EPISODE 535

[INTRODUCTION]

[0:00:00.3] JM: Humans organize into groups. There are lots of group types; religions, corporations, national governments, state governments, citizenries, clubs, musical bands, every group has governance. Governance defines the rules and the ways that rules change. The United States requires citizens to pay taxes. A corporation requires you to show up to work, but the corporation has to pay you a salary.

Most groups today are managed by people. If you break a law, you have to go to court and sit in front of a judge and a jury. Those are the people who decide how you will get punished. If you work at a corporation and you have a problem with your manager, you go to HR and HR arbitrates that problems.

These organizations that we're discussing are centralized. There's a governing body who sets the rules. If there's an ambiguity, the person who happens to be empowered at any given time gets to decide how that ambiguity is resolved.

Power is centralized in that governing body. These organizations are run by people, the governance of the organization is enforced only to the extent that the human governance carries out its duties.

A decentralized autonomous organization is a group that can run without either centralized or human intervention. It is decentralized and autonomous. It is a DAO, a Decentralized Autonomous Organization.

Aragon is a platform for running and managing Decentralized Autonomous Organizations. Luis Cuende is the founder of Aragon and he joins the show to explain what a DAO is and why people would want to create them. What kinds of organizations can you, or can you not create on a blockchain. Can you create any type of organization? Or are there restrictions? We also talk about the engineering of Aragon and the structure of its ICO, which raised 25 million dollars via token sale. It's a great episode. I think you're going to like it.

I want to mention that meetups for Software Engineering Daily are being planned. You can go to softwareengineeringdaily.com/meetup if you want to register for an upcoming meetup. In March, I'll be visiting Datadog in New York and HubSpot in Boston. In April, I will be at TeleSign in LA. I would love to see you at one of these meetups.

With that, let's get on with this episode.

[SPONSOR MESSAGE]

[0:02:29.5] JM: Users have come to expect real-time. They crave alerts that their payment is received. They crave little cars zooming around on the map. They crave locking their doors at home when they're not at home. There is no need to reinvent the wheel when it comes to making your app real-time.

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

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

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

[INTERVIEW]

[0:04:12.8] JM: Luis Cuende is a co-founder of Aragon. Luis, welcome to Software Engineering Daily.

[0:04:17.2] LC: Yeah, thank you for having me.

[0:04:18.9] JM: Your company Aragon is focused on changing the way that organizations can form. Around the world, we have a lot of different kinds of organizations. We've got corporations, we've got churches, we've got governments. What causes organizations to form?

[0:04:37.2] LC: Well, it's interesting because really the first person to think deeply about that was Nobel Prize Ronald Coase. That was many decades ago. Basically, he argues that firms or organizations form due to transaction cost, like it's easier to operate with a group of people that you know and that you trust, instead of a random group of people, or not even a group of people, but just random freelancers or contractors that they may do, like using the internet.

Now while that's also a very powerful way to organize, like just having projects that are for example open source and jumping into those without knowing each other and just knowing each other from the internet, there is also value having like organizations or firms. It's also true that also an open source project can be considered an organization, because it's just like a band of people organizing around the same vision or goal.

It is really that. It is really just a bunch of people organizing towards a common goal. Therefore, optimizing everything they have to achieve that goal.

[0:05:39.2] JM: Some organizations have formal incentives. A job pays you a salary, that is a formal incentive for a worker to work a job at a company. In other organizations, the incentives can be less formal, like in my family I know that there are incentives in place. I know that I'm going to be rewarded if I fulfill the tasks that I am incentivized to fulfill, but it's a little bit less clear what those incentives are. Maybe it's being a good brother, or being a good son, whatever those things mean. In return, maybe that behavior is reciprocated to me.

There are two examples there; example of somebody working for a salary and the example of somebody being a member of a family where in one case you have very clearly defined incentives. In another case you have less clearly defined incentives. In both cases, the organization seems to function relatively well. Is it important for an organization to clearly define its incentive structure?

[0:06:43.2] LC: I believe so, because if you made them transparent, if you define them for example like the smart contract in which you know how everything works, you know all the actions you can perform and the results, or the outcomes they have in terms of rewards for example. That really incentivizes people to participate if they actually know what they are getting in return for their work.

I mean, there are scenarios for example again with open source software, incentives are not that clear historically and people just contribute a lot times, because of altruism and they don't really expect anything was changed.

