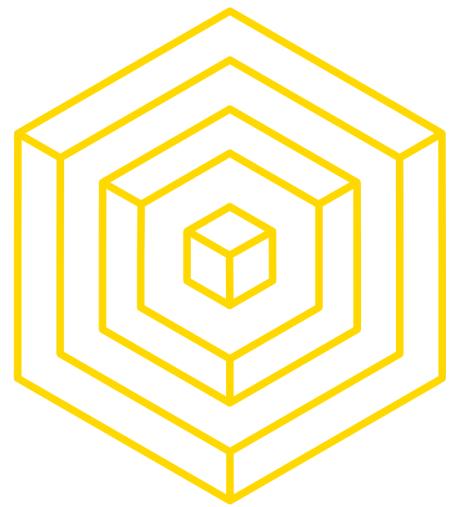


ETHEREUM AND SMART CONTRACTS: ENABLING A DECENTRALIZED FUTURE

Brian Ho
Gillian Chu

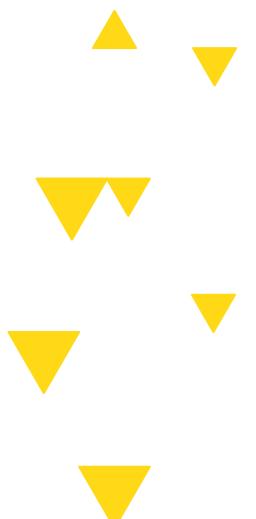


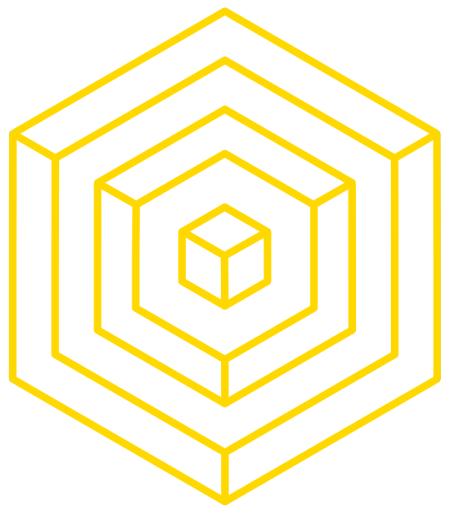
BLOCKCHAIN
AT BERKELEY



LECTURE OVERVIEW

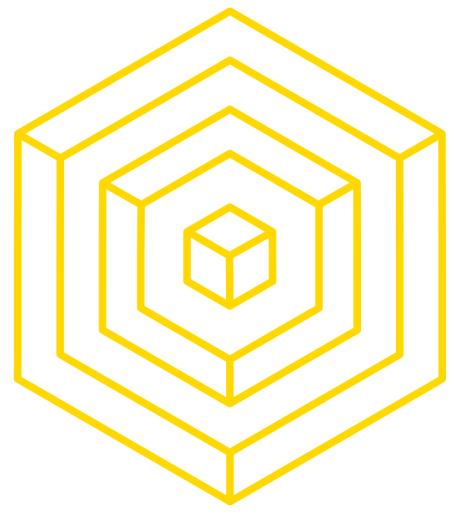
- 1 ► SMART CONTRACTS
- 2 ► ETHEREUM
- 3 ► EVM
(ETHEREUM VIRTUAL MACHINE)
- 4 ► USE CASES
- 5 ► STATE OF THE ECOSYSTEM





1 SMART CONTRACTS

BLOCKCHAIN FUNDAMENTALS LECTURE 5



BITCOIN REVIEW

PROPERTIES OF BITCOIN

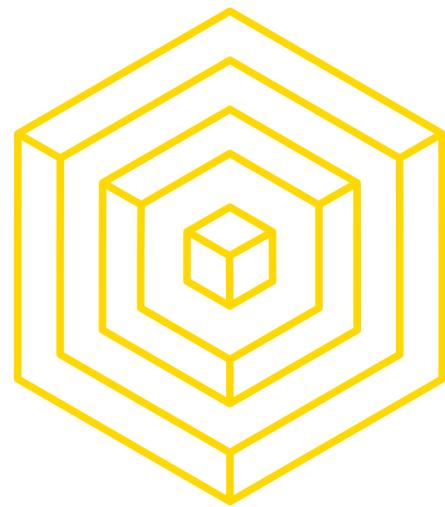
... but first, a question:

What makes Bitcoin so special?



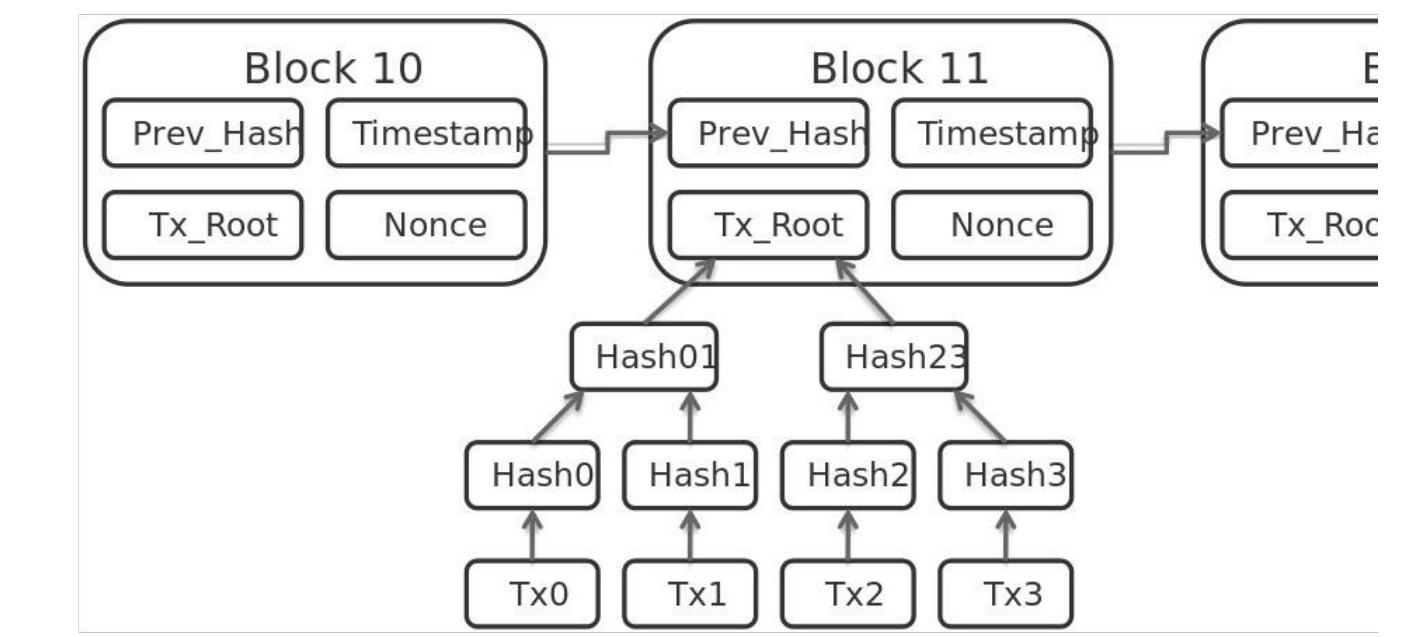
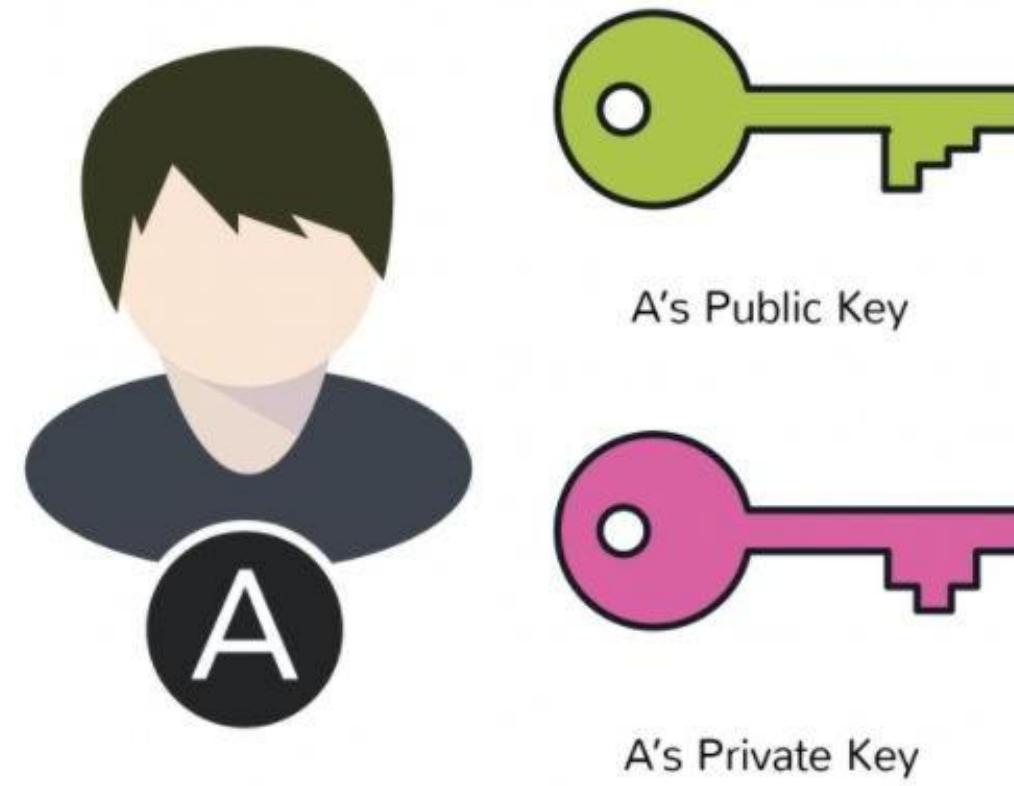
AUTHOR: GLORIA ZHAO

BLOCKCHAIN FUNDAMENTALS LECTURE 5



A DISTRIBUTED NETWORK

BITCOIN'S BARE BONES



Cryptographic
Identities

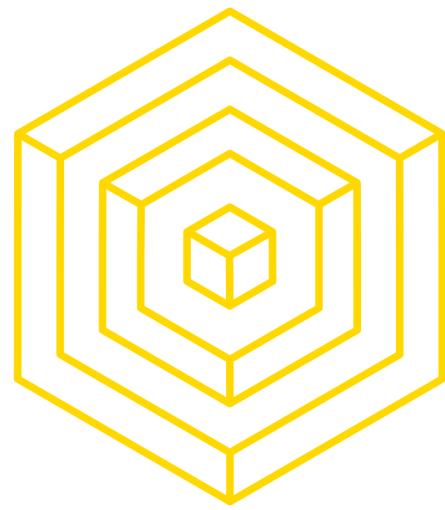
Consensus
Protocol

Blockchain



AUTHOR: GLORIA ZHAO

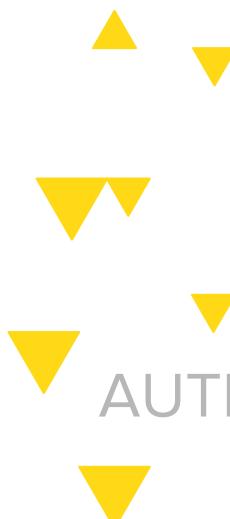
BLOCKCHAIN FUNDAMENTALS LECTURE 5



A DISTRIBUTED NETWORK

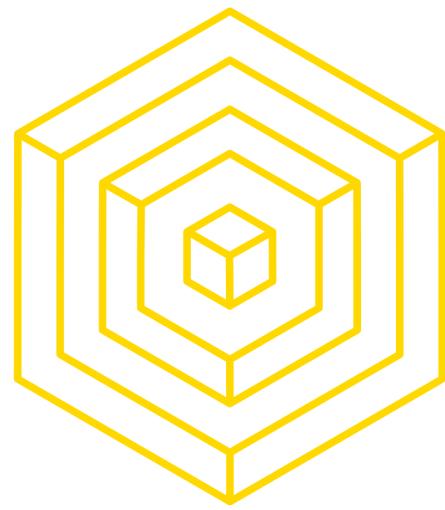
TRANSFERRABLE BENEFITS OF BITCOIN

- **Pseudonymous**, cryptographic identities allow for accountability
- **Democratic** decisions made through consensus protocol that **doesn't require trust**
- **Immutable** ledger of truth
- **Uncensorable**, cannot be controlled by any one party
- **Distributed**: no central point of failure



AUTHOR: GLORIA ZHAO

BLOCKCHAIN FUNDAMENTALS LECTURE 5



SMART CONTRACTS

CONTRACTS

con·tract

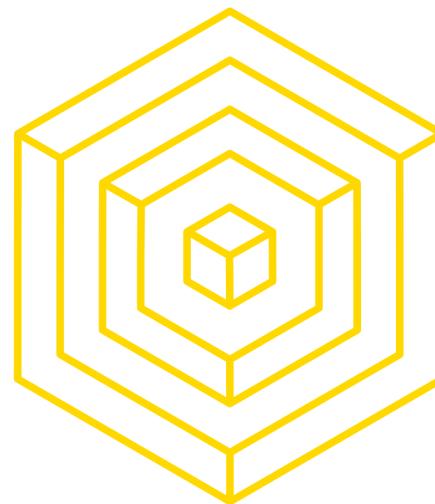
(noun) /'käntrakt/

1. a written or spoken agreement ... that is intended to be enforceable by law.



