# Peer To Peer Networks

**Professor Y.N. Singh**

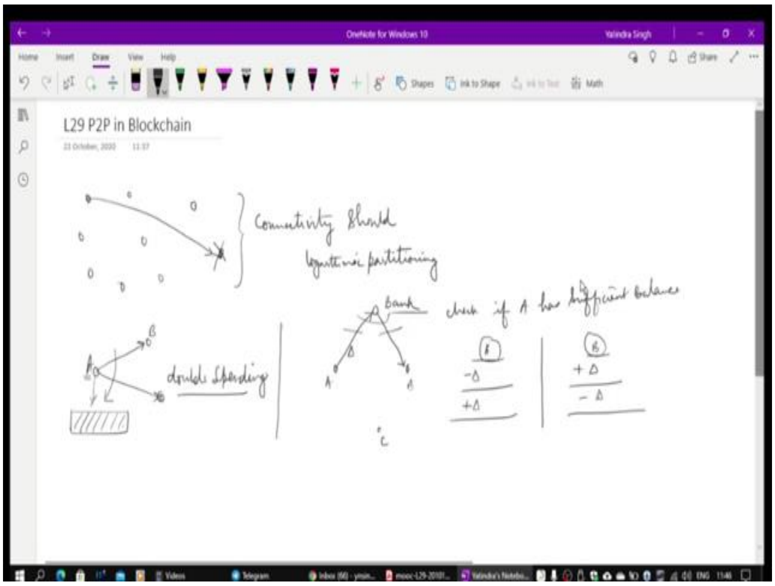
**Department of Electrical Engineering**

**Indian Institute of Technology, Kanpur**

**Lecture 29**

**P2P in Blockchain**

(Refer Slide Time: 00:15)



L29 P2P in Blockchain

Welcome to lecture number 29; we will be talking about peer to peer-based system called Blockchain. It is a perfect example of how the Blockchain has been used. Now basically, the idea is that when you read about Blockchain, look into anywhere in the literature, nobody will tell you that what kind of peer to peer network is underlined, what kind of DHT algorithm is in use.

And whether it is a Kademlia or a pastry or tapestry, or it will not be evident. The only important thing is that if nodes were there, there might be many nodes in this who are participating; what they will do, I will explain. But if this guy is alive and if some other node is alive, there is should always a network connection reachability should exist between them; only if they die off, then the reachability will not be there.

But, when it comes back, it should be able to get connected to the network again. The important thing is that connectivity should be maintained; that is an essential thing, and this is what is achieved by Logarithmic Partitioning based routing tables. It will be a structured system for how long this thing will work; again, what people tried was they tried the first important thing that let me explain what a difference is.

So, if you actually in the real world, if you have banknotes or currency. So if you give a currency from a to b, you send an actual transfer of the money. Now, what the question is, a cannot do the spend the same money again because it is not there with him, it is only with B. While suppose it would have been some random string, and you said I am okay. I will transfer this money to b; you transfer that random string to b but, you can always return a digital copy of this; it is a digital transfer basically; it is nothing but a string.

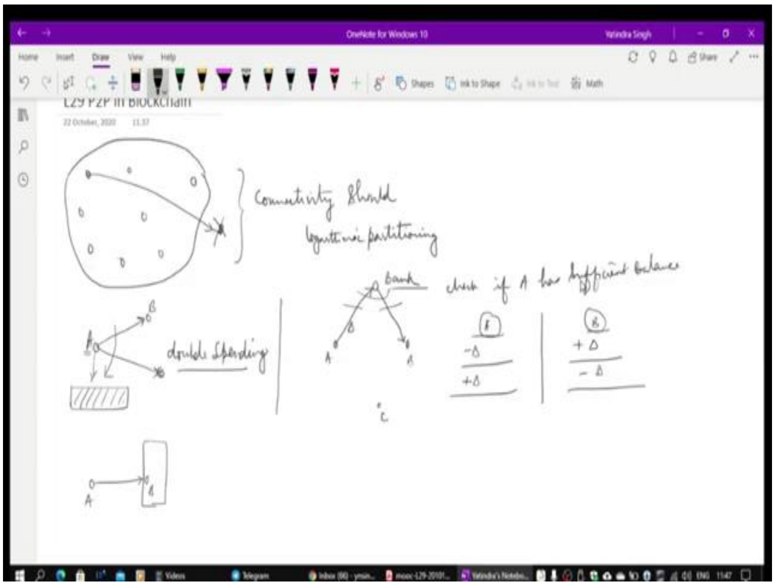
Now, the problem with this is that even before b spends, you can give it to somebody else and then buy something and give the money you get the stuff you can give it to so. You can do multiple spending of the same money, so we called it a double-spending problem. So that was one reason why this was not feasible in the digital domain. Typically, the transactions happen in the digital domain; when you want to transact to a to b, you always depend on some third party, which we called the bank. First of all, I will now communicate to the bank and say that you kindly transfer the money to b.

