Podcast Transcript

Blockchain University Episode 3: What You Need to Know About the Most Common Blockchain Networks

Date: November 18, 2020

Guest: Rob Musiala, Jordan Silversmith **Host:** Amy Kattman

Run Time: 23:25

For questions and contact:

Robert A. Musiala Jr.

Counsel
Chicago
T: +1.312.416.8192 | rmusiala@bakerlaw.com

Jordan R. Silversmith

Associate
New York
T: +1.212.847.6968 | jsilversmith@bakerlaw.com

Kattman: Blockchain technology is widely anticipated to disrupt major industries and business operations over the next several years, but with all of the hype in the blockchain market, at times it can be difficult to separate fact from fiction and identify the real value in this new technology. To help bring things into focus, we’ve crafted a five-part series, BakerHostetler Blockchain University, to introduce blockchain from a technological, market and legal perspective. In the third episode in the series, we focus on the differences and similarities between Ethereum Network, Hyperledger and other key blockchain networks. I’m Amy Kattman and you’re listening to BakerHosts. Our guests today are Rob Musiala and Jordan Silversmith. Rob is Counsel in the Digital Assets and Data Management Group and Co-leader of the Blockchain Technologies and Digital Currencies team. Jordan is an Associate at BakerHostetler whose practice focuses on financial litigation, blockchain technology and cryptocurrencies. Welcome to the show, Rob and Jordan.

Musiala: Thanks, Amy. Nice to be here.

Silversmith: Thank you, Amy. Pleasure to be here.

Kattman: In the last episode, we focused on the Bitcoin network, but there are many more blockchain networks out there, each with their own unique features. Today, we're going to provide an introduction to some of the more commonly used blockchain networks. Rob, lets begin with you. What is the Ethereum Network, and where did it come from?

Musiala: So, the Ethereum Network is what I like to call public blockchain 2.0. In the last episode of our podcast series we talked about the Bitcoin network, and one of the interesting aspects of the Bitcoin network is that the code base, the code that operates the network, is open source. It's publicly available. And as we mentioned last time, Bitcoin was launched back in 2009. So what happened was, beginning in 2009 technologists all over the world became fascinated with Bitcoin, and they could see that the code base was publicly available for them to take a look at, and a group of technologists, after studying the Bitcoin open source code base for several years decided, in around the 2014 timeframe, to launch their own sort of version of Bitcoin that they called Ethereum, and what they did was they essentially copied the Bitcoin code base, which was publicly available, and they tweaked it. They made certain updates, revisions, added certain features that they thought would be useful, and by and large they tried to add some features that would make it more friendly for building different types of business applications and technology platforms on top of the blockchain as opposed to just a payment system, which is what Bitcoin was intended for.

So, they did this, and they launched it in 2014 and they called it the Ethereum Network. Again, it's essentially, has a lot of the similar features as Bitcoin because a lot of the code was copied directly from the Bitcoin source code but with a lot of important updates, and we'll talk more about those updates. The Ethereum Network is operated by a Swiss non-profit foundation called the Ethereum Foundation, and we'll talk about some of the unique characteristics, but one I want to talk about now is this concept of distributed autonomous organizations, D-A-Os or DAOs. The reason why I bring it up is that a DAO is essentially intended to be a self-operating business organization. So, by programming computer code into the Ethereum blockchain using what are called smart contracts, the idea is that you can create an organization that effectively runs itself, and the first of these DAOs was launched shortly after the Ethereum Network launched. It was intended to be sort of an automated investment organization, where it would invest cryptocurrencies in certain projects. This DAO was hacked. In 2016 the very first DAO on the Ethereum Network was hacked. The hacker ran off with about $50,000,000 worth of cryptocurrencies, and in response to that the Ethereum Foundation and the folks that created Ethereum decided to essentially erase that theft by performing a coordinated 51% attack.

Now, we talked a little bit about 51% attacks in the Bitcoin podcast, and that's where if over, if 51% or more of the computer nodes supporting a blockchain all are acting in concert, they can actually change the transaction ledger, and that's exactly what, in the early days, happened with Ethereum. The computer nodes supporting the Ethereum Network all grouped together and coordinated together to essentially erase this $50,0000,000 hack. But not everyone agreed that that was the right action. A whole other group of the node supporting the Ethereum Network said, no, it's not right that we erase transactions, even if it was a theft. So, what ended up happening as a result of that is that the Ethereum Network actually split into two different blockchains. So, it split into what is now known as the Ethereum Network, and that is the more commonly used one, on the one hand, which erased the $50,000,000 hack, and on the other hand, the Ethereum Classic Network which kept the record of the hack, and the Ethereum Classic Network is still actually used fairly widely to this day.