What's true is that if we actually for example incentivize those contributors to actually be rewarded, then I'm sure the number of contributions would actually go up pretty quickly. What I would say is yeah, there are a lot of systems that don't really have clear incentives at work, or that they have very obscure, even – their incentives is like a black box and you don't really know how they work, then I think that the systems are not transparent in terms of incentives will totally outperform them.

[0:07:46.8] JM: I tend to agree with you. If it weren't so taboo, I would probably try to setup a system of smart contracts among my family to better regulate proper behavior.

[0:07:59.8] LC: Well, I mean you could technically do that. I don't know how that will play out. If you do so, please do other comment or something.

[0:08:06.5] JM: Can I use Aragon for that?

[0:08:08.3] LC: Well, you could, even for your – I don't know, for your marriage or whatever. Yeah.

[0:08:12.1] JM: Okay. Let's use that as a use case to just get us into what Aragon is. Aragon is for setting up Decentralized Autonomous Organizations. That term has been overloaded in different places. I guess, before we jump into an example of using an Aragon set of contracts to create a family, let's define that term. What is Decentralized Autonomous Organization mean to you?

[0:08:39.8] LC: Decentralized Autonomous Organization or just DAO is a set of smart contracts, or a set of incentives that are built on chain that means on a blockchain and that incentivize different rewards or ways of corporation. A DAO could be something so simple such as Bitcoin. Basically, Bitcoin organizes people to mine Bitcoin and to contribute computing power to secure the network. It organizes people to do that in a pretty seamless and transparent way. They're autonomous, like there is no sensor party that can take down the whole system. That's a very basic example of a DAO.

It can be even things that are more complex. It can be for example like the DAO which was the first DAO project in the wild and there basically was like a VC fund, so to say. People put money there and then you fund [inaudible 0:09:29.4], basically by letting people vote what to fund. It can be really a lot of different things. I think that can only call definition use an entity that can run by itself, and obviously with human intervention that it tends to organize humans in a way that it achieves its own goal.

[0:09:49.8] JM: We'll scratch the family example for now, and maybe we could return to it a little bit later. Let's talk more about DAO. What are the problems that could be solved if we had more wide widespread use of decentralized organizations?

[0:10:04.0] LC: I think one of the miss accountability and it goes from the small firms, like for example number of hits and knowing what the funds are spent for, to huge things, for example like government spending, like right now didn't really know where your taxes go and if you know, they're probably paying like the next what like the private market will offer for the services the government gives you and stuff like that.

It really helps with accountability and it also helps with decision-making. There are a lot of use cases where you need to coordinate a huge number of people, like maybe on soft versions around common goods, for example think about Twitter or Facebook, they are run by centralized organizations, centralized companies. The issue with that is that for example Facebook was able to help. Maybe they didn't mean to, but to help Russians take over the US elections and all of that.

If Facebook was a decentralized autonomous organization, you will have things such as decentralized filter and system to know what's fake news or not and people in the world will actually vote. Whereas, right now Facebook is probably one of the biggest nations in the world and it is controlled by one guy and like a management board and that's it. The data common goods at social networks, or open source projects and all of that can really benefit from involving millions of people in their decision-making process.

You can have a DAO and let people vote on how Twitter should work, or Facebook should work, or even nation states. I believe DAOs are very good for that, just aligning millions of people to have the same incentives and also to vote on very important topics, instead of having like a dictatorship model.

[0:11:48.9] JM: Okay. Now that we have a basic idea of what a decentralized organization is and how it can help people, what's Aragon? Why did you start Aragon? What can people do with Aragon?

[0:12:02.8] LC: Aragon is the easiest way to organize people around decentralized autonomous organizations. You can create your own DAO and you can invite people. You can give them tokens. You can also do voting and enable them to vote on very important decisions, so just for example where do the funds of the DAO go and how they are spent?

The cool thing is that the operating system that goes – that powers Aragon – I say operating system, because it's a very complex and modular approach, is a framework called Aragon OS, in which basically you can define any permission for any app for a smart contract and let other apps consume it. For example, you can have like a vault app, which has all your assets and tokens. Then give permission to another finance app to withdraw app to 50 tokens per day or stuff like that.

You can really create very complex and modular governance mechanisms, which is the first time in history you can do that, because previously you could only sort of try out democracy or I don't know things, like dictatorship with people. Now you can try out this new governance mechanisms on chain and with fully online.

