriers that you sort of – Start to realize like, “Wow! We've just lived with for so long.” This idea that money is controlled by governments and that it’s all set up and everything we have to sort of conform to what structures are out there.

Ever since I got into it and got deeper into it I realized that this stuff is incredible and it's going to go to some pretty crazy places. And I think it behooves everyone to start being more aware of what you can actually do on a blockchain like Ethereum. The innovation, the composability that I mentioned before, it feels like we're at a point of like early Internet. Like it feels like we're kind of in the mid to late 90s, perhaps maybe even closer to .com, but like people are starting to figure out this stuff has got real, real value and we just need to figure out how to make it work. We're not quite there yet. I mean, we're not there with the Internet either, but I feel like this stuff is a big deal and it really behooves you to spend some time and get involved in it.

[00:50:12] JMeyerson: Cool. Justin, thank you for spending some time with us and educating us about synthetics.

[00:50:17] JMoses: Yeah, not a problem. Awesome. Thanks for having me, Jeff.

[END]