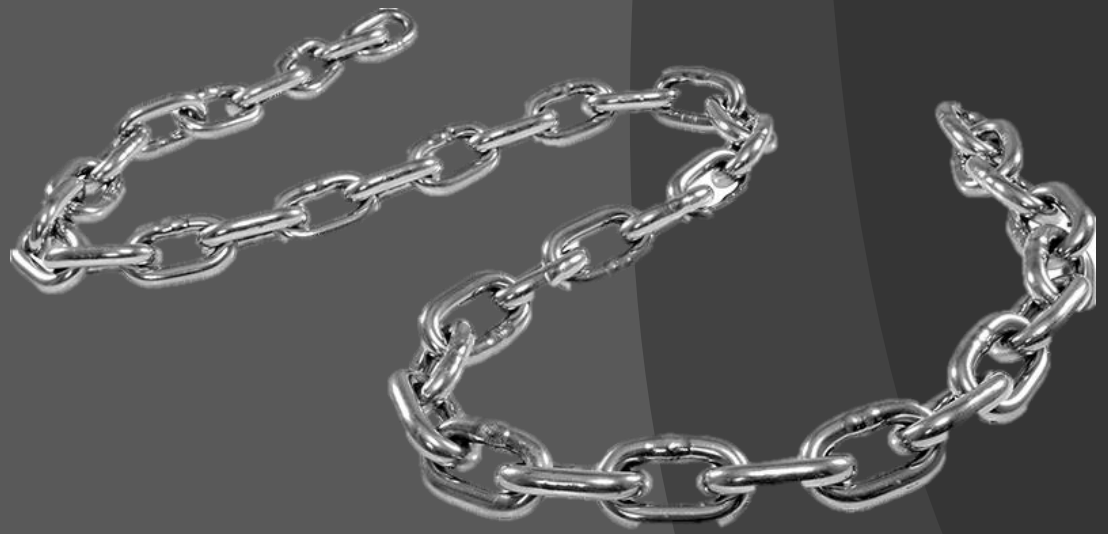
The blockchain and you

Cloudbrew

Benny Michielsen
Hans Peeters



Benny Michielsen

benny.michielsen@infosupport.com

@bennymichielsen

Consultant

MCT

Less talk more code

Programming, Motherfucker

Hans Peeters

hans.peeters@infosupport.com

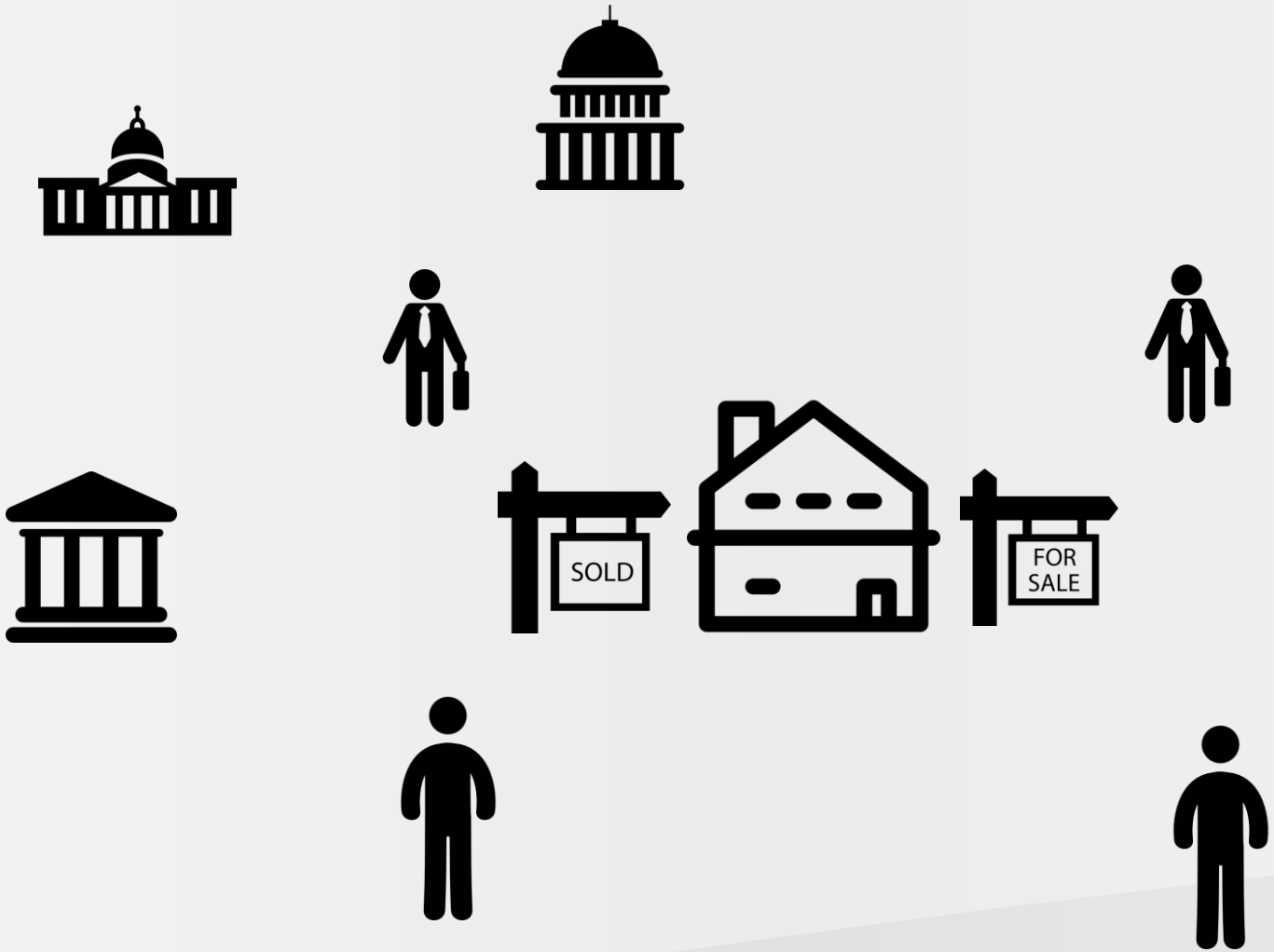
@hspeeters

Consultant

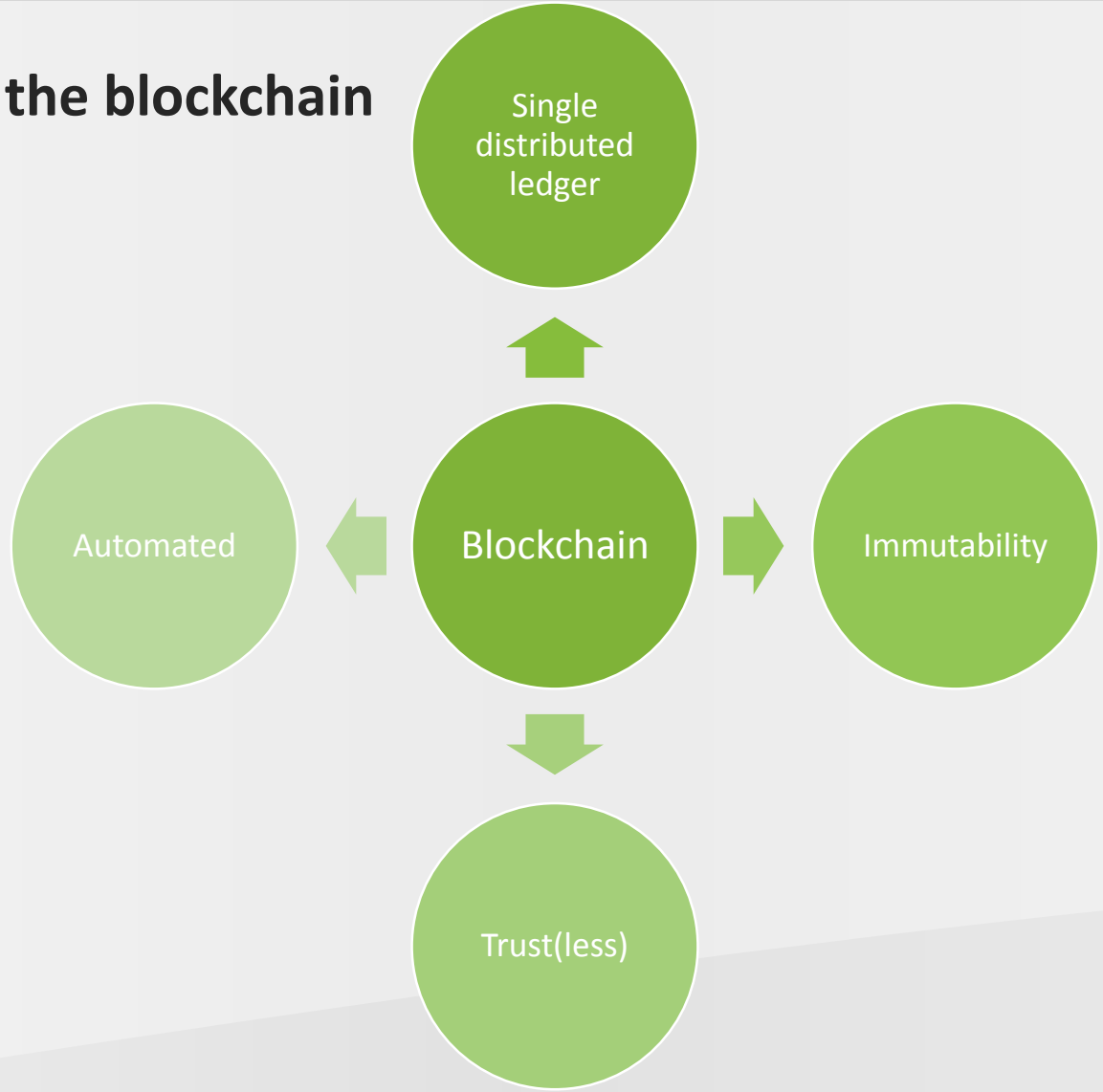
MCT

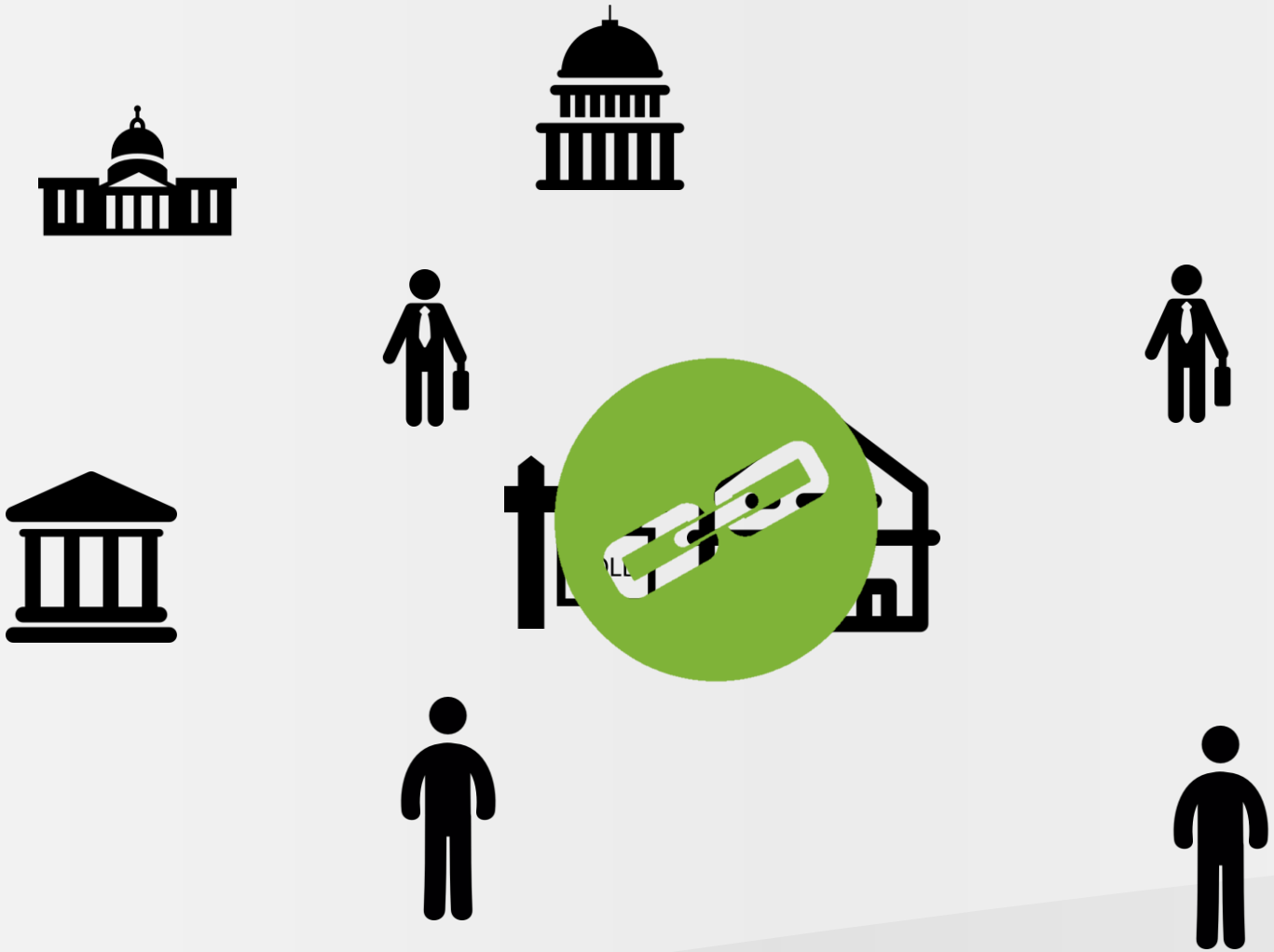
Lean

Mean


SOLD
FOR SALE

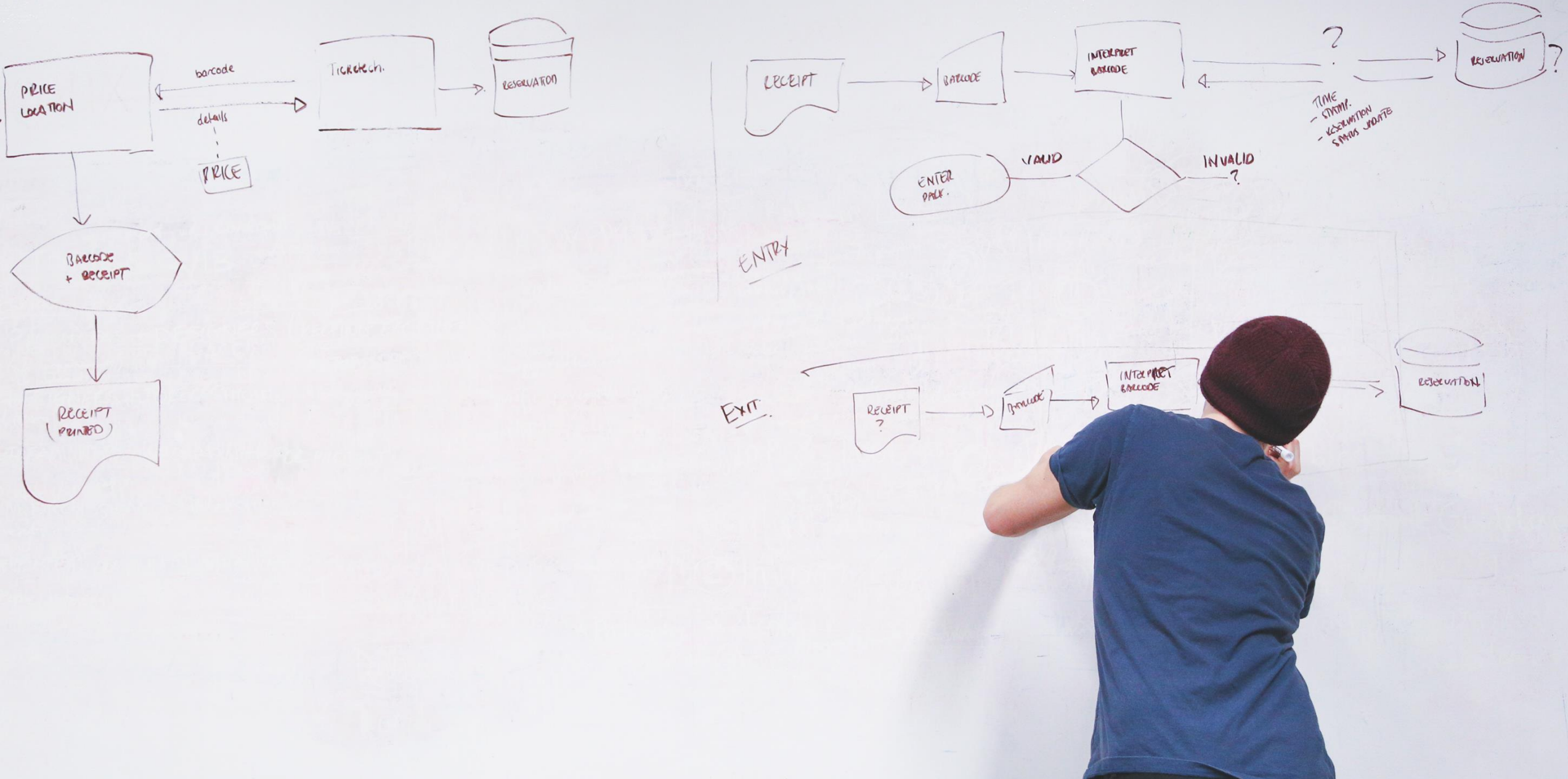
Core properties of the blockchain

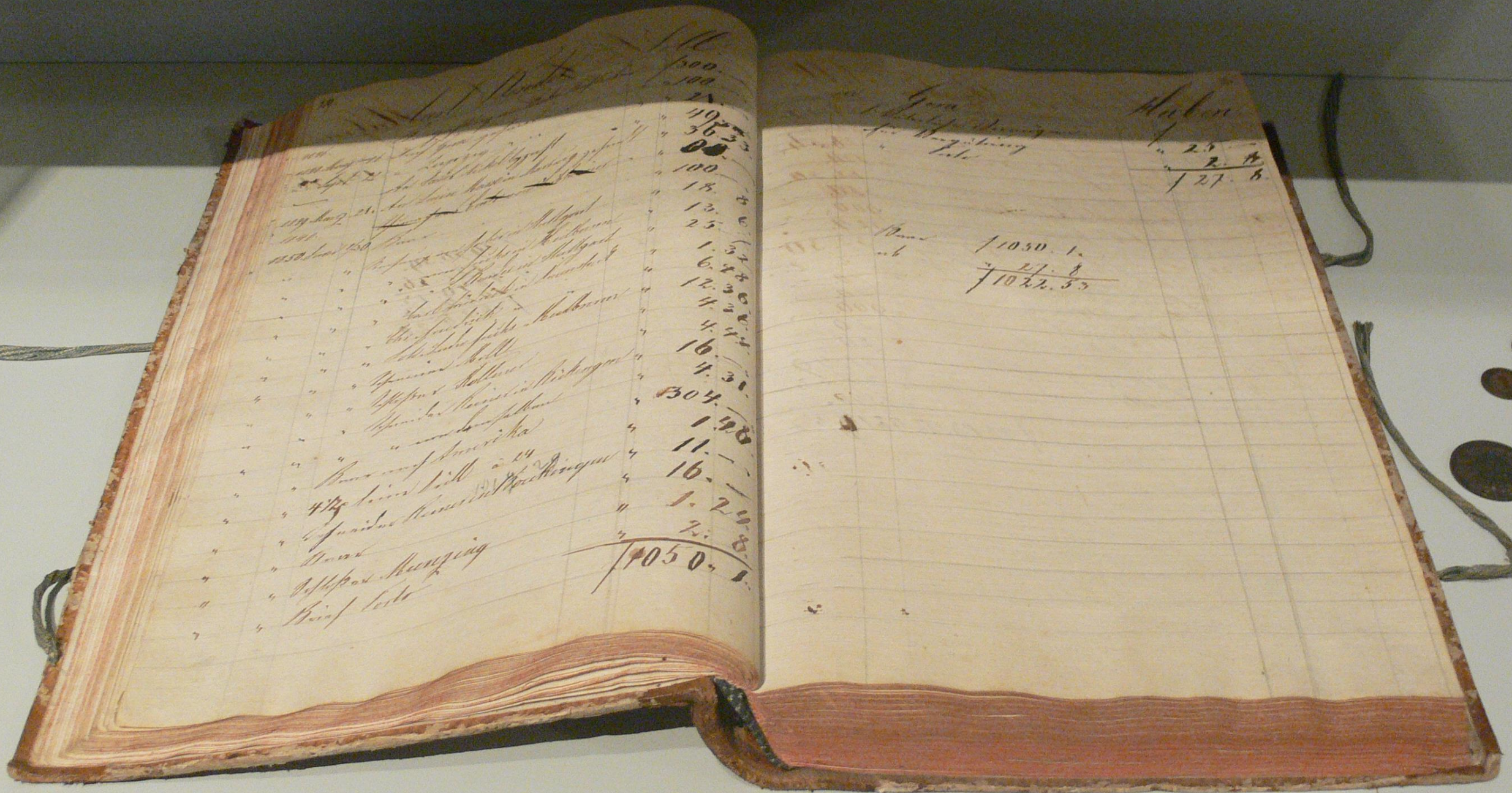




Blockchain technology can innovate in

- Trust industries
- Finance
  - Reduce transaction costs
- Company management
  - Transparency
  - Decentralized organizations
- IoT
  - Autonomous agents that transact with eachother
  - No central trust party needed
- Art
  - Micropayments for music
  - Certificates of authenticity
- Democracy
  - Liquid democracy
  - Voting
- Privacy, regain control of your data
  - Internet behaviour
  - Health data
  - Government identity
- New business models
  - Metering economy
  - Prediction markets
  - Real sharing economy
- ……!


PRICE
LOCATION
barcode
details
PRICE
RESERVATION
BARCODE
+ RECEIPT
RECEIPT
(PRINTED)
RECEIPT
BARCODE
INTERPRET
BARCODE
RESERVATION
TIME
- STATUS
- RESERVATION
STATUS UPDATE
ENTER
PARK.
VALID
INVALID
?
ENTRY
EXIT
RECEIPT
?
BARCODE
INTERPRET
BARCODE
RESERVATION



Identify

- Where are my funds located?
- Where can I send funds to?

- Public Key Cryptography
- Random number to create a private key
- No central registry

- Unlikely to create private key twice 10^{48}
  - All atoms in the earth: 10^{50}

- Important to keep your private key safe!

Identify

- Mobile Apps

Airbitz, Bither, breadwallet, Green Address, Coin.Space, Copay

- Desktop Apps

Bitcoin Knots, Bitcoin Core, Armory, Electrum, mSIGNA, Copay

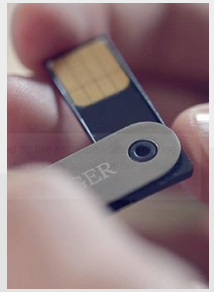
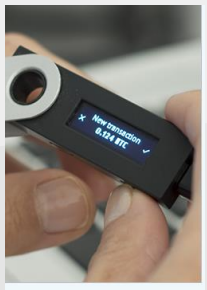
Bither, MultiBit HD, BitGo, Green Address

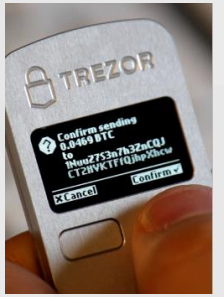
- Web Apps

BitGo, Green Address, Coin.Space, Xapo, Coinapult, Circle

Coinbase

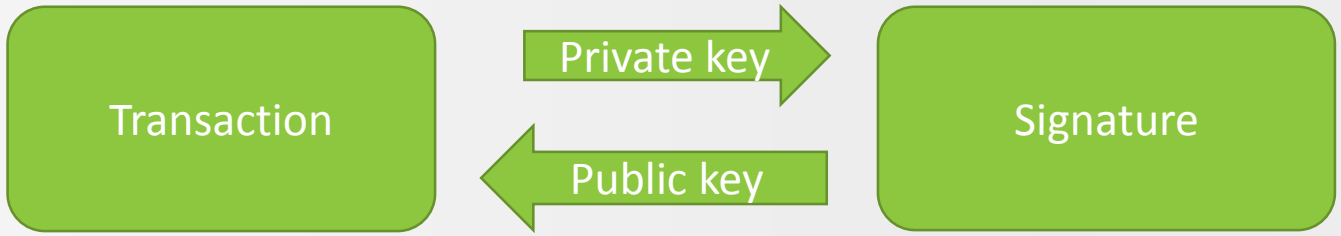
- Hardware



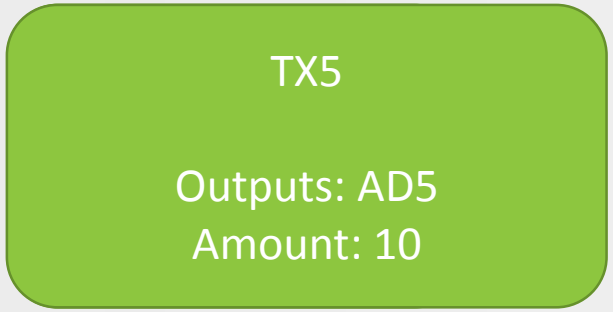
Signing Transactions

- How to validate integrity and source of a transaction?
- Public Key Cryptography

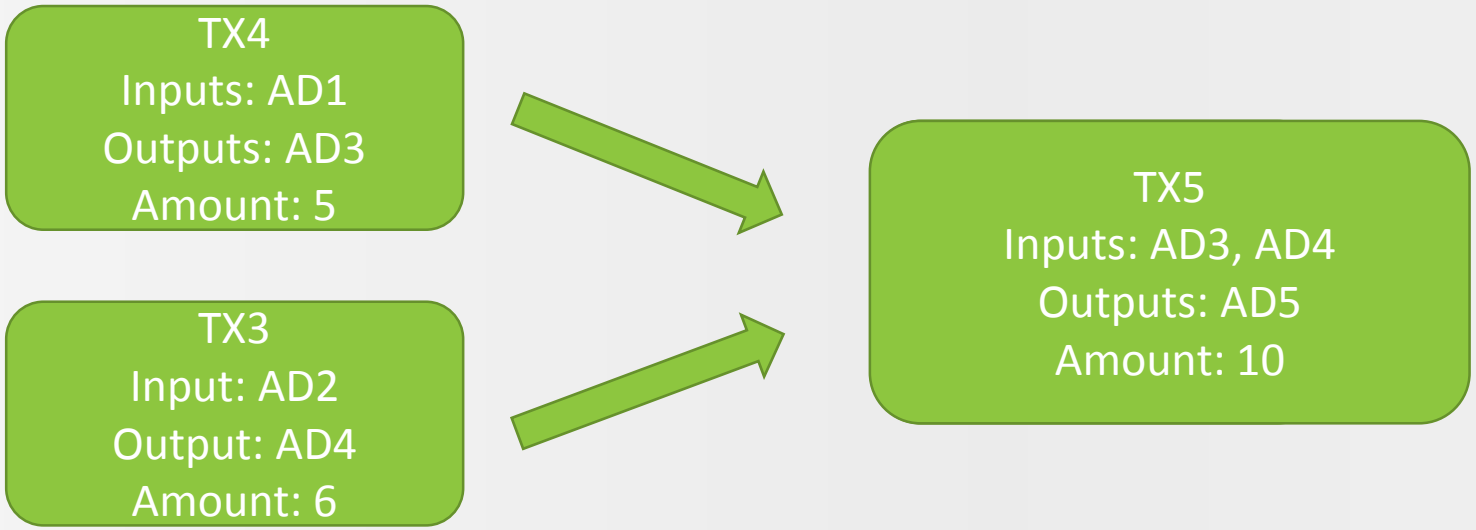


Transactions

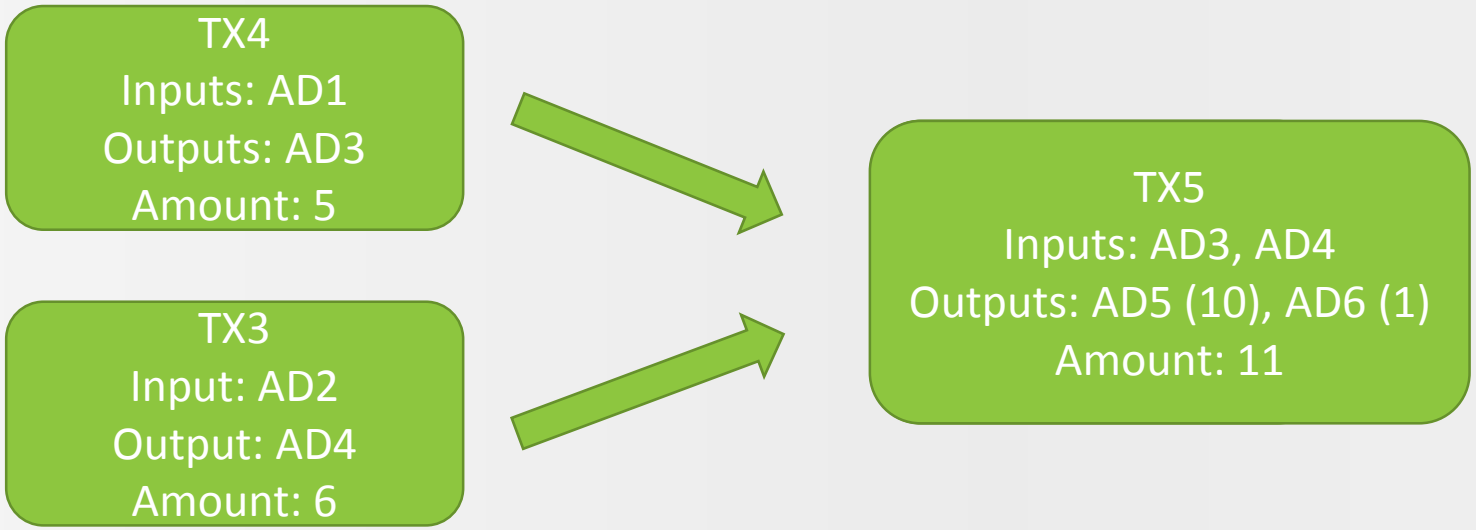
Transactions



Transactions



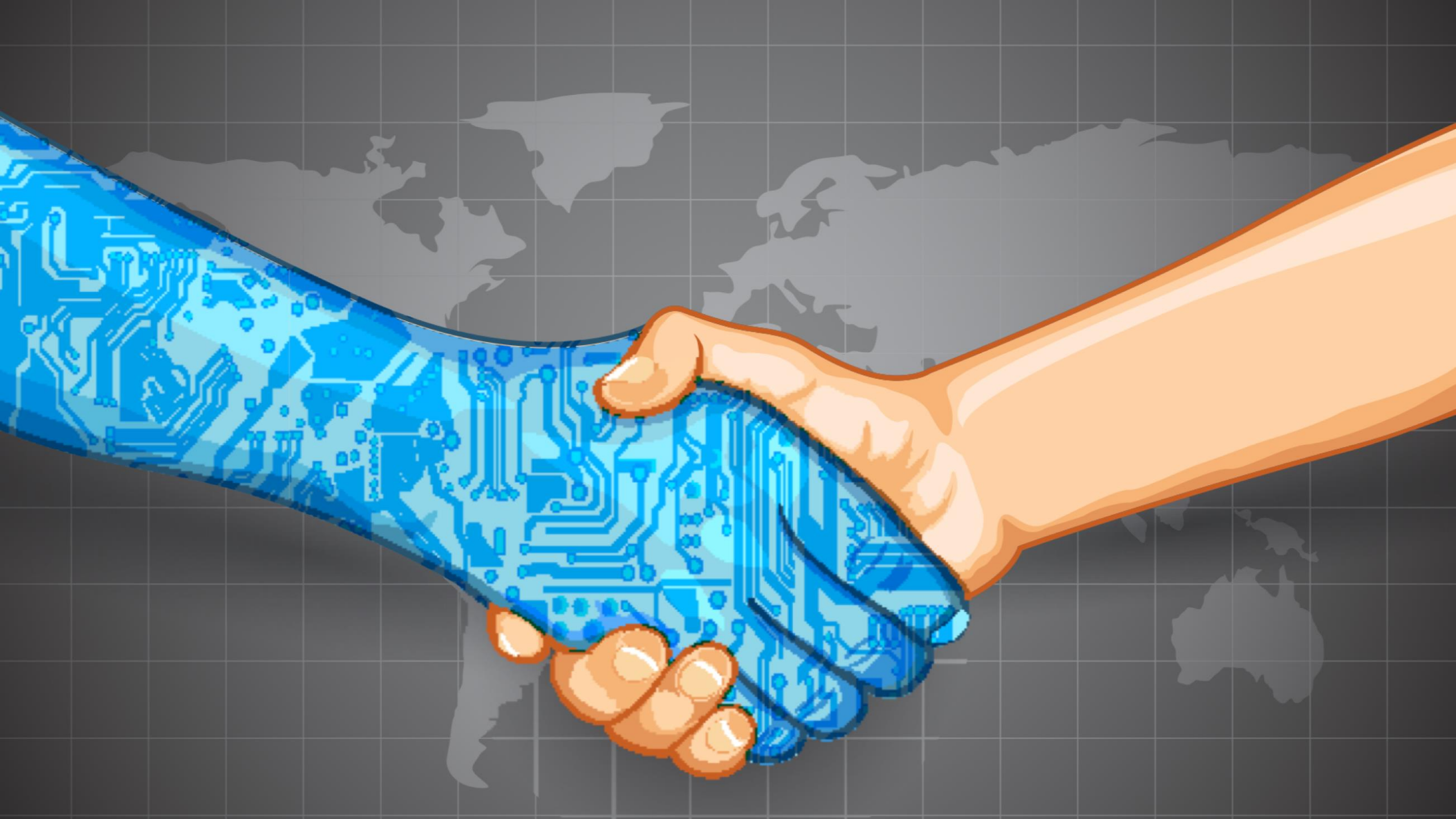
Transactions



Miners / Bookkeepers

- Who keeps track of the transactions?

- Miners
  - Receive transactions
  - Group transactions to form a new block
  - Hash previous block + hash new block + random number < puzzle hash
  - Goal: guess random number
    - Very compute intensive
    - Are rewarded for finding the number
  - One quintillion hashes per second




ZCASH


I THINK WE SHOULD BUILD A BLOCKCHAIN
UH-OH
DOES HE UNDERSTAND WHAT HE SAID OR IS IT SOMETHING HE SAW IN A TRADE MAGAZINE AD?
WHAT COLOR DO YOU WANT THAT BLOCKCHAIN?
I THINK MAUVE HAS THE MOST RAM.
E-mail: SCOTTADAMS@AOL.COM
© 1995 United Feature Syndicate, Inc.(NYC)

Ethereum

- Bitcoin has limited possibilities to program against

- Ethereum blockchain
  - Programmable by design
  - Distributed “computer”
  - Develop smart contracts/dapps
  - Most popular development blockchain

- Each node runs contracts and verifies result
  - Calculation costs gas (payed for with Ether)
- Not fast, but very reliable
- Deploy contract to address
- Trigger contract by sending ether to that address

```
pragma solidity ^0.4.0;

contract MyGame {

    enum State { None, Rock, Paper, Scissor, Lizard, Spock }

    address public player1;
    address public player2;
    address public winner;
    State public lastPlayedHand1;
    State public lastPlayedHand2;

    function MyGame(address player1addr, address player2addr{
        lastPlayedHand1 = State.None;
        lastPlayedHand2 = State.None;
        player1 = player1addr;
        player2 = player2addr;
    }

```

```
20      function playHand(State hand){
21          if(winner != player1 && winner != player2)
22          {
23              if(msg.sender == player1 && lastPlayedHand1 == State.None)
24              {
25                  lastPlayedHand1 = hand;
26              }
27              else if(msg.sender == player2 && lastPlayedHand2 == State.None)
28              {
29                  lastPlayedHand2 = hand;
30              }
31              if(lastPlayedHand1 != State.None && lastPlayedHand2 != State.None)
32              {
33                  declareWinner();
34              }
35          }
36      }
37
```

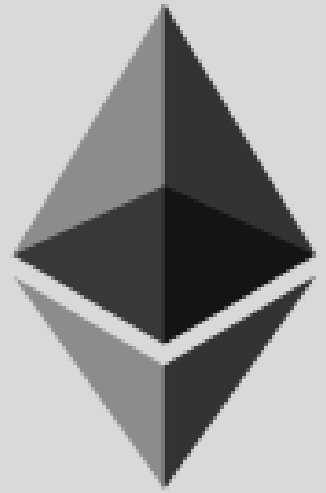
```
38      event Winner(address winner, address loser, State winnerState, State loserState);
39
40      event Draw(State draw);
41
42      function declareWinner(){
43          if(lastPlayedHand1 == lastPlayedHand2){
44              var playedState = lastPlayedHand1;
45              hands.push(Hand({player1Hand: lastPlayedHand1, player2Hand: lastPlayedHand2}));
46              lastPlayedHand1 = State.None;
47              lastPlayedHand2 = State.None;
48              Draw(playedState);
49          }
50          else{
51              State winningHand;
52              State losingHand;
53              address loser;
54              //find the winner
55               Winner(winner, loser, winningHand, losingHand);
56          }
57      }
```

Create and deploy

Solidity compiler:

- Input:
  - Smart contract source
- Output:
  - ABI
  - Binary

- Truffle
  - Provides easier javascript integration
  - Handles the details
  - Focus on app and contract development



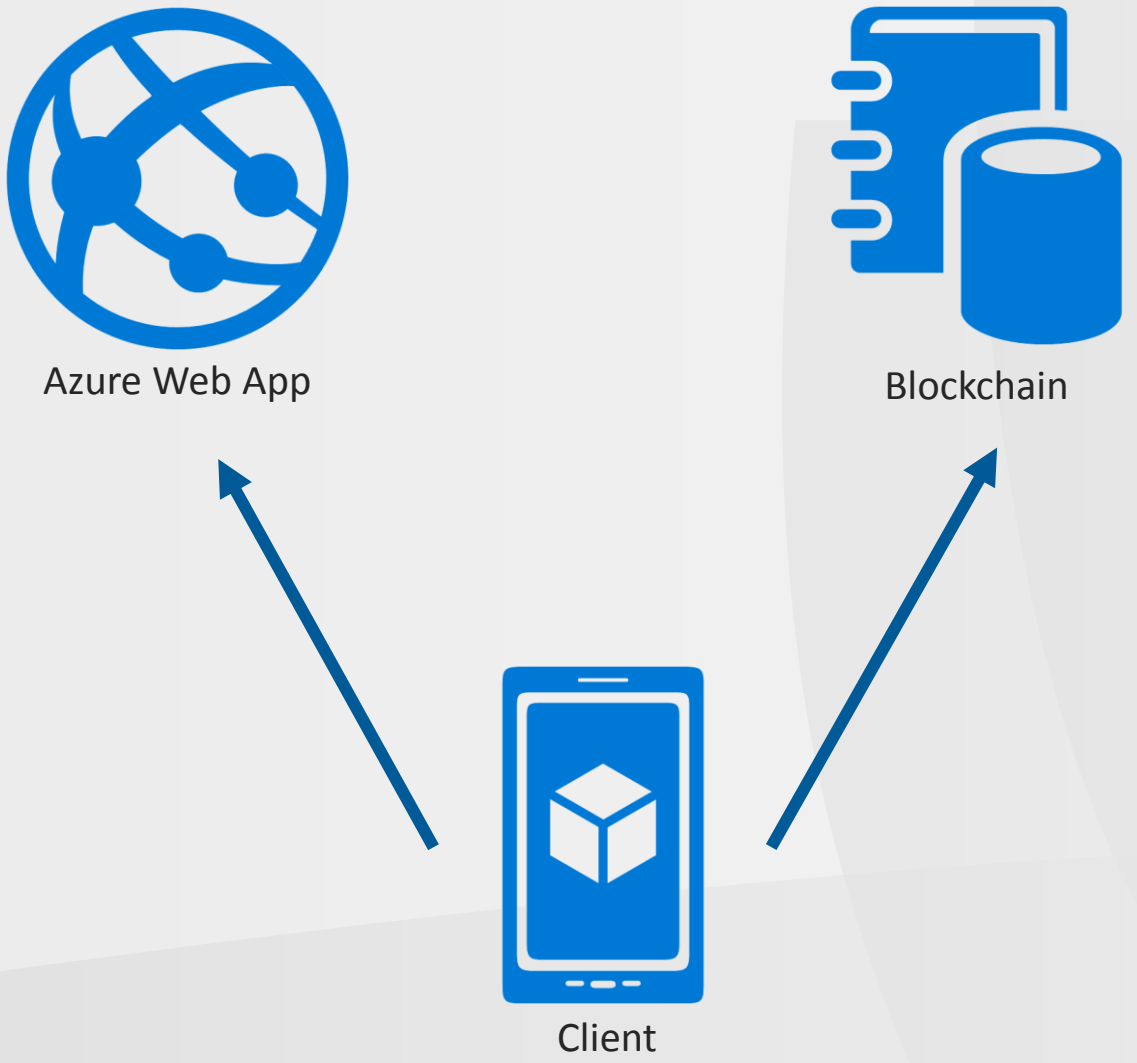
```
var gameInstance;
Game.new(player1, player2, {from: player1, gas: 400000, gasPrice: 20000000000})
  .then(function(gameInstance){
      gameInstance = Game.at(gameInstance.address);

     gameInstance.Winner(function (winnerError, winnerResult) {
        if (!winnerError) {
          console.log(winnerResult);
          console.log('winner: ' + winnerResult.args.winner);
          console.log('loser: ' + winnerResult.args.loser);
          console.log('winner: ' + winnerResult.args.winnerState);
          console.log('loser: ' + winnerResult.args.loserState);
        }

      return gameInstance.playHand(1,{from: player1, gas: 400000, gasPrice: 20000000000});
  })
  .then(function(){
    return gameInstance.playHand(2,{from: player2, gas: 400000, gasPrice: 20000000000});
  })
  .catch(function(error){
    console.log(error);
  });
```

Why on the blockchain?

- Distributed hosting
- Everyone can check the game code
- Everyone can verify there was no cheating

Our dapp architecture

- Web app in azure
- Client generates key
- Then client connects directly to the blockchain

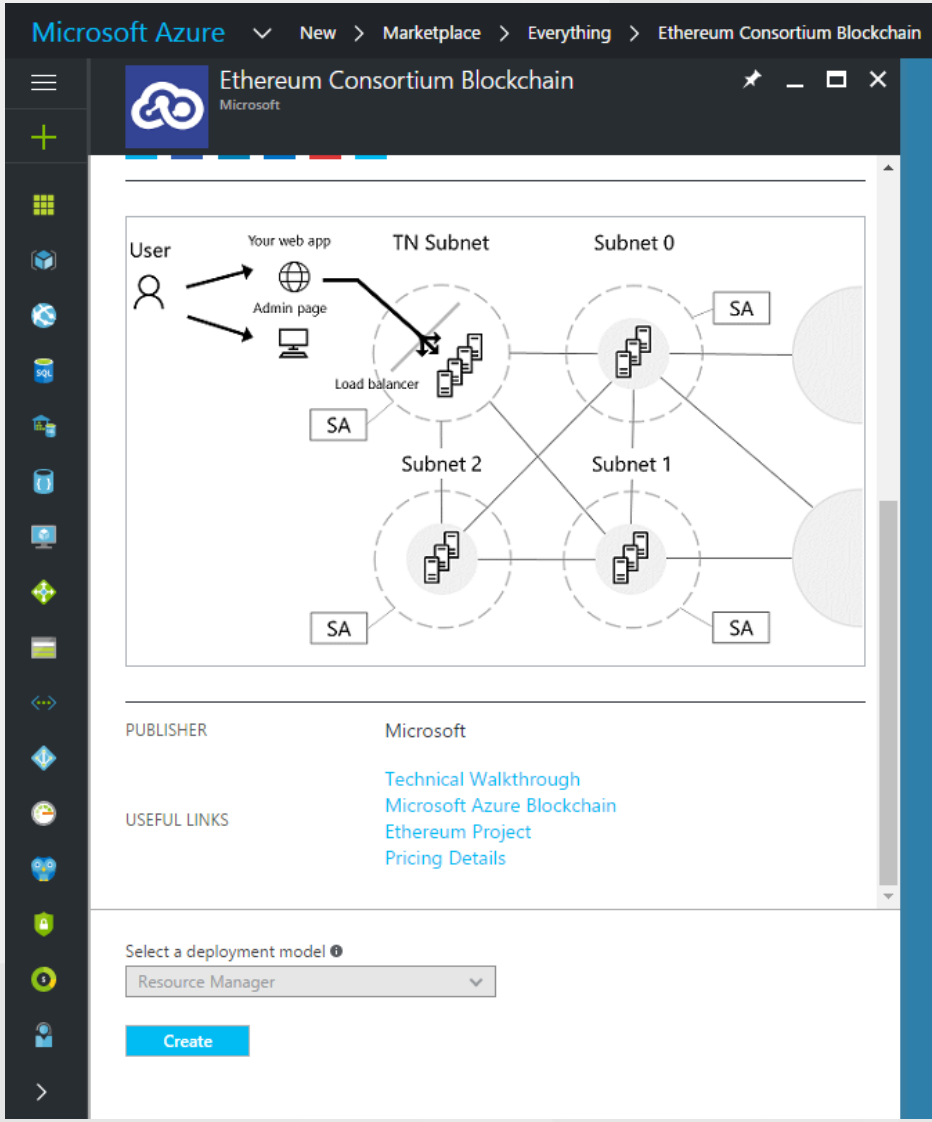


Azure BaaS – Blockchain as a Service

- Ecosystem for blockchain development/test/deployment
- Open to all blockchain partners
- Fail fast: evaluate different platforms quickly

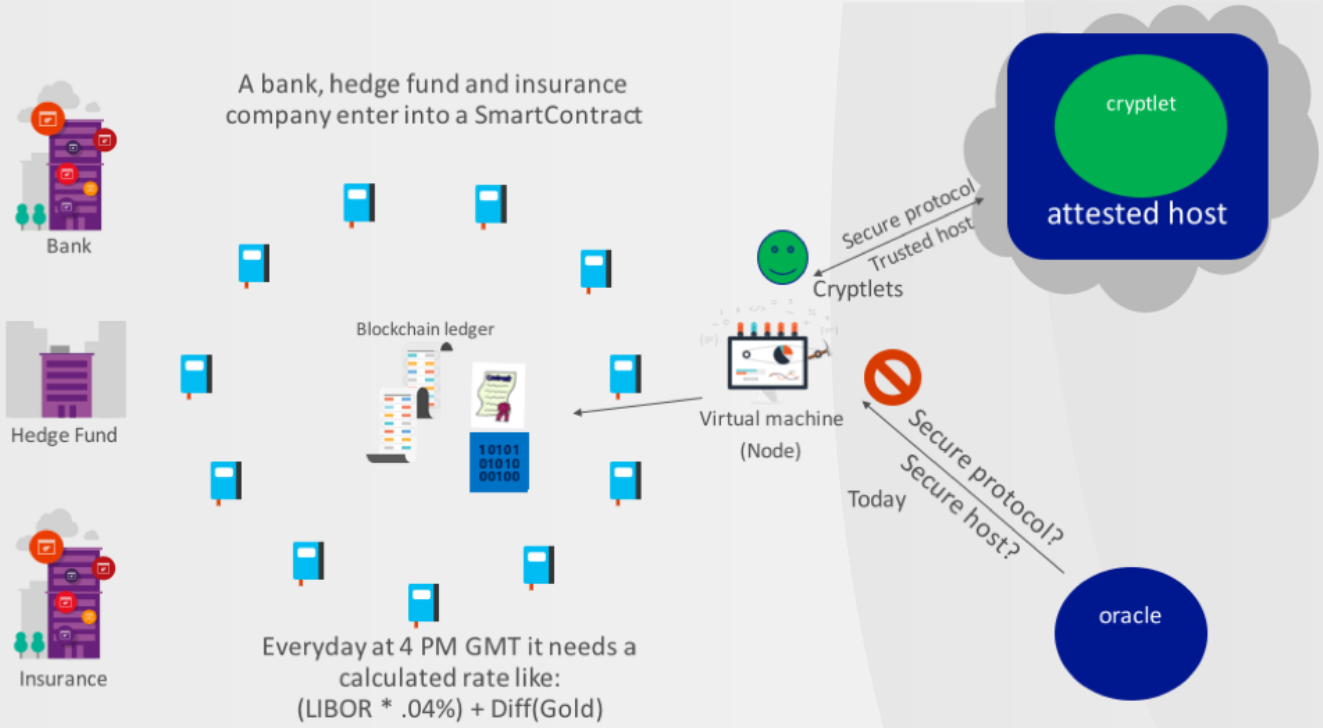
- Dev/Test labs
- Easily deploy private, public, permissioned, or consortium blockchains

- Works already!



Azure BaaS – Project Bletchley

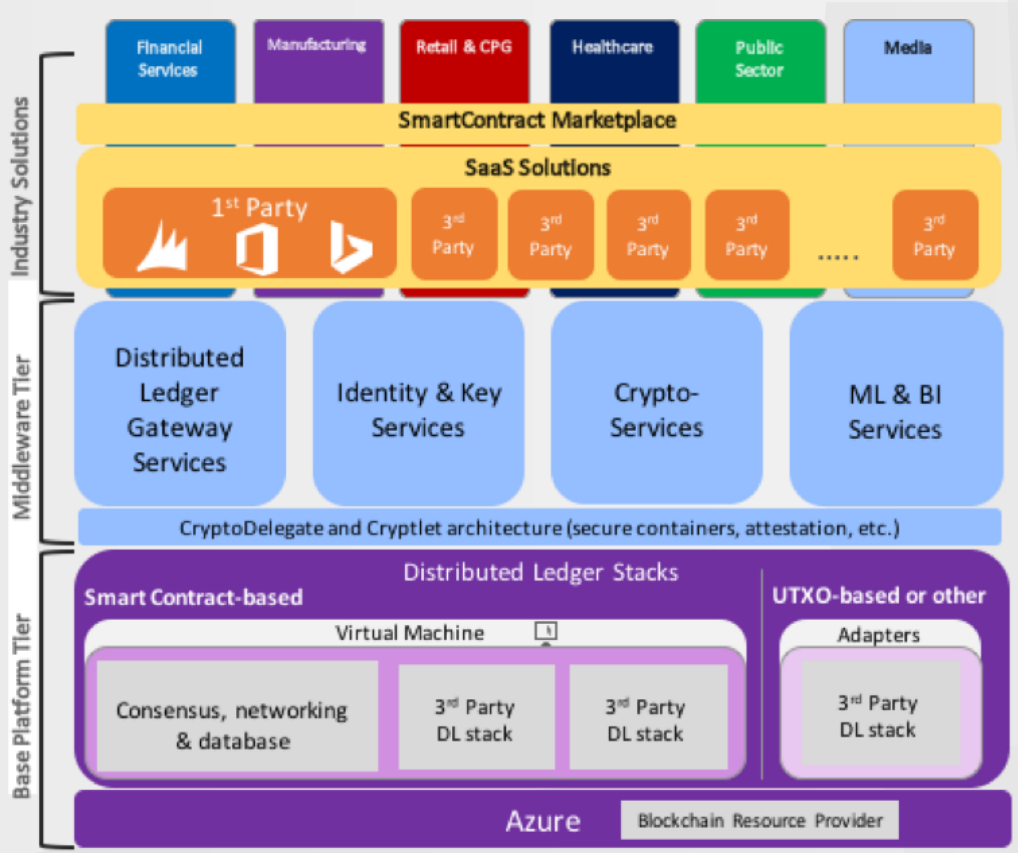
- Blockchain 1.0
  - Simple state machine/distributed database
  - Bitcoin
  - No programming
- Blockchain 2.0
  - Smart contracts
  - Ethereum
  - Oracles for outside interaction: not secure/trusted
- Blockchain 3.0
  - Cryptlets for secure outside interaction



Azure BaaS – Project Bletchley

- Blockchain fabric: more middleware for blockchains
  - Identity and certificate services
  - Encryption services
  - Cryptlet services,
  - Blockchain gateway services
  - Data services
  - Management and operations

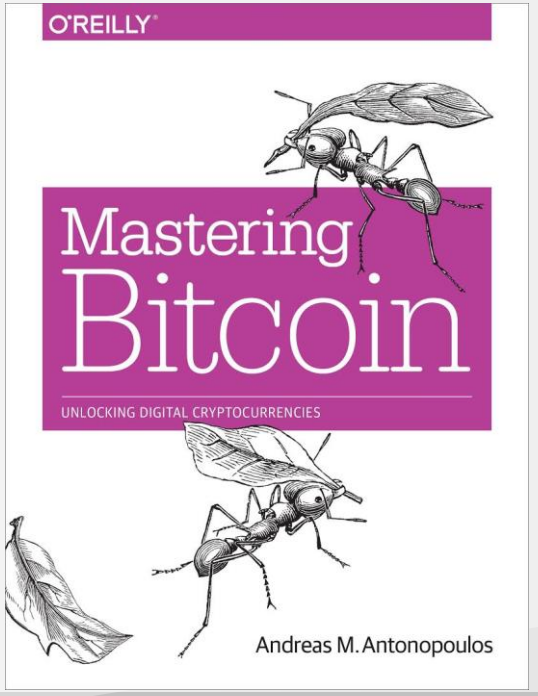
- Work in progress

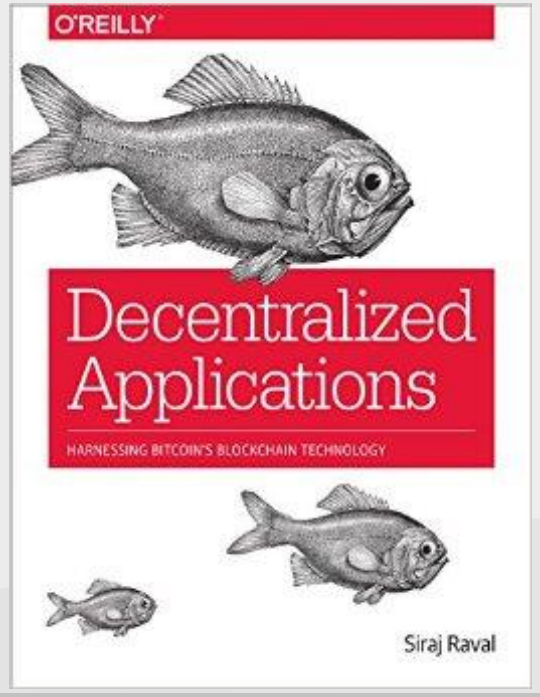


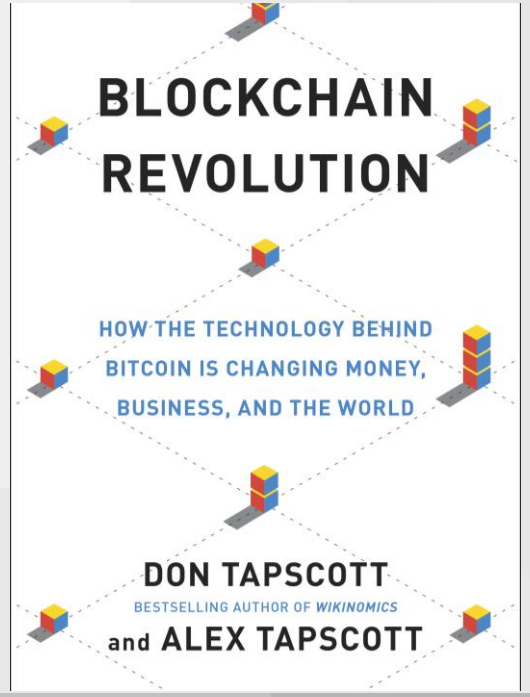

Demo

Resources

- https://bitcoin.org/en/bitcoin-paper
- https://www.ethereum.org/
- https://github.com/Azure/azure-blockchain-projects/blob/master/bletchley/bletchley-whitepaper.md







@bennymichielsen
bennym@infosupport.com

@hspeeters
hansp@infosupport.com