[0:13:12.0] JM: When I think about the different decentralized models that have had success, I think of Linux, I think of Bitcoin and I think of Ethereum. Linux and Ethereum both in a sense have had a centralized authority figure, even though the development takes place on a centralized basis, there is a centralized authority figure. Bitcoin does not have a centralized authority figure and you see positives and negatives for those approaches.

Certainly, with the centralized authority figure, there is used to be a tendency towards faster development. I mean, obviously we have a very limited set of sample cases, so I don't mean to extrapolate this to everything. With Bitcoin, we have perhaps more aggressive adoption, because I think people have maybe more trust in Bitcoin, because it almost seems like this total – it is this headless organization. It is truly decentralized.

Whereas, Ethereum or with Linux, you could argue is somewhat centralized. I mean, Linux at this point probably not. In any case, I'm just trying to illustrate that there are different ways to run a decentralized autonomous organization. In some cases, you want the head, and in some cases you don't want a head. Does Aragon support a variety of different leadership models?

[0:14:35.2] LC: Yeah, exactly. You can do that with Aragon. You can go from a dictatorship thing to a multi-sig, which means having a board, like multiple people that can decide. Or just voting and have all your token holders decide on things. Yeah, it's very flexible. I mean, I believe that dictatorship, it's a very good model for a lot of projects and we are trying to combine the good parts of having governance and not having it.

If you think about Bitcoin for example, Bitcoin works well because it has no governance, so they move pretty slow and that is downside. Yeah. You said, like people really trust it because it's a headless organization. Whereas, Ethereum perfectly are a sort of have and whereas, [inaudible 0:15:19.3] is decentralized.

You have Vitalic who everyone really trusts and likes. I believe you can really combine the trust of having to hold this organization with also taking fast decisions with having a very clearly defined governance mechanism. Yeah, we definitely support different governance models. You can really build whatever you want with Aragon.

[SPONSOR MESSAGE]

[0:15:50.3] JM: LiveRamp is one of the fastest-growing companies in data connectivity in the Bay area. They're looking for senior level talent to join their team. LiveRamp helps the world's largest brands activate their data to improve customer interactions on any channel or device.

The infrastructure is at a tremendous scale. A 500 billion node identity graph generated from over a 1,000 data sources, running a 85 petabyte Hadoop cluster and application servers that process over 20 billion HTTP requests per day. The LiveRamp team thrives on mind-bending technical challenges. LiveRamp members value entrepreneurship, humility and constant personal growth.

If this sounds like a fit for you, check out softwareengineeringdaily.com/liveramp. That's softwareengineeringdaily.com/liveramp. Thanks to LiveRamp for being a sponsor of Software Engineering Daily.

[INTERVIEW CONTINUED]

[0:16:57.4] JM: Well, we can return to talking about the high-level decentralized autonomous organization management and creation a little bit later. Since this is Software Engineering Daily, we should talk a little bit about software, because I've done other shows of this flavor where we end up talking a little too much about the decentralized Uber network, or the decentralized Airbnb, or the day when the government falls and the world descends into utopian crypto-anarchy. I want to talk about engineering. What have you build so far with Aragon?

[0:17:33.2] LC: Well, we have built a lot of different stuff, because in the Ethereum ecosystem and in crypto in general, there are a lot of things that are really early state, so you got to build it yourself. When we got started with Aragon, it was like a simple way to manage like companies but on the blockchain. Then we figured out over, we can create a framework for people to create new reach apps that is post permissions to other apps, such following the Unix philosophy having one app that does one thing well and that's another app consume their functionality.

We built our own, like smart contract framework called the Aragon OS. Then we figured out that, like people were signed in Ethereum transactions without knowing what they were signing. You could be signing a pay log that said that, or your ether goes to 0X, 000, which means that you're burdening your ether, or it goes to a hacker, or whatever.

Basically, we figured out we needed a way for users to understand what they were signing, like the pay loads. We created a DSL for a smart contract, like human language translations so we can decode that X input to something human readable. You are sending your tokens to this person by signing that from a section. We also did that. Then we figured out, "Okay, this is beautiful, but what happens, well we need to update Aragon." Because we could update Aragon and basically change how everything works and steal all the ether and all the funds from an organization.

We figured out, we need a way to update it securely and also in a decentralized way that doesn't depend on us. We have created a package manager, which is called APM. Basically uses IPFS and other decentralized technology to push a lot of these data packages and also smart contract to repos and those repos can have any coordinate mechanism, so you can have new versions of the software vetted by a community of like they can hold, there is an incentivized not to provide malware inside the software. We also created that.

