Agenda of the Day

1. Smart Contract intro
2. We will play with the Python Code for Mining and also understand our own python code for Blockchain Arch.
3. Ethereum end to end theory
4. Few resource we will see for ethereum
5. Metamask setup
6. Discord Link for you all

Intro of the Python Script which we developed

1. Mining
   a. Does increasing the Number of 0 in the beginning will increase the time yes or no

Assignment 4 - Will be graded

Test mining for 10 Zero and submit the Python Notebook file for grading and check how much time does it takes

How you can develop Apps/ dAPPs on Blockchain ?

Ethereum -

- Cryptocurrency
- Smart contracts
- 2015 it got formulated
- It is 2nd widely used Cryptocurrency after Bitcoins
- You can use SOLIDITY Programming Language for Deploying Smart Contracts here

Bitcoin vs Ethereum

1. Bitcoin is calculator & Ethereum is your Smartphone
2. Bitcoins are only for Currency purpose but Ethereum is for storing smart contracts
3. Ethereum has got https://entethalliance.org/

What is Solidity Programming Language ?

- Solidity is a programming language which is used for
- Creating smart contracts
- It is still development phase
- Statically typed language
- Extension of the Solidity programs are - .sol
- We can build our own Solidity code in Remix IDE or Visual Studio code.
- http://remix.ethereum.org/ - IDE for smart contract Development

- **Ethereum**

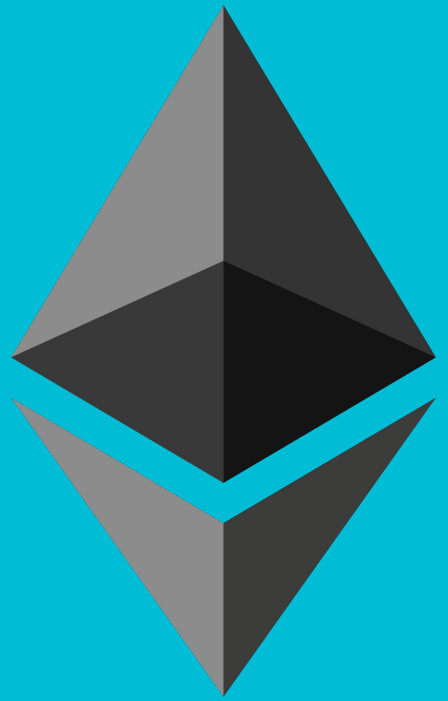
- Hyperledger Fabric / ChainCode
- R3 Corda
- Blockchain Database

Smart Contracts

- **Whats a Contract ?**
  - A legal bond which makes sure that, some assets is on someone name, or a legal bond with some conditions

- **What Are smart Contracts ?**
  - A Contract which is deployed on a Blockchain Network as an asset record, which also has a power to automatically change the ownerships/ attributes when some certain case is achieved

INTRODUCTION TO ETHEREUM



Please

Hold complex questions until the end of the presentation!

1.

WHAT IS A BLOCKCHAIN?

A brief overview of a fairly complex topic!

Some portions of the topic have been simplified

This presentation has been edited for content and to fit your screen!

FIRST, WHY DO WE CARE ABOUT BLOCKCHAINS?

A blockchain allows for trustless transactions between multiple parties.

Or, more importantly, it allows transactions without trust of a third party intermediary!

BLOCKCHAIN =

DISTRIBUTED LEDGER
+ CONSENSUS

BLOCKCHAIN =

DISTRIBUTED LEDGER + CONSENSUS

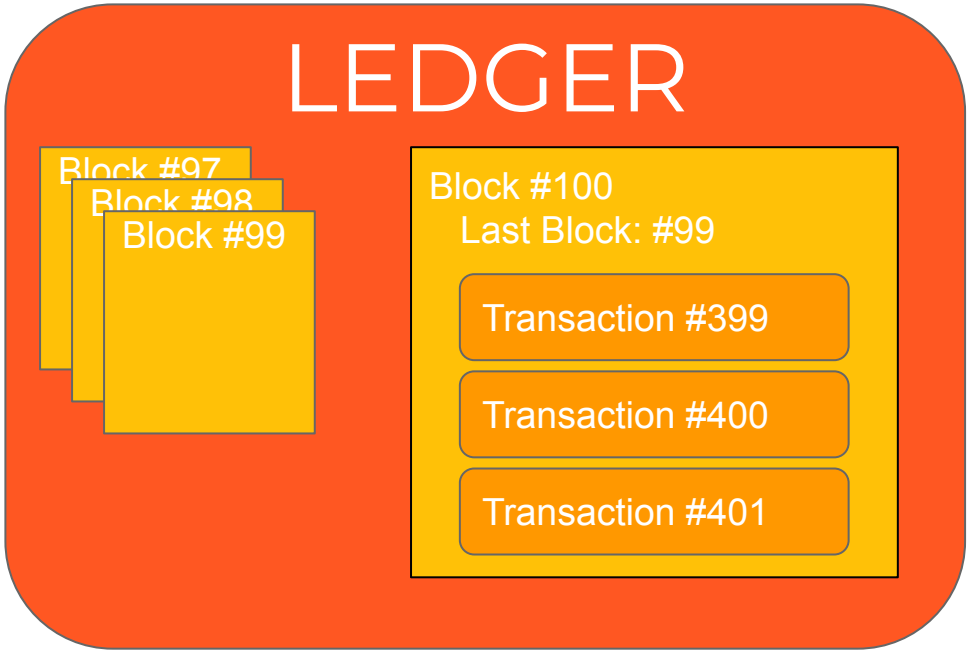
A list of transactions between accounts (a ledger) are stored on distributed “nodes”. New transactions are periodically added into a block. Nodes use an agreed upon protocol to reach consensus on when a new block is appended to the chain of previous blocks.