AUTHOR: PHILIP HAYES

BLOCKCHAIN FUNDAMENTALS LECTURE 5

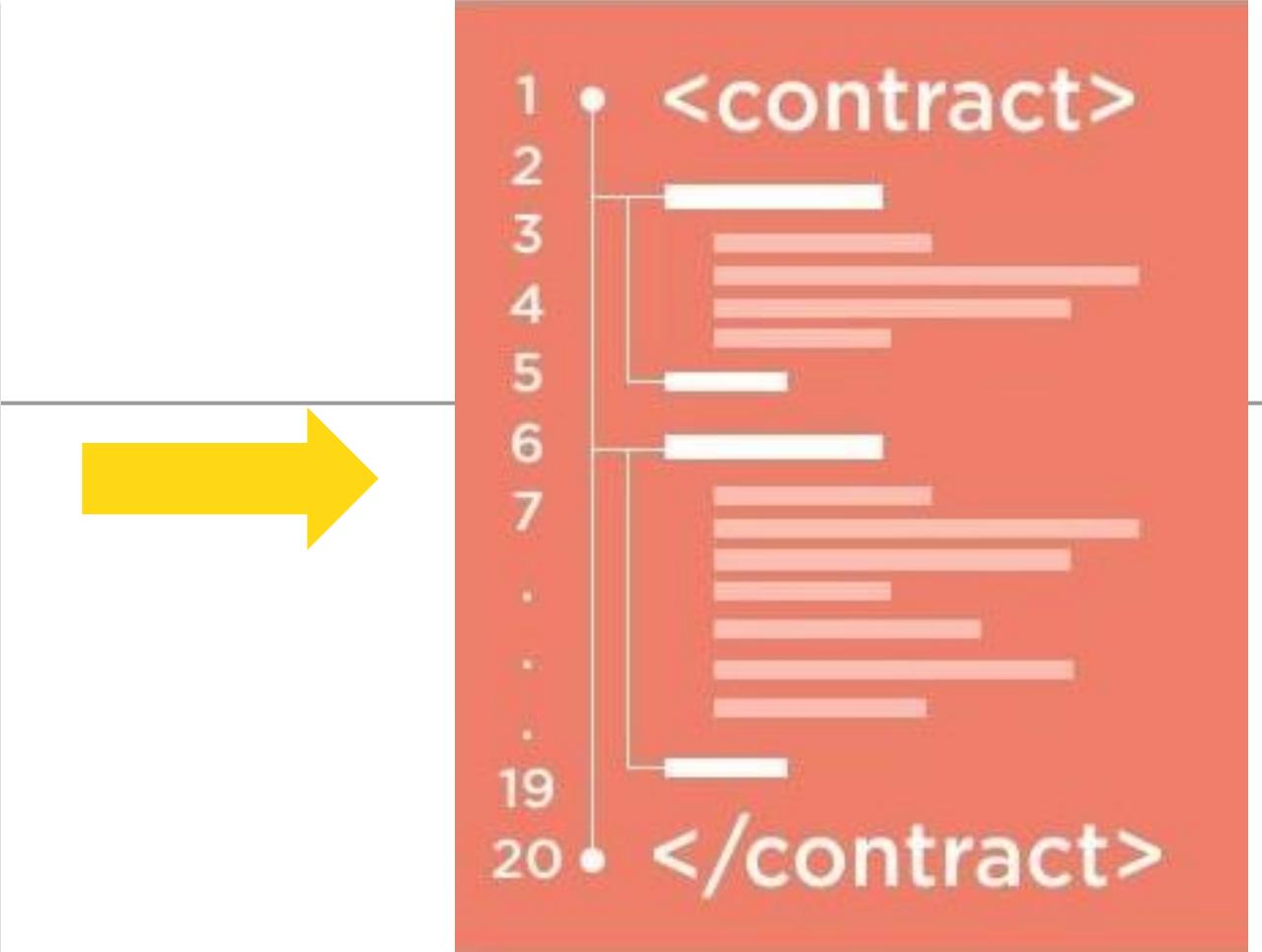


SMART CONTRACTS

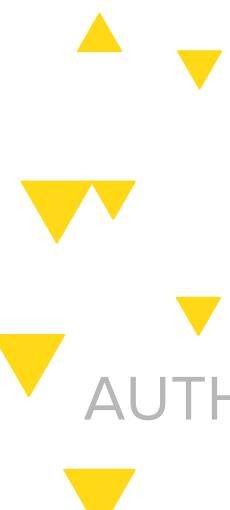
CONTRACTS

smart con·tract

(noun) /smärt 'käntrakt/

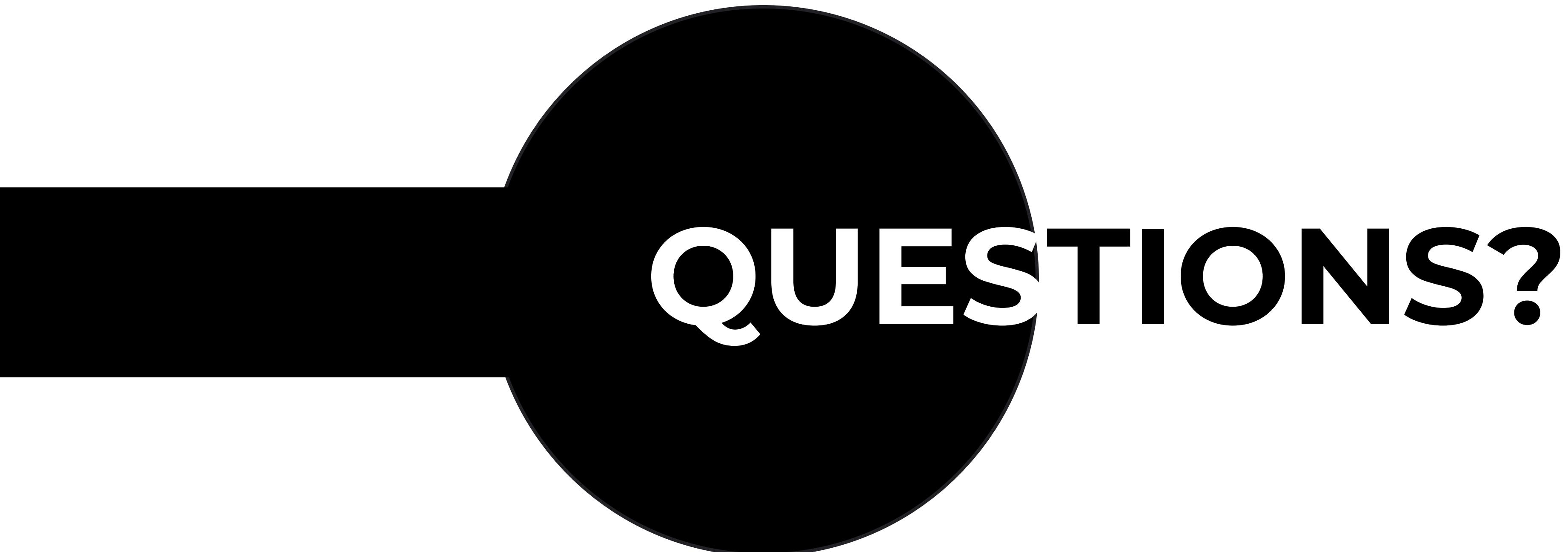


1. code that **facilitates, verifies, or enforces** the negotiation or execution of a digital contract.
 - a. **Trusted entity** must run this code

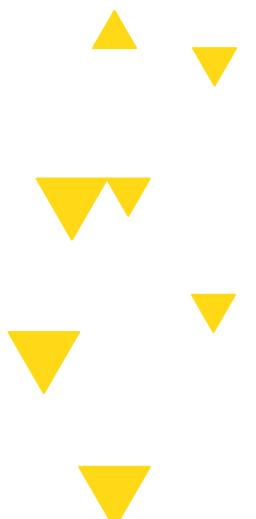


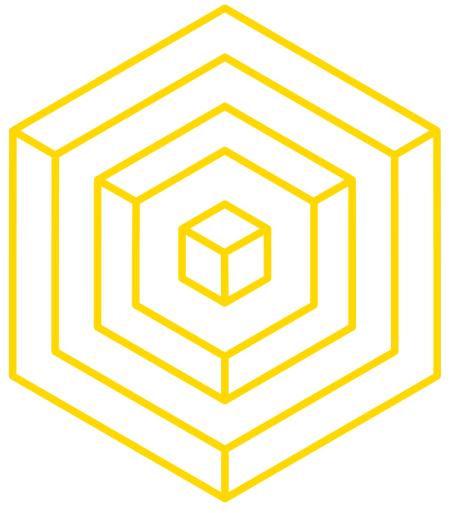
AUTHOR: PHILIP HAYES

BLOCKCHAIN FUNDAMENTALS LECTURE 5



QUESTIONS?





2 ETHEREUM

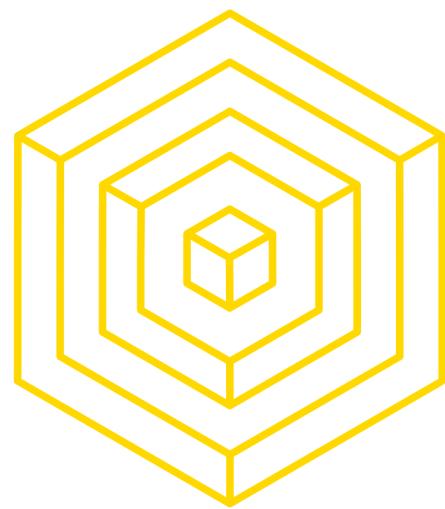
BLOCKCHAIN FUNDAMENTALS LECTURE 5



ethereum

HOMESTEAD RELEASE

BLOCKCHAIN APP PLATFORM



WHAT IS ETHEREUM?

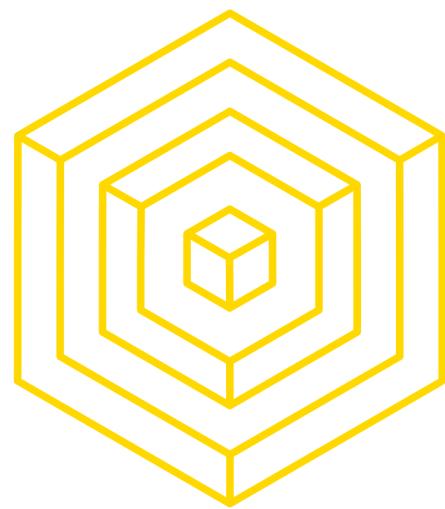
HIGH-LEVEL OVERVIEW

- Ethereum is a **decentralized** platform designed to run **smart contracts**
 - Like a distributed computer to execute code
 - Account-based blockchain
 - **Distributed state machine - transactions change global state**
 - transactions == state transaction function
- Ethereum has a native asset called **ether**
 - basis of value in the Ethereum ecosystem
 - needed to **align incentives**, given as mining rewards



AUTHOR: PHILIP HAYES

BLOCKCHAIN FUNDAMENTALS LECTURE 5



WHAT IS ETHEREUM?

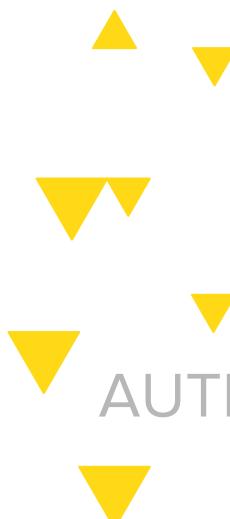
WHO WOULD WIN?

Bitcoin

- First successful cryptocurrency
- Trustless
- Immutable
- Uncensorable
- Pseudonymous
- No central point of failure
- One-CPU-One-Vote

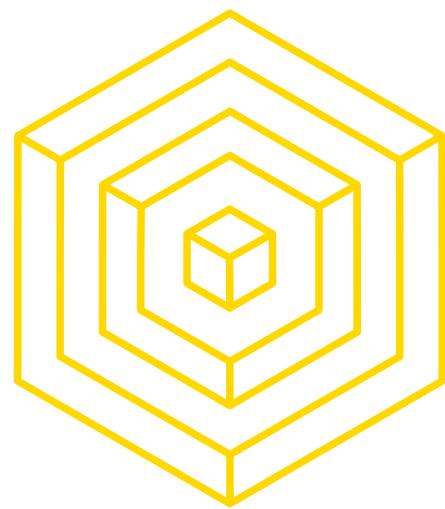


1 turing-complete boi



AUTHOR: GLORIA ZHAO

BLOCKCHAIN FUNDAMENTALS LECTURE 5



WHAT IS ETHEREUM?

COMPARISON WITH BITCOIN

Bitcoin

- The “Gold Standard” of blockchains
- Asset: bitcoins
 - Primary purpose of the Bitcoin blockchain
- Simple and robust
- Stack-based, primitive scripting language, not Turing-complete
- UTXO-based
- Will likely remain Proof-of-Work

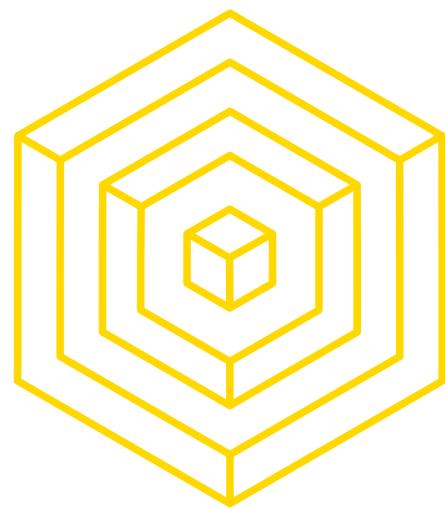


AUTHOR: PHILIP HAYES & GLORIA ZHAO



Ethereum

- Smart Contract Blockchain Platform
- Asset: ether
 1. Fund computation
 2. Align incentives
- Complex and feature-rich
- Turing-complete scripting language
- Account-based
- Planning to move to Proof-of-Stake



WHAT IS ETHEREUM?

COMPARISON WITH BITCOIN

Misc. Implementation Details

- Block creation time: ~15 sec vs ~10 min
- Proof-of-Work: Ethash vs SHA-256 (currently ASIC resistant)
- Exchange Rate: \$841.01 (2018-02-24)