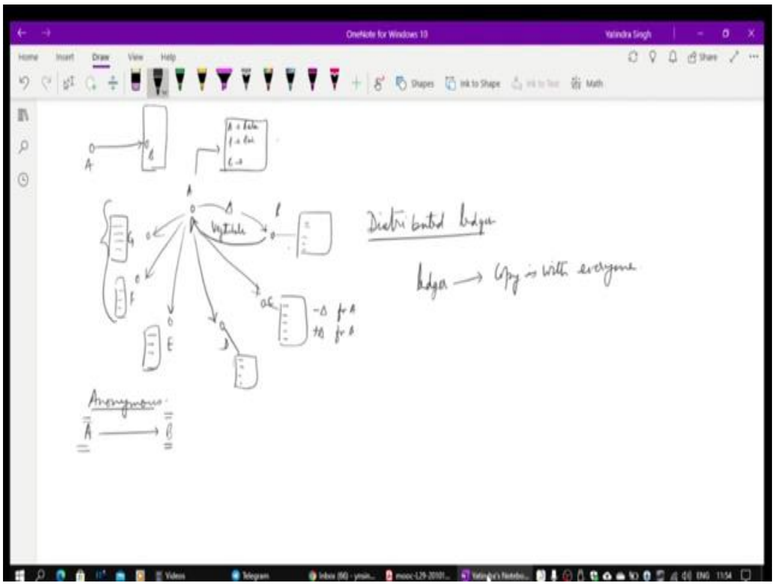
Some amount says delta amount has to be transferred, so what bank will do is check if there is sufficient balance. Once has sufficient balance, only then a transfer to b is basically what it will do. It will maintain the ledger for it will maintain a ledger for B, the bank maintains both ledgers, it will reduce the amount by minus delta here. It will increase the amount by delta here.

So, the balance will get updated in the bank now, a goes and wants to do a transaction with c it is to again talk to the bank they cannot do a direct transfer. So most of the digital transfers are done in this fashion; they depend on the bank but, the problem here is you depend on the bank and bank tomorrow can make an entry, may reverse these entries it may find out something wrong and may reverse it that is it and you are lost.

So, you depend on the faith of the bank as of now. So people wanted a system because now this depends on the bank. The bank will depend on the government regulator. Depending on the country, it is possible to do a transaction where you can do direct peer-to-peer transfer the way you do cash. So cash here, all transactions can be tracked; who communicated to whom the bank has the information so the governments can find out all the transactions.

(Refer Slide Time: 04:50)





But if you do a cash transaction, a gives some 100 rupees to b, so there is no record there only cash is there now with him, there is no record the cash was with whom earlier, all earlier traces are gone. So whoever has the value has one record, who has the value, who has the currency at that point of time that is more important and avoids double-spending.

So, people thought can do distinct in the electronic domain, so; the idea is pretty simple now after this. If all these people in Blockchain, how we call it a blockchain, I will explain that all these nodes, if they start maintaining the balance of everybody's life, will be straightforward to let me give the principle things works. So, we can now make another system where all nodes keep track of; of course, the issue is that there are many nodes how to do it and how to do it NMC.

There is a small village, and everybody maintains a notebook, so everybody maintains a notebook, so their names are A, B, C and D. You can now also realize why peer to peer communication becomes essential here. So, everybody maintains a notebook where say A is having this much balance, B also has this much balance, so everybody balance is recorded here.

When everybody maintains the same copy of the notebook, so records are there now, A and B decided they want to do some transactions, so A will sell something, say maybe some vegetables or something, and transfer the money. So what A; how the transfer will execute? So, there will be a will check whether B will check whether the what is a balance in his notebook, how much A has?

So, A has a sufficient amount of money perfect, so A and B both of them will agree that they have sufficient money is there, and this much money has to be transferred. So, say amount delta as to go from here to here now what they will do is once they agree for delta transfer, either of them can communicate to most likely it will be A. Because it is money which is getting transferred, I will tell everybody that kindly reduce this much delta amount from my balance and add it to the balance of B.