LET'S START WITH A TRANSACTION THAT OCCURS BETWEEN ACCOUNTS

An **example transaction** could be:

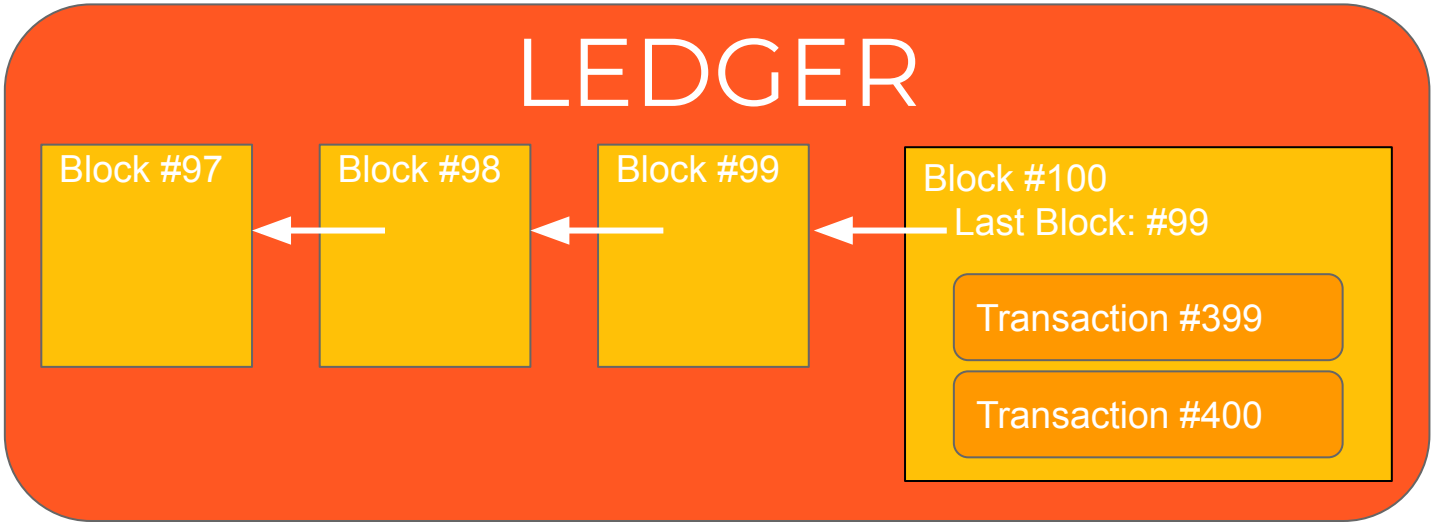
Account A will send 10 tokens to Account B

A LEDGER IS A LIST OF TRANSACTIONS (GROUPED INTO BLOCKS)



