Blockchain Basic - Key Takeaways

Below you will find a number of key points from this course. Defined terms are underlined.

Week One: Defining a Blockchain

The blockchain technology supports methods for

- a decentralized peer-to-peer network
- a collective trust model among unknown peers
- a distributed immutable ledger of records of transactions.

Decentralization means the network operates on a user-to-user (or peer-to-peer) basis.

A **Distributed Immutable Ledger** means the data doesn't sit on one all-powerful server and the data stored in it cannot be deleted or edited

Transactions bring about transfer of value in Bitcoin Blockchain. The concept UTXO defines the inputs and outputs of such a transaction.

Once a block is verified and algorithmically agreed by the miners, it is added to the chain of blocks, viz., the blockchain.

An **Unspent Transaction Output (UTXO)** can be spent as an input in a new transaction.

The main operations in a blockchain are transaction validation and block creation with the consensus of the participants. Yet, there are many underlying implicit operations, as well.

A **Smart Contract** provides the very powerful capability of “code execution” for embedding business logic on a Blockchain.

Significant innovations such as smart contracts have opened up broader applications for blockchain technology. Private and permissioned- blockchains allow for controlled access to the blockchain, enabling many diverse business models.

In a **Private Blockchain**, access to the Blockchain is limited to selected participants.

Permissioned or Consortium Blockchain has the benefits of a public blockchain with allowing only users with "permission" to collaborate and transact.

Week Two: Ethereum Blockchain

Smart contracts add a layer of logic and computation to the trust infrastructure supported by the blockchain.

Smart contracts allow for execution of code, enhancing the basic value transfer capability of the Bitcoin Blockchain.

Solidity is the high level programming language code for writing smart contracts that run on EVM.

Ethereum Virtual Machine (EVM) is a special structure where code is deployed on after being translated into byte-code.

Accounts are basic units of Ethereum protocol: external owned accounts and smart contract accounts.

An Ethereum transaction includes not only fields for transfer of Ethers but also for messages for invoking smart contract.

Externally Owned Accounts, or EOA, are controlled by private keys.

Contract Accounts, or CA, are controlled by code and can be activated only by an EOA.

An Ethereum block contains the usual prev block hash, nonce, transaction details, but also details about gas (fee) limits, the state of the smart contracts and runner-up headers.

Transaction Validation involves checking the timestamp and nonce combination to be valid, and the availability of sufficient fees for execution.

Miner Nodes in the network receive, verify, gather and execute transactions.

Any transaction in Ethereum, including transfer of Ethers, requires fees or gas points to be specified in the transaction.

Gas Points are used to specify the fees instead of Ether for ease of comparison using standard values.

Miners are paid fees for security, validation, execution of smart contract as well as for creation of blocks.

Week Three: Algorithms and Techniques

Elliptic Curve Cryptography (ECC) family of algorithms is used in Bitcoin as well as Ethereum Blockchain for generating the key pair.

Rivest-Shamir-Adelman (RSA) is a commonly used implementation of public-private key in many applications, except Blockchains because of its need for a more efficient and stronger algorithm.

Hashing transforms and maps an arbitrary length of input data value to a unique fixed length value.

The following are two basic requirements of a hash function.

- make certain that one cannot derive the original items hashed from the hash value.
- make sure that the hash value uniquely represents the original items hashed.

A combination of hashing and encryption are used for securing the various elements of the blockchain. Private-public key pair and hashing are important foundation concepts in decentralized networks that operate beyond the trust boundary.

Asymmetric cryptography uses public-private key pairs to encrypt and decrypt data.

Week Four: Essentials of Trust

A **Merkle tree** is constructed by hashing paired data (the leaves), then pairing and hashing the results until a single hash remains

Proof of work is a protocol that has the main goal of deterring cyber-attacks such as a distributed denial-of-service attack (DDoS) which has the purpose of exhausting the resources of a computer system by sending multiple fake requests.

Well-defined processes for handling exceptions improve trust in the blockchain.

Forks are mechanisms that add to the robustness of the Blockchain framework.

Well-managed forks help build credibility in the blockchain by providing approaches to manage unexpected faults and planned improvements.

Soft fork and hard fork in the Blockchain world is like the release of software patches and new versions of operating systems respectively.

A **Soft Fork** is a fork where updated versions of the protocol are backwards compatible with previous versions.

A **Hard Fork** is a change of the protocol that is not backwards compatible with older versions of the client. Participants would absolutely need to upgrade their software in order to recognize new blocks.

Ommer Blocks contribute to the security of the main chain, but are not considered the canonical "truth" for that particular chain height.

Blockchain Platforms - Key Takeaways

Below you will find a number of key points from this course. Defined terms are underlined.

Week One: Permissioned Blockchains

Permissioned Blockchain allows only nodes with permission to transact and take part in the blockchain operations.

Permissioned blockchain is also known as a consortium blockchain based on its common usecases in specific vertical business domains such as the auto or food services consortiums.

Linux Foundation's Hyperledger is an ecosystem supporting not only blockchain protocols but it also supports the framework and tools for active engagement and collaboration of developers, businesses and other stakeholders.

The goal of the Hyperledger Project is to promote the development of a safe, reliable, efficient, innovative, quality-driven, open-source components and platforms to support enterprise adoption of the blockchain technology.

Hyperledger has **five** frameworks: Fabric, Sawtooth, Indy, Iroha, and Burrow

There are no cryptocurrency In the **Hyperledger** protocol

Chaincode is the smart contract code in Hyperledger that defines a set of assets and provides the functions for operating on the assets and changing their states. It also implements application specific rules and policies.

Hyperledger Fabric is a permissioned business blockchain

Here is the list of services offered by the Fabric:

1. Identity services
2. Policy services
3. Blockchain services and
4. Smart contract services

Identity services module manages the identities of entities, participants, and ledger objects such as smart contracts. In the case of Fabric it is called chaincode.

Policy services module manages access control, privacy details, consortium rules and consensus rules.

Blockchain services module manages

- the peer-to-peer communication protocol,
- distributed ledger maintaining the global state,
- global state replicated at many participants, and
- pluggable consensus algorithm (PBFT, or POW)

Smart contracts services module provides a secured and lightweight sandboxed environment for the chaincode to execute.

APIs allow application programs to call into the underlying services. **SDKs** help in code development based on these APIs. **CLI** is the command line interface for invoking these APIs for testing purposes.

Peers are nodes that initiate transactions and maintain the state of the ledger. There are three types of peer nodes:

1. **Endorsing peers** receive and validate transactions, sign them, and return them to the creating application. They are called endorsers.
2. **Ordering peers** collect signed transactions, order them into blocks, and send them to the committing peers. This is also known as ordering service.
3. **Commiting peers** receive the blocks created by ordering service, validate conditions such as double spending and signatures, and then commit them to the ledger.

Channel provides segregated fabric for a group of entities to transact privately. Channels also provided the ability to support multi-lateral transactions among competing businesses and regulated industries through cross-chain chaincode.

An **identity** determines the role of the entities and the permissions they have for accessing the resources in the blockchain network.

Consensus is the agreement on the next block of transactions to be added to the chain and the extensive validation and verification of the order and correctness of the transactions including double spend and other conditions.

Microsoft Azure's main goal is to accelerate blockchain deployment. Azure BaaS features include:

- A Collection of ready to deploy ledgers
- Blockchain network with multiple nodes, with hashing, mining, the consensus among the nodes, and the distribution of replicated ledger to all nodes
- Preconfigured network configurations for developing business logic
- Tools and infrastructure in a single place
- Data security and scalability of the cloud platform and
- Single Node Ledger and Multi Node Ledger

Week Two: Decentralized Application Platforms

Augur is a trustless, decentralized prediction market platform based on blockchain technology.

Roles participants can play in the prediction market:

1. Market creator who places the prediction query, sets the expected outcomes, pays fees and escrows, establishes the rules, and designates the initial set of reporters.

2. Trader who places the bets on the expected outcomes and takes part in the pre-reporting phase of the process. Traders buy and trade shares that bet on the odds of the outcomes. The trading currency is currently ETH.

3. Reporter who reports on the outcomes. Understand that outcomes do not have to be binary (Yes or No). The reporter can be a designated reporter or an open reporter based on the phase of the process.

Grid+ is a Dapp platform implemented on Ethereum blockchain that has created an energy ecosystem by integrating blockchain and AI.

Energy Retailer: Grid+ will operate as a commercial electricity retailer in deregulated markets.

Smart Agent: At the user household, Grid+ smart agent is a computing device that hosts the software for the blockchain transactions, multi-signature crypto-wallet, with PKI security and off-chain payments for faster confirmations.

Intelligent electricity usage: Electricity trading is a complicated process with many intricacies; Grid+ manages these by coding the efficient price options using smart software.

ERC-20 Token payments: A special ERC20 compliant token called BOLT has been created for payment purposes.

Integration to IOT devices: A Smart agent can be integrated into other intelligent agents such as NEST and electric batteries (Telsa Powerwall)

Remote control: Grid+ enables integration of mobile phones and computing devices to allow remote control of its operation.

Week Three: Challenges and Solutions

In **Proof of Stake (POS)**, the full node with the most at stake or most coins is chosen for adding the next block. That is why is it is called Proof of Stake. The idea is that the node with most at stake will not be malicious and risk its stake for forking the network.

In **Practical Byzantine Fault Tolerance (PBFT)**, nodes vote to elect a leader, and that leader adds the next block to the chain. This leader adds the block of validated transactions.

Scalability is the ability of a system to perform satisfactorily at all practical levels of load. Load in the context of the blockchain could be: transaction times, number of nodes, number of participants and accounts, and other attributes of the blockchain.

Escrow is “a contractual agreement in which a third party receives and disburses money or documents for the primary transacting parties, with the disbursement dependent on conditions agreed to by the transacting parties..”

Week Four: Alternative Decentralized Solutions

IPFS is a decentralized model for file transfer in contrast to the centralized namespace and transfer provided by the http family of protocols.

Bitswap protocol manages the block exchanges involving the nodes accordingly.

Hashgraph is a trust model that provides a consensus layer that addresses the transaction latency, and energy wastage, fairness and also provides a computationally strong algorithm for Byzantine Fault Tolerance, and eventual consistency. Forks are mechanisms that add to the robustness of the Blockchain framework.

Elements of the hashgraph include:

- Event
- Transactions
- Directed acyclic graph: DAG The hashgraph
- Witness
- Famous Witness
- Round: Round Created, Round Received
- Consensus by voting by the witnesses of the next round and
- Gossip protocol

A **round** consists of all the events between the oldest participant and youngest participant of the round.

Bitcoin has had a butterfly effect on technology, its concept has opened up a Pandora's box of technology and efforts ushering in a technological and possibly a social “revolution.”

Decentralized Applications - Key Takeaways

Below you will find a number of key points from this course. Defined terms are underlined.

Week One: Decentralized Applications (Dapps)

A Dapp, or decentralized application, solves a problem that requires blockchain services and blockchain infrastructure for realizing its purpose.

Blockchain server represents the infrastructure and the functionality the blockchain provides.

The design of a Dapp has a ____ front-end, a blockchain back-end and the code connecting the two.

API or Application Programming Interface is a convenient and standard way to expose a set of functions related to a specific data set and services.

An API publishes a set of functions or methods that can be used programmatically to invoke operations, access data and store data.

Two well-known examples of API are:
- Twitter API to access tweets that can be filtered by query terms
- Google Maps API that allows for applications to “embed” the map features such as geolocation in their own applications, leveraging and reusing the power of google map API.

There are two major categories of APIs:

- The first one is for management APIs that includes admin, debug, miner, personal, and txpool. They support methods for management of the geth node.
- The second one is the Web3 APIs: web3, eth and net. They support methods for development of Dapps.

Week Two: Truffle Development Environment

Applying the general design process requires five steps:

- Step 1: Design the Ballot.sol;
- Step 2: Illustrate modifiers with just one modifier “onlyOwner” referring to the qualified person as the chairperson. Recall that in the design of a smart contract you can use modifiers to represent global rules.
- Step 3: Add a tester code for the required problem specification and run the tests to make sure they all pass.
- Step 4: Add the user interface component and
- Step 5: Test the complete application by interacting with the interface.

Basic Truffles Commands:

- Truffle Init: Initializing a template or base directory structure for a Dapp
- Truffle compile: Smart contract compilation
- Truffle develop: Personal blockchain for testing with a console
- Truffle migrate: Migration scripts for deploying smart contracts
- Truffle test: Test environment for testing the deployed contract

Metamask will link to the local blockchain created by Truffle to manage the accounts, acting as a bridge between the application front-end and the blockchain node that hosts the accounts.

Node.js which will serve as your web server for the Dapp front-end

Smart contracts are like your hardware chip. Once deployed, they are final and cannot be updated unless special provisions or escape hatches are built-in.

Positive tests - making sure, for a given valid input, it performs as expected. We will test the complete ballot cycle of deploy-register-vote-winningProposal

Negative tests - making sure it handles invalid inputs or situation appropriately. We will code only 2 of the many negative tests possible.

test.js has 4 positive tests and 2 negative tests.

Asserts in the positive tests have three parameters: expected value, actual value, a string representing the test being performed.

Contracts has the solidity contracts

Migrations has the migration scripts

Test has the test scripts

Build has the json artifacts generated by the compile process

Src has the web assets such as js, css, index.html

Node_modules has the node.js modules

The **json** files and **js** files are configurations files.

Week Three: Improving the Smart Contract Design

Memory is transient memory in RAM

Storage refers to the persistent store in a permanent storage device like your hard drive.

Solidity provides a function called **“selfdestruct”** to delete or kill a smart contract.

Libraries are special smart contracts with no Ether balance, no payable functions, and no state to be stored on the blockchain.

A smart contracts ownership may change or deleted

Coin sc has two data items.

- The first one is the address of the minter or owner;
- The second is the mapping between the account address and the COIN balance

Coin has a mint function that only the minter can execute to mint new coins for a given address. You have to be in Account1 in Metamask or minter’s account to mint.

Coin has a transfer function that can transfer Coins from one account to another. You have to be in Sender’s account in Metamask to be able to transfer.

It is possible, a complex application could be composed of many smart contracts; one instantiating the other or inheriting others

A transfer function may be used to implement a transfer in ownership of the Smart contract.

Oraclize is described as a data carrier between the web resources (APIs, and URLs) and a smart contract.

usingOraclize is a smart contract that provides minimally a query function to access external sources.

Week Four: Application Models and Standards

Initial Coin Offering (ICO) uses a blockchain to record the distribution of the Coin, receive funds, specify rules, and to enforce any preconditions and policies.

Token is similar to a Dapp Coin, but its offering is typically associated with an asset or utility.

Decentralized Autonomous Organization (DAO) is an investment instrument deployed as a smart contract on Ethereum blockchain whose main goal is to showcase an autonomous organization without traditional corporate governance structure for decentralized, anonymous fundraising and automatic investing.

The **Decentralized Marketplace** facilitates meetings of sellers and buyers.

Fintech, short for Financial Technology, has great potential for innovative Dapps whose domain ranges from decentralized investment instruments to micropayment channels.

EIP or Ethereum Improvement Proposal is a means to manage the protocol specification, improvements, updates, client APIs and contract standards.

EIP handles issues in four different categories, including:

- Core or core ethereum protocol
- Network or network level improvement
- Interface or Interfaces such as ABI, RPC and
- ERC or Application level convention and standards

Ethereum Request for Comments (ERC) is a solution draft proposed by the ERC document and discussed on the github, gitter and sub-reddits community.