So everybody will now make an entry of minus delta for A. It will do plus delta for B, so everybody does it, and everybody confirms this so, once everybody has confirmed, so B will now do the confirmation. Yes, the money has been with each one of them. It has been confirmed so, now the money has been transferred, and no currency has been done; it is only the ledger entries that have been updated everywhere.

And then you can get this particular whatever vegetables he was trying to sell. Consequently, an updated ledger shows that balance with A has gone down the balance with B is high. Now, I want to again communicate with somebody else to do a transaction; they will check their ledger. Balance is reduced now, so it cannot do double spending every time you spend; balance is reducing. This is the principle; the only problem is now supposed A says let me make fraud, and I want to make an earlier entry to the change.

So, I paid maybe 5 days back 100 rupees to B, and I would like to deny that so he will remove that entry. Now once that entry has to be removed, it has to be removed everywhere.

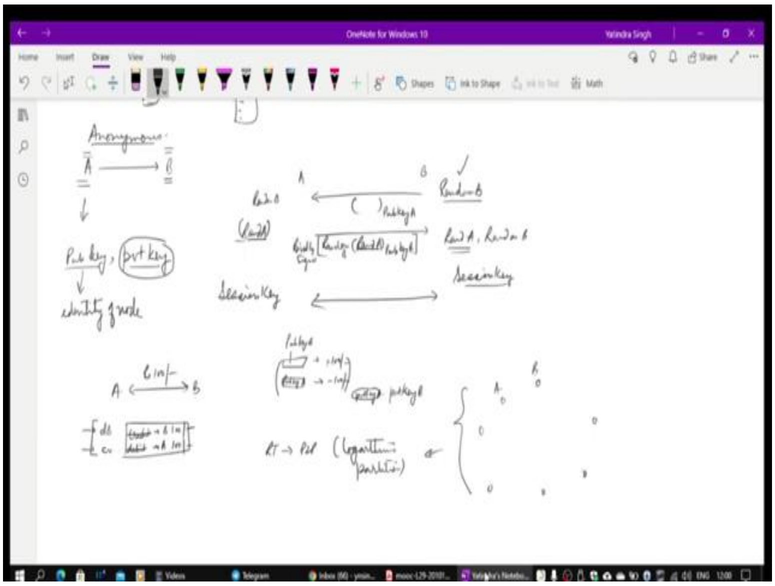
And B want to ensure that even if people collude, removing this particular entry will be very difficult; of course, you need a collision of more than 50 percent of people has to agree.

Yes, 100 rupees were never paid to B, and they have to make the upgradation in their ledger in these records. These records themselves can be made immutable by making it costly to do it because you do such so much energy has to waste in computing, and it takes a lot of time it is better not to do it. So this incentivized that so, that is the idea in the Blockchain so, this is a distributed ledger this as of now it is not a blockchain I can call it I should not even call it distributed ledger.

So, that is the word people used, but usually, a distributed ledger means the same ledger of part of the ledger is part of the ledger with this node part with other nodes it is the same ledger distributed to everyone. There is only one ledger, and the copy is there with everybody. That is a way copy is with everyone. That is how we should have to identify this.

In this case, so far, everybody else you also wanted anonymity, so whenever the money is transacted, this was probably the one reason this got invented that when you kept transferring money, a transfers money into B, B should not know who is transferring. A should not know who is transferring to whom. So they become anonymous. Anonymous, in a sense, there is no unique user ID being looked at but then how you will then identify A and B.

(Refer Slide Time: 10:47)



So, this was the same thing that we had done in the previous lecture; every node will generate a public and private key pair. And public and private key pair, in this case, this public key is what is become the identity of the route, and he needs to know the private key so that he can prove to somebody else that he is the person who is identified by this public key.

This can be done using SSL; its principle is pretty the same as generated, and B will generate some random number. And it will send this random number back to this guy, and this will be encrypted with the public key of A and the same thing which we had done earlier A will be who the only person who can decrypt it using the private key.

So, A knows this random number, and I can call it random B so, this random B can decrypt, and then he can generate another random number that will be A. It can use this random number; a can be now encrypted by transferring back with the public key of B.

So, this random number a can go here; of course, it is possible to send random B, but random b you can even send unencrypted does not matter. Somebody can tamper with this, but of course, this can be digitally signed by random a by the private key of a. So, signatures can be done so tampering can always figure out so, even if somebody knows random b, it would not be able to crack the security.

