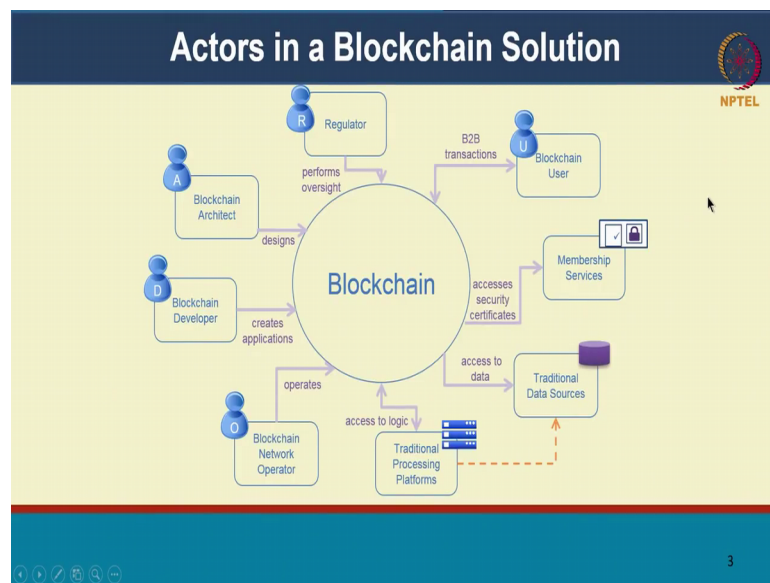


**Blockchains Architecture, Design and Use Cases**  
**Prof. Sandip Chakraborty**  
**Prof. Praveen Jayachandran**  
**Department of Computer Science and Engineering**  
**Indian Institute of Technology, Kharagpur**

**Lecture- 20**  
**Blockchain Components and Concepts**

Hello everyone, welcome to the next lecture of our Block Chains course on Architecture, Design and Use cases. So, this next lecture is going to be about some of the basic block chain components and concepts and how some of these components interact with each other. So, this is give you the foundation for us to go deeper in and learning about a hyper ledger fabric itself and how it is architected.

(Refer Slide Time: 00:36)



Who are some of the actors in a block chain solution? There is a block chain architect who is going to design, how this block chain solution is going to be built. We will figure out what are some of the information that needs to get stored, what are some of the transactions and the business logic and needs to be embedded onto the network, how the network itself should be created and so on.

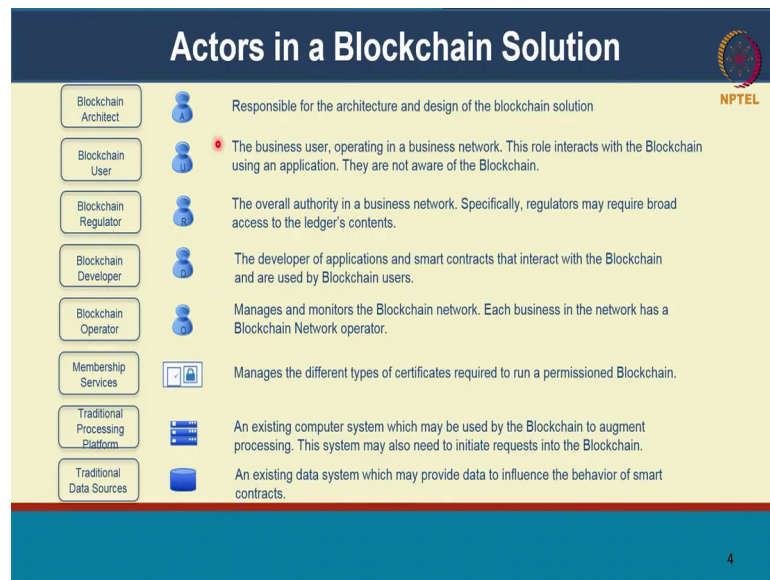
There is a block chain developer who is going to take what has been architected and they are going to develop the actual code that will run on the block chain network itself. There will be a network operator, a network operator who is going to manage and run the block

chain network. There may be traditional processing platforms, there may be other systems of record that the block chain connects to and might get information from and send information to that those are traditional processing platforms that could be data sources, there could be external databases that are also part of the overall solution. So, those are a part of the overall solution as well an important component here is membership services

So, the membership service defines or provides the identity for users to come and transact on the block chain. So, think of it, as when you open an account with the bank, they give you a username password, a kind of a login right to, for you to access web services or when you get a account with Gmail or something, you get a username password. The membership service is actually going to do a little more than that. It is not just a username password, but it is actually going to give you a certificate that will a digital certificate that will allow you to transact on the on the network. So, we will talk about some of those details as well and there is a block chain user who is, who is going to perform the business transactions on the block chain.