Blocks contain an indeterminate number of transactions

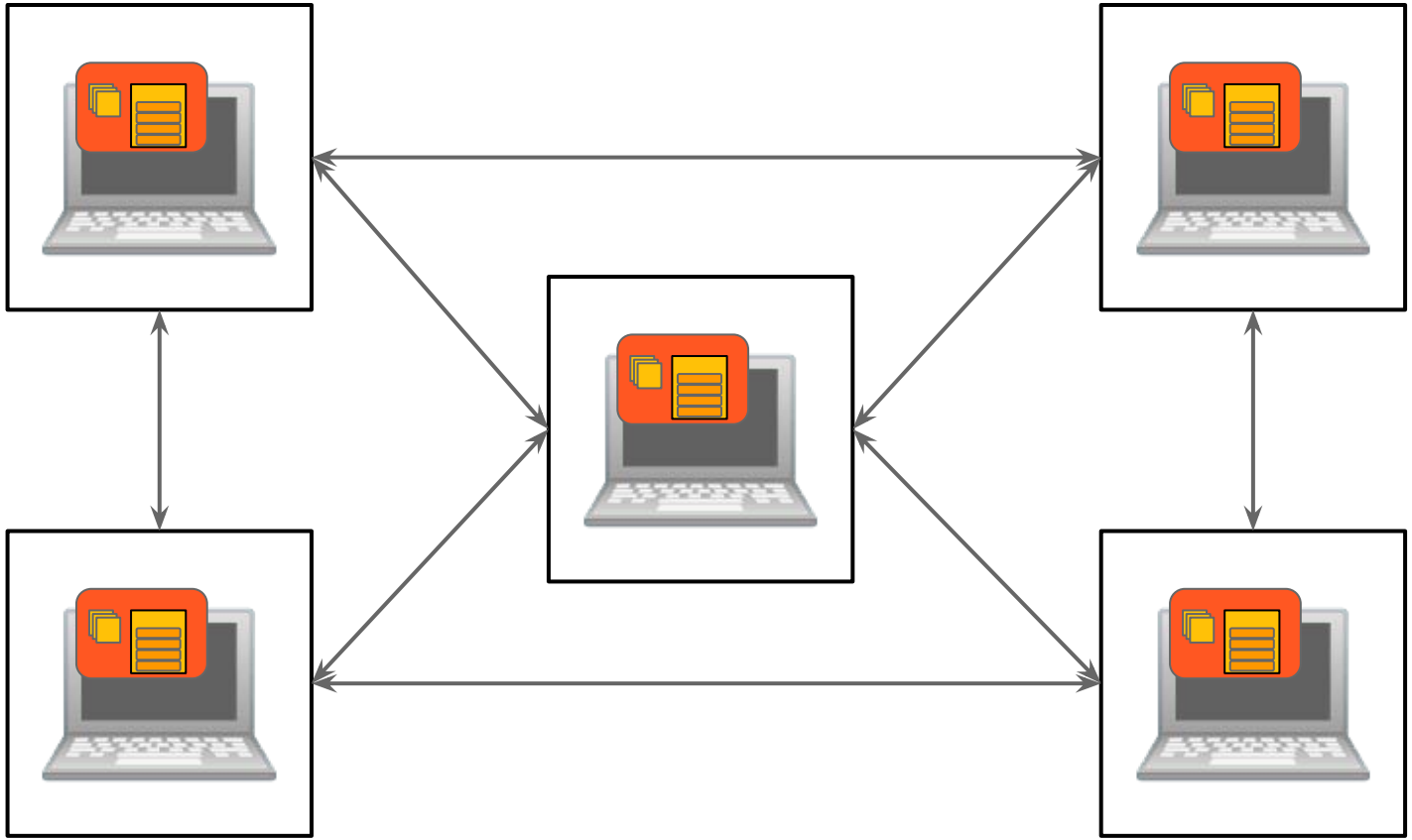
BLOCKS ARE CHAINED TOGETHER



The ledger is a chain of blocks! Each block is created with a pointer to the previous block creating a blockchain!

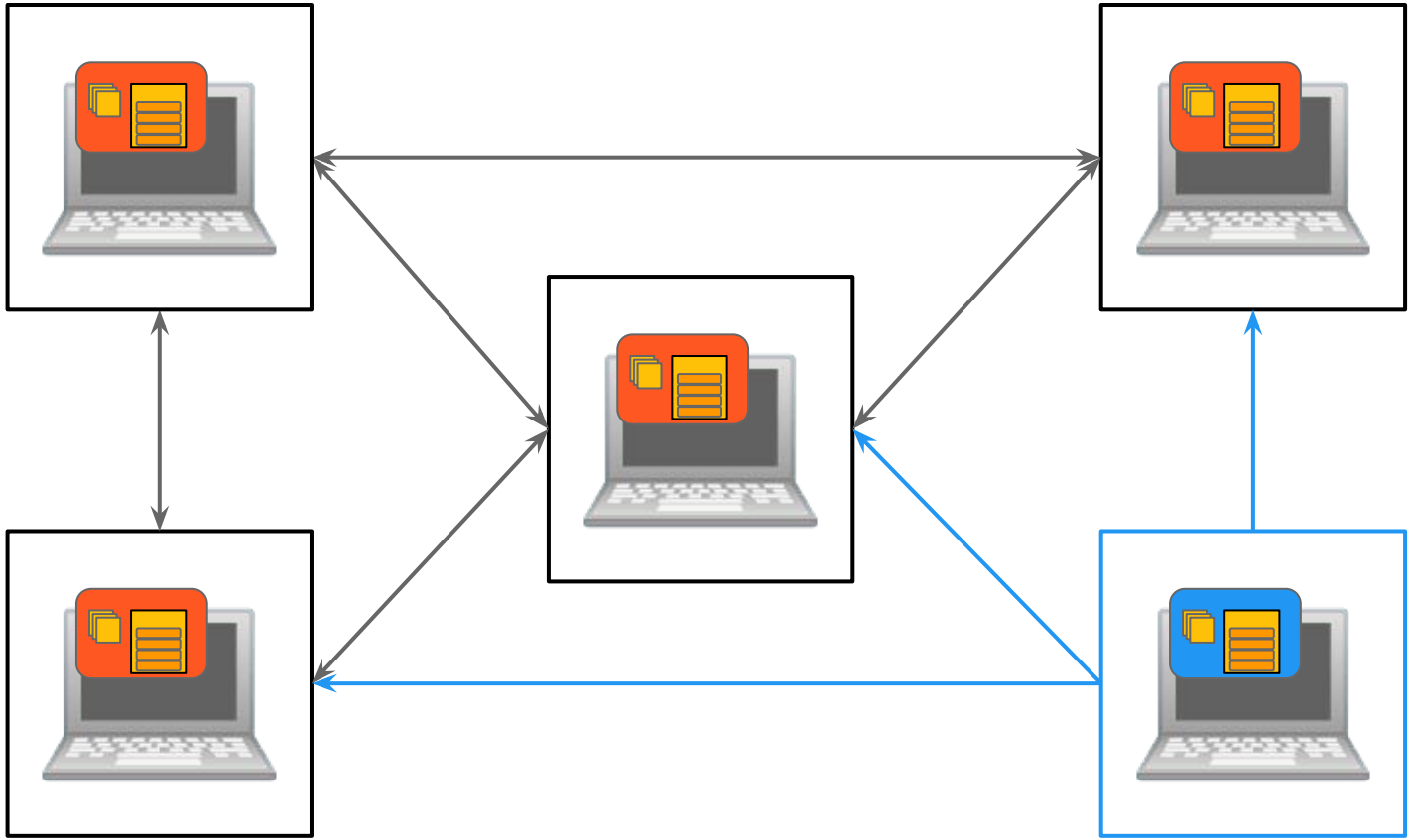
Blocks are generated on a time interval (e.g. every 5 minutes)

THE LEDGER IS COPIED AND DISTRIBUTED AMONG NODES



A node is a server (peer) running on a device

EACH NODE HAS A COPY OF THE LEDGER AND AT LEAST ONE OF THEM WILL CREATE THE NEXT BLOCK!