So, when this random B comes here and the only guy who knows actually what is a random B is this and this, if this guy is not the general person, random B must be different than this random b. So, when B figures out both confirm, so A is a confirm person, nobody can confirm this otherwise. Only A can do this check and random A is not known to anybody it comes here. It random A and random B both are now available to him, random A is kindly known to B and A, it has A, B has a private key so it has it can decrypt it. So, it can combine these two and generate a session key.

And it can start this guy also knows this also has the session key by this time; nobody else can make the session key because rand A is only known to A and B, and they can now communicate and confirm whether rand a has been correctly received by B. So, A and B both are genuine, then only this session key can be established, and they can communicate that is how the identity is checked by each other, and they can do the transactions.

So, for being anonymous again, the same mechanism will be used herein Blockchain. So now, when a transaction has to happen, we took this village example where, of course, the anonymity was not there. You know that who is transacting with whom. Random keys have replaced those identities, so we do not know who has the unique ID except when the communication is happening through surveillance agencies can figure out from which public key belongs to which particular IP address, that much address they can get it.

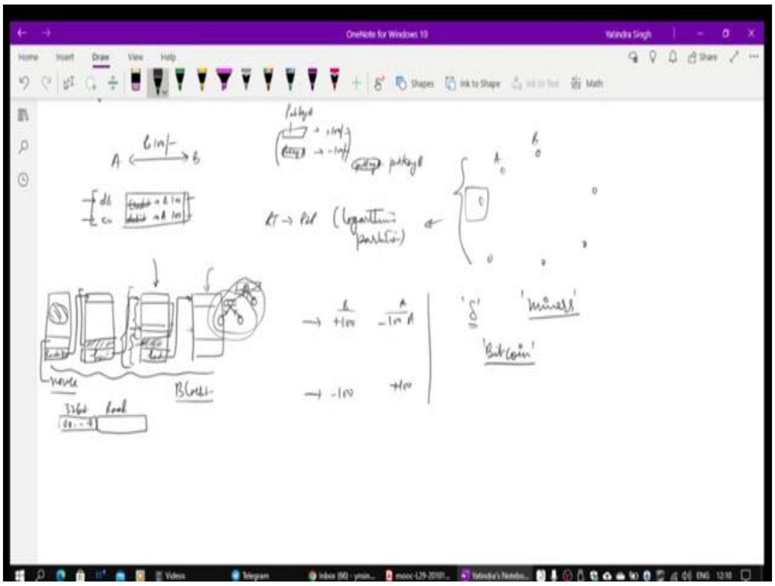
But mostly, everything is through the encrypted channel; it is challenging. Of course, once you localize, you can find out who the person is sitting there and catch him; that is circumstantial evidence. So, A and B both now transact that is an important thing here let us come back here; they say 100 rupees has to be transferred. So they will now make a transaction that credit to B should happen of 100 rupees and debit of should happen to A. it is a bank account this should be reversed, this should have been an actual debit entry, and there should have been a credit entry.

So bank accounts that are the way it happens debit mean an increase of your bank account and credit means a decrease of your bank account so, just being by accounting standards, I have to be correct. So, I have put it in this way, but it does not matter. This means the money is going from A to B; that is the way you should remember, this is only for the notational thing.

So once they make this record, they both will digitally sign this transaction so, how they will sign; so, everybody knows so the id of node B has a public key of B, and it says that reduce 100 rupees from my bank, increase 100 rupees here. This is the public key of a node ID, reducing 100 rupees here then they will sign, so this will be private key will be used for this by a. Private key of b they both sign. This transaction can now be distributed to everybody.

So, this A and B are sending it to everybody, but how they will get it; you will now be using your routing table in peer to peer network based on logarithmic partition. So in few hops, you will do broadcast routing, and everybody will have this record, and once this record is there, everybody can now put it in their transaction block. Now that is where the Blockchain comes because people should not refuse.

(Refer Slide Time: 17:08)



So what they will do is most of these people make a record now; another important thing they do is they now complete this record transaction which is happening before they are not still confirmed. So everybody has to confirm and lock it so that this cannot be reversed.

So locking is very important here, so there is a previous block that will maintain, so there is always some initial block. There is a hash value that is computed. There are two essential things to the hash. There is something called norms is a randomly generated number, and this can be anything arbitrary, but this has to be such that the hash value will have the first 32 bits to be 0, for example.