Also a UI toolkit, because when people create apps for Aragon, there is a lot of stuff that is very dependent on creating decentralized publications. For example, everything is asynchronous, because you have to wait until blockchain actually validates your transaction. We also codify UI toolkit.

At this point, we are creating an operating system almost for the different pieces, which I mean it's a lot of work, but it just had to be done and crypto is very early. When something is not built, you have to build it yourself.

[0:20:14.1]JM: If I heard you correctly, you said that Aragon started as merely a DSL that sat on top of Ethereum?

[0:20:21.0] LC: Well, no. It just started as a way to create just companies, but on Ethereum like traditional companies on Ethereum. Then we went from that to being like a super, super broad way to create any decentralized organization, or even like a blockchain protocol.

[0:20:37.7] JM: Did it start as a set of tools before it mature into a whole platform and a tokenized set of infrastructure?

[0:20:46.2] LC: We actually started as a very user-friendly app. Then we actually built on the developer tool in. First of all, it was like very user-focused and now it's – I mean, it's both users because users can use it, but also developers because developers can create apps into with Aragon. That's very developer-focused.

[0:21:04.2] JM: Okay. The first version was just an app where you could spin up a company, but it was a company that was managed directly by Ethereum's smart contracts. I mean, you drew up a DSL that defined what you wanted the Ethereum smart contracts to be, but it was in human readable language rather than solidity. Then it was only later on that you turned into a tokenized platform, is that right?

[0:21:33.0] LC: Well basically, the first version which is a way to create like companies and very traditional companies, like you had your cap table and you have your words on voting or whatever. Now it's super extendable, flexible and not for only that niche, but can be extended. I think a good analogy is like Emaxx. It just started being as a text editor, and now it's just – you can do crazy stuff on Emaxx. I think that's a good analogy probably.

[0:21:58.8] JM: Understood. Describe a little bit more the interaction between Aragon and Ethereum as it stands today.

[0:22:07.8] LC: Today, all the Aragon organizations are based from Ethereum smart contracts. Ethereum is a touring complete, smart contract like blockchain so you can create any application on top of it. Basically all the Aragon apps are smart contracts themselves and it is truly 100% built on Ethereum. Which doesn't mean that is not compatible with other blockchains. There are other blockchains that also have the virtual machine that Ethereum has. Right now, it's very, very, very based on Ethereum.

[0:22:38.4] JM: Now in Ethereum, code is law. Once a contract is deployed, it cannot change. If you wanted to program mutability into your organization and you wanted to build that into Ethereum, an Ethereum contract, you would have to make a forwarding contract that would point to other subcontracts, because the contracts themselves are immutable, so if you wanted to change your organization, you would have to change the forwarding contract, you would have to change the address that its forwarding different functionality to.

This seems like an interesting challenge for Aragon, because obviously corporations change over time, so you have to program mutability into the contracting and sub-contracting system. Tell me a little bit about how mutability works in Aragon.

[0:23:35.7] LC: Yeah, exactly. That was another theme that we have to build ourselves, because no one built it before us that was applicability. Basically how it works as you can describe, we have forwarders and so forwarders basically take an action and forward it to any other entity. A good example of how forward is worked, like we have voting app for example. If you want to get an action on Aragon, but you don't have the permissions to do so. For example, you got a random token holder but you want your decentralized organization to send some funds to whatever, to another contract in Ethereum, but you don't have the permissions to do so.

What you do is you forward your intent to perform their action to the voting app. Then if the voting passes the voting, which actually like has permissions to execute the withdrawal of the fund, performs that action for you. That's how a little bit how forward is working in Aragon. Then for applicability, we actually have to implement this proxy, where you basically update the contract that it actually points to.

Basically, you have to contract. You have like a proxy contract and then you have the actual contract that has the code of the app. How that works in Aragon is Aragon has a kernel. This kernel is basically an ACO, an Access Control List, such as the one that you have on Linux or other Unix-face operating systems.

Basically, this ACO for example knows that there is a voting app. The kernel asks the ACO, “Hey, where is this voting app located?” The ACO basically says, “This voting app is here.” Then you query the voting app and you are actually querying the proxy, not the voting app itself and then the proxy basically – from the proxy you go to the voting app itself, which is the one that contains the code.

This is a bit abstract. There is a blog post on our blog on Aragon OS, if you use Aragon OS 3.0, which is the version that implements all these. Basically, you are able to have multiple versions of a smart contract and the kernel keeps track of which version is the one that actually contains the code, because all the other generic proxies are just forwarders. You call them and actually what happens is that they will call the smart contract that actually contains the code and not the proxy, which is again empty cell so to say.

