EXTROPY.IO

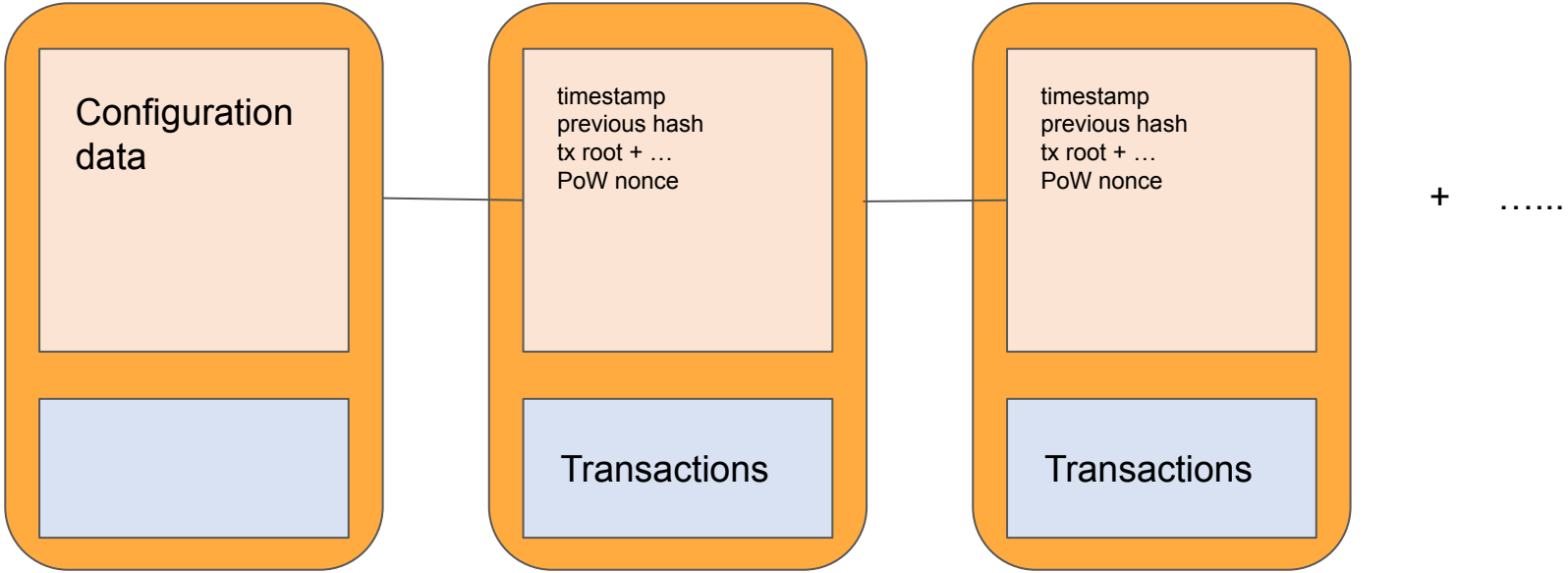
Introduction to Ethereum

Week 1
Lesson 3

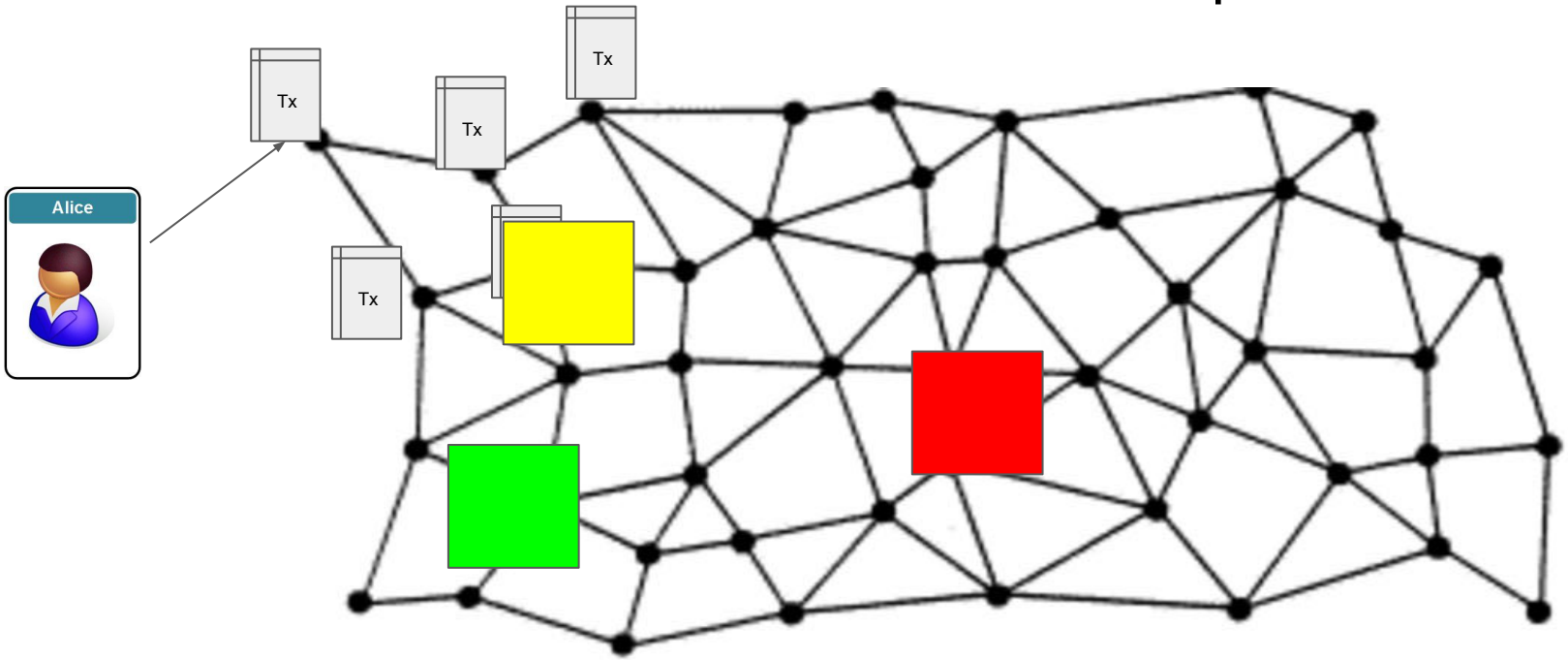
Lesson Plan

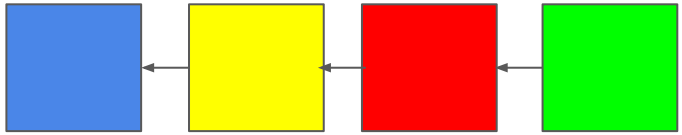
Review of Consensus Mechanisms
Hard and Soft Forks
Transactions in Bitcoin
Ethereum History
Smart Contracts
EVM Languages
Ethereum Clients and Mining
Connecting to a test network
ETH 2.0

Blockchain Data structure

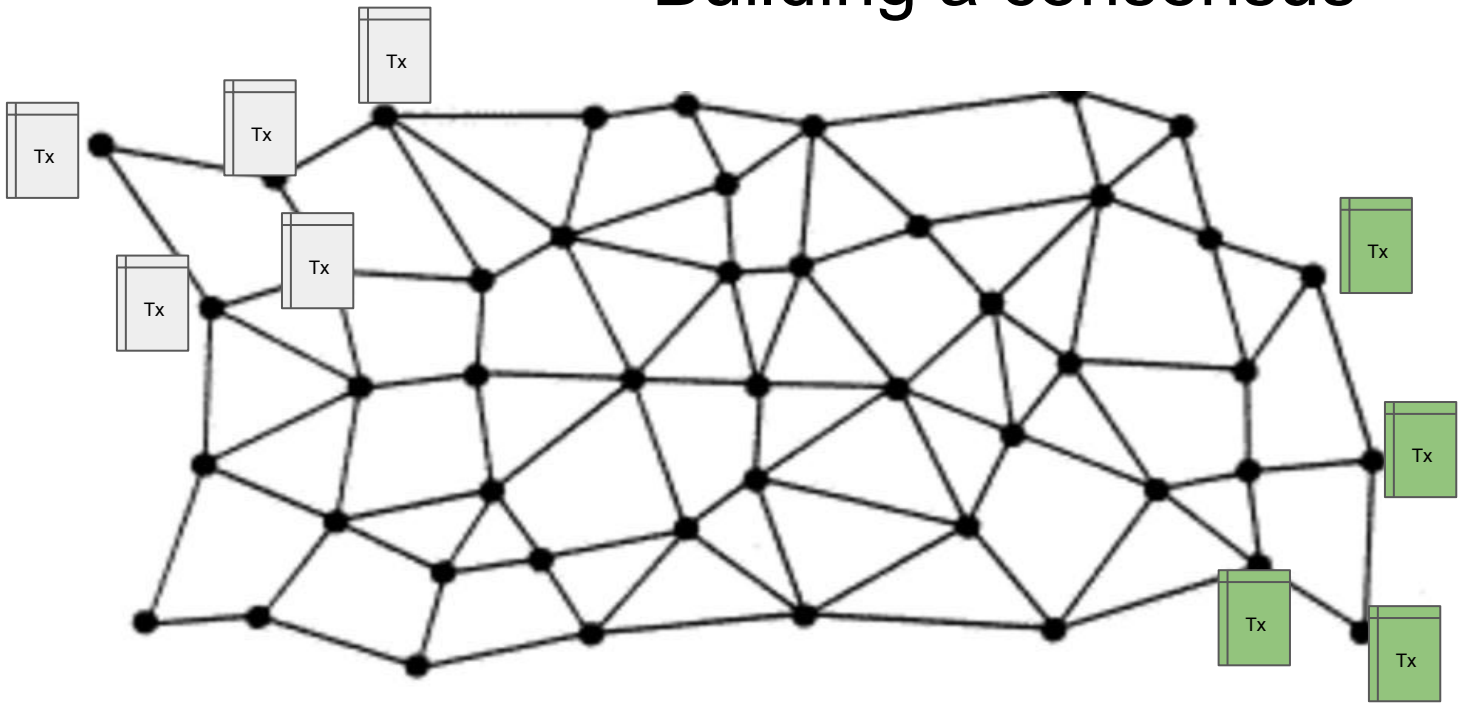


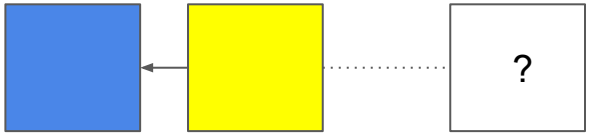
Network point of view



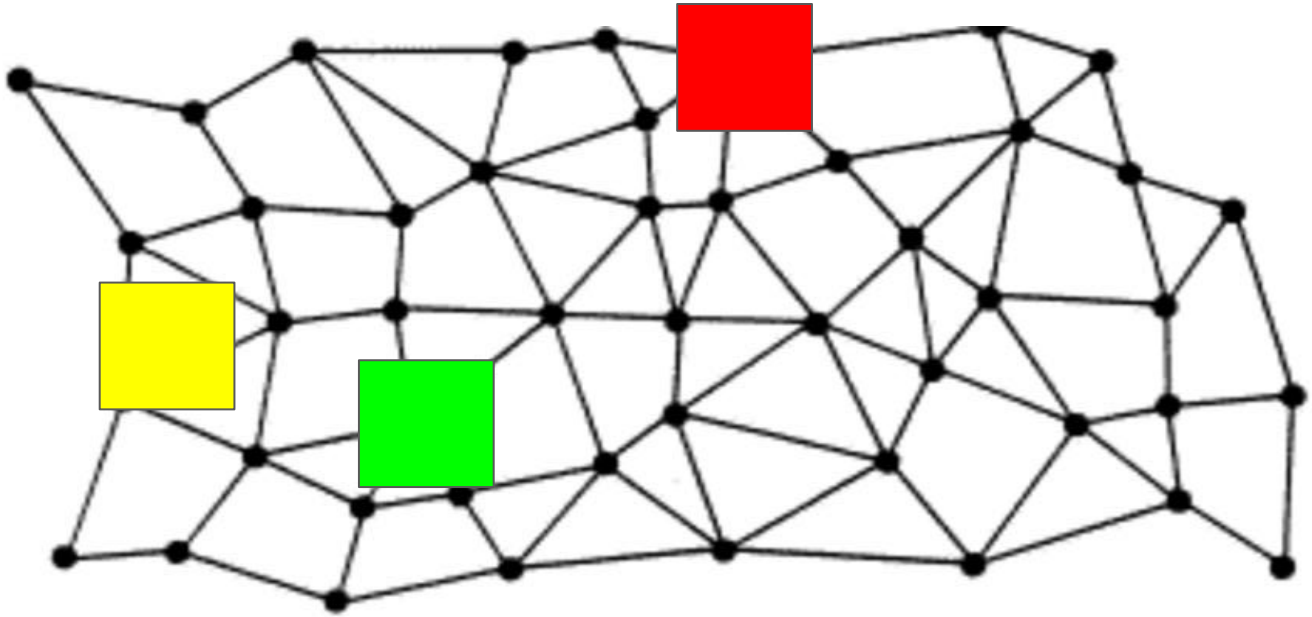


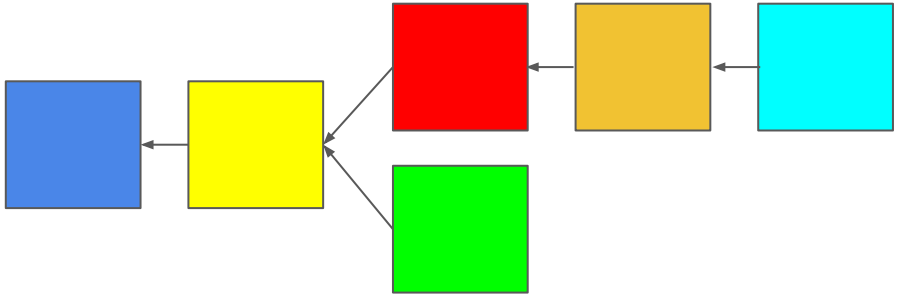
Building a consensus



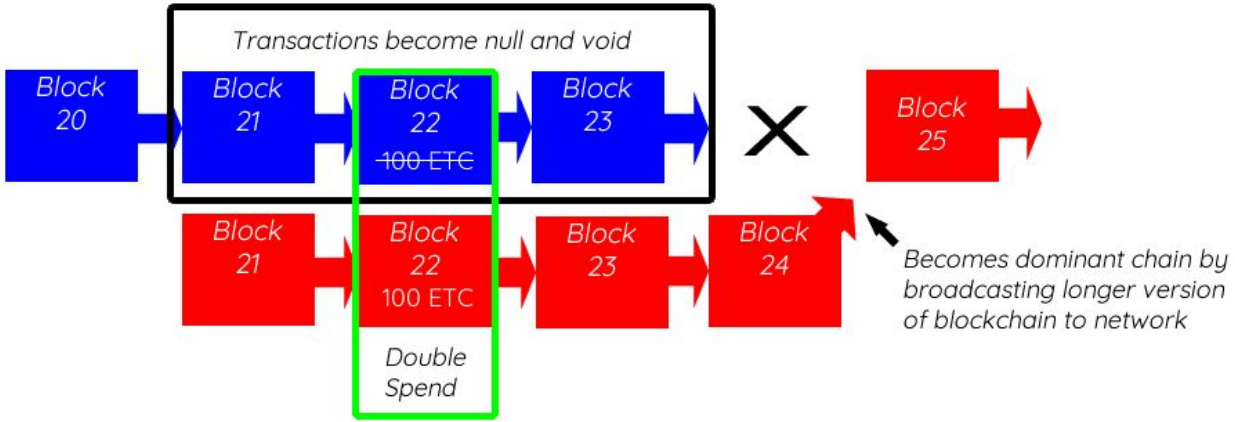


EXTROPY.IO





51% Attack (double-spend)



Original (honest) blockchain <50% hash power

Malicious blockchain >50% hash power

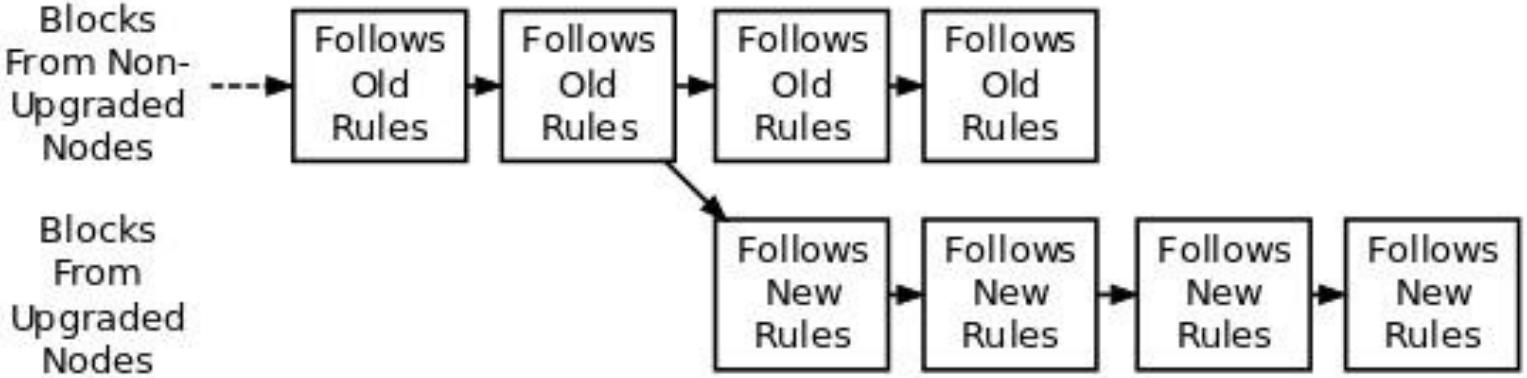
Some class questions

1. If we can choose a miner at random in Proof of Stake, why didn’t we use that mechanism in Proof of Work and leave out the puzzle part ?

2. Is a 51% attack possible in Proof of Stake ?

3. Can a block have no transactions ?

Hard Forks and Soft Forks



A Hard Fork: Non-Upgraded Nodes Reject The New Rules, Diverging The Chain

Soft Fork

A soft fork is a forwards compatible upgrade

Non upgraded nodes can still validate blocks produced to the new specification. If non upgraded nodes produce blocks however they will be rejected.

For example the SegWit change on Bitcoin.

Transactions in Bitcoin

This transaction was first broadcast to the Bitcoin network on August 20, 2021 at 12:41 PM GMT+1. The transaction currently has 4 confirmations on the network. At the time of this transaction, 0.01231900 BTC was sent with a value of $579.29. The current value of this transaction is now $578.70. Learn more about how transactions work.

Hash	58e11562ee7330afcb159de030fa7760df700569ca0d59810adb...				2021-08-20 12:41
	1C1v1uK2y7D1SjntNvjLBZNrdF9awXCH8y	0.01281900 BTC	→	3LRye9PgPYHoMiUHjuBxZ6iQFD4U4nku8w	0.01231900 BTC
Fee	0.00050000 BTC (264.550 sat/B - 66.138 sat/WU - 189 bytes)				0.01231900 BTC 4 Confirmations

In bitcoin, there are **no coins, no senders, no recipients, no balances, no accounts, and no addresses**.
All those things are constructed at a higher level for the benefit of the user, to make things easier to understand.

From : MasteringBitcoin

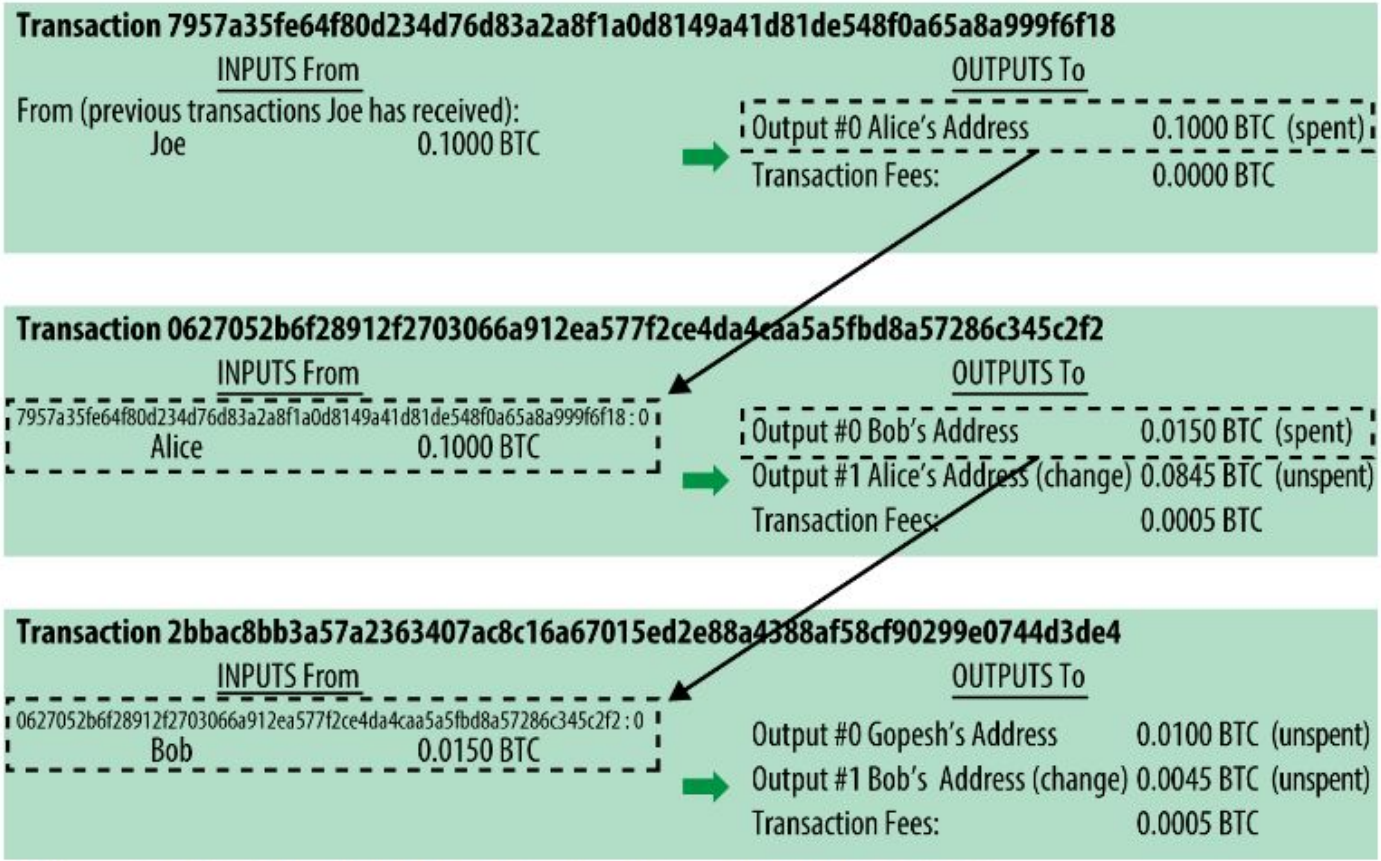


Figure 4. A chain of transactions, where the output of one transaction is the input of the next transaction

From [Mastering Bitcoin]

What is Ethereum?

- Blockchain proposed by Vitalik Buterin (2013)
- Virtual Machine (EVM)
- Turing Complete Language (Smart Contracts)
- Cryptocurrency: **Ether**
- Defined by a formal specification: *Yellow Paper*
  - *Combine English and mathematical (formal) explanations.*

- **Community driven by EIP = E**thereum **I**mprovements **P**roposals

EXTROPY.IO

Ether denominations

Table 1. Ether denominations and unit names

Value (in wei)	Exponent	Common name	SI name
1	1	wei	Wei
1,000	10^3	Babbage	Kilowei or femtoether
1,000,000	10^6	Lovelace	Megawei or picoether
1,000,000,000	10^9	Shannon	Gigawei or nanoether
1,000,000,000,000	10^{12}	Szabo	Microether or micro
1,000,000,000,000,000	10^{15}	Finney	Milliether or milli
1,000,000,000,000,000,000	*10^{18}*	*Ether*	*Ether*
1,000,000,000,000,000,000,000	10^{21}	Grand	Kiloether
1,000,000,000,000,000,000,000,000	10^{24}		Megaether

EXTROPY.IO

Ethereum History

2013

[what]

Ethereum is a next-generation distributed cryptographic ledger that is designed to allow users to encode advanced transaction types, smart contracts and decentralized applications into the blockchain. Ethereum will support custom currencies or "colored coins", financial derivatives, and much more, but unlike many previous networks that attempted to accomplish the same thing Ethereum does not attempt to constrain users into using specific "features"; instead, the ledger includes a built-in Turing-complete programming language that can be used to construct any kind of contract that can be mathematically defined.

To find out more about how Ethereum works, visit the technical whitepaper at http://ethereum.org/ethereum.html or the FAQ at https://wiki.ethereum.org/index.php/FAQ

“

EXTROPY.IO

what can you make out of ether?

Create a Currency

Make your own cryptocurrency on top of the Ethereum blockchain

Savings Wallet

Use multiple keys and withdrawal limits to protect your funds

Financial Derivatives

Speculate on financial assets at high leverage, or use hedging to protect yourself from volatility

Decentralized Organizations

Create decentralized companies or organizations that operate entirely on the blockchain

Name Registration

Register your name and your website

Data storage

Securely pay for nodes to archive your data, or earn money by renting out your hard drive

Frontier is coming - what to expect, and how to prepare

Posted by Stephan Tual on July 22, 2015

Research & Development

We are only days away from launching 'Frontier', the first milestone in the release of the Ethereum project. Frontier will be followed by 'Homestead', 'Metropolis' and 'Serenity' throughout the coming year, each adding new features and improving the user friendliness and security of the platform.

What is Frontier?

Frontier is a live, but barebone implementation of the Ethereum project. It's intended for technical users, specifically developers. During the Frontier release, we expect early adopters and application developers to establish communities and start forming a live ecosystem. Like their counterparts during the American Frontier, these settlers will be presented with vast opportunities, but will also face many dangers. If building from source and command lines interfaces aren't your cup of tea, we strongly encourage you to wait for a more user-friendly release of the Ethereum software before diving in.

Hard Fork Completed

Posted by Vitalik Buterin on July 20, 2016

Research & Development

We would like to congratulate the Ethereum community on a successfully completed hard fork. Block 1920000 contained the execution of an irregular state change which transferred ~12 million ETH from the “Dark DAO” and “Whitehat DAO” contracts into the WithdrawDAO recovery contract. The fork itself took place smoothly, with roughly 85% of miners mining on the fork:

2016
Tangerine Whistle and Spurious Dragon to fix DOS

2017
Byzantium - delay difficulty bomb

2019
Istanbul / Constantinople - Solidity and gas cost changes

2020
Muir Glacier - Difficulty bomb delay
Beacon chain genesis

2021
Berlin / London - EIP1559

Next - Altair - Beacon chain upgrade

EXTROPY.IO

Ethereum Viewers

http://ethviewer.live/

https://ethstats.net/

https://txstreet.com

Smart Contracts

What are Ethereum smart contracts ?

- Pieces of code running in the Ethereum Virtual Machine
- The environment is highly restricted for security and determinism
- Each contract has code, state and optionally a balance of Ether
- They may be written to represent a contract between parties but they need not.
- They are written in a high level language then compiled into bytecode to run in the EVM.

EXTROPY.IO

How smart contracts are created?

Let’s take this simple contract as example.

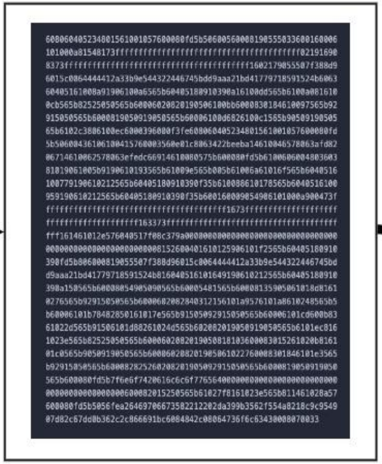
```
pragma solidity >=0.4.21 <0.9.0;

contract Score {

    uint public score;
    address owner;

    event Score_set(uint);

    constructor() public {
        score = 5;
        owner = msg.sender;
        emit Score_set(99);
    }

    modifier onlyOwner {
        require(msg.sender == owner, "not allowed");
        _;
    }

    function setScore(uint new_score) public onlyOwner {
        score = new_score;
        emit Score_set(new_score);
    }

    function getScore() public view returns (uint) {
        return score;
    }

}
```

Solidity smart contract code

```
pragma solidity >=0.4.21 <0.9.0;

contract Score {

    uint public score;
    address owner;

    event Score_set(uint);

    constructor() public {
        score = 5;
        owner = msg.sender;
        emit Score_set(99);
    }

    modifier onlyOwner {
        require(msg.sender == owner, "not allowed");
        _;
    }

    function setScore(uint new_score) public onlyOwner {
        score = new_score;
        emit Score_set(new_score);
    }

    function getScore() public view returns (uint) {
        return score;
    }

}
```

① **Compile**

contract bytecode

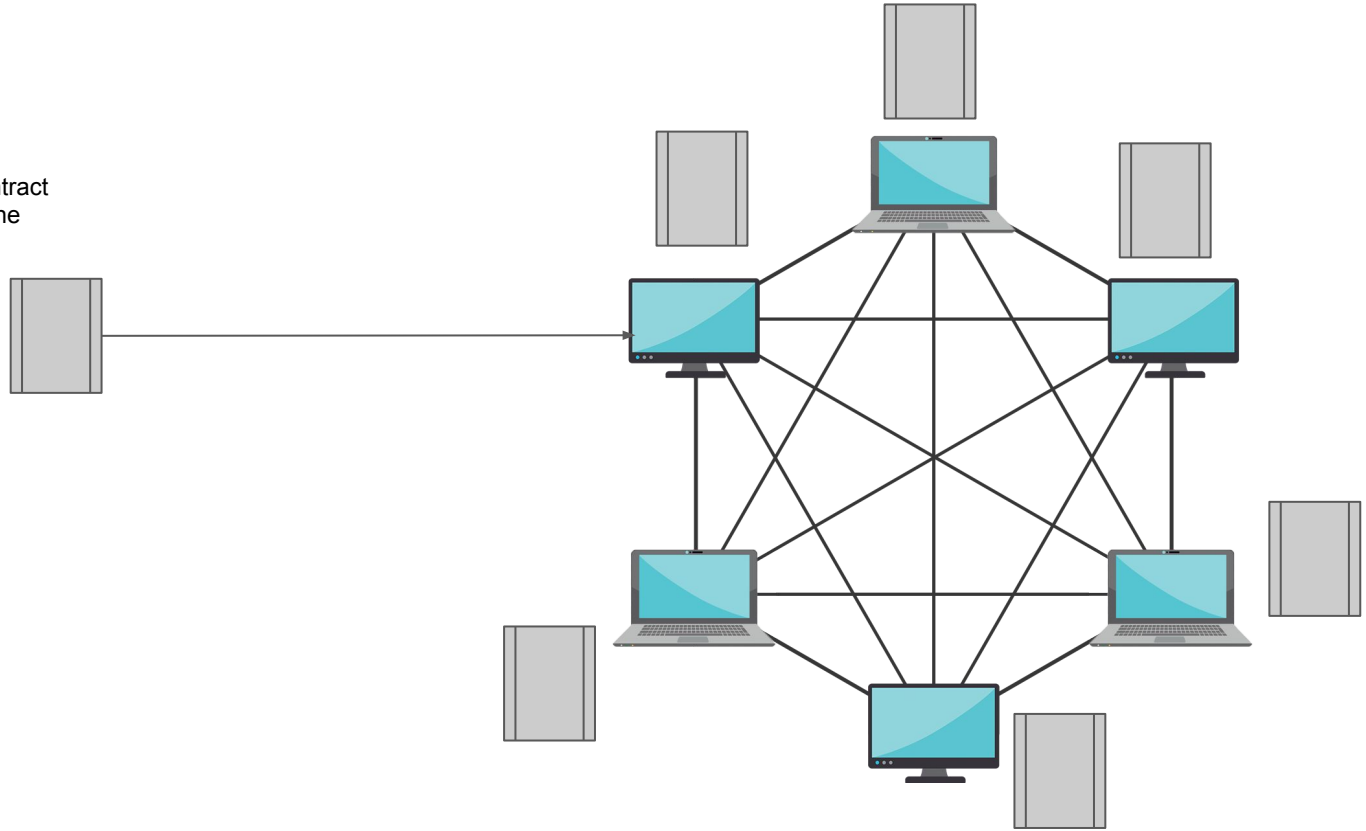


② **Deploy**

contract address