So they will be all zeros for any specified pattern remaining will be the hash value, so now this is done through group force approach to keep on trying till you get 32 bits as zeros in the hash value is when you see now I have got the correct norms. So, this hash value is being used as one of the parts of the next block and then there a norm here and then there a hash value computation this is being used in for the next block again, so when the hash is this hash is computed this hash over this all values, so previous hash also. When I computed this hash by identifying norms, I cannot modify anything here because if I do it, I have to do this whole chain has to be computed, and this is going to by everybody.

So you will be adding, for example, here block; this block is not confirmed, so everything has been confirming. You have now maintained the entries here, so you add the entry here; more is done here. They keep on doing end the thing; for example, they start doing some hashing combining them. So as and when a transaction happened, a new hash is computed, which will be recursively used.

So when an entry is made, it is not; it is difficult to move. However, of course, everybody can do it; it is still not locked. When the time comes after a certain period, then norms is everybody starts computing norms. They start all doing group force thing until they find out an appropriate hash with first 32 bits zero that is the difficulty level, so higher number bits you want to be zero will be higher difficulty more time it will take.

And never gets it first immediately inform everybody else to know what the norms were selected: this guy and everybody will verify. If the hash first will have the first 32 bits as zero if everybody gets it is; yes everybody has got the same set of transactions, and they will just put that hash value, and they will now lock this particular block new block start after this.

This actual computation takes a lot of effort, so it is impossible to change somewhere else, so naturally, if you have made an entry, you made 100 rupees to b n minus 100 rupees a; they remain there, you cannot do much. If you want to reverse it, you create another entry somewhere later in the block where you will do 100 here and 100 here; you can only make the correction entry.

So ledgers normally, even in the banks, you never do a correction; you always do a rectification entry. Once entries are made, made; if it is made incorrectly, you make a correction entry. All records are maintained to do auditory, so not any point of time when A and B before they are doing you do the transaction. They can check the audit trill if they know the earlier balance of a, they can find out this whole audit trill can be verified by doing this hashing business. And they can find out what all transactions were done there, so what is going to be a current balance; whether the guy pay it to me or not?

I will agree only on this transaction, and it will be communicated to everybody else, and when the block gets from the transaction gets confirmed, a new balance will come. When I assume it is the confirmation that has happened, I will give him whatever I had to transfer to him. If I found balance was not there by that time, balance became negative. When these entries are made one by one entry that I was talking about, you start tracking using this particular hash value, which I have computed whenever every transaction is added.

You cannot revert in between, and any new value that will check even through this transaction what is the current balance of a being maintained.

So, you will only accept the transaction when a has sufficient balance, and the moment this is done, it is locked, so now you can only look into the next block, so you have to only search through these transactions to ensure that how much balance is there. At the end of it, I know what a balance is, so balance for every node you can find out from the previous block wherever he has done the transaction.

So, I get this because I am creating a chain, a chain consisting of blocks, and blocks will have all the transactions done in a certain period. And they have been locked by computing the norms, so changing this block at most nodes is the most challenging thing, so it is done once it is done.

So, there is no bank involved; it is anonymity, and you can now do the, you have done the transactions also. Now, what happens if a node leaves; that is why if a node leaves, the remaining nodes will keep on doing the job, they will keep on maintaining when he comes back he will you find out he will lagging he will talk to the majority of the people and will update its blockchain part to the latest head.

So longest chain it will pick up and will go there, and if you are a new guy who is joining in, you can copy what the Blockchain which majority people have and then start working there onward actually in that.

Now, the guy who computes his norms first is a winner. He should be given something, so there is by default, there is an algorithm that says some amount of some delta money will be transferred and credited to this account; it is going to be generated, so that is like mining the money. That is the reason why all these nodes are also called minors. Not everybody does mining; some people only do the transactions, so they will talk to the minor and do it, but they would not compute the nodes.

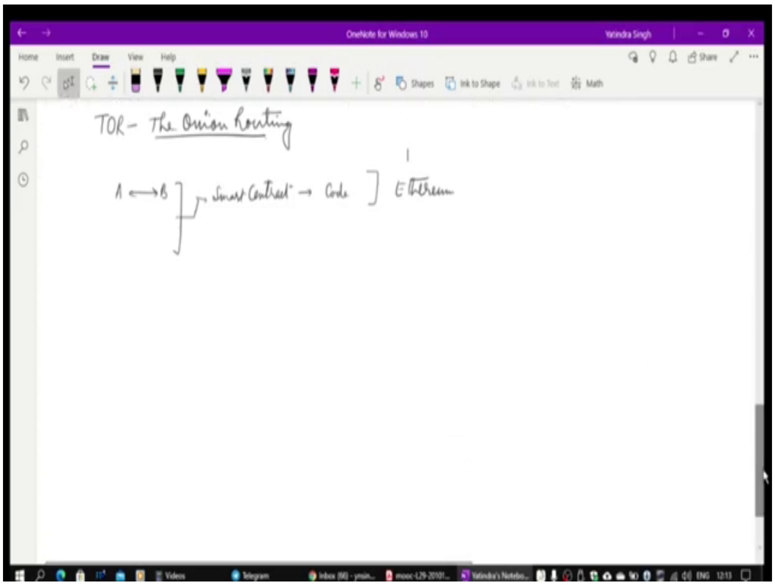
So, people who have the minors, the majority of that essentially will always figure out what is balance with everybody else, so there can be other participants also but, they may not be the minors. Instead of rupees here, you replace it with Bitcoin; that is what is used everywhere.

Now, since there is nothing else except private and public key as an identity for a node, if you lose your private key, the public key's balance is also lost unless you somehow recover that private key, which has happened with people, challenging and only a challenging part.

Usually, you have to keep a copy of your private key someplace always to recover your bitcoins and use them. As far as the peer-to-peer network is concerned, only an important part is connectivity, so everybody has to be connected if they are alive, so the logarithmic partition ensures that.

So, routing tables are mutually exchanged, and you keep on updating your routing tables. You can optionally use neighbour tables to keep on optimizing the network. Once this is done, this is very popular for doing transactions but transactions anonymously as of now if you I know public key I can find out IP address I can look the person.

(Refer Slide Time: 25:23)



People went even one step ahead, and they built something called TOR, TOR stands for The Onion Routing. I will describe this in the coming lectures and probably the last in the series. This provides anonymity; it is complicated to crack that who is the guy who is sending it. And at every periodic step, it keeps on changing; your identity will remain the same, so only two nodes A and B, when they are transacting they know each other's identity so that you can do this transaction between A and B.

A and B can find any one of them can send it to everybody all the minors and minors can update it so this was a pretty popular way of transacting the money there. There is also something called bit-coin exchanges that are there, so you essentially do it because you do not participate directly. You buy that bit-coin money from there, or you sell it back there depending on the demand and supply.

Now, there is one more step people have thought of now because remember what is happening when A and B are transacting. They are making some contract for something than a consequence, and the contract gets fulfilled the transaction of money. People went ahead and said, okay, we could split this process into two separate things; one is what we call smart contracts.

So the agreement can be signed between them can be locked and put in this blockchain structure, and it is a programmable thing it is code being provided, so there certain conditions if then else kinds of things are matched some transactions are going to happen. They will be automatically happening, so all minors keep track of the smart contract. Whenever the smart contract condition is matched, more than 50 percent of peoples will see it; they will immediately do the make ledger entry without even A and B telling them.

So, that is what also now people have been trying this came to Ethereum, now one of the problems which remain is that need a lot of computing power to compute norms, and we call it a proof of work, somebody has worked, he has given the proof. He gets credit for that minor's reward, which is a small amount of bitcoin that has been allocated to whoever cracks the value of the norms.

So, but this is a computation-intensive thing. It becomes harder and harder as time goes by, and secondly, all records are infinitely increasing. So this record, this Blockchain will keep on infinitely increasing as time goes by, so everybody has to keep a more massive and more extensive database for this. So your disc space will keep getting consumed, but you have small records as basically a sending it to b some hash values and other stuff, so it not much it not a much it is not big data, but ultimately it will accumulate.

So, something it has to sustain will be non-sustainable longer, so people have thought about the norm even to try some other mechanisms. So they call it proof of stake, so you hold a larger stake you proof that is when you can allow making the changes in the record. Just like some in society, the guys were well off. They are essentially being honoured, and the people have more faith in them. They can be used as intermediate for all the transactions records is typically the way it happens in villages similar kind of concept.

So, but the important thing is that this was the transacting anonymously any value without any intervention on the bank regulatory bodies or government was the reason why this bit-coin of Blockchain came. People had been trying earlier, but, ultimately, when this proof of what came the Blockchain came, the bit-coin was becoming feasible that time.

And but they all underline network is a peer to peer system where the routing tables are managed. You do a broadcast here; it is not a DHT routing being used to send to somebody else excepts A and B when they are talking or negotiating; they can use DHT routing. Still, once transactions are done for transaction confirmation, it is broadcast, which is used.