[0:26:00.0] JM: Let’s go back to the high level. I started Software Engineering Daily in 2015, let’s go back to 2015. Let’s pretend I had not started my company yet and I was starting Software Engineering Daily and I’ve got myself and my co-founder. Let’s say I’m going to hire one employee and they’re going to have some four-year vesting schedule and I don’t want to go through the work of setting up an LLC in the United States, because it’s annoying. You got to fill out some paperwork, so I decided to do it through a DAO. If I wanted to setup Software Engineering Daily through Aragon, what would I do?

[0:26:41.0] LC: You will create a token and that will be the union. For engineering token for example, or something like that and then you will give out a vesting grant to your employee. Basically, that vesting grant could be revocable by you in case the employee leaves or whatever. Then the employee would automatically get their tokens vested.

The funny part with all of these is that you don’t have to do anything. Each block, an Ethereum block which happens each 14 seconds the employee will actually get tokens as the stock. Also, you could payroll, which is something very interesting, because we’re actually going to do payroll internally with an Aragon app starting next month.

With new parallel app that we build, you can say I want 30% of ether for example and 30% of the token itself and 30% of a stable coin that takes its value to dollars, so you actually get some percent of your salary in dollars. Then you can actually withdraw your proportional part of the each block each day, or each week or whatever.

You have to wait until the end of the month, because if you think about it, like payroll is just really done at the week, or the month, because it's annoying. Someone has to do it and you don't want to be that person every single day. With the smart contract, you can do that. You can withdraw your salary each 14 seconds, each block, or each day or whatever and it's perfectly fine.

[0:28:02.8] JM: It's a really perfect example of streaming money, which is one of the things that people really look forward to once some of the scalability challenges and adoption challenges are overcome in terms of cryptocurrencies. Payroll for example, you could just orchestrate that through a smart contract. Delve a little bit deeper into that. Smart contract development is not something we've covered in-depth, yet we've got some upcoming episodes on it.

What you're saying is any organization that sets up on top of Aragon, a DAO, let's say Software Engineering Daily, first of all you mentioned that we can have this token issuance that will vest over time and it will vest programmatically. The token in that case really would resemble equity in the company. Like we would codify that in comparison to some of the ICOs that we've seen, where your token doesn't necessarily represent equity.

In this case, you're really talking about the token representing equity in the Software Engineering Daily decentralized autonomous organization, much like I might give away equity in my LLC, or MIC corporation. There's really not much different there. Then payroll, I think is really interesting example, because payroll is such a perfect use case for setting up a smart contract. If I'm looking into my Software Engineering Daily DAO dashboard on Aragon and I just want to do one click setup payroll, you could very easily imagine a smart contract being spun up that has certain parameters for fulfilling payroll.

Is that how it would work? Would it literally spin up a brand new smart contract, or does Aragon itself have some payroll contract that's already setup and all you would have to do is parameterize some mappings for Software Engineering Daily, so that Software Engineering Daily has payroll on the same smart contract? Would you want to setup your own Software Engineering Daily payroll smart contract, or reuse some internal one within Aragon?

[0:30:09.6] LC: While we're doing the case is when you deploy your own DAO, you are deploying your own smart contract. Then this rises a very interesting point, which version of the smart code that are you using? Are you using your own version, or are you using some old or canonical version?

You can actually do both. You can either, like it's still open source so you could deploy your own smart contract for the payroll, or you could use a canonical version that we offer in our code repos and you can just use a version, and that's it, which is very easy.

In any case, like everything is open source so you could actually do our implementation and when you deploy in your instance of the payroll app, you can create a new smart contract and parameterize as you want. It's very interesting, because also all of these operations take a lot of what's called gas in Ethereum, which is basically the computation that it takes for the network to actually perform all the operations in the smart contract.

We have setup this way of creating new smart contracts to create cold factories. Factories basically contain a lot of code. When you call it, they replicate that on their into another smart contract, which is way cheaper than actually deploying a new version. In that sense, if you use our own smart contract in our code repo, it's much easier and cheaper for you.

[0:31:29.3] JM: The token question, where when I setup Software Engineering Daily as a DAO and I issue a token over a vesting schedule to my co-founder and my employee. Actually, let's say we have a disagreement. Let's say we have a disagreement over what equity the parting employee should get.