One node creates the next block according to a set of rules

EXAMPLE CONSENSUS PROTOCOL: PROOF OF WORK

This is only one example of a consensus protocol

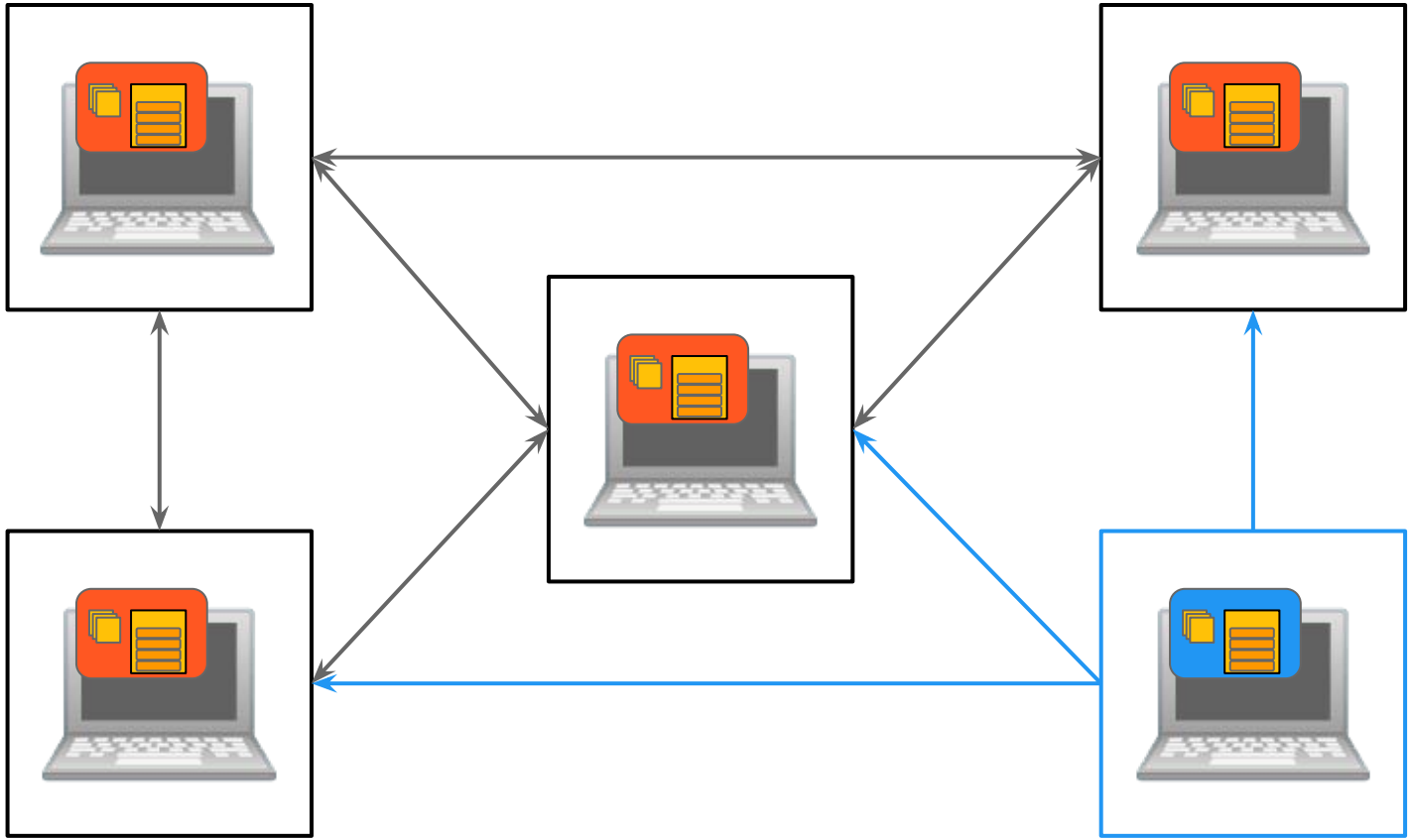
Different methods or protocols exist for distributed nodes to reach consensus

Assignment - 5 - Checked by LU

Difference B/W

Proof of stake and Proof of Work

NODES ATTEMPTING TO CREATE NEW BLOCKS ARE USUALLY CALLED “MINERS”



The first Miner to solve a hard math problem creates the next block and is rewarded

THE MAJORITY OF PoW BLOCKCHAINS INCENTIVIZE MINERS WITH REWARDS

E.g. Bitcoin

Miners currently receive 12.5 BTC plus all included transaction fees

New block:
Every ~10 minutes

E.g. Ethereum

Miners currently receive 5 ETH plus all included gas fees (more on this shortly)

New block:

Every ~15 seconds

A BRIEF HISTORY OF BLOCKCHAINS

- **2008**: Bitcoin and blockchain idea gifted to world by “Satoshi Nakamoto”
- **2009**: Bitcoin client released (open source)
- **2011**: Litecoin, first “altcoin,” released (based on bitcoin source code)
- **2014**: Ethereum whitepaper released/crowdsale
- **2015**: Ethereum “Frontier” launched

2.