▼
▼
▼
▼

AUTHOR: PHILIP HAYES

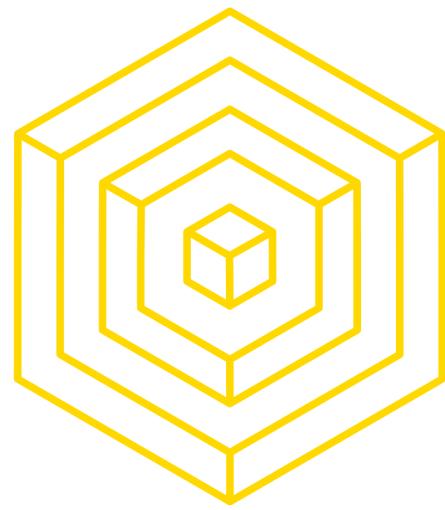
|
Apr '17

|
Jul '17

|
Oct '17

|
Jan '18

BLOCKCHAIN FUNDAMENTALS LECTURE 5



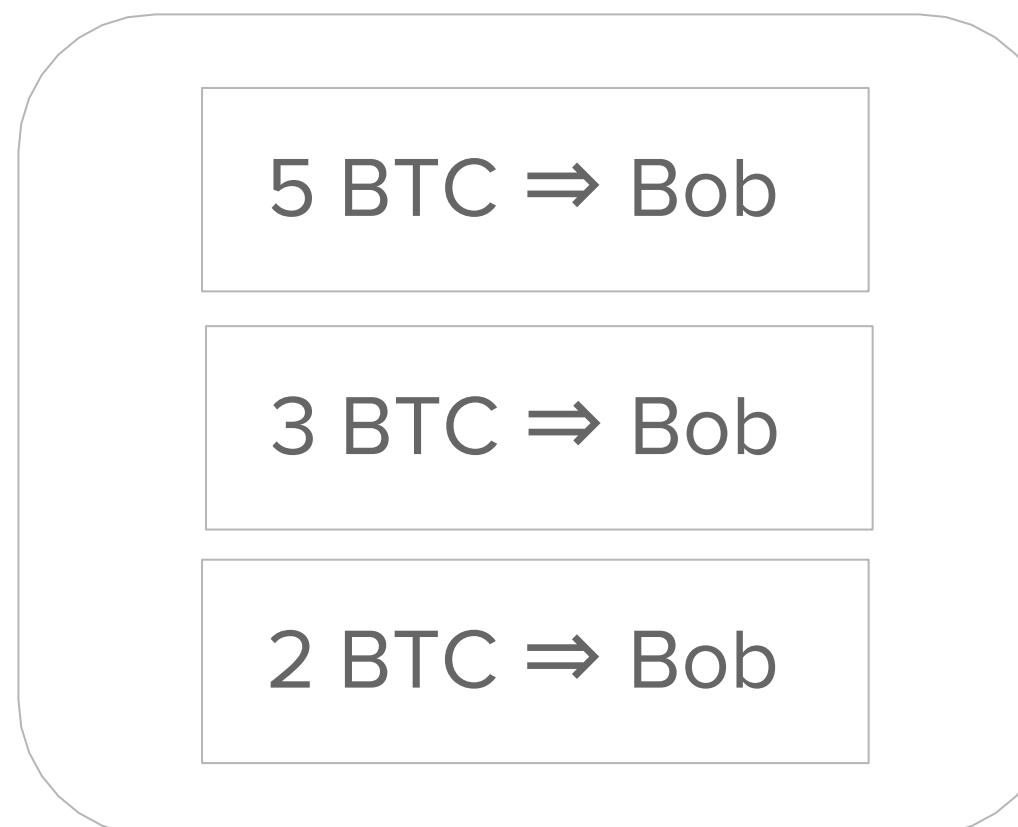
ETHEREUM ACCOUNTS

ACCOUNTS VS UTXO MODEL

Easy to make transactions and prevent double spending

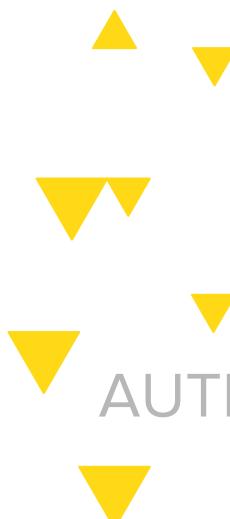
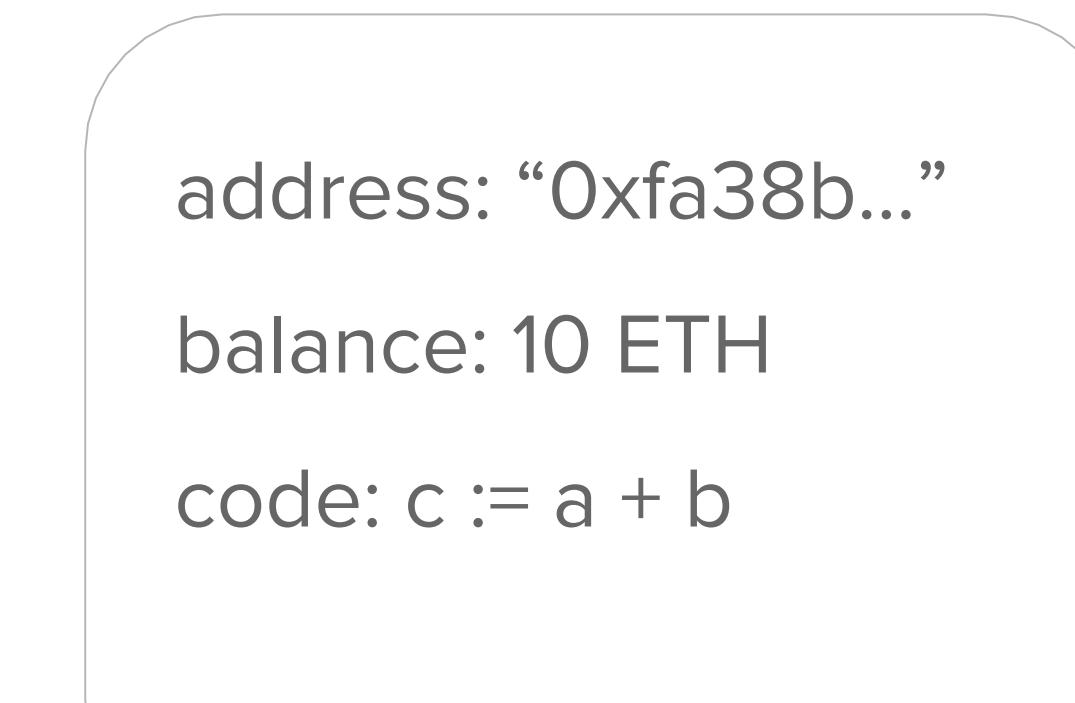
Bitcoin:

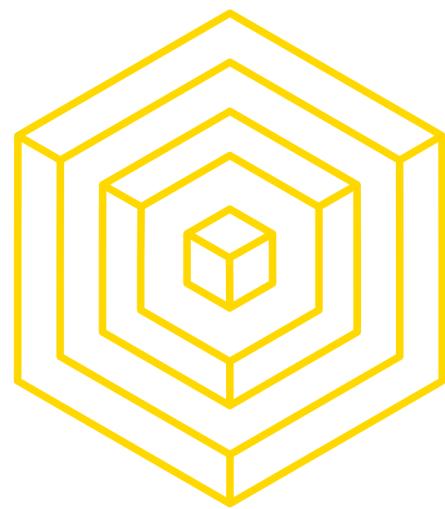
Bob owns private keys to set of UTXOs



Ethereum:

Alice owns private keys to an account





ETHEREUM ACCOUNTS

ACCOUNTS RATIONALE

Easy to make transactions and prevent double spending

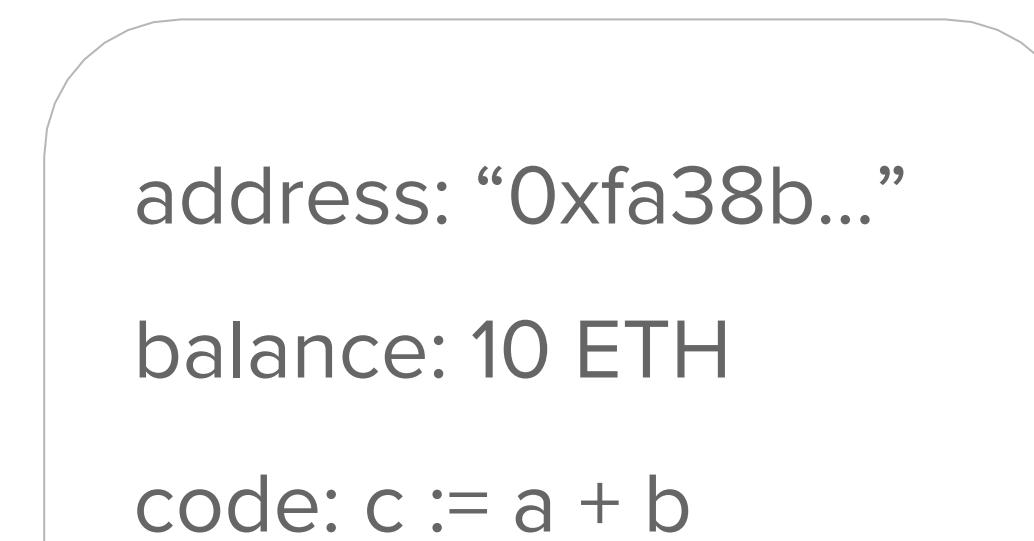
Bitcoin:

Bob owns private keys to set of UTXOs



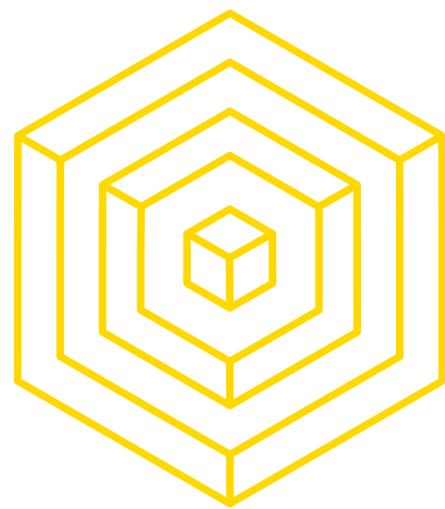
Ethereum:

Alice owns private keys to an account



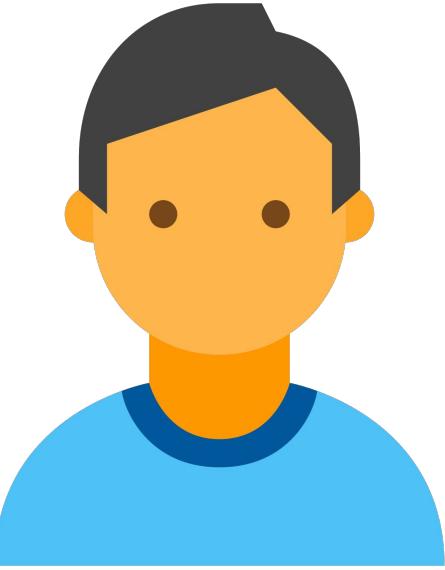
Space-efficient to update balances instead of storing UTXOs

Easier to look up balance and transfer between accounts when programming



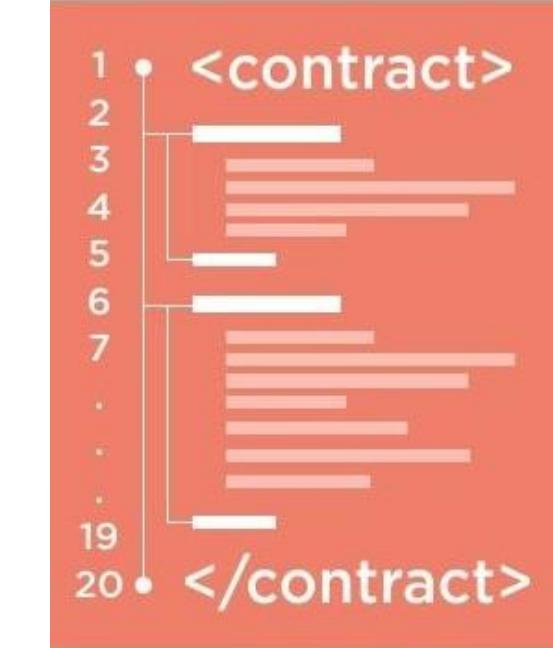
ETHEREUM ACCOUNTS

ACCOUNT TYPES



Externally Owned Accounts

- Owned by some external entity (person, corporation, etc.)
- Can send transactions to transfer ether or trigger contract code
- Contains:
 - Address
 - Ether Balance



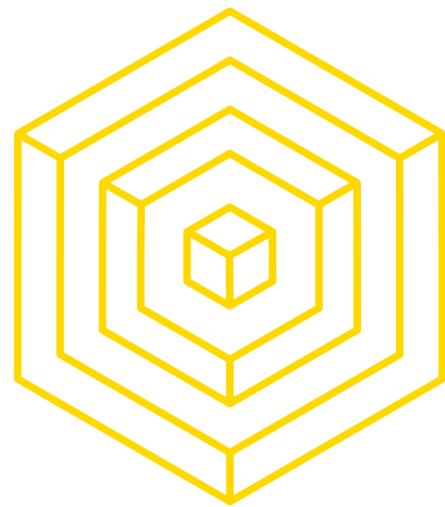
Contract Accounts

- “Owned” by contract
- Code execution triggered by transactions or function calls (msg)
- Contains:
 - Address
 - Associated contract code
 - Persistent storage



AUTHOR: PHILIP HAYES & GLORIA ZHAO

BLOCKCHAIN FUNDAMENTALS LECTURE 5



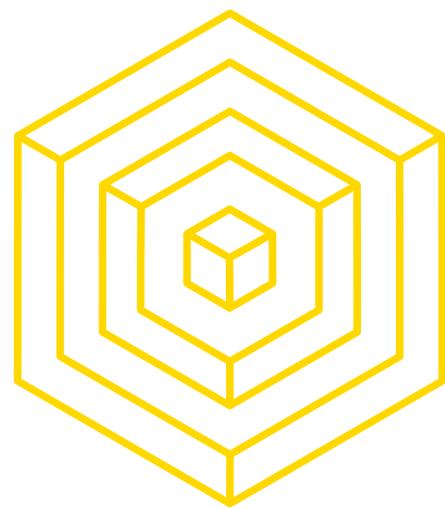
ETHEREUM SMART CONTRACTS

CONTROL

Smart Contracts in Ethereum are like autonomous agents that live inside of the Ethereum network

- React to external world when "poked" by transactions (which call specific functions)
- Have direct control over:
 - **internal ether balance**
 - **internal contract state**
 - **permanent storage**



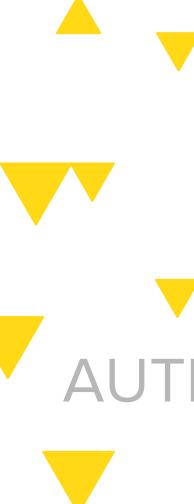


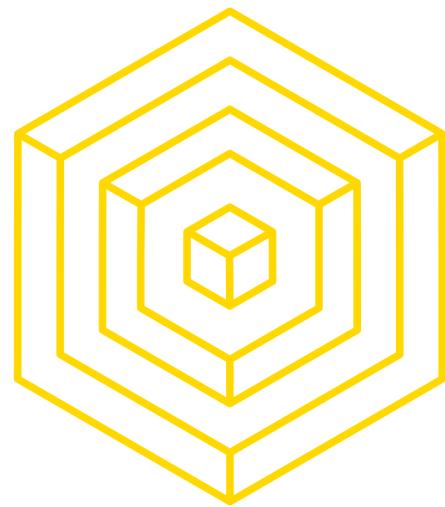
ETHEREUM SMART CONTRACTS

SMART CONTRACTS IN ETHEREUM

Ethereum Contracts generally serve four purposes:

- **Store and maintain data**
 - Data represents something useful to users or other contracts
 - Ex: a token currency or organization's membership
- **Manage contract or relationship between untrusting users**
 - Ex: financial contracts, escrow, insurance
- **Provide functions to other contracts**
 - Serving as a software library
- **Complex Authentication**
 - Ex: M-of-N multisignature access

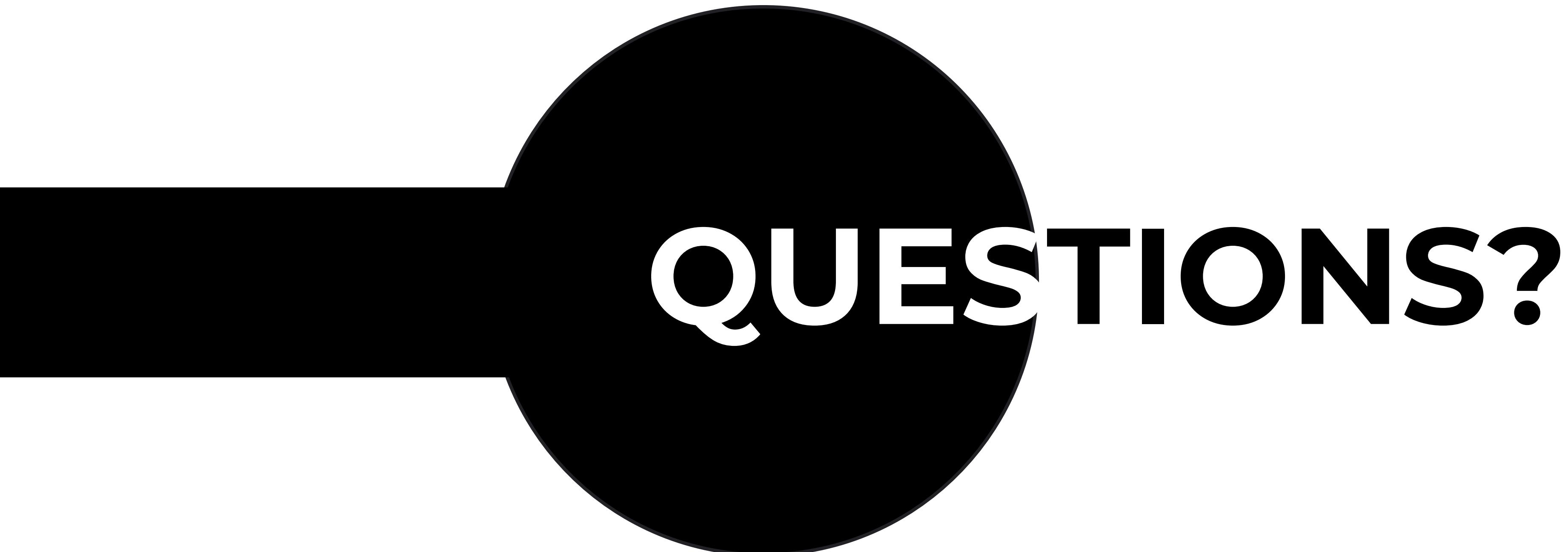




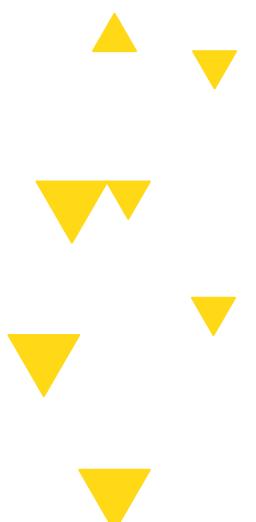
ETHEREUM SMART CONTRACTS

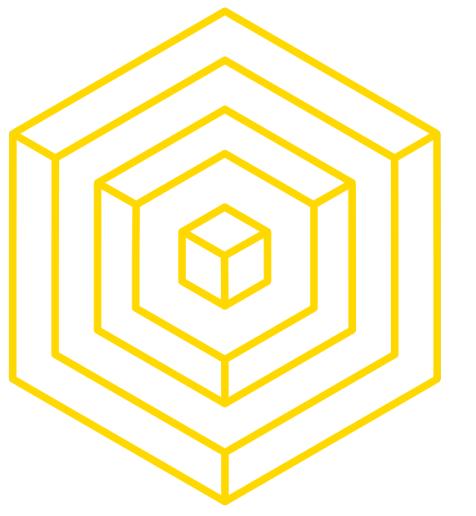
SAMPLE BETTING CONTRACT

```
contract Betting {  
    address public owner;  
    address public gamblerA, gamblerB, oracle;  
    uint[] outcomes;  
  
    struct Bet {  
        uint outcome;  uint amount;          /* Defines a Bet */  
        bool initialized;  
    }  
  
    mapping (address => Bet) bets;  
    mapping (address => uint) winnings; /* Keep track of every gambler's bet */  
    ...                                /* Keep track of every player's winnings */  
  
    function makeBet(uint _outcome) payable returns (bool) { ... }  
    function makeDecision(uint _outcome) oracleOnly() { ... }  
    function withdraw(uint withdrawAmount) returns (uint remainingBal) { ... }  
    ...  
}
```



QUESTIONS?