In the centralized company world, the argument could be mediated by somebody who is very familiar in LLC law, or in corporate law, whatever and they could regulate that. In the decentralized world, what happens there is we have a disagreement over the equity, and also this begs the question what even defines that token being worth something. I guess, if the

Software Engineering Daily DAO also offers the tokens on the public market then there is an open market for them, but how do these tokens that I am spinning up for Software Engineering Daily to DAO, how do these have value? How do they have a legally binding value and how are legal issues disintermediated?

[0:32:39.9] LC: Yeah. Those are very interesting questions. One of them, like the value of the tokens, if they are not publicly tradeable then it will be the same as, for example how startups are valued in the sense that they have investment rounds, or how companies are valued, like private companies are valued in terms of some parameters just a revenue and all of that.

It's really a very liquid market. I think that tokens can really help have – provide a lot of liquidity to those tokens. Ideally, every single token will be tradeable, and then therefore, you get a market price. That's one thing.

For the other things like how did you resolve dispute in the smart contract? That's very interesting, because smart contracts are in the middle of humans and machines. They represent human intent, but they are securely machines. Therefore, there are a lot of subjective issues that humans know about that machines don't really know about.

For that, we are building what's called the Aragon network, which is basically the first fully detailed jurisdiction. We noticed that there are multiple things that nation states provide for companies and for people that are actually very well thought out. One of them is dispute resolution. You can go to court and make sure that if someone does something to you that is not correct, you can go and actually resolve a dispute.

We want to actually create a decentralized court, in which that just can be anywhere in the world and they can participate in a chance of having a small stake in the system. If they actually perform the right calls, they get rewarded. If not, they get slashed. I think that's an interesting way of looking at it is, let's replicate the things that work well in the traditional world, which are probably like judges and courts.

I mean, they are very slow and they're something that we want to fix with our decentralized court mechanism. We want cases to be resolved in ours and not like months or years, because that's the bottleneck in the traditional system. Yeah, I think you could have a clause, your smart contract which says this parameter can be changed by an external party, which is this arbitration of smart contract, which can actually change things if and only if a set of judges actually approves that changes and gives you – basically makes you win the sentence.

[SPONSOR MESSAGE]

[INTERVIEW CONTINUED]

[0:36:26.5] JM: Okay. Let's go a little bit deeper. In traditional equity markets, if I remember correctly and please correct me if I'm wrong, but I believe what is ultimately giving value to a share of a company is that you have voting rights, or voting rights/ownership right. If you own all the stock in an organization, you can take control of that company.

If Google owned all of the shares of Facebook for example, Google could have the power to redirect all of the free cash flows of Facebook into Google's coffers. Because of that fact, that is what leads to these disparate shares of Facebook having value on the open market is because it's ultimately a share of an organization. It's a voting right. It's a percentage ownership. By having preponderance of those, you can have control over the free cash flows of that organization.

Now with tokens, this can become a little bit more confusing, because tokens can have utility, as well as representing ownership. Actually, in most cases they purport to have utility, whether or not they actually have utility as quite more of an open question. In the case of IPFS for example, Filecoin purports to be utility. You can use it to pay for essentially storage, but it's not having shares in the Filecoin protocol explicitly. It's not explicitly shares the utility token.

When I spin up Software Engineering Daily, the DAO my token – Software Engineering Daily is a podcast, so it's not like – I mean, maybe I could do, okay you can spend that token on merchandise in the store or something, but that's not really what we're saying here. We're actually saying this is an ownership stake. If somebody were to somehow acquire all of the Software Engineering Daily DAO tokens, they should be able to redirect the free cash flows of Software Engineering Daily into their own organization.
If Vox Media wants to acquire Software Engineering Daily and make it part of their podcast network, there should be some hypothetical route to them acquiring all of the Software Engineering Daily DAO tokens and then redirecting all of the cash flows.

Help me understand that entire picture. Are you replicating that same system of organizational management and acquisition and shareholding, or are you thinking of doing something completely different?

[0:38:58.8] LC: We're not very opinionated at this point. Right now, with Aragon you can really do both. For example, if you want to replicate how companies work today, then what will you is you will create a token and then give token holders permissions to with rough hands if 51% of the voting power actually the size to withdraw funds. You can also give them as I said voting power over decisions.

Then you have a fully fledge equity over dividends and all of that. You can do that, or you can really do a utility token, like you are talking about Filecoin. You can do the same thing where your token is actually like a ticket to a service or something like that. You can really do both. The beautiful thing about it is it depends how you save your permissions, like you have multiple apps and depending on how you connect them, how you assign permissions to each app and how you assign permissions to token holders to what they can do, you can really program whatever you want.