INTRODUCTION TO ETHEREUM

WHAT IS ETHEREUM?

WHAT IS ETHEREUM?

OPEN SOURCE

Like Bitcoin, Ethereum is a public blockchain no one controls or owns.

WHAT IS ETHEREUM?

OPEN SOURCE

Like Bitcoin, Ethereum is a public blockchain no one controls or owns.

PROOF OF WORK CONSENSUS

The Ethereum Whitepaper specifies the PoW rules and 4+ major clients exist and run the “nodes”.

WHAT IS ETHEREUM?

OPEN SOURCE

Like Bitcoin, Ethereum is a public blockchain no one controls or owns.

PROOF OF WORK CONSENSUS

The Ethereum Whitepaper specifies the PoW rules and 4+ major clients exist and run the “nodes”.

ETHEREUM VIRTUAL MACHINE (EVM)

Transactions are *more than just values*, but “Turing” complete programs that run when blocks are processed by nodes.

ETHEREUM VIRTUAL MACHINE

General Purpose

Supports Bitcoin-like value transactions (e.g. send 10 ETH from account A to B) or more complex applications.

ETHEREUM VIRTUAL MACHINE

General Purpose

Supports Bitcoin-like value transactions (e.g. send 10 ETH from account A to B) or more complex applications.

Smart Contracts

Turing complete languages (e.g. *Solidity*) allow the Ethereum transactions to be programmed to do different operations.

ETHEREUM VIRTUAL MACHINE

General Purpose

Supports Bitcoin-like value transactions (e.g. send 10 ETH from account A to B) or more complex applications.

Smart Contracts

Turing complete languages (e.g. *Solidity*) allow the Ethereum transactions to be programmed to do different operations.

Decentralized Apps (DApps)

GUIs for smart contracts allow users to interact in ways similar to web 2.0 (HTML/JS/CSS).

THINK:
WORLD COMPUTER

The Ethereum blockchain is the first
“decentralized world computer” to ever exist!

TRANSACTIONS AND CONTRACTS

Two types of Accounts

- **Externally owned accounts** (controlled by people/keys similar to Bitcoin)
- **Contract accounts** (controlled by smart contract code)

TRANSACTIONS AND CONTRACTS

Transactions can include more than just value transfer; they can include programming or bytecode that does things (smart contracts)

Transaction #3512
If (Account A == Member) && (Date >= 4Q):
Dividend = 25% the current value of Contract C:
Transaction:

Bare with me! It's okay if you don't fully "get" this program.

TRANSACTIONS AND CONTRACTS

Submitting transactions and contracts to the blockchain has an associated “gas” cost paid in Ether based on the complexity of the operations.

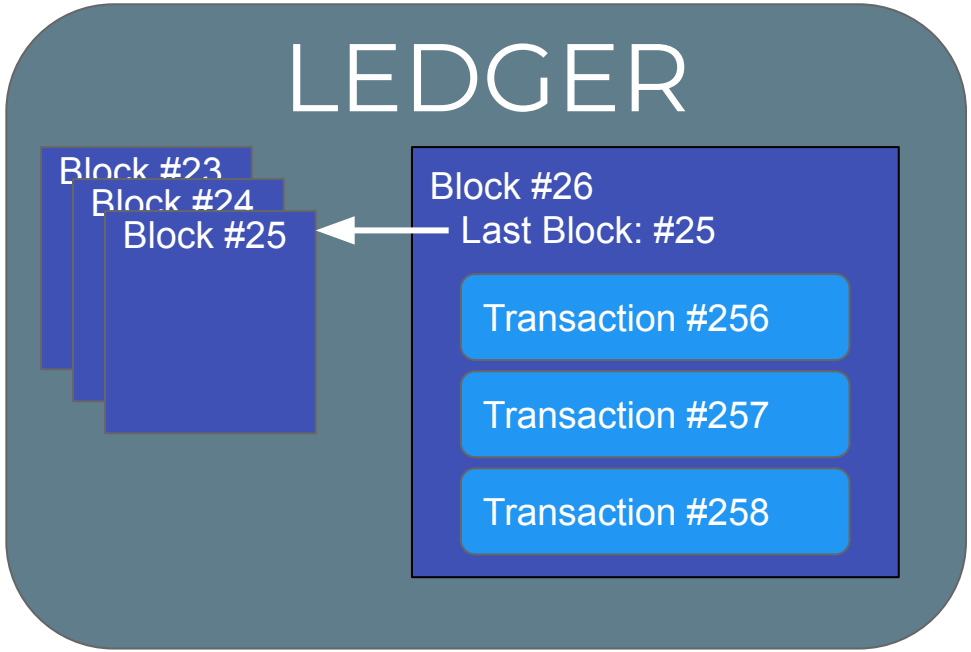
Transaction #256
If (Account A == Member) && (Date >= 4Q):
Dividend = 50% the current value of Contract C:
Transaction:

IF	+70 Gas
AND	+30 Gas
EQUAL?	+40 Gas
DIVIDE	+120 Gas
SEND	+20 Gas

TOTAL COST = 280 Gas

Ethereum gas cost changes with time just like with gasoline

TRANSACTIONS AND CONTRACTS



Transactions are written to blocks and mined just like other blockchains

TRANSACTIONS AND CONTRACTS

Think of the Ethereum Ledger like a SPREADSHEET

Cells can just have a ***value*** or they can ***give the result of a macro/script.***

(Ignore this if you're not the CPA type!)