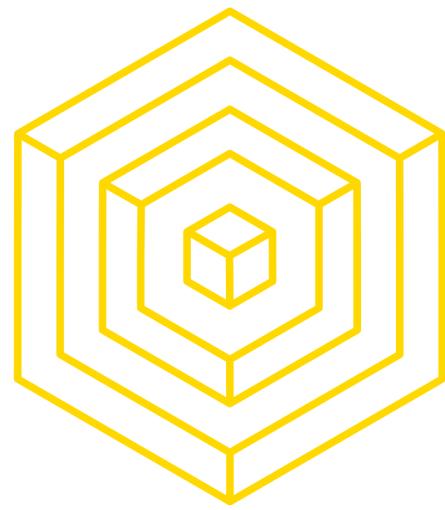




3

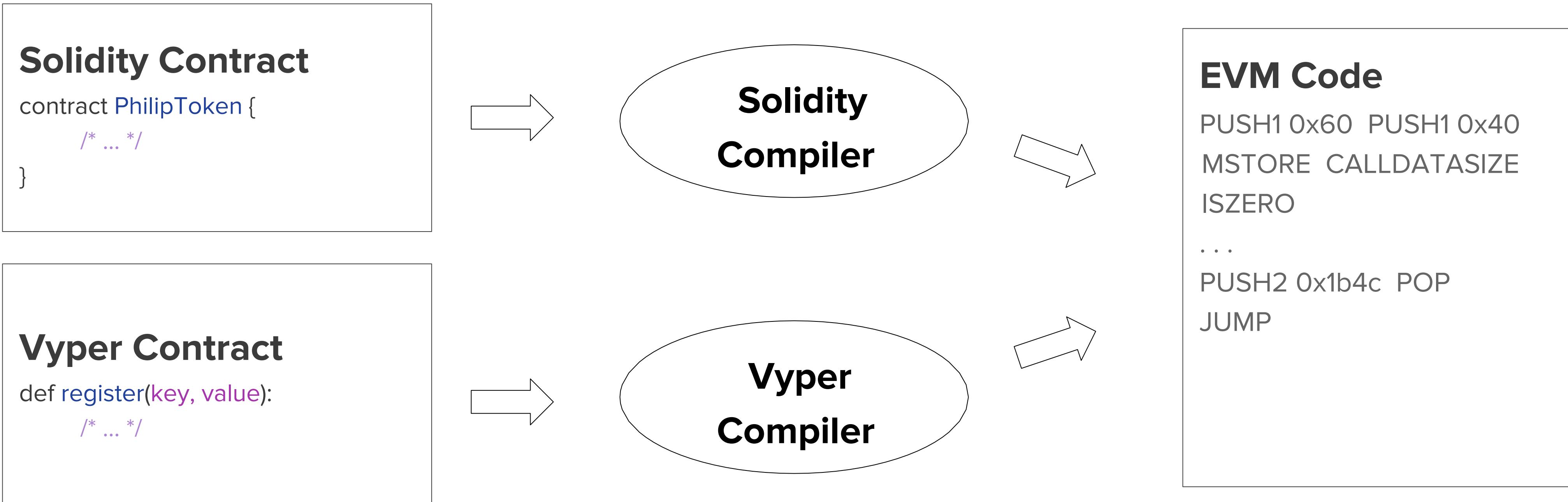
ETHEREUM VIRTUAL MACHINE

BLOCKCHAIN FUNDAMENTALS LECTURE 5



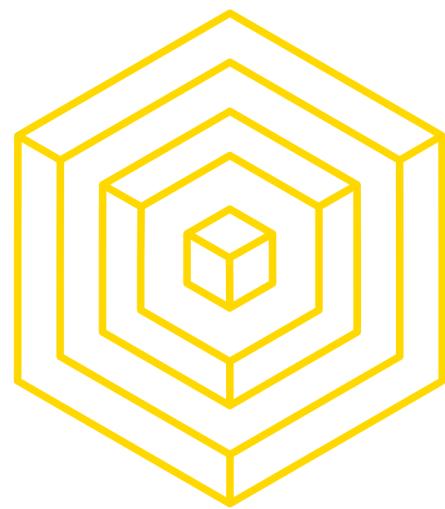
ETHEREUM VIRTUAL MACHINE

COMPILEATION AND PROCESS



AUTHOR: PHILIP HAYES & GLORIA ZHAO

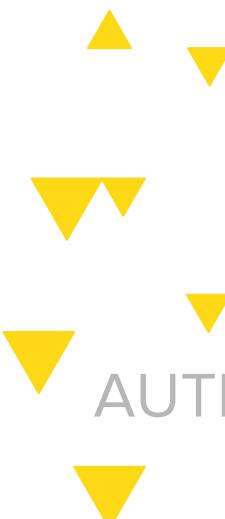
BLOCKCHAIN FUNDAMENTALS LECTURE 5



ETHEREUM VIRTUAL MACHINE

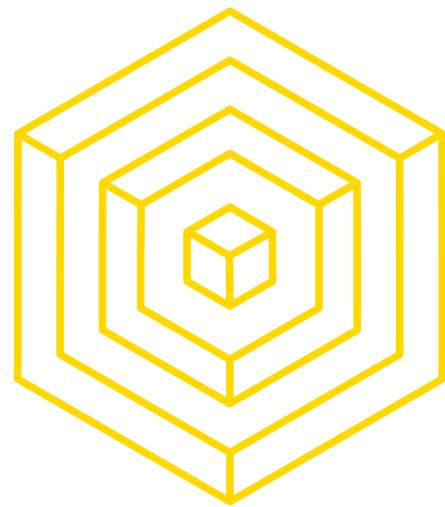
DISTRIBUTED VERIFICATION & CONSENSUS

- Ethereum is a “distributed computer”: **every node** executes Ethereum smart contracts, then come to consensus on the new network state
- Ethereum’s distributed consensus protocol is **Proof-of-Work**
 - Miners competitively create blocks by executing EVM code and searching for solution the the mining puzzle
 - PoW is **competitive**, meaning only **ONE** miner is able to add block to the Ethereum blockchain and receive gas and transaction fees
 - Think of PoW as process of “randomly” selecting one node’s execution result as the correct one



AUTHOR: GLORIA ZHAO

BLOCKCHAIN FUNDAMENTALS LECTURE 5



ETHEREUM VIRTUAL MACHINE

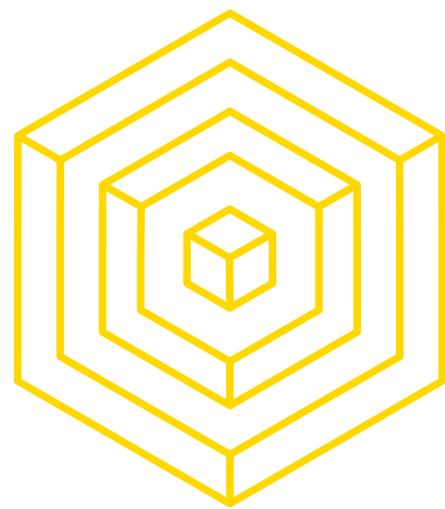
HIGH-LEVEL OVERVIEW

- Every Ethereum node runs EVM as part of its block verification procedure
- Network consensus removes the need for Trusted Third Party
 - Violation of contracts requires subverting the entire network
- Secure Peer-to-Peer agreements that live on the blockchain forever
- The **EVM (Ethereum Virtual Machine)** runs contract code
- Contract code that actually gets executed on every node is EVM code
 - Our complex features are made possible by the fact that we can compile contract code into something more simple
 - **EVM code:** low-level, stack based bytecode language (i.e. JVM bytecode)



AUTHOR: PHILIP HAYES

BLOCKCHAIN FUNDAMENTALS LECTURE 5



EVM GAS AND FEES

HIGH-LEVEL OVERVIEW

Immediate Issue:

What if our contract has an infinite loop?

Every node on the network will get stuck executing the loop forever!

- By the *halting problem*, it is impossible to determine ahead of time whether the contract will ever terminate

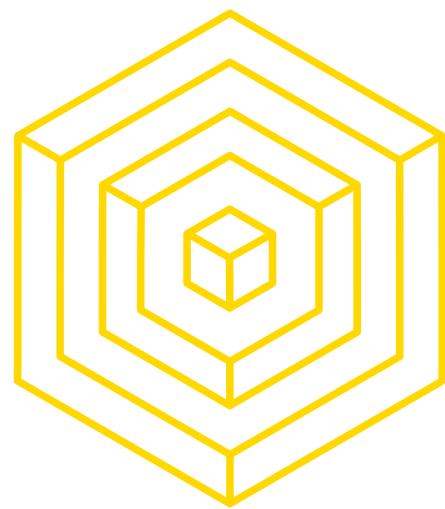
- ⇒ Denial of Service Attack!

```
function foo()
{
    while (true) {
        /* Loop forever!
    }
}
```



AUTHOR: PHILIP HAYES

BLOCKCHAIN FUNDAMENTALS LECTURE 5

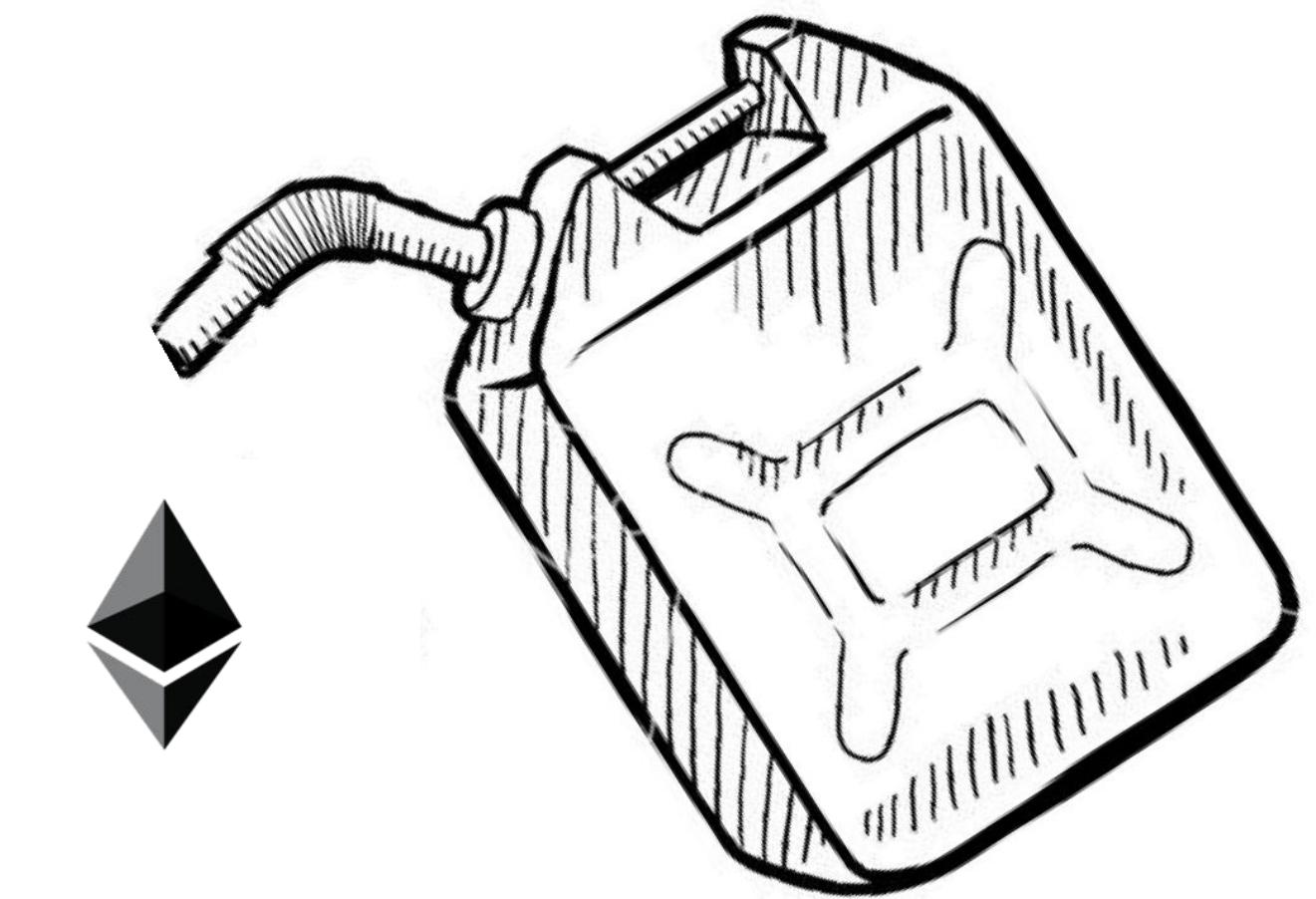


EVM GAS AND FEES

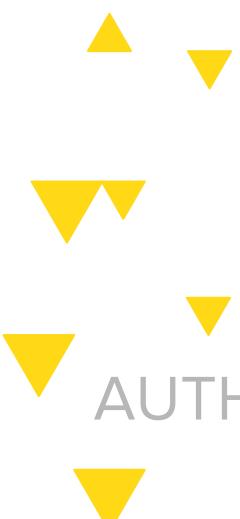
HIGH-LEVEL OVERVIEW

Ethereum's solution:

- Every contract requires “gas”, which “fuels” contract execution
- Every EVM op-code requires some gas in order to execute
- Every transaction specifies:
 - the `startgas`, or the maximum quantity of gas it is willing to consume
 - the `gasprice`, or the fee in ether it is willing to pay per unit gas