You can even if you want to replicate some of the way some public companies work, for example Google which has multiple kinds of stocks, you can do that with Aragon too. You can create multiple tokens and then one of them have dividends plus voting power and the other one only has dividends power and not voting power, for example.

You can really create these combinations. It's all very simple. There is a permission center face and you basically can assign permissions to each of the apps to what they can do in other apps, and also what token holders can do, if they can withdraw funds, if they can withdraw dividends, if they can do a lot of these stuff.

I think that a lot of interesting models will come out not on the traditional company's front, but on the more experimentation front of what you can do with new kind of assets that are not equities, but are something else.

[0:40:52.5] JM: I want to come back to the question of cash flows, because I've heard in multiple places when you're talking about traditional companies, if you're evaluating what the worth of a company should be, like the market value of a company, you should try to evaluate what the present value of the future cash flows of the company should be.

Again, that's maybe entirely because if you acquire that organization, you would get to redirect all of the cash flows, present and future to your own coffers. Is that to say that in order for these autonomous organizations, these decentralized autonomous organizations, that people start within Aragon, in order for these to be tokenized and for that token to be binding in a way that would be meaningful, you would need to have all of the payments and the cash flows of these organizations coming through the Aragon network?

[0:41:54.7] LC: Well, you'd certainly need them to be compatible with Ethereum network. Right now, that means like Ethereum tokens. I mean, probably the Ethereum guys are in other communities are working on cross chain compatibilities with probably all their assets and other chains like Bitcoin, [inaudible 0:42:12.6], all of these assets.

Also, there are people working on Oracles that can fetch data and interact with APIs in the traditional web so to say, like in the internet. In the future, you could be able to actually own other assets. For example, like real estate or stuff like that. As of today, you can – or your DAO can only own assets issued on the Ethereum network.

[0:42:40.8] JM: I see. Help me understand, Software Engineering Daily gets paid by advertisers and that money comes in to my bank account, just a money transfer, or I get a check. How far do you think we are from those payment rails being properly setup to have the cash flows coming through Ethereum, or coming through a side chain that's compatible with Aragon, or maybe this is just a – happens to be a really terrible example that is very far from spinning up. In fact, I'm pretty sure at this point it is.

What is an organization that – what do you think will be the first set of organizations that will make a lot of sense to set up on Aragon? I don't mean to be hard on the project. I think everything were still in very early days. I believe in your vision. I'm just trying to get a sense for how far we are from production use cases, where Aragon really makes a lot of sense?

[0:43:46.0] LC: I think we are super, super early, especially for the cases you've been seeing interacting with sort of the – the traditional world and bank accounts and all of that, that will take a lot of time to setup; probably a couple years even.

In the short-term, I'm very excited about Apple use cases. One of them is cryptoprojects, like basically talking sales and projects that have done ICOs as well, where you have a large number of token holders that have contributed also large number of funds. Right now the idea is controlled by a single entity, or by two or three people.

That can be problem. We have seen a pool of these projects just ran with their money, or the founder is just getting rich overnight and then cashing it out and just don't provision on the project anymore.

With DAOs, you can really fundraise having really strict rules on how you can spend that money. Also you can have transparency, because everything is feasible on change. You can go to a blockchain explorer and see what the founders are spending their money into. That's something that we are actually doing ourselves. If you go to transparency that Aragon went, you can see basically our expenses, like our day-to-day expenses.

I think cryptoprojects are going to be the first users. We have seven projects that are really committed to using these with the token holders this year, so that will be interesting. Then we have another second use case, which I'm very excited about, which is open source projects. Right now with open source, you don't really have any truly sustainable path. You can do consulting or stuff like that, but it's really hard.

With open source projects, it's really easy to build in this decentralized governance. For example, create a token that represents that governance of the open source project, in terms of features and roadmap and all of that. Then have the founding team and maintainers and contributors own that token and if they do their job well, that token will appreciate in value because it represents governance power over the project. If the project goes well, people will actually want to buy it.

Therefore, you can provide a very sustainable path for open source projects, which I believe will be huge, because I mean if open source is working great and not having formal incentivization scheme, imagine if actually people were incentivized to contribute and maintain open source projects.