So, these users could belong to multiple organizations that are participating in this block chain network and optionally there can also be a regulator. The regulator might have only read, only access onto the network where they see, they have some oversight into whether the transactions being performed are legitimate or not whether they are compliant with policies set by the regulator. So, all that can be done in real time with a block chain platform.

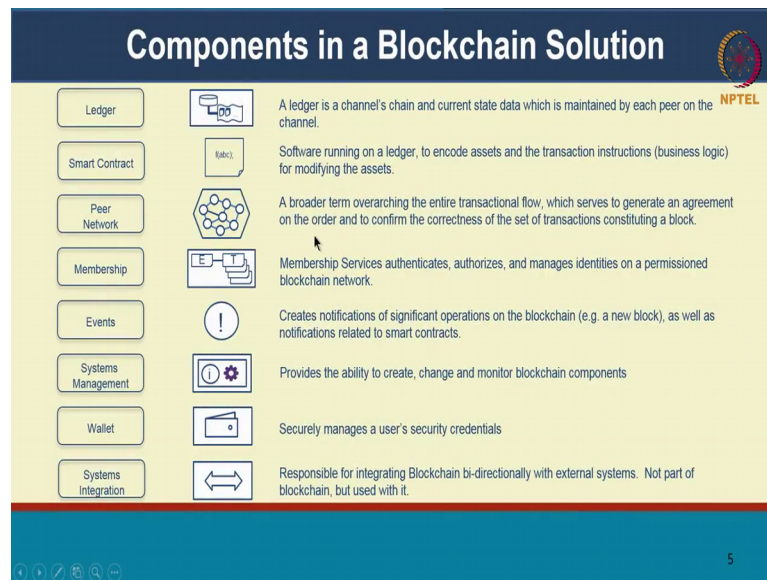
(Refer Slide Time: 02:47)



And yeah I talked about some of the actors here right. So, this again lists the same set of actors. I think the key wants to remember are the architect the developer. So, some of you who are students, who are trying to build a first block chain project, you are probably going to play the role of the architect and the developer, you might also play the role of a network provider, so you might set up the network yourself.

We will look at some of those examples, we you will be setting up a membership service. So, this could be again a separate organization that provides this membership service, so those things are possible. So, all of these can be distinct users and, but it can be one user or one organization can play multiple roles as well.

(Refer Slide Time: 03:28)



So, what are some of the components of a block chain solution right. So, we talked about some of these components, but I think it is worth going through some of them again. So, there is a ledger. So, every node in the block chain network, so this is a peer to peer network. So, every node in the network is going to maintain a ledger of all transactions and these transactions is going to somehow maintain the state of the data that is being stored on the block chain network.

So, this is going to be maintained at each peer, there is a notion of channel we will talk about, but think of your ledger as storing all the state information, its replicated across all the nodes in the network. The smart contract is the business logic that I mentioned about; you can encode your own business logic as code as functions and each invocation of this function becomes a transaction on the block chain. So, those are smart contracts, there is a peer to peer network we talked about that, a membership service, so this is going to provide identities for the users to transact on the block chain and important notion is the notion of events

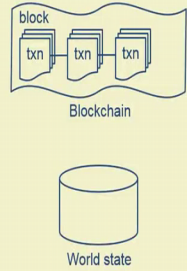
So, whenever a transaction happens on the block chain you can create a event notification. So, the block chain will tell you that this particular transaction has committed, will give you the details of the transaction and you can use that to integrate with existing systems of record. So, you can use that to trigger some other transaction that might be internal to your organization, so all that is possible. So, we will see

examples of how to use some of these events as well. There is big there is going to be notions of systems management, so this block chain network is a distributed system, its actually running across multiple organizations.

So, it requires new ways to create manage and monitor these applications. So, you are going to be monitoring at the system level, at the application level ah, there is a lot of interesting system management questions to be answered. There is a notion of wallet, each user as I mentioned has a digital certificate and its going to be performing transactions using those digital certificates, and there is means to be a place where the user can securely store that private information. So, that digital certificate contains the private identity of that individual. No he should not be sharing that information with anybody else and that is securely managed in a wallet, and I talked about systems integration

(Refer Slide Time: 05:47)

### A ledger often consists of two data structures



- Blockchain
  - A linked list of blocks (a hashchain)
  - Each block describes a set of transactions (e.g. the inputs to a smart contract invocation, output, identities/certs)
  - Immutable – blocks cannot be tampered
- World State
  - Stores the most recent state of smart contracts / output of transactions
  - Stored in a traditional database (e.g. key-value store)
  - Data elements can be added, modified, deleted, all recorded as transactions on blockchain

6

Now, let us look at some of these concepts you know a little more detail. Now the ledger consists of you can think of it as two important data structure. The first data structure is what you would call the block chain itself, so this is the append only log of all transactions that have happened. So, you can think of it as a linked list of blocks and a block as you know is a group of transactions that have been put together. And when I say a linked list of blocks, how are they linked with each other? There is a hash that is computed on each block that is part of the next block. So, this is actually a hash chain,

this is very similar to how Bitcoin also builds its block chain. So, every block of transactions is, there is a cache of it that is computed and that hash is added on to the next block, so that is a hash chain that is getting created. So, this is a pictorial representation, it connects transactions together, but all these blocks are also chained together ah.

And these hash chaining gives you some certain properties of immutability right the. If you modified or tampered with a previous block then the hash on the next block will not match. So, it will be very hard for someone to tamper with previous blocks, and also the fact that these blocks are maintained in a decentralized fashion, means that you will have to take down or manipulate a large net number of nodes in the network, a large fraction of nodes in the network to be able to tamper with this, so it gives you a lot of immutability properties. So, that is the block chain aspect of the ledger itself. So, this is the sequence of transactions that every node in the network is going to maintain

The second data structure is the world state. So, what is the world state? So, this is the information that your smart contracts can work with. It stores the most recent state of smart contracts that are the outputs of transactions. So, in the Bit coin world you can think of the world state as maybe like the account balances. So, the fact that this particular user owns so many bit. The bitcoin block chain does not explicitly store that, but because of the fact that we are looking to handle much more generic smart contracts, we allows the ability for you to maintain arbitrary state information.

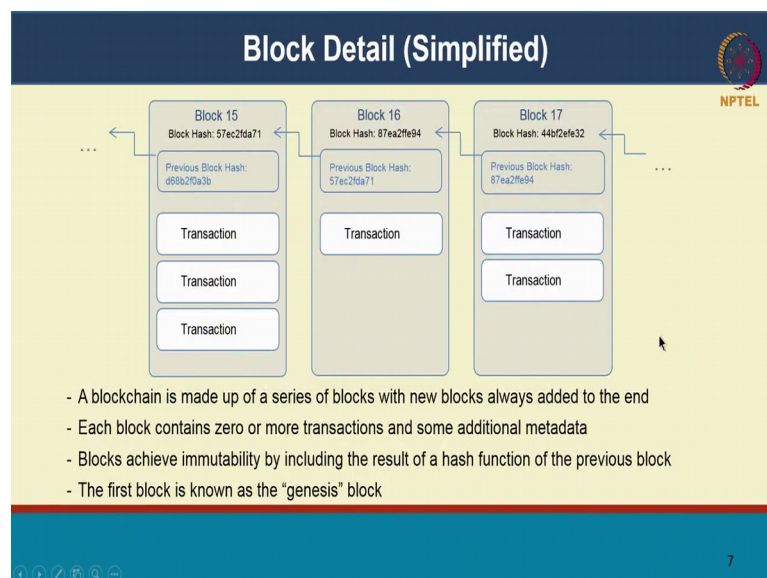
So, these are all stored as key value pairs and we have other databases support as well, but this is, think of it as you go to store some state on the database, each transaction is going to manipulate this state and the output of transactions is also recorded on the block chain ok. So, the transactions include what state was read and what state was written right, so that is the world state that is also going to get main maintained. The other way to think about of the world state is, think of the block chain transactions as modifying data or a period of time ah, it the keeps modifying this data.

The world state think of it as just the most recent view of the block chain itself. The historical values are all there in the block chain, but the most recent values of every piece of information you are storing is stored in the world state, and this allows you to add

modify and even delete elements, so you can go delete and entry in the world state, what this means is a new transaction gets added which marks this data element has deleted.

The previous fact that a data element was created, modified are all part of the block chain that cannot be tampered with, but you can go add another transaction that says this element is now deleted, which means that any transaction that tries to operate on this deleted element will fail. Because, this element does not exist anymore, but the record of all previous transactions on even on the deleted element, all the previous transactions up to the point it was deleted will be a get recorded on the block chain

(Refer Slide Time: 09:25)



So, this shows us the block itself in a very simplified format, what are some of the aspects that you are going to store in the block. Like I mentioned every block has a hash and this hash is getting chained together, so block 15 has a particular hash and that hash is getting stored in block 16. Likewise block 16 there is going to be a particular hash and that hash is in block 17 and so on. So, that is the hash chain itself.

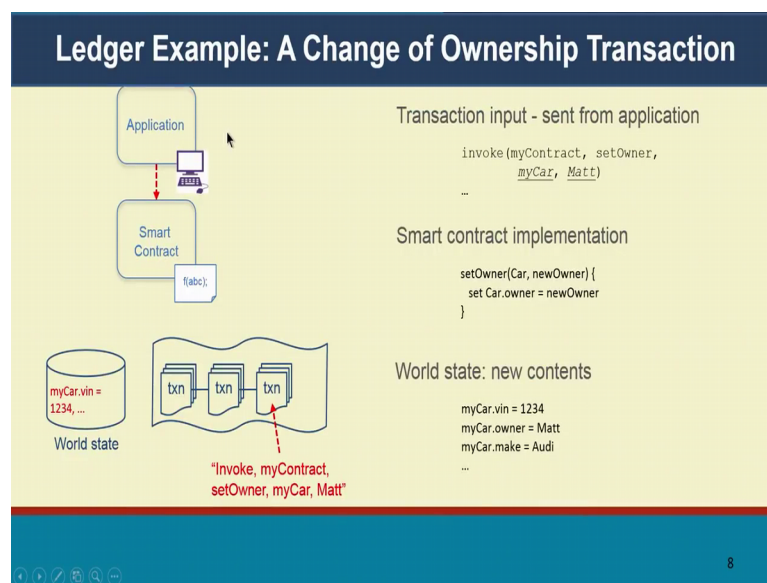
Apart from that the block is a sequence of transactions, it is a set of each block can have multiple transactions in there, and that each transaction is going to say which smart contract got invoked which what was the specific function that got invoked, what are the, which user invoke this transaction will be some notion of their identity which will be there in the in that transaction there will be other details as well that are part of the transaction right.



So, all of this gets embedded into a block ah. Each block can have zero or more transactions, the reason I say zero transactions is, the very first block that you are creating in the block chain is called the genesis block. So, that has special information about the configuration of the network itself who the participants are, and in certain other configuration information.

So, that genesis block does not have any user transactions. So, for that reason it could be zero or more transactions, but all the other blocks in the in the block chain will have multiple transaction. The immutability like I mentioned has achieved because of two properties, because of the hash chaining and also because of the fact that these blocks are getting added, through consensus in a decentralized fashion. So, no one single organization unilateral, unilaterally manipulate this block chain itself

(Refer Slide Time: 11:13)



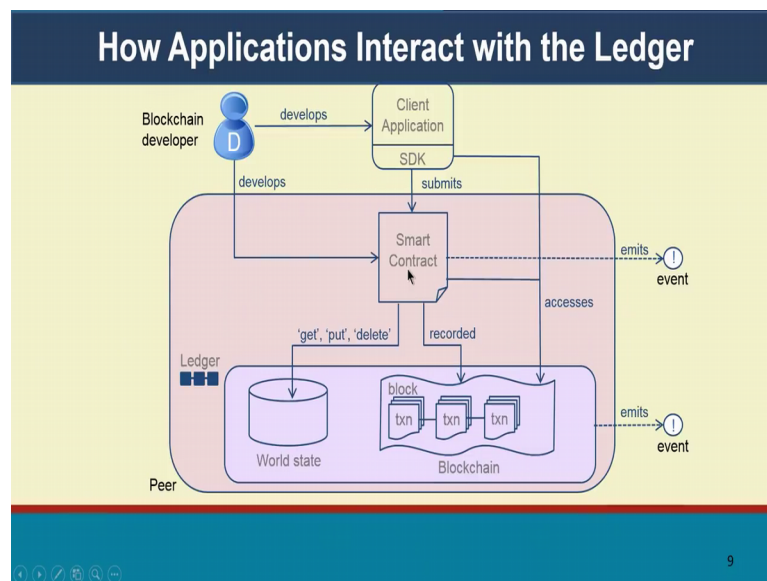
So, the, here is an example of the ledger itself and how our transaction might work and for familiarity sake, we are going to pick just the transfer of ownership itself. This is just the kind of transaction that that Bitcoin does. So, we are going to take the same example, but transactions can be much more complex and can manipulate arbitrary data elements like I mentioned right. So, let us take an example. So, here is a smart contract, it takes as input down here is the smart contract that takes as input, let us say a particular owner we are going to take a car for example, and we are going to set the owner of that car to matt right.



So, there is a set owner function, so in this in, this, this is a very simple smart contract function, all it does is it sets the owner of the car to a new person, so new person in this in this particular example is going to be matt. So, what are the some of the inputs to this function. The input is basically the car itself, so the car data element and the owner of that car where the new owner of that car.

So, these are the two inputs to the to the function and what you are setting is that the car element itself is getting modified, so the car dot owner is getting modified to new order. So, that will be the final output of that transaction, so this car that car object has been modified. Now if you look at the world state, now the my car is now set with these elements. So, this is how the my car object has been modified. So, this is getting stored on in the world state, the transaction that contains this information saying this particular contract was invoked and these are the arguments for that contract, this was the output that got written from that contract. All of this gets written as a block chain transaction in the block chain right itself, it is one of the transactions in the block

(Refer Slide Time: 13:06)



So, now let us look at how external applications could interact with the block chain ledger. So, what are some of the key things right. So, there is a block chain developer, it could be you and me, we are developing both the client application that will interact with the block chain and we are they also developing their smart contract itself. So, this is the business logic that runs inside the block chain in a decentralized fashion. So, in the

previous example that set owner function was this part of the smart contract and that is the one we are writing, and there is a client application that is going to invoke these functions over a period of time. So, the client application is going to be the one that says matt is now the new owner of this car, and we going to set matt as the owner, so we invoke that smart contract with those elements.

So, the client application will use an S D K to submit transactions onto the block chain, and the client application can also separately access the block chain directly, to see whether those transactions are actually committed or also look at historic transactions. So, it, you can go and ask give me all the transactions that happened in block 40 or you can go and ask, for this particular car object, tell me all the changes it has gone through over a period of time right. So, you can get all that history back from the block, like I said every transaction that you are invoking gets recorded on the block chain and all the elements that are getting modified, whether it is a its a read, it is a get or it is a write we call it a put or it is a delete.

So, whatever you are doing on the data elements will get recorded onto the world state and parts of the world state are whatever is getting modified in the world state is going to be part of the transaction, so that is how the block chain itself is constructed. And this is all if we see this all of this is part of a one peer, which means that every peer in the network is going to be performing this function when the client application submits this to the smart contract, that smart contract gets executed on all the nodes. All of the nodes simultaneously will update their world state, will agree that this output is actually valid and this is all consistent across everybody and then the block will get added with that legitimate transaction onto the block chain.

And once you commit the transaction on to the block chain in the enterprise setting we are going to call that a final transaction. So, unlike the bitcoin world, folks in the block chain can happen. In this world there is only going to be one serial sequence of transactions and blocks, there can never be folks and this sequence is, once its written on to the block chain its fine no one can change it and there will be you can get up event emitted out of it saying, this transaction has now simultaneously been committed on all the nodes in the block chain and an event can be generated.

Now, you can use this event in; however, way you want right. So, this can be used internal in an organization to perform additional processing that that organization needs to perform and so on right, so those things can be done.

(Refer Slide Time: 16:01)

### Blockchain Events

- In computing, an event is an occurrence that can trigger handlers
  - e.g. disk full, fail transfer completed, mouse clicked, message received, temperature too hot...
- Events are important in asynchronous processing systems like blockchain
- The blockchain can emit events that are useful to application programmers
  - e.g. Transaction has been validated or rejected, block has been added...
- Events from external systems might also trigger blockchain activity
  - e.g. exchange rate has gone below a threshold, the temperature has gone up, a time period has elapsed...

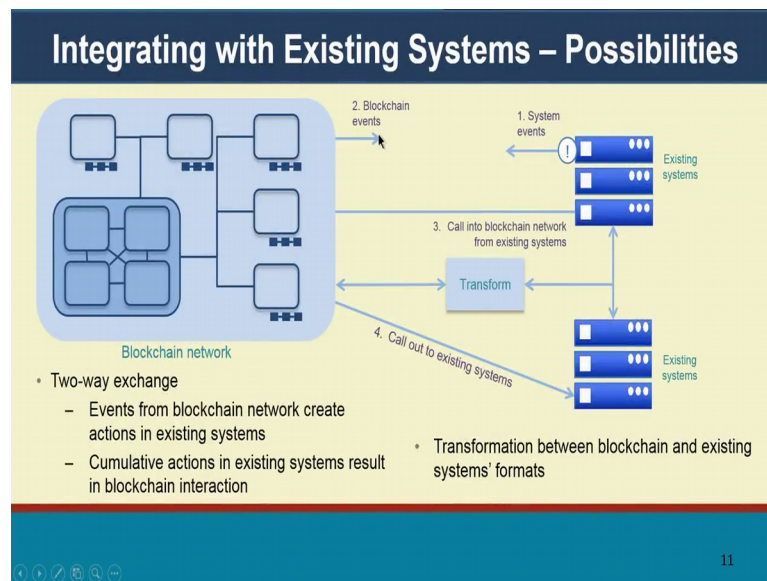
The diagram illustrates the flow of a blockchain event. It starts with a sequence of transactions (txn) represented by document icons. A 'New txn' is added, which triggers a 'New txn event handler' (represented by a document icon). This handler then sends a signal to an 'Application' (represented by a computer monitor) which displays 'Transfer confirmed'.

So, the events again I think I have covered most of these aspects. The I think the important aspect here is this can all be used for asynchronous processing; we do not have to wait for the block chain to commit. So, when the client application submits a transaction onto the block chain, it might take a few milliseconds few hundreds of milliseconds in the enterprise world, we do not have to wait even for that amount, you can be doing your own processing otherwise for other purposes and you can get an event back from the block chain to say this event has now committed. So, that allows you to process things in an asynchronous fashion, you do not have to wait for the block chain to commit, and using that those events you can perform additional processing.

And this event will tell you also even if the block chain if the transaction was rejected for any reason, it could be rejected for a multitude of reasons, it might be that the user who performed this transaction was not authenticated or did not have the right access permissions to perform the transaction or it could be a host of other reasons why the transaction may be may be invalidated and you will get that information as well in the event. If it was rejected the application can always try to perform that transaction again if it feels that it is a legitimate transaction.

Its, you can also integrate other applications to trigger block chain activity directly. For instance you can connect an I O T device to a block chain and whenever the I O T device or let us say a gateway detects that, let us say there is a serious event; that means, everyone to be aware of then it can automatically perform that transaction on block chain and get can get a confirmation back through the events, saying everyone has now been apprised of the fact that this event happened or this particular transaction has been committed

(Refer Slide Time: 17:43)



So, this block chain events are central to integrating with existing systems and when I, when we look at some of the we are building will become apparent why integrating with existing systems is such a key factor, the key part of enterprise applications. Block chain is never going to be operating by itself in a silo, it is definitely going to be exist integrating with existing systems in each of the organizations that are participating in the block chain net, and the integrations can both be in terms of sending information to block chain, as well as getting information out of block chain. So, that is the two way exchange that is, that is explained here ok.

(Refer Slide Time: 18:21)

**Fun Reading**

- Integrating Blockchain with ERP for a Transparent Supply Chain, Infosys white paper: <https://www.infosys.com/Oracle/white-papers/.../integrating-blockchain-erp.pdf>
- Introductory video to Hyperledger Fabric (3 mins): <https://www.youtube.com/watch?v=JuxH9OYXcQQ>
- Hyperledger Fabric Explainer (3 mins): <https://www.youtube.com/watch?v=js3Zjxbo8TM>

NPTEL

12

So, with that we come back to our fun reading section, so the hopefully you have got some of the basic concepts and components of a block chain platform, and here there is a some interesting reading that you can look at. So, there is a nice white paper by Infosys on how you could interact, how you can integrate some of your ERP systems in an enterprise, with a supply chain, it explains some of the importance of integrating systems and how you can automate some of these processes that were all manual previously. In the enterprise world companies have done very well in automating their internal systems, but it when it comes to interacting with enterprises outside of their organization with other enterprises, then lot of the things break down to manual processes paper documents and so on. And block chain uniquely enables you to automate some of those cross organizational processes and some of these integrations are pivotal to them

The next one is a interrupt free video hyper let your fabric, its only 3 minutes. So, over the next four lectures we are going to look at fabric in quite some detail, we will also go through some demos of fabric itself, and here is another video explaining some of the concepts in fabric. So, I encourage you to look at some of these videos and we will meet again at the next lecture to delve deeper into hyper ledger fabric

Thank you.