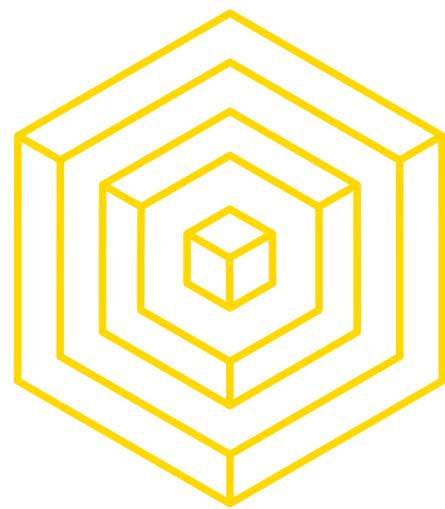


EVM



AUTHOR: PHILIP HAYES

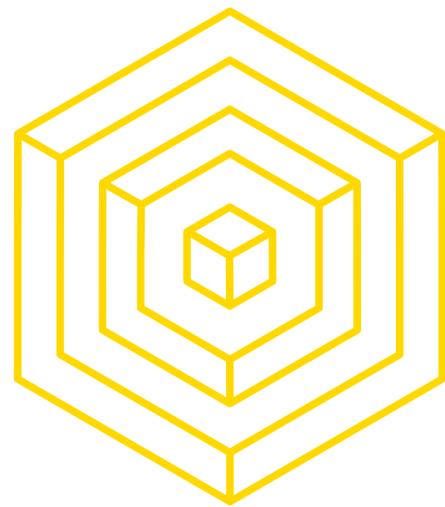
BLOCKCHAIN FUNDAMENTALS LECTURE 5



EVM GAS AND FEES

HIGH-LEVEL OVERVIEW

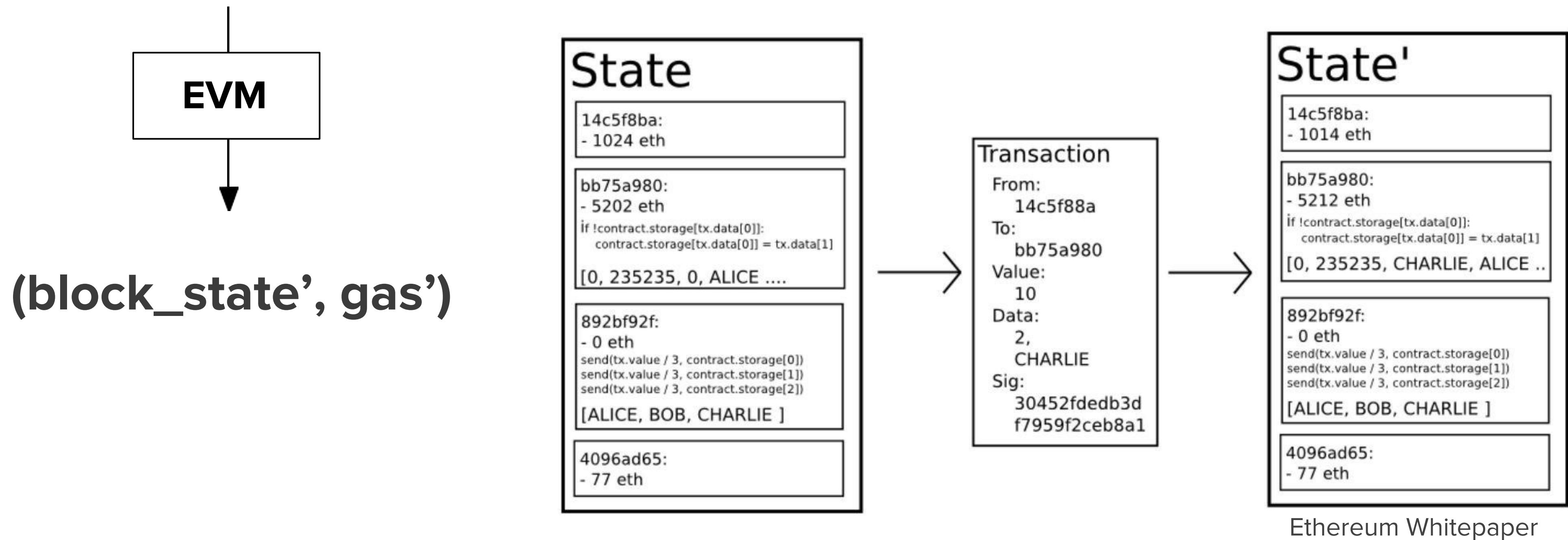
- At the start of the transaction
 - `startgas * gasprice` (units = ether) are subtracted from the sender's account (the one “poking” the contract)
- If the contract **successfully executes** ...
 - the remaining gas is refunded to the sender
- If the contract execution **runs out of gas** before it finishes ...
 - execution reverts
 - `startgas * gasprice` are not refunded
- Purchasing gas == purchasing distributed, trustless computational power
- ▼ ▼ ● An attacker looking to launch a DoS attack will need to supply enough ether to fund the attack

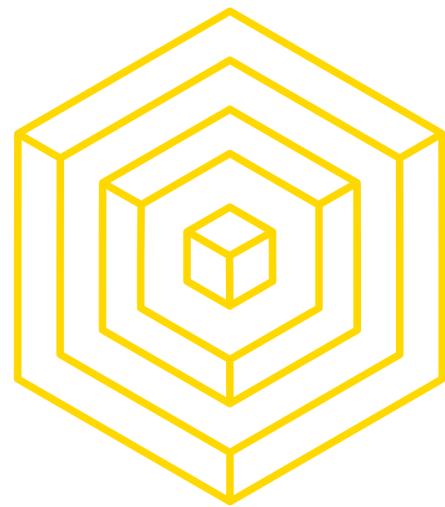


ETHEREUM NETWORK STATE

STATE TRANSITION FUNCTION

(block_state, gas, memory, transaction, message, code, stack, pc)





ETHEREUM CONCLUSIONS

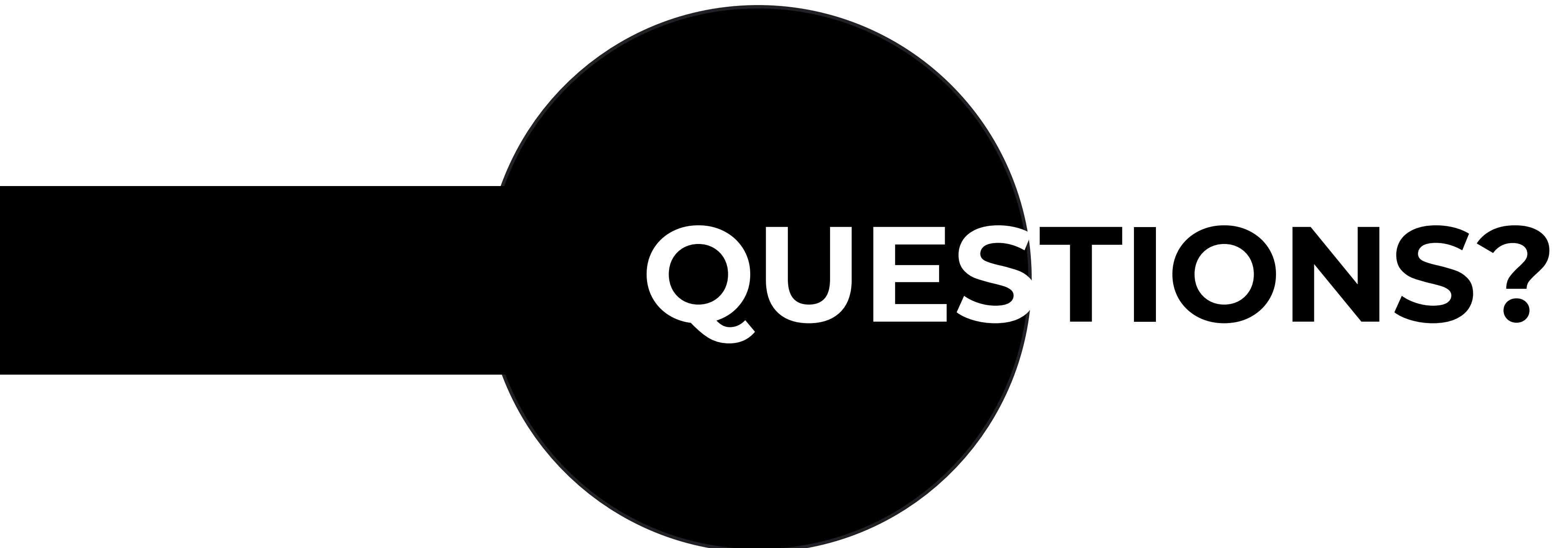
IT'S NOT FOR EVERYTHING

- Ethereum is not about optimising efficiency of computation
- Its parallel processing is redundantly parallel
 - efficient way to reach consensus on the system state without needing trusted third parties
- Contract executions are redundantly replicated across nodes
 - ⇒ expensive
 - creates an incentive not to use the blockchain for computation that can be done off chain

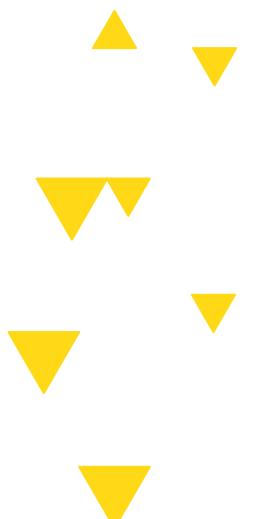


AUTHOR: PHILIP HAYES

BLOCKCHAIN FUNDAMENTALS LECTURE 5



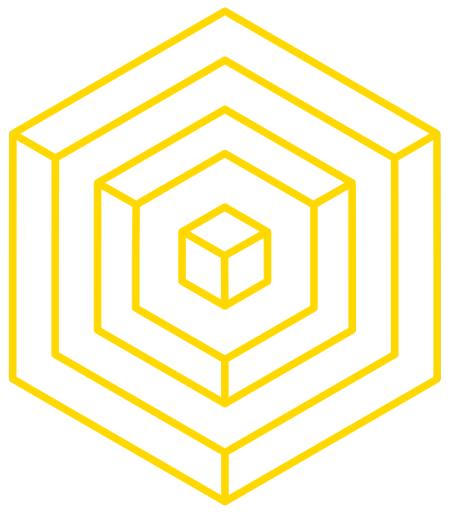
QUESTIONS?





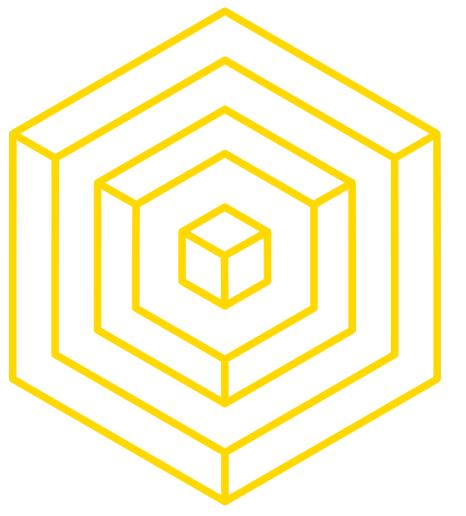
BLOCKCHAIN
AT BERKELEY

**BREAK
SECTION**



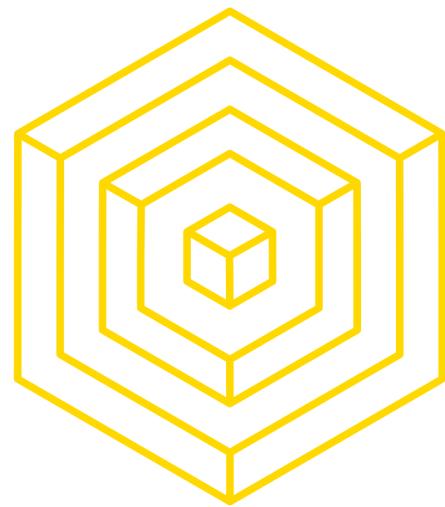
4

USE CASES



4.1

BASIC USE CASES



BASIC USE CASES

SMART ASSETS

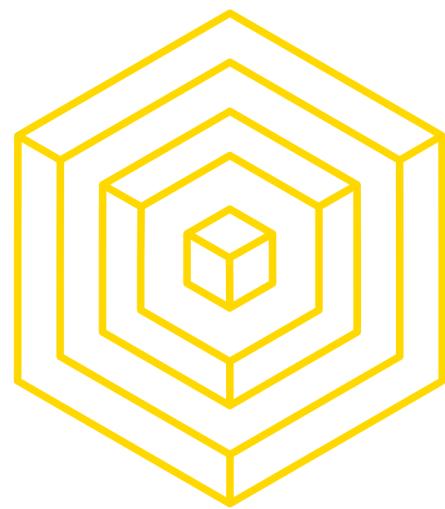
- Token System Implementation
- Database with one operation
 - Ensure Alice has enough \$\$ and that she initiated the transaction
 - Subtract X from Alice, give X to Bob

```
def send(to, value):
    if self.storage[msg.sender] >= value:
        self.storage[msg.sender] = self.storage[msg.sender] - value
        self.storage[to] = self.storage[to] + value
```



AUTHOR: GILLIAN CHU

BLOCKCHAIN FUNDAMENTALS LECTURE 5



PUBLIC REGISTRY: Namecoin

BLOCKCHAIN FUNDAMENTALS

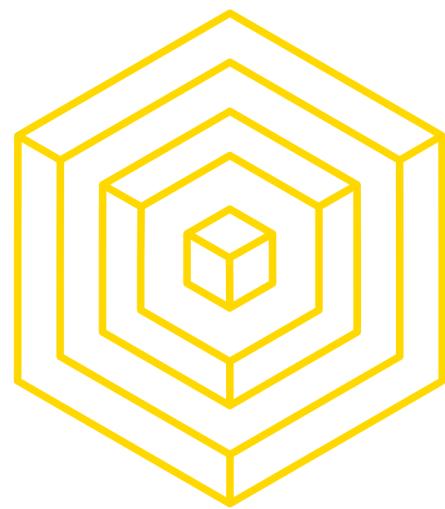
- DNS System
 - Maps domain name to IP address
 - “gillian.chu” => “12.34.56.78”
- Easy to implement in Ethereum

```
def register(name, value):  
    if !self.storage[name]:  
        self.storage[name] = value
```



AUTHOR: GILLIAN CHU

BLOCKCHAIN FUNDAMENTALS LECTURE 5



DOCUMENT OWNERSHIP

BLOCKCHAIN FUNDAMENTALS

“Proof-of-Existence”

- Proves ownership of a certain document without revealing it
- **Timestamps** verify ownership later

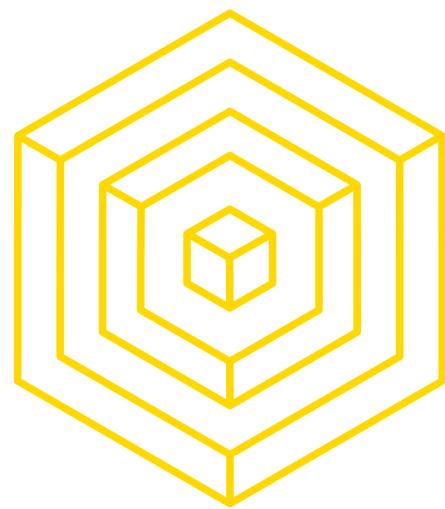
Use Cases:

- Rent server space to store documents
 - Proof document is unmodified via hash values
 - **Integrity** guaranteed



AUTHOR: GILLIAN CHU

BLOCKCHAIN FUNDAMENTALS LECTURE 5



STANDARD BOUNTIES

ETHEREUM SMART CONTRACTS

ETHDenver

- Incentivize answers to StackOverflow Q's
- Makes use of StandardBounties contract
- Uses Metamask Integration



AUTHOR: GILLIAN CHU

BLOCKCHAIN FUNDAMENTALS LECTURE 5

Graph Search vs Tree Search

The screenshot shows a Stack Overflow question with the following details:

- Question Summary:** "I am curious whether there is the basic difference between *graph search* and *tree search* versions regarding DFS, A* searches, in *artificial intelligence*?"
- Upvotes:** 71
- Tags:** search, graph, tree, a-star, depth-first-search
- Share Options:** share, edit
- Timestamps:** edited Jan 7 '17 at 21:20, asked May 21 '12 at 6:02
- User Information:** nbro (4,856), Rayhanur Rahman (356)
- Content:** "I've posted an in-depth discussion of A* (and tried to make it as accessible as possible) at the [libGdx AI project](#). I hope you or somebody else finds it useful! – [EntangledLoops](#) Oct 12 '15 at 16:15"
- Comments:** add a comment
- Answers:** 5 Answers
- Sort Options:** active, oldest, votes

▲ Judging from the existing answers, there seems to be a lot of confusion about this concept.