[0:46:11.7] JM: Speaking of which, so Aragon had an ICO as you said and you raised I believe 25 million dollars' worth of ether in the token sale. Was there a lockup period for those coins after you did a token sale? Are you talking about the transparency and stuff. Is there some dispensation period, or do you get all the cash in your hands immediately?

[0:46:35.3] LC: I mean, for the tokens, like of the ANTO, which is Aragon's token, which we actually like as founders have a vesting period. Then for the ether we raised, it didn't have any sort of period where you squared upfront.

It's funny, because I mean we would have loved to actually do this fundraise using a DAO and then having vested over the fence, over like different milestones and all of that. It's ironic, because we are doing that ourselves so we couldn't do that in the first place. Our rough map is that when we have the Aragon network deployed and we are able to actually give governance over the project to our token holders, then we will release a big part of the funds we raised to the token holders, so they can actually like – they will be the ultimate owners of the funds and they will be able to do whatever they want with it, which will be an amazing government experiment.

It will probably take a couple years though, because we don't want to put hundreds of millions of dollars under the control of a super bleeding at governance experiment, but we will slowly transition the funds to the token holders.

[0:47:37.0] JM: Fascinating. You're talking about transitioning the control of the actual ether you raised to the token holders.

[0:47:43.2] LC: Yeah, exactly.

[0:47:44.3] JM: Interesting. What about taking some cash off the table? Have the founders gone to take a million or two or just –

[0:47:53.3] LC: No.

[0:47:54.9] JM: Nothing like that?

[0:47:55.8] LC: No. Not at all. I mean, I have seen that in like little projects in the space and it just started well, like – we make sure that the people we hire are actually voting for the money, because I mean, we get that a lot of times and we will raise a lot of money and people then are like, "Yeah, I want to work on this project, because they might pay well." Actually not true. We actually want to underpay people, because we actually want to get the best and the most motivated.

That also goes for the founders, like we haven't taken any money off the table and like I will never do that. Basically, because it's just unfair to – if people give you money upfront, which is very ironic, because historical in the startup ecosystem for example, you work in a project and then maybe some other startup or whatever buys you and then you get the cash.

Here, you are getting a bunch of cash upfront. You have to be very careful not to fuck up your incentives, because they actually gave you the cash to work on the project. Not to go on and buy Lambos or whatever. First work on it and then do whatever you want with it. First, you have to actually deliver what you promised and that's what we are building and basically have found doing.

[0:49:03.7] JM: Yeah. Okay. Well, I know we're up against time and this has just been a really fascinating conversation, so I'm sure we'll have to do another one in a couple years or something, or you have some more developments. I want to close off on one other question about corporations, because I think having decentralized organizations as a way to structure your organization would be awesome, but corporations in the past, they have not been run democratically. There's no corporation that I knew of that is really run democratically.

They could've instituted internal voting systems. They could've mock decentralization in, organizations, incorporations. Why don't you think that has been done in corporations? Or have corporations just been – is that just a structure in the past where you have wanted somebody at the top? I don't know. What are your theories for why there have not been totally democratized decentralized corporations in the past.

[0:50:04.3] LC: Well, there have been a couple of cases where you have structures that are closer to co-ops and people like democratically and have very democratic purposes, the issues that it was very impractical. It was very slow and very bureaucratic. Also, I mean you receive an easy to forge votes in like government, like a government election. Think about forging votes for a company, it would be realistic.

It was really hard and slow. I mean, today you can really vote in 14 seconds until your transaction confirms and totally impossible to fake, impossible to forge. I think that is the real game changer here, that there were things that were not practical. Same thing for example, like having a company in which you have a million shareholders. This is just very impractical and only a few companies actually can do that by go in public and all of that.

Today, we can do that – you can do an airdrop of your token and then you have a lot of people that are a part of your organization. I think that is a huge change. We are automating human organization and that's huge for a lot of use cases that we couldn't previously try out because it was so inefficient.

[0:51:13.4] JM: All right. Luis Cuende, thanks for coming on the show.

[0:51:16.3] LC: Thanks for having me.

[END OF INTERVIEW]

[0:51:20.7] JM: Your enterprise produces lots of data, but you aren't capturing as much as you would like. You aren't storing it in the right place and you don't have the proper tools to run complex queries against your data.

MapR is a converged data platform that runs across any cloud. MapR provides storage, analytics and machine learning engines. Use the MapR operational database and event streams to capture your data. Use the MapR analytics and machine learning engines to analyze your data in batch, or interactively across any cloud, on premise, or at the edge.

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

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

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

[END]