So, I bring that up because one of the things we learned during the launch of the Ethereum Network is that it actually is possible to go about a coordinated process where you take one blockchain and split it into two, this is called a forking event. Ethereum, you know, in addition to all the other great things that it has introduced, showed us, kind of unwittingly, that it is possible to undergo these coordinated forking events where one blockchain is split into two, and those events have happened several times since that first forking event happened with Ethereum. But that's a little bit of background on the Ethereum Network and where it came from.

Kattman: Thanks, Rob. Jordan, what are some of the key characteristics of the Ethereum Network?

Silversmith: So, I think it would be helpful if we used an image for this. So, let's use a vending machine. Now, what happens when you want to get a snack from a vending machine? You look for the snack you want in the machine, you enter the unique code for the particular snack, you insert money into the machine and usually it comes out. That's something like how a smart contract works. Smart contract is a kind of computer program that's unique to blockchain networks, and it's intended to automatically execute an event or action pursuant to the terms of the contract without the cost of intermediators, arbitrators, enforcement or other kinds of externalities. Ideally, it's a simplified and legally verifiable transaction that, in its own way, imprints logic into the Ethereum blockchain. Smart contracts and Ethereum take the form of executable computer programs that are made out of code and data that are then deployed in digitally signed transactions on those blockchain networks. There are three essential features or objects of a smart contract: the signatories, the subject of the agreement and the specific terms. The signatories, of course, constitute the parties, two or more, who work together to come to terms and sign using a digital signature. The second object is the subject of the agreement, which can only be an object that exists within the smart contract's environment. Of course, any contract has to have specific terms. Those terms need to be mathematically described and these terms will include the requirements expected from the participating parties as well as all the rules, rewards and penalties associated with those terms.

A smart contract will also automatically enforce those rules. It's like everything you need to write and implement a contract, the paperwork, communication, middlemen and agents, all in one automatically executable program. Now, I'm gonna use the vending machine example again. What do you use to buy candy? Coins usually, or money in general, but coins, and that's close to what ERC-20 tokens are. ERC-20 tokens, like Bitcoin and other cryptocurrencies, are blockchain based assets that have value and can be sent and received. What makes ERC-20 tokens different from those other cryptocurrencies, though, is that ERC-20 tokens do not run on their own blockchain. Instead, they are issued on the Ethereum Network. ERC-20 tokens define a list of rules that all tokens on Ethereum must abide by, which has turned the token into kind of the technical standard used for all smart contracts in the Ethereum blockchain for token implementation.

Now, Rob mentioned DAOs earlier, but let's talk about them a little more. Using our vending machine metaphor, imagine you create a vending machine, and then it not only gives you candy in exchange for money, but also automatically reorders its supply, pays rent, cleans, and when you put money into the machine you and its other users can influence what snacks it stocks. That's similar to what a DAO is. Like Rob mentioned, DAO is an abbreviation for Decentralized Autonomous Organization. It's a completely automated company that manages itself and whose rules are written into the Ethereum blockchain and encoded as a smart contract. Now, once it's encoded as a computer program, a DAO is transparent and controlled by its members. Then the company operates entirely or mostly off the code contained in its digital charter. Now, DAOs are still an emerging new concept. Rob had mentioned the first DAO called the DAO, which was launched in May 2016 and was subsequently hacked, losing over $50,000,000 in Ether. Now, the legal status of DAOs is unclear, but neither the hacking of DAOs or their ambiguous legal status has really stopped developers from working on developing them. This past May the number of active DAOs was actually reported to be up 660% since the same time in 2019.

Now, let's talk about one more thing. Let's leave behind our vending machine metaphor and we'll talk about gas fees. Gas fees, or gas, as it's just called on Ether. Gas fees are transaction costs in which you pay to submit transactions on the Ethereum Network. More technically, gas refers to the unit that measures the amount of computational effort required to execute specific operations on the Ethereum Network, whether it's doing something like transferring Ether, sending tokens, or working on a smart contract. In the same way that a car currently needs gas to run, the Ethereum Network needs gas fees to operate. In the same way that when people are traveling more and more and there are more cars on the road, gas can be in greater demand and its prices go up. Ethereum gas fees increase as the network gets busier. So those were just four kind of concepts I thought would be useful to get into.

Musiala: And Amy, just to highlight a couple of things. With regard to smart contracts, I have two soundbites that I always like to repeat that I think are helpful. The first is that smart contracts are neither smart nor contracts. They are not legal contracts. They are instead if-then type statements, the ability to program code into Ethereum so that when one type of transaction happens, another type of transaction automatically happens. Another soundbite I like to repeat is that you should always get a contract for your smart contract. In other words, get a real, legal contract to govern the operations of your smart contracts, and the reason for that is that smart contracts are only as smart as the person who programmed them. So, you know, smart contracts, like any computer code, are programmed by humans, humans are flawed, and so smart contracts can be susceptible to hacks just like any other computer code can, which is why it's important not only to audit them from a technical perspective, but also to define the legal parameters that govern their operations, especially in the event that they don't function exactly the way that you intended them to. The other quick note I want to make is that, with regard to ERC-20 tokens, people may have heard of these things before, they have created a whole bunch of very complicated legal issues. We will, we're not going to get into those today, but we will get into those in a later podcast.

Kattman: This is all great information. Let's talk about Hyperledger. Jordan, what is Hyperledger? Where did it come from and how is it used?

Silversmith: Sure. Hyperledger, or I think, as it's formally known the Hyperledger Project. It's an umbrella term for a collection of open source blockchains that was started in 2015 by the Linux Foundation for enterprise grade blockchain deployment. Other companies have contributed to the Hyperledger Project subsequently, including Intel and IBM, but the role played by the Linux Foundation, which hosts the project, cannot be understated. The Linux Foundation has been around since 2000, when two groups, Open Source Development Labs and the Free Standards Group, merged to standardize Linux. While standardizing, supporting and popularizing Linux was and remains central to what they do, in recent years the Linux Foundation has expanded its support of funding for various collaborative open source software projects. One of those projects is Hyperledger. The Linux Foundation announced the creation of the Hyperledger Project in December 2015. Its goal is fundamentally about collaboration among enterprise level companies, developing blockchains and distributed ledgers to improve those systems' reliability and performance so that, eventually, technological, financial and supply chain companies can use these systems for global business transactions. Now, to that end, and differing a little bit from Ethereum, Hyperledger has a focus on data management rather than payments and is built for enterprises rather than individual use. The Hyperledger Project focuses on enterprise grade blockchain technologies by using a permission network. So instead of needing to run proof of work mechanisms like on public blockchains, on Bitcoin and Ethereum, participants using Hyperledger are all global businesses, therefore known quantities to each other, so they can skip the proof of work concept step and move on to more pressing problems, whether it's collaborating on cross industry issues or recording trades or transactions into the blockchain.

Now, within the Hyperledger Project there are actually 16 different types of Hyperledger code protocols, each designed with their own unique features. What happens is that a company will write the original code and they subsequently donate it to the Linux Foundation, at which point it becomes part of the Hyperledger Project. That code becomes publicly available, and a company who wants to use it can copy it from Hyperledger and alter it for their own purposes. Now, I want to highlight a few of them. Probably the most prominent Hyperledger code protocol is IBM's Hyperledger Fabric. Fabric was the first code protocol to emerge from the Hyperledger incubation stage, becoming active in 2017. It's an enterprise grade distributed ledger platform whose focus is on modularity and versatility for a wide variety of industry use cases. Because Hyperledger Fabric is private and requires permission to access, businesses can segregate information, like prices. Plus, transactions can be sped up because the number of nodes on the network is reduced. It was launched by the Linux Foundation in December 2015 and Fabric 2.0 was actually updated and released this past January.

There are a few other Hyperledger code protocols I'd like to highlight. Hyperledger Sawtooth was created by Intel. It's used a lot with Fabric, and it's a distributed ledger software that, along with Fabric, was generally designed for data management solutions for enterprise. Hyperledger Burrow is another one. It was originally contributed by a startup company called Eris, later known as Monax, which was arguably the first permissioned blockchain. Its primary function is to execute Ethereum smart contracting code on a permissioned blockchain network. There's also Hyperledger Indy, which was contributed by the Sovereign Foundation, who wrote the code and donated it to Hyperledger, at which point it became Hyperledger Indy. Now, the Sovereign Foundation, if you know anything about them, they're focused on the concept of self-sovereign identity, and that's the main thrust of Hyperledger Indy as well. The Hyperledger Indy blockchain code implements decentralized identity solutions on the blockchain. Developers using Hyperledger Indy can use these tools and libraries from it to create identity solutions that are actually interoperable across jurisdictions and agencies. So, like I said, there are a lot of code protocols that are part of the Hyperledger Project, but I just wanted to highlight those few and there will more to come, I'm sure.

Kattman: That I'm sure. Thanks, Jordan. Rob, how is Hyperledger different from Bitcoin and Ethereum?

Musiala: In general, as Jordan mentioned, Hyperledger is targeted at traditional businesses and is focused more on data management, not necessarily payments. Ethereum, on the other hand, while it has features that can be used in business applications, it's targeted more at the startup community, and it does have those payment features similar to Bitcoin. So, in Ethereum you have its native cryptocurrency, Ether. Hyperledger does not have a native cryptocurrency. To dig a little bit deeper, Ethereum is what is called a public permission list blockchain. It's public in the sense that anyone can access it and use it, and it's permission-less in the sense that anyone can donate their computer processing power to operate a node on the network and co-host that network, earning rewards in the cryptocurrency, Ether. In contrast, Hyperledger is a private permissioned network. It's private in the sense that it is invitation only in terms of who can use it, and it's permissioned in that it's invitation only in terms of who can operate a computer node co-hosting that network.

Another key difference is in the consensus algorithm. So Ethereum, like Bitcoin, uses a proof of work algorithm whereby transaction blocks on the blockchain are verified and validated once 51% of all the computer nodes co-hosting that blockchain have validated the transactions. In contrast, Hyperledger uses what's called a follow the leader consensus algorithm, and that's where the various nodes on the blockchain appoint one node as quote, the leader node, and when a block of transactions is up for verification the leader node will validate that block according to predefined criteria, and then all of the other nodes will just automatically validate that block as long as the leader node has done it, so they essentially follow the leader and validate any blocks that the leader validates unless they take specific actions otherwise. So, this makes the transaction validation process a lot faster, but it also means that you have to really trust that leader node because if the leader is doing something wrong or validating transactions that are incorrect, there's a risk that everyone else is just gonna go along and validate those incorrect transactions.

One other difference I want to point out is that Ethereum is supported by and large by private hardware that is controlled by individual private parties, and it's that private hardware that hosts the nodes supporting the network, whereas with Hyperledger it's very much supported by and compatible with cloud service providers. So, you know, if you're a company that is co-hosting a Hyperledger blockchain, instead of having your own hardware to operate your node, you'll actually usually subcontract that function out to one of the major cloud providers like AWS, Microsoft Azure, SAP Cloud, Oracle Cloud or one of the other large cloud providers. Then lastly, as Jordan mentioned, Hyperledger does not have a native cryptocurrency, whereas Ethereum does have Ether as it's native cryptocurrency, as a payment mechanism. So those are some of the major differences.

Kattman: So, we've talked about Ethereum and we've talked about Hyperledger. What other blockchain networks are important to know about, Rob?

Musiala: There are lots of blockchains, but there are just a couple that I want to mention that I think are important to know about. Ripple Labs is a startup based in San Francisco. They founded the Ripple Network blockchain. It's a blockchain that's being used in back end payment systems, mostly for international remittance applications. The R3 Corda blockchain is another blockchain that is used in back end financial transactions. Corda is interesting because it was founded by a group of over 100 banks from around the world who came together to create their own private permission blockchain, the Corda blockchain, that is being used in all sorts of pilots and live applications now in the banking sector. It's also starting to be used outside of financial transactions as well. Another one is the Quorum Blockchain. JPMorgan was initially a part of this R3 Corda Group, this blockchain consortium, but very early on they left and they decided to sort of go it alone in terms of their blockchain development, and they started the Quorum Blockchain, which interestingly was based off of Ethereum, but Quorum is a private permission blockchain. JPMorgan actually recently sold the Quorum Blockchain to a blockchain development firm called ConsenSys. A fourth one I want to mention is called Stellar. Less popular than the first three I mentioned, Stellar is also used primarily in payment applications. Then, a last one is called VeChain. You probably won‘t hear about this one too much here in the U.S., but I like to mention it because it’s one of the more popular blockchains for enterprise applications that is being used in the foreign markets and in particular in the Asia-Pacific market. So those are a couple of the other blockchains that folks ought to know about.

Kattman: Thank you very much, Rob and Jordan, this has been very insightful.

Musiala: Thanks, Amy, always nice to be on the show.

Silversmith: Thanks, Amy, it was a pleasure being here.

Kattman: If you have any questions for Rob or Jordan, their contact information is in the show notes. Be sure to check out our weekly Blockchain Monitor blog, where Rob is an editor and Jordan is a frequent contributor. In our next episode of BakerHostetler Blockchain University, we will discuss non-financial use cases. As always, thanks for listening to BakerHosts.

Comments heard on BakerHosts are for informational purposes and should not be construed as legal advice regarding any specific facts or circumstances. Listeners should not act upon the information provided on BakerHosts without first consulting with a lawyer directly. The opinions expressed on BakerHosts are those of participants appearing on the program and do not necessarily reflect those of the firm. For more information about our practices and experience, please visit bakerlaw.com.