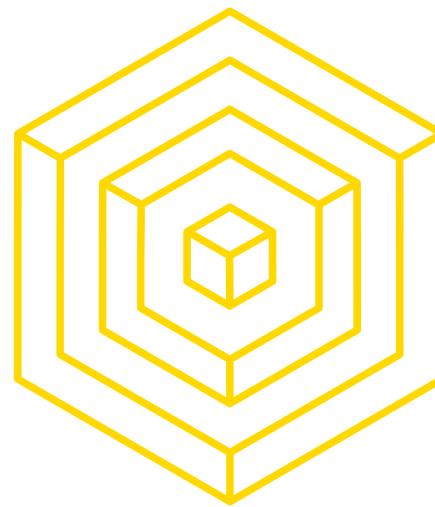
131 The Problem Is Always a Graph

▼ The distinction between tree search and graph search is not rooted in the fact whether your problem is a tree or a graph. It is always assumed you're dealing with a graph. The distinction lies in the *traversal pattern* that is used to search through the graph, which can be graph-shaped or tree-shaped.

If you're dealing with a tree-shaped *problem*, both algorithm variants lead to equivalent results. So you can pick the simpler tree search variant.

Difference Between Graph and Tree Search

Your basic graph search algorithm looks something like the following. With a start node `start`, directed edges as `successors` and a `goal` specification used in the loop condition. `open` holds the nodes in memory, which are currently under consideration, the *open list*. Note that the following pseudo code is not correct in every aspect (2).



STANDARD BOUNTIES

ETHEREUM SMART CONTRACTS

ETHDenver

- Incremental
- Make
- Uses



AUTHOR: GILLIAN CHU

Bounties . NETWORK

Create a New Bounty

Title
Difference between Prim and Dijkstra graph algorithm

Description
My question is, why we are checking if vertex belongs to Q (v in Q), i.e. that vertex doesn't belong to tree, whereas in Dijkstra algorithm we are not checking for that.

Any reason, why?

Payout Method
ETH

the token which will be used to pay out the reward

Associated Files
Upload

any files required by bounty hunters

When to Activate
Now

The requirements for a bounty can only be edited while it is in the draft stage

Bounty Category
x Code x Questions

the types of tasks being bountied

Payout Amount (ETH or whole tokens)
0.1

the reward amount for completing the task

Contact Info
brian.ho@berkeley.edu

for bounty hunters to be able to contact you off-chain

Bounty Deadline (UTC)
02/25/2018, 11:59 PM

the deadline for submitting any bugs

Deposit Amount
0.1

To activate, you must deposit enough to pay the bounty at least once

Graph Search vs Tree Search

most popular
is free for 1-year

Microsoft Azure

TRY AZURE FREE →

Compare between graph search and tree search versions
since?

asked May 7 '17 at 21:20
bro 8,856 6 36 79

asked May 21 '12 at 6:02
Rayhanur Rahman 356 1 4 4

ied to make it as accessible as possible) at the libGdx AI
useful! – EntangledLoops Oct 12 '15 at 16:15

active oldest votes

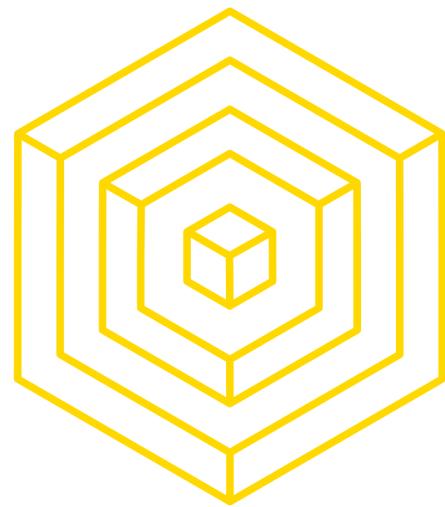
to be a lot of confusion about this concept.

search is not rooted in the fact whether your
ed you're dealing with a graph. The distinction lies
rough the graph, which can be graph-shaped or

both algorithm variants lead to equivalent results. So

Tree Search

ing like the following. With a start node `start`,
specification used in the loop condition. `open` holds
consideration, the `open list`. Note that the following



STANDARD BOUNTIES

ETHEREUM SMART CONTRACTS

ETHDenver

- Incentivize
- Make
- Use

Bounties .NETWORK

PROFILE

You have posted **0** bounties

0	0	0
DRAFT	ACTIVE	DEAD

0	0
EXPIRED	COMPLETED

MY PROFILE

MY BOUNTIES

SIGN UP TO RECEIVE BOUNTIES NOTIFICATIONS

email address

SIGN UP

Graph Search vs Tree Search

New Bounty

most popular tree for 1-year

Microsoft Azure TRY AZURE FREE →

between graph search and tree search versions?

'17 at 21:20 asked May 21 '12 at 6:02 Rayhanur Rahman 356 1 4 4

to make it as accessible as possible) at the libGdx AI !! – EntangledLoops Oct 12 '15 at 16:15

active oldest votes

be a lot of confusion about this concept.

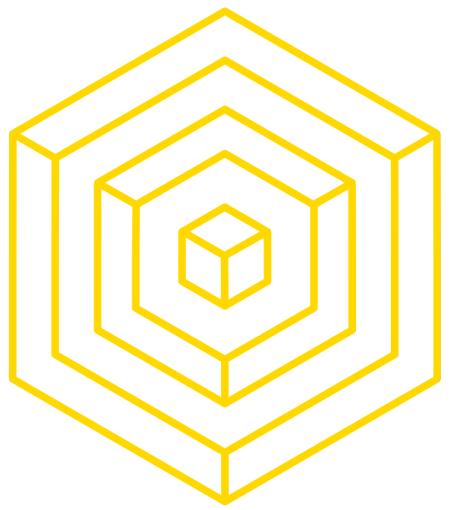
search is not rooted in the fact whether you're dealing with a graph. The distinction lies in how the graph, which can be graph-shaped or

algorithm variants lead to equivalent results. So

Tree Search

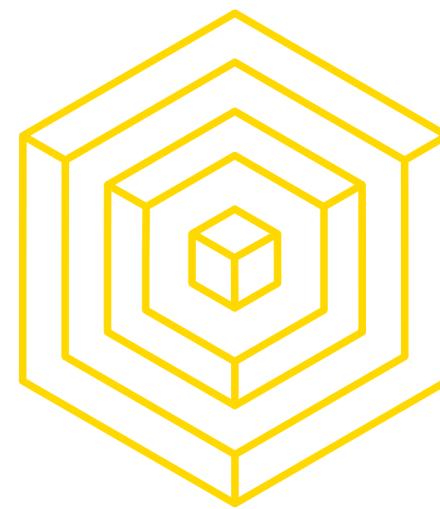
like the following. With a start node `start`, ification used in the loop condition. `open` holds consideration, the `open list`. Note that the following

AUTHOR: GILLIAN CHU



4.2

ADVANCED USE CASES



DECENTRALIZED LAND TITLES

BLOCKCHAIN FUNDAMENTALS

Problem:

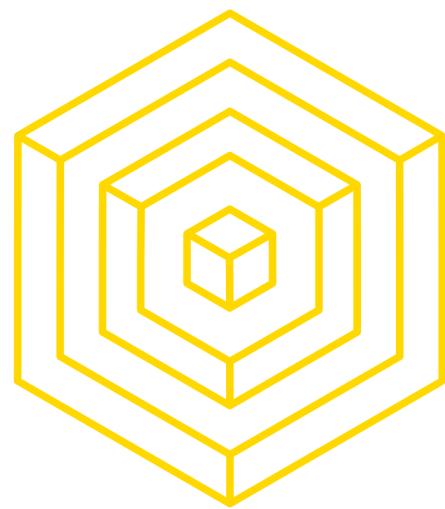
- Flawed paperwork, forged signatures, **unclear documents**
- Lacking **central** government authority

Pitfalls:

- Corrupt officials accepting **bribes, tampering** with records
- Government cannot support land record authority
- Citizens **mistrustful** of NGO/multiple NGOs



AUTHOR: GILLIAN CHU



DECENTRALIZED LAND TITLES

BLOCKCHAIN FUNDAMENTALS

The Blockchain Solution:

- Hashes & Digital Signatures
- Transparency
- Immutability
- Limits Centralization



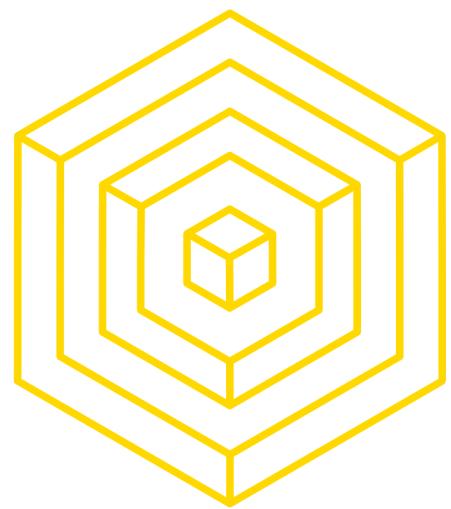
BTCMANAGER.com

*Simple mechanism to transfer ownership, like
making a transaction on Bitcoin*



AUTHOR: APARNA KRISHNAN

UPDATED: GILLIAN CHU



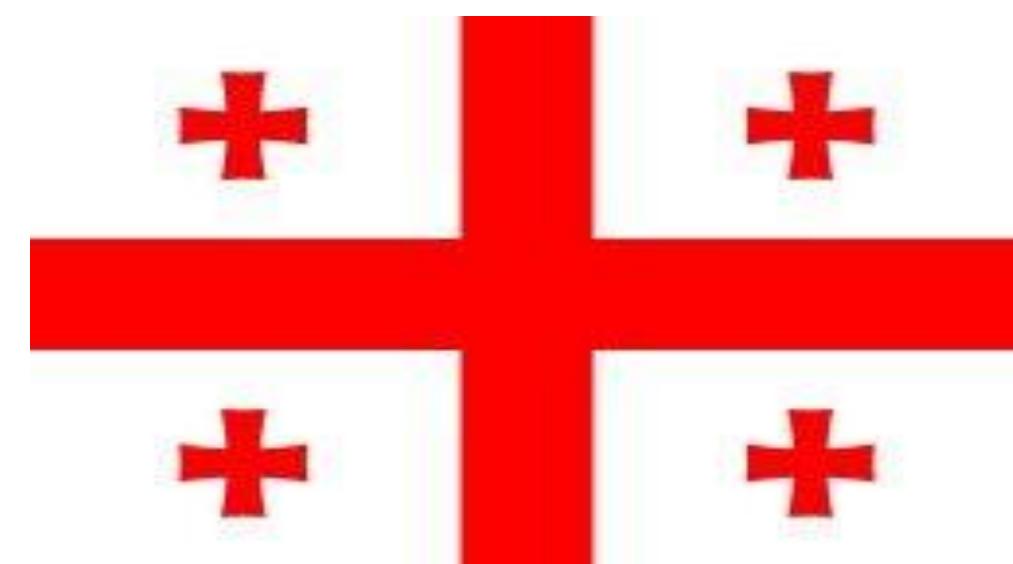
DECENTRALIZED LAND TITLES

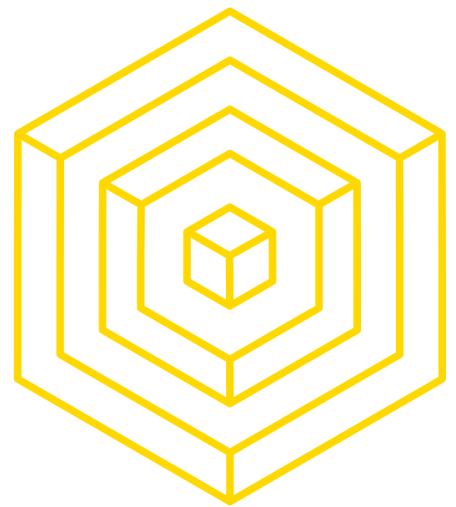
BLOCKCHAIN FUNDAMENTALS

Caveat:

The blockchain is only as good as the information fed into it.

Countries investigating: **Georgia, Ukraine, Sweden**





DECENTRALIZED LAND TITLES

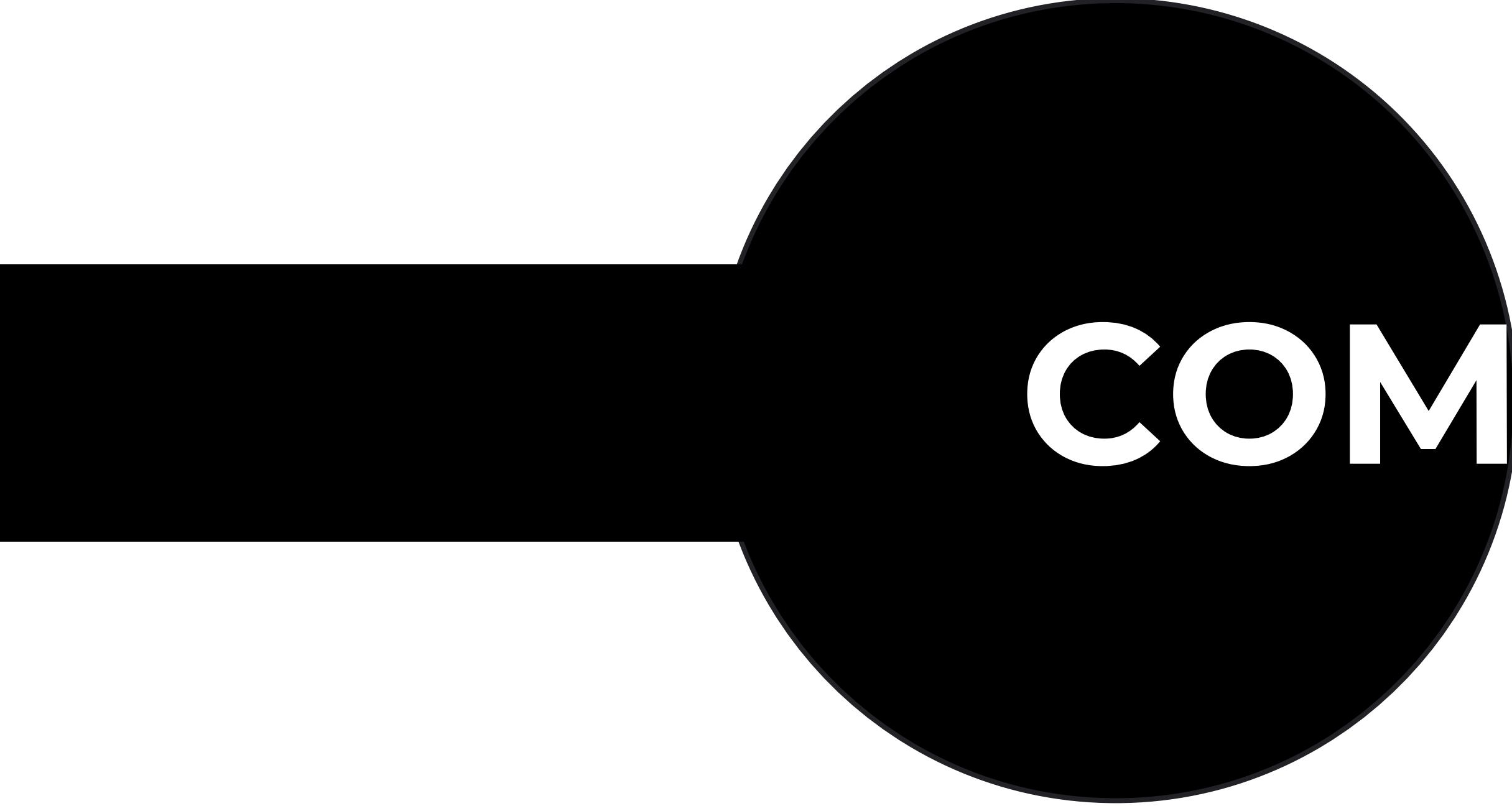
BLOCKCHAIN FUNDAMENTALS

Caveat:

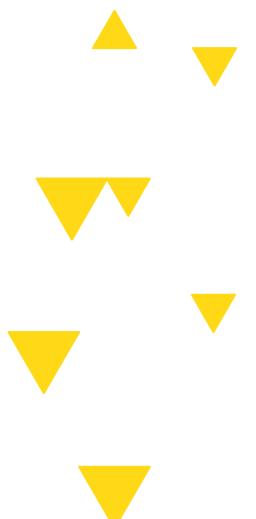
The blockchain is only as good as the information fed into it.

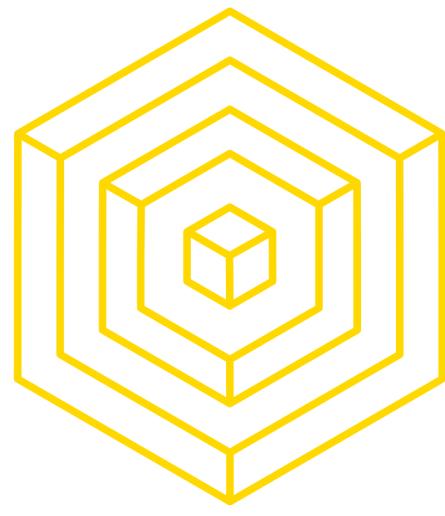
Countries investigating: **Georgia, Ukraine, Sweden**





COMMENTS?





PREDICTION MARKETS

A QUICK DEFINITION

Draws on the wisdom of the crowd

Ex: “Who will win the 2020 Presidential Election? Zuckerberg or Trump?”

1. Replace shares with bets
2. Random oracles report


Let's Play Politics™

[Markets](#)
[Analysis](#)
[About](#)

Who will win Arizona's 8th District GOP primary?

[TRADE NOW](#)



The Prediction Market for Politics

PredictIt is a real-money political prediction market, a stock market for politics. A project of Victoria University of Wellington, PredictIt has been established to research the way markets can forecast future events. Our job is to study the wisdom of the crowd, yours is to use your skill and knowledge to get ahead.

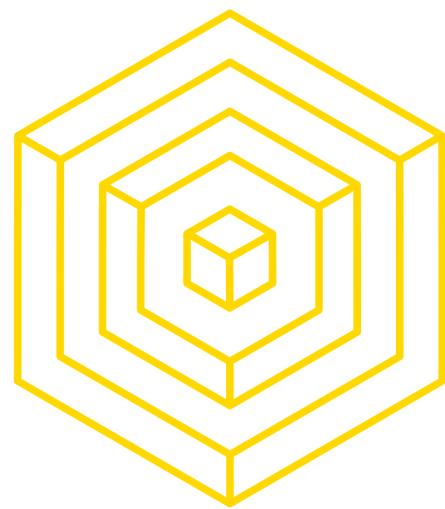
So you think maybe you know better than the rest of us? [Sign Up](#) to test your wits and be a part of this select group. Let's Play Politics!

To learn more about how we support academic research, visit our [Research Page](#).

Centralized Prediction Market



AUTHOR: GILLIAN CHU



PREDICTION MARKETS

AND DECENTRALIZING THEM....

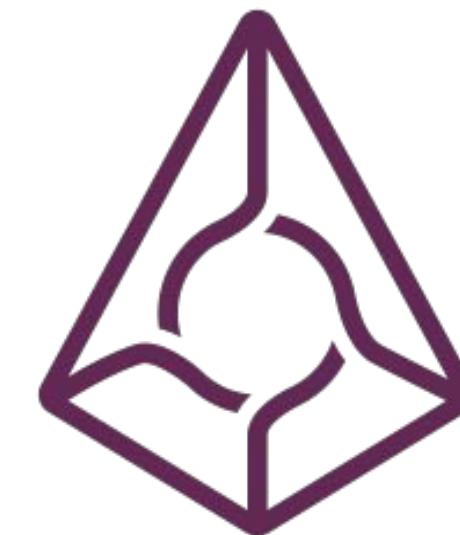
“A service that never crashes, a service that’s completely transparent...”

Benefits:

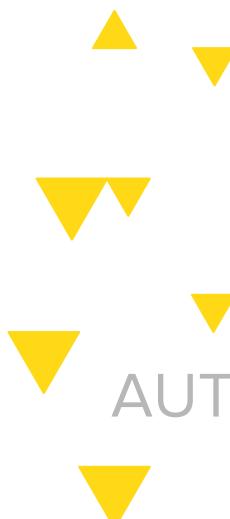
- no restrictions on market creation
- shared liquidity pool
- censorship-resistant
- automatic, trustless payments



GNOSIS

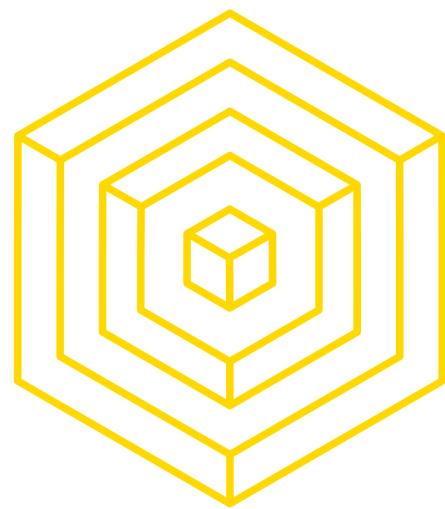


augur



AUTHOR: GILLIAN CHU

BLOCKCHAIN FUNDAMENTALS LECTURE 5

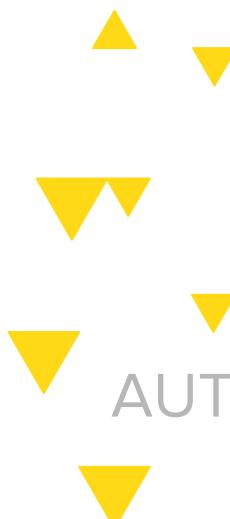


PREDICTION MARKETS

AND DECENTRALIZING THEM....

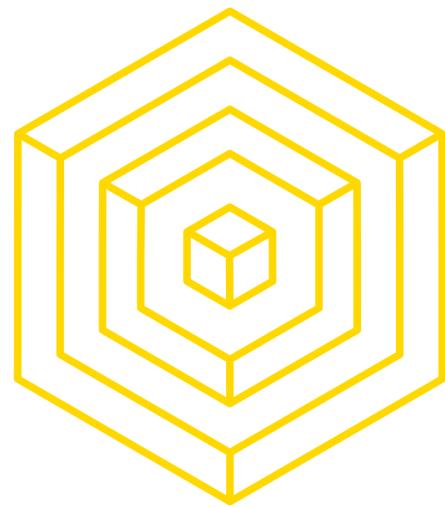
Use Case 1:

- Cost-effective way to “buy” information
- Bet for/against to incentivize those with insider information



AUTHOR: GILLIAN CHU

BLOCKCHAIN FUNDAMENTALS LECTURE 5



PREDICTION MARKETS

AND DECENTRALIZING THEM....

Use Case

- Co...
- Bet fo...

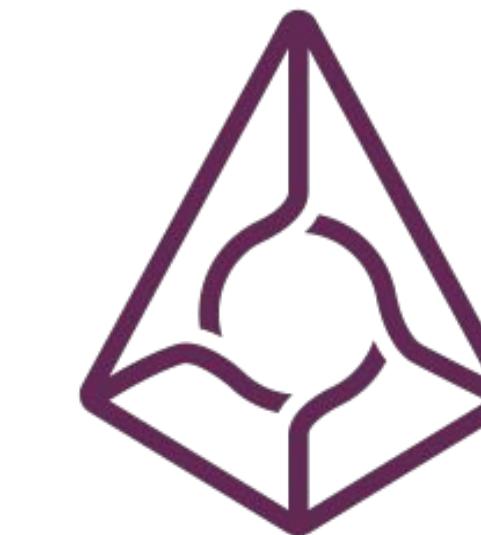
inform...

Will this movie
be a flop?

der



GNOSIS

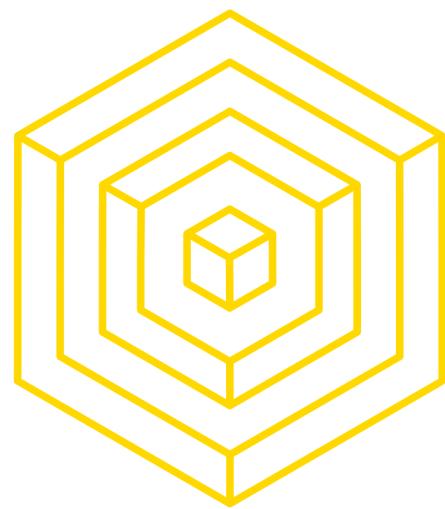


augur

▼
▼
▼
▼
▼

AUTHOR: GILLIAN CHU

BLOCKCHAIN FUNDAMENTALS LECTURE 5



PREDICTION MARKETS

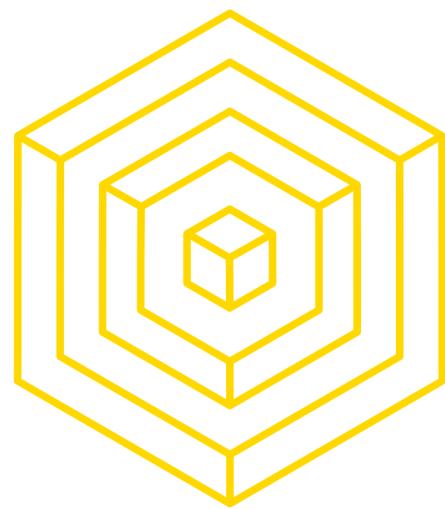
OTHER USE CASES

- **Hedging & Insurance**
 - “Will my house burn down?”
- **Security Bug Bounty**
 - “If my company is hacked, will the hacker reveal the vulnerability in a whitehat manner?”
- **ICO Signaling**
 - “Will my ICO deliver products on time?”



AUTHOR: GILLIAN CHU

BLOCKCHAIN FUNDAMENTALS LECTURE 5



PREDICTION MARKETS

FUTARCHY & CRYPTOECONOMICS

Vote Values, but Bet Beliefs

An Introduction to Futarchy

Posted by [Vitalik Buterin](#) on August 21st, 2014.

How manipulation-resistant are Prediction Markets?

Our Undertaking in Empirical Cryptoeconomics



Matt Liston [Follow](#)
Crypto native
Apr 28, 2016 · 3 min read

An Introduction to Cryptoeconomics and Futarchy experiments on Gnosis

Over the next few months we will be running several experiments on the Gnosis prediction market platform thanks to a generous Ethereum DEV grant. These experiments will test cryptoeconomic hypothesis and methods related to market manipulations and Futarchy and were inspired by Vitalik's reddit post on Empirical Cryptoeconomics. Each experiment consists of a market on

▲
▼

120

💬

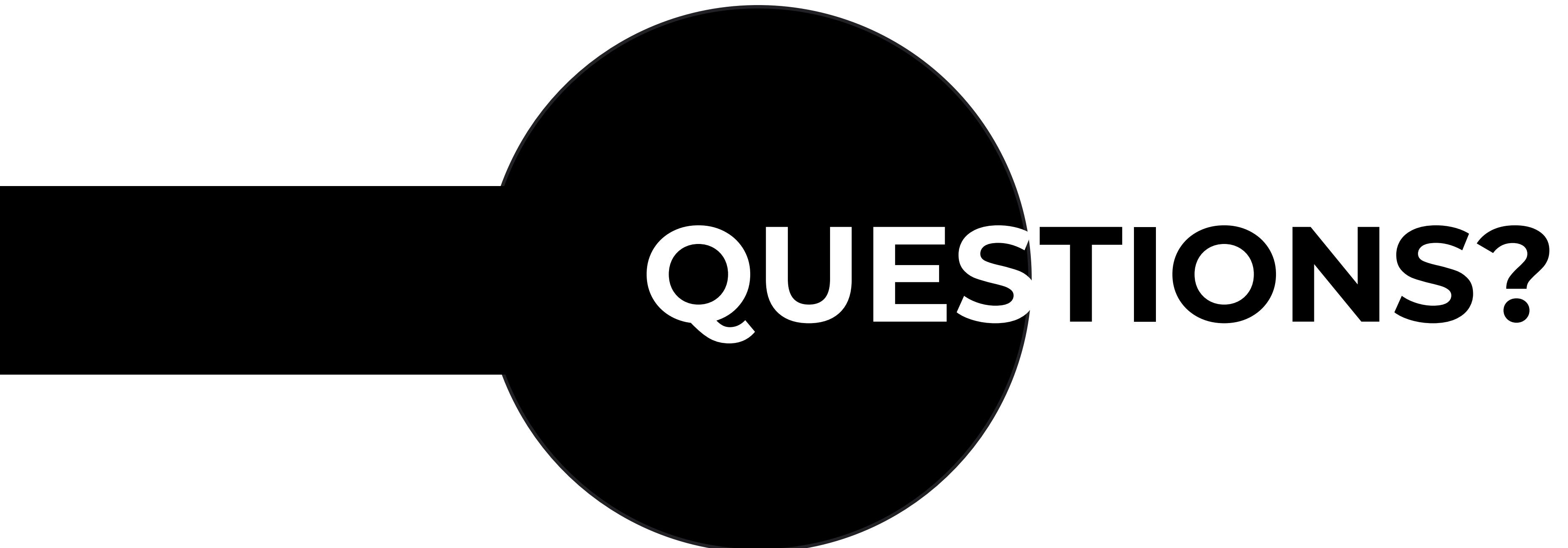
Empirical Cryptoeconomics self.ethereum

Submitted 2 years ago * by [vbuterin](#) Just some guy

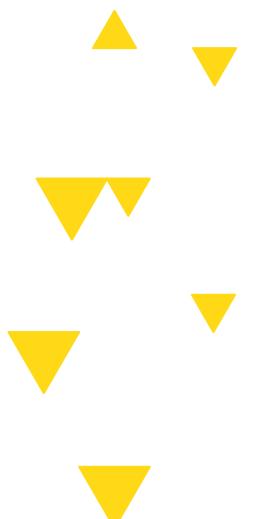
Now that the Ethereum infrastructure is becoming increasingly mature, I thought that it would make sense to try to do some empirical tests of just how effective

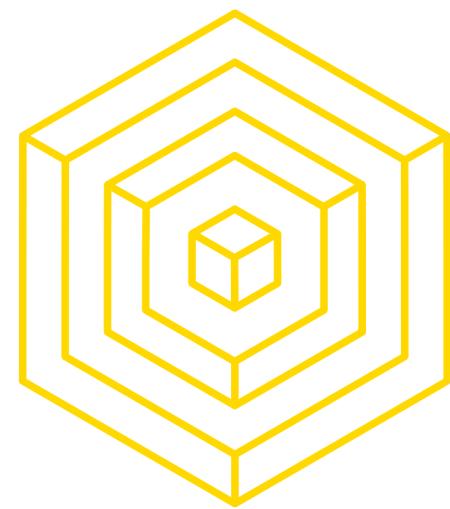
AUTHOR: GILLIAN CHU

BLOCKCHAIN FUNDAMENTALS LECTURE 5



QUESTIONS?





SUPPLY CHAIN AND PROVENANCE

BLOCKCHAIN FUNDAMENTALS

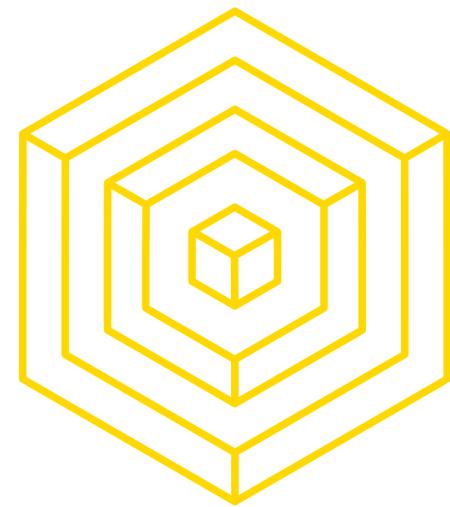
Problem: “Blood” diamonds

The **Kimberley Process** is a governmental effort requiring participants to certify the origin of their diamond.

- Corrupt officials take bribes to sign certifications
- Complex supply chains mask actual trails

Everledger: provenance of diamonds





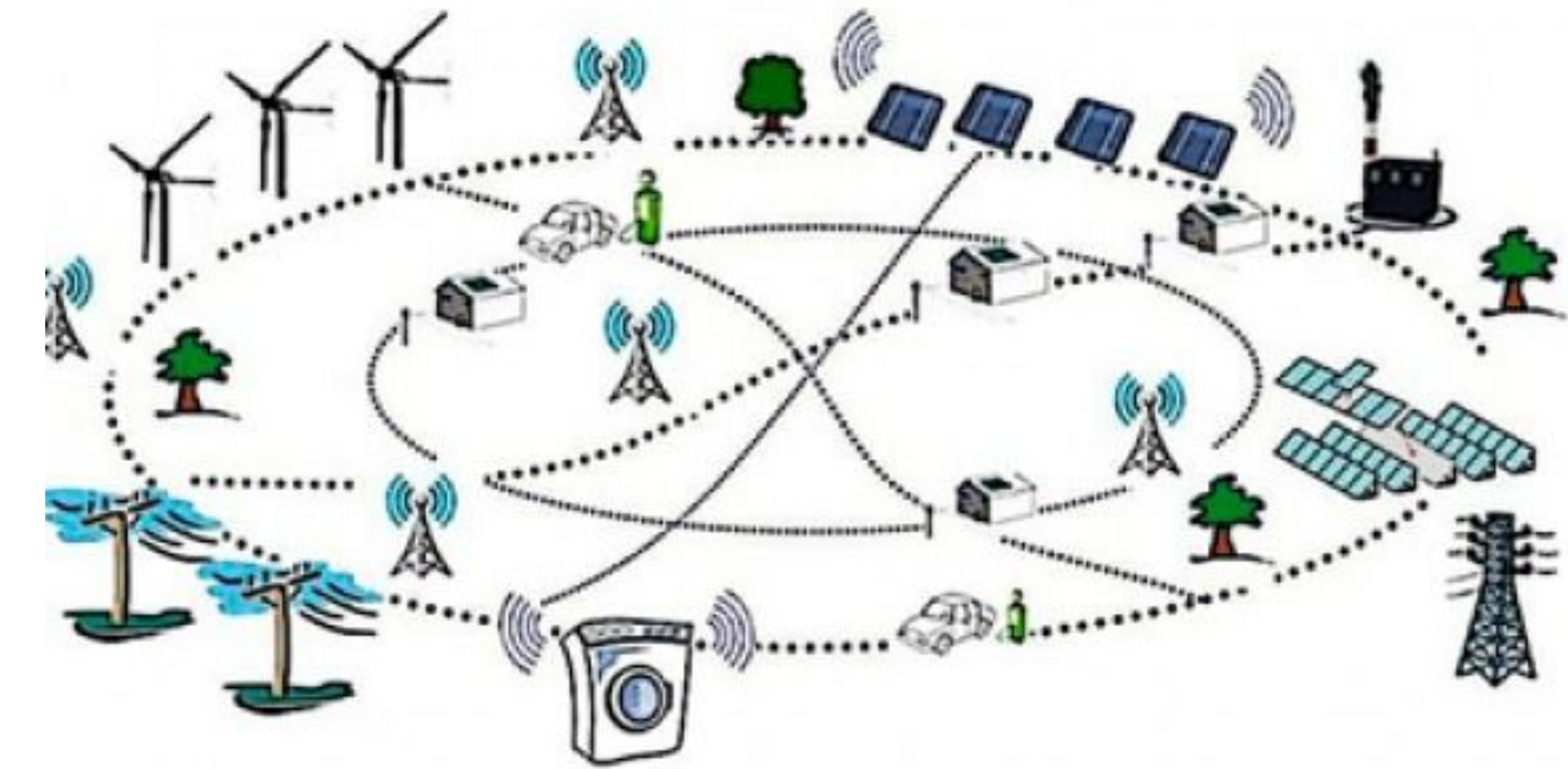
SMART ENERGY GRIDS

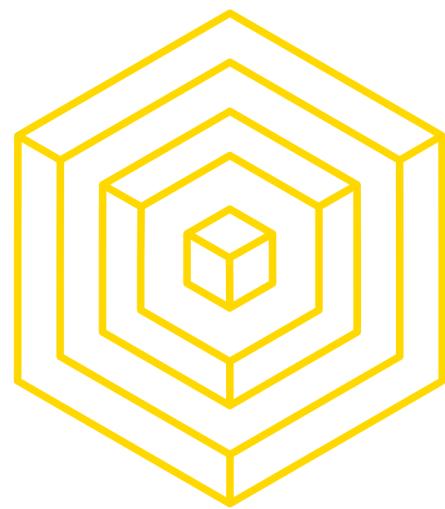
BLOCKCHAIN FUNDAMENTALS

Setup: Imbalances in energy across a community of houses
i.e. House A has excess heat, House C lacks heat

Problem: Need custom infrastructure from A \Rightarrow B \Rightarrow C

Solution: Ethereum smart contract for financial commitments





AFTER HOURS TRADING

BLOCKCHAIN FUNDAMENTALS

Tokenizing shares of stock

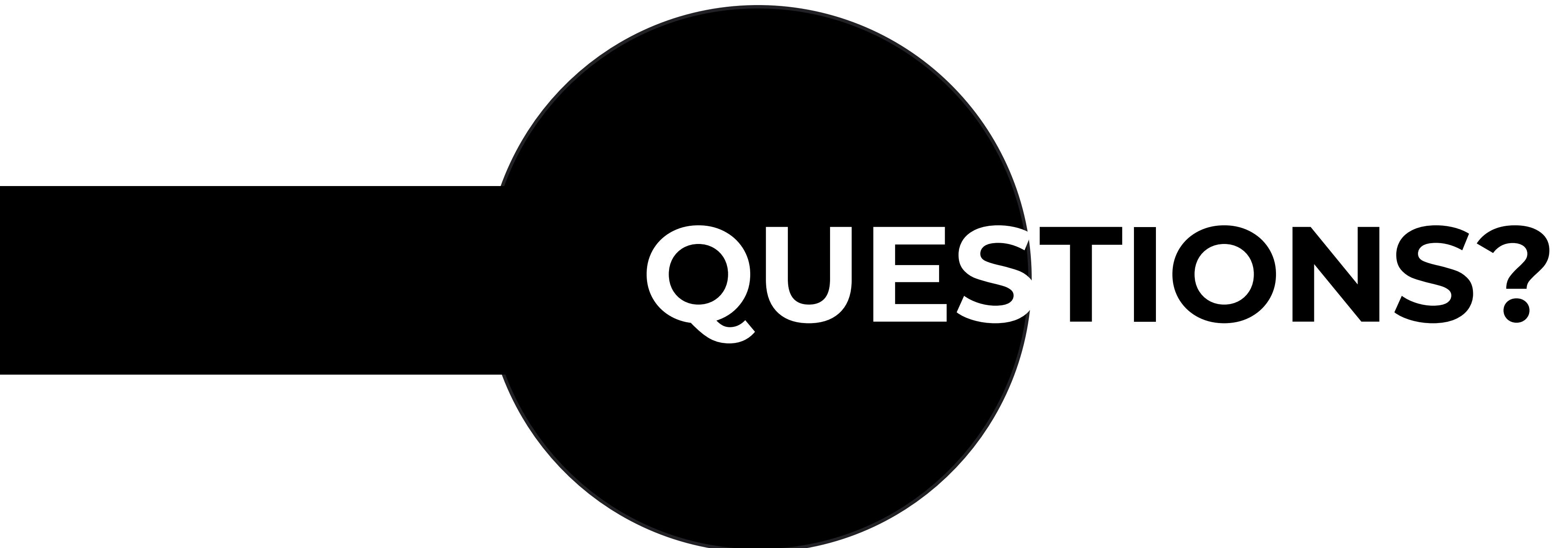
Problem: “Liquidity Problem”

Regulation restricts trading to only using limit orders

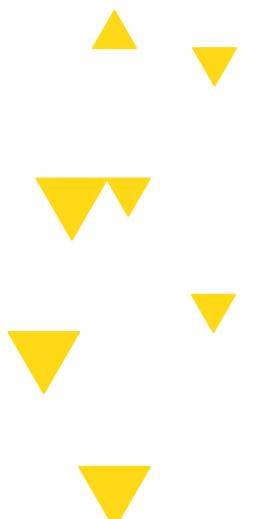
Blockchain Solution:

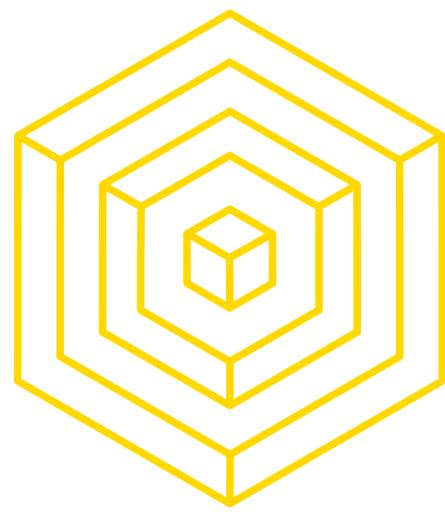
- Stock is subtracted from user's account and converted into stock tokens
- Stock tokens can be traded worldwide (even after market close!)
- Redeemable via a legal contract





QUESTIONS?





COMMENTARY ON USE CASES

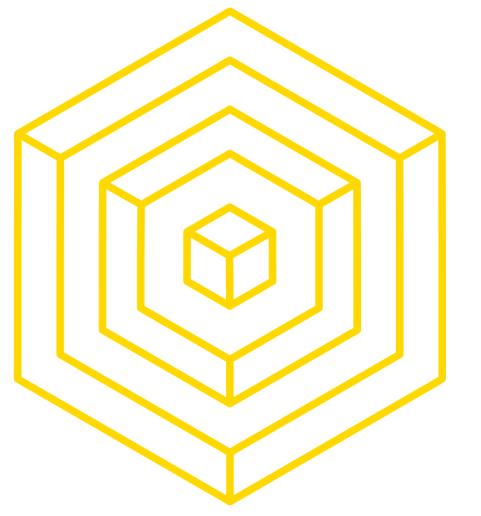
BLOCKCHAIN FUNDAMENTALS

Blockchain vs. The Internet



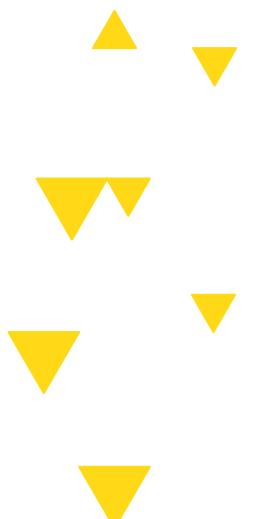
UPDATED: GILLIAN CHU

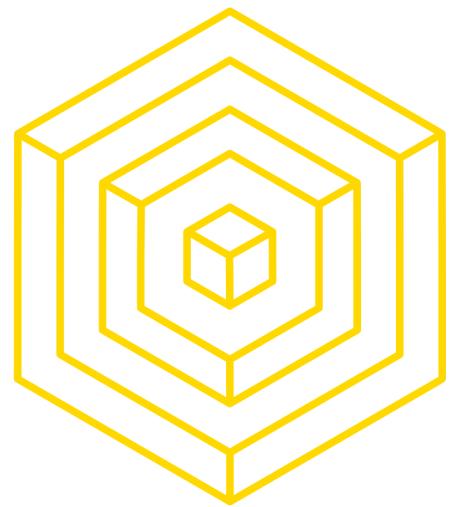
BLOCKCHAIN FUNDAMENTALS LECTURE 5



5

STATE OF THE ECOSYSTEM





THE COMMUNITY LEADERS



Vitalik Buterin



Bobby Lee



Roger Ver



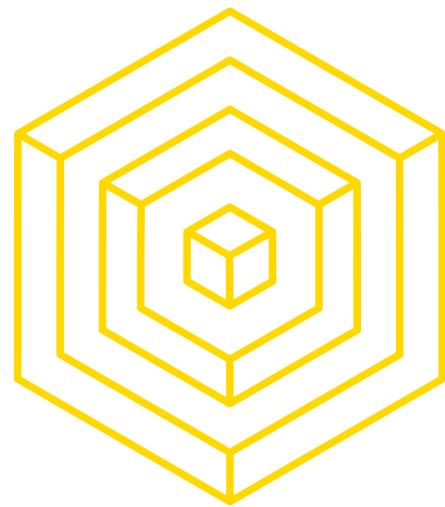
Nick Szabo



Marc Andreessen



Andreas Antonopoulos



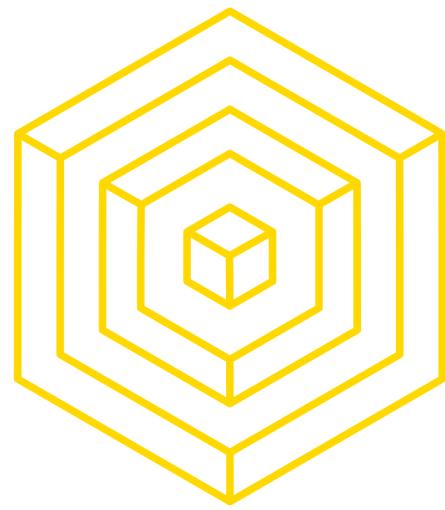
CONSENSYS

BLOCKCHAIN FUNDAMENTALS



AUTHOR: GILLIAN CHU

BLOCKCHAIN FUNDAMENTALS LECTURE 5



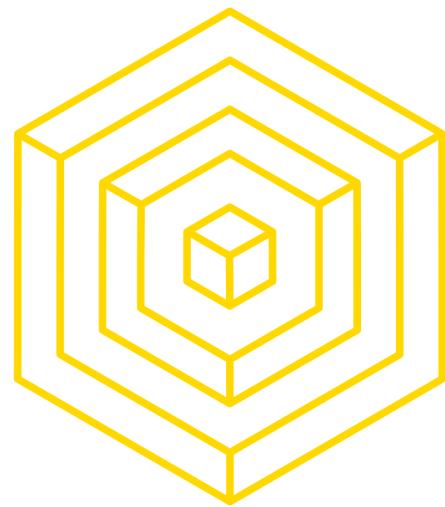
CONSENSYS

BLOCKCHAIN FUNDAMENTALS



AUTHOR: GILLIAN CHU

BLOCKCHAIN FUNDAMENTALS LECTURE 5



HOMEWORK

READINGS

- **Finish** reading the Ethereum Whitepaper
 - after “Miscellanea and Concerns”
- Selfish Mining:

<https://bitcoinmagazine.com/articles/selfish-mining-a-25-attack-against-the-bitcoin-network-1383578440/>

- Optional: [Take a look at these cool Ethereum Applications](#)
- Optional: [Ethereum Yellow Paper](#)

HOMEWORK

- Come up with a dApp (decentralized app) idea or blockchain use case! Try to be as creative and specific as possible, and justify the use of a blockchain. Due by next Saturday, March 3rd at 11:59 PM. Check Piazza for submission link.