Trans-action	Account A	Account B	Account C (Contract)
#256	10	10	0
#257	5	5	10
#258	10	5	Dividend(A)

THIS IS AN EXAMPLE SMART CONTRACT WRITTEN IN SOLIDITY

```
contract MyToken {
    /* This creates an array with all balances */
    mapping (address => uint256) public balanceOf;

    /* Initializes contract with initial supply tokens to the creator of the contract */
    function MyToken(
        uint256 initialSupply
        ) {
        balanceOf[msg.sender] = initialSupply;  // Give the creator all initial tokens
    }

    /* Send coins */
    function transfer(address _to, uint256 _value) {
        if (balanceOf[msg.sender] < _value) throw;       // Check if the sender has enough
        if (balanceOf[_to] + _value < balanceOf[_to]) throw; // Check for overflows
        balanceOf[msg.sender] -= _value;                 // Subtract from the sender
        balanceOf[_to] += _value;                        // Add the same to the recipient
    }
}
```

Bare with me! It's okay if you don't fully "get" this program.

DAPPS

DECENTRALIZED APPLICATIONS

USE CASES FOR DAPPS AND PROJECTS IN DEVELOPMENT

Decentralized Exchange

Convert between crypto-currencies and tokens

EtherDelta, EtherEx

Prediction Market

Utilize the wisdom of the crowd to predict future events

Augur, Gnosis

Distributed Computing

Enable users to lease out spare compute cycles (think: uber for your computer, dAWS)

Golem

Identity Management

Retain ownership of your online identity, metadata, and relationships.

uPort

Crowd Sale Platform

Create a custom token for trade, payment, customer loyalty, etc. and take it to market.

Ethereum, Firstblood

Digital Asset Management

Manage, buy, sell physical objects cryptographically tied to digital tokens.

Digix

See more upcoming DApps at:

Dapps.Ethercast.com

USE CASES FOR DAPPS AND PROJECTS IN DEVELOPMENT

Internet of Things

Enable IoT devices to interact and exchange value with each other

Slock.it

Digital Governance

Blockchain-chartered companies, voting / election monitoring and transparency

Otonomos, Colony

Energy Management

Buy and sell energy to the grid or neighbors directly from your solar panels

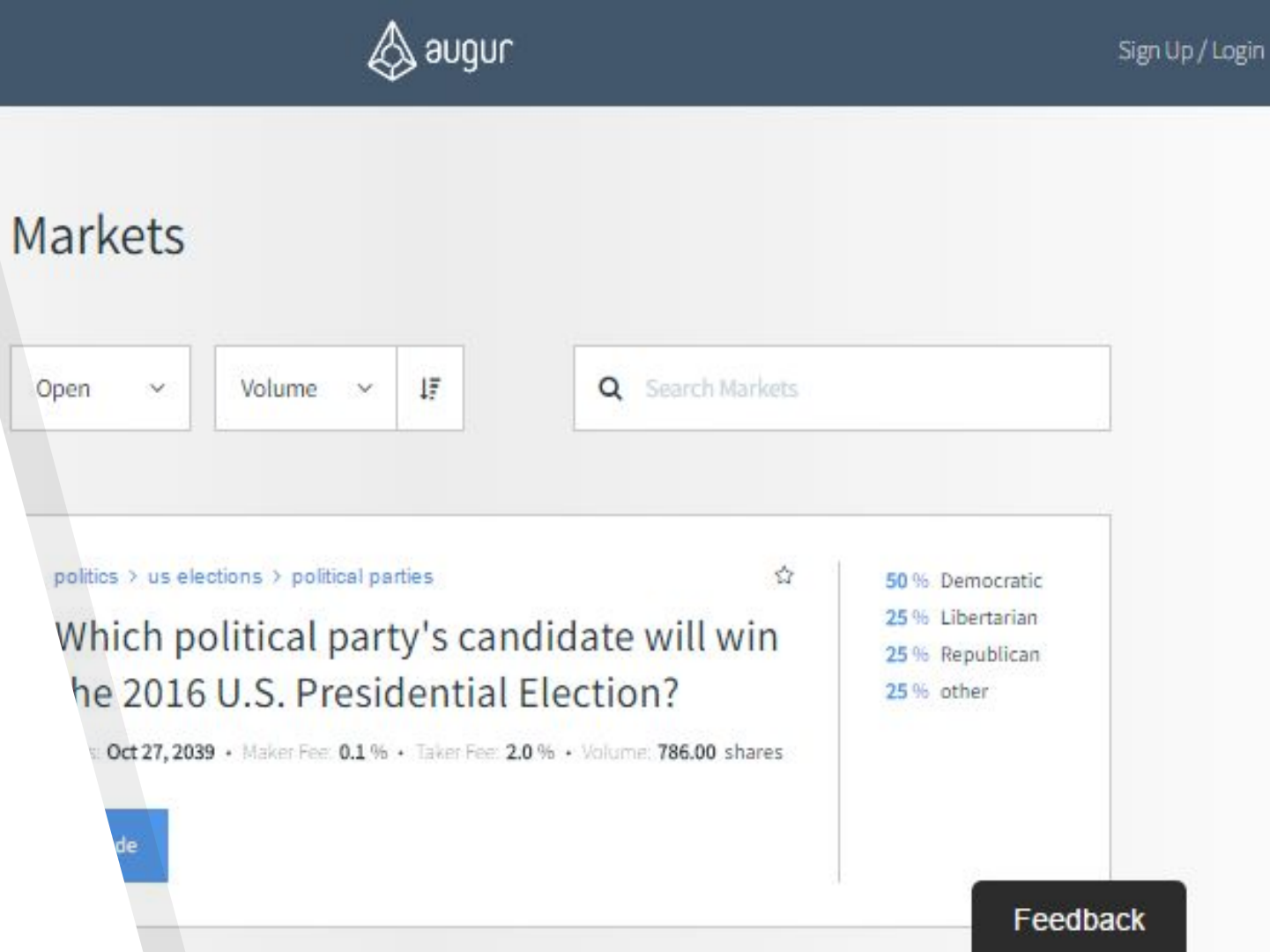
Transactive Grid

And many more to come...

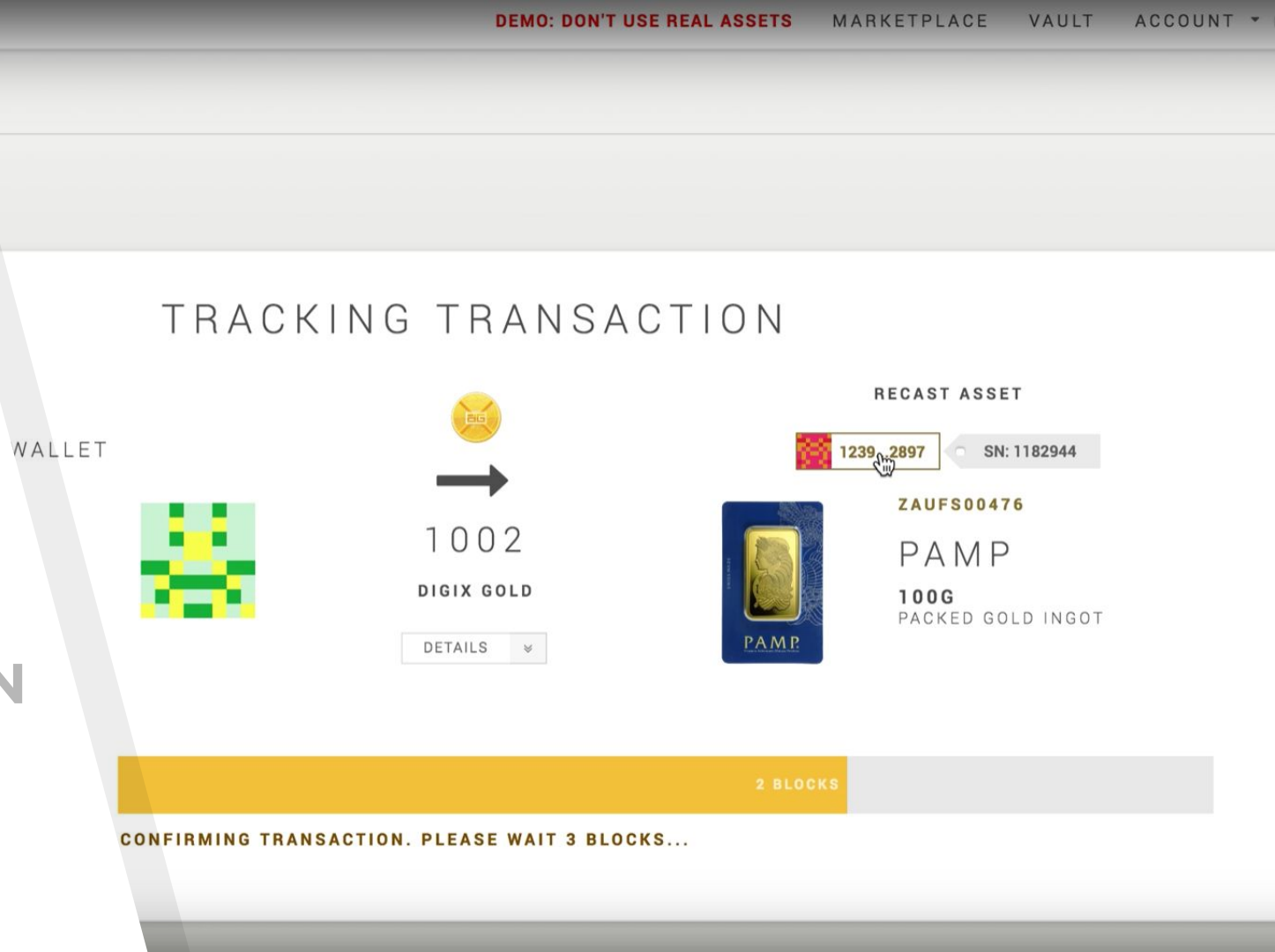
See more upcoming DApps at:

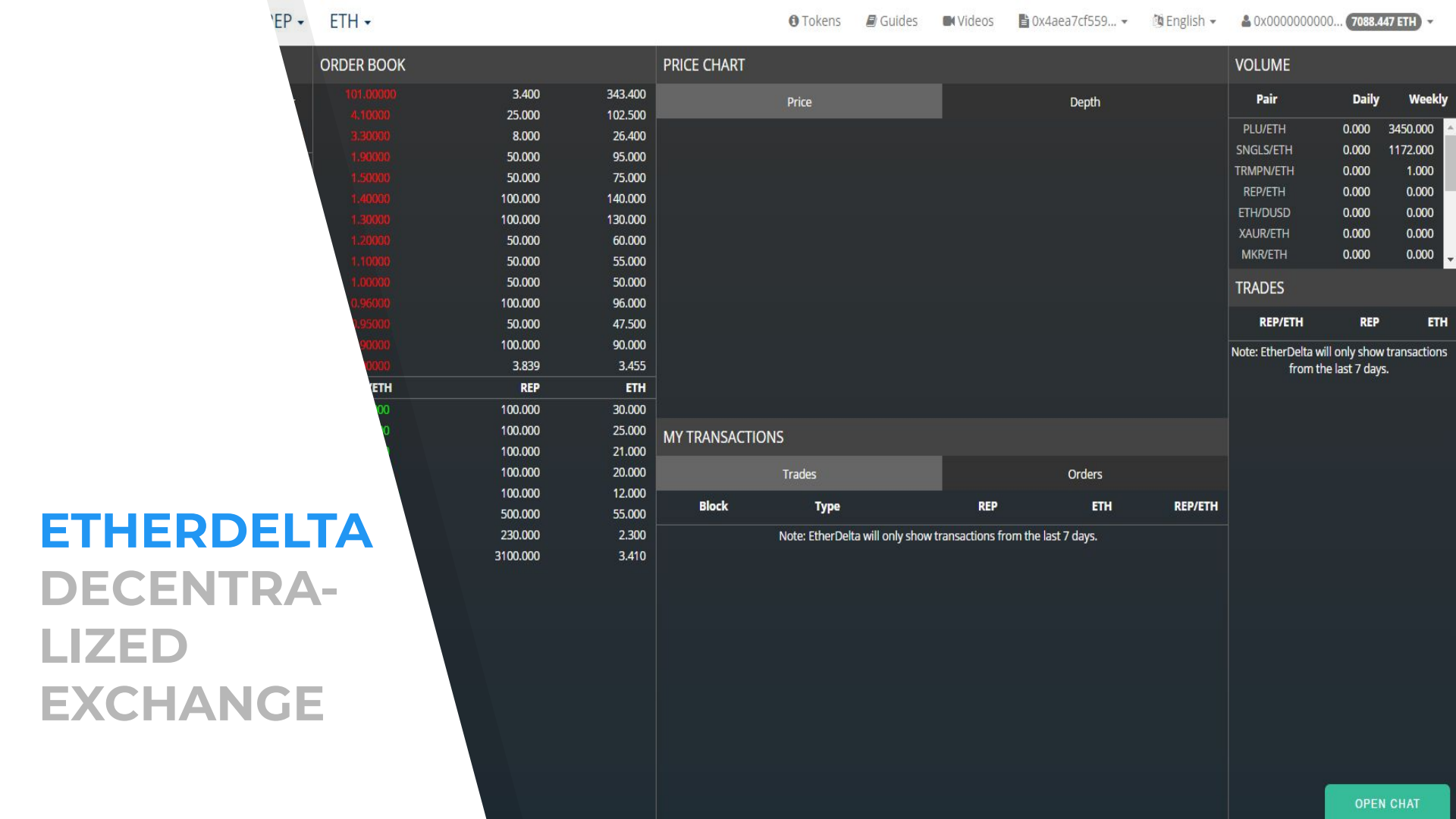
Dapps.Ethercast.com

AUGUR
PREDICTION MARKET



DIGIX
BLOCKCHAIN DIGITAL ASSETS





ETHERDELTA

DECENTRA-LIZED EXCHANGE

3.

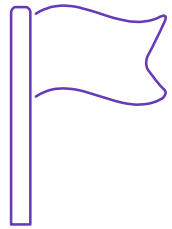
ETHEREUM'S FUTURE FEATURES & ROADMAP

ETHEREUM **ROADMAP**

Past	Present	Future
Olympic Testnet	Homestead	Metropolis
Launched May 2015	Launched Pi Day 2016	TBA (2017/2018)
Frontier		Serenity
Launched July 2015		TBA

Each Ethereum release is a hard fork.

Ethereum has forked 5 times. It will fork again for EIPs with consensus.

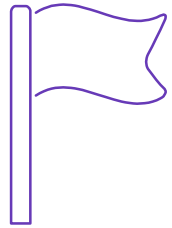


CASPER

PROOF OF STAKE

No more need for mining.

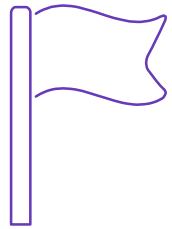
Waste less electricity!



SWARM

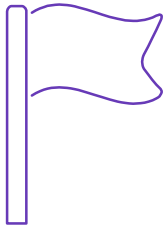
FILE STORAGE

Decentralized file storage system directly built-in to Ethereum



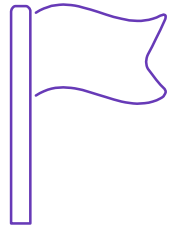
WHISPER MESSAGING

Secure communications whether you're talking to a friend, or need one contract to talk to another quickly.



RAIDEN
STATE CHANNELS

Supports off-chain transactions, with
on-chain reconciliation.
(For all ERC-20 compatible tokens, too!)



SHARDING

TRANSACTION GROUPS

Increase simultaneous transaction handling exponentially.

Read about Types in Ethereum and share the summary in community and submit link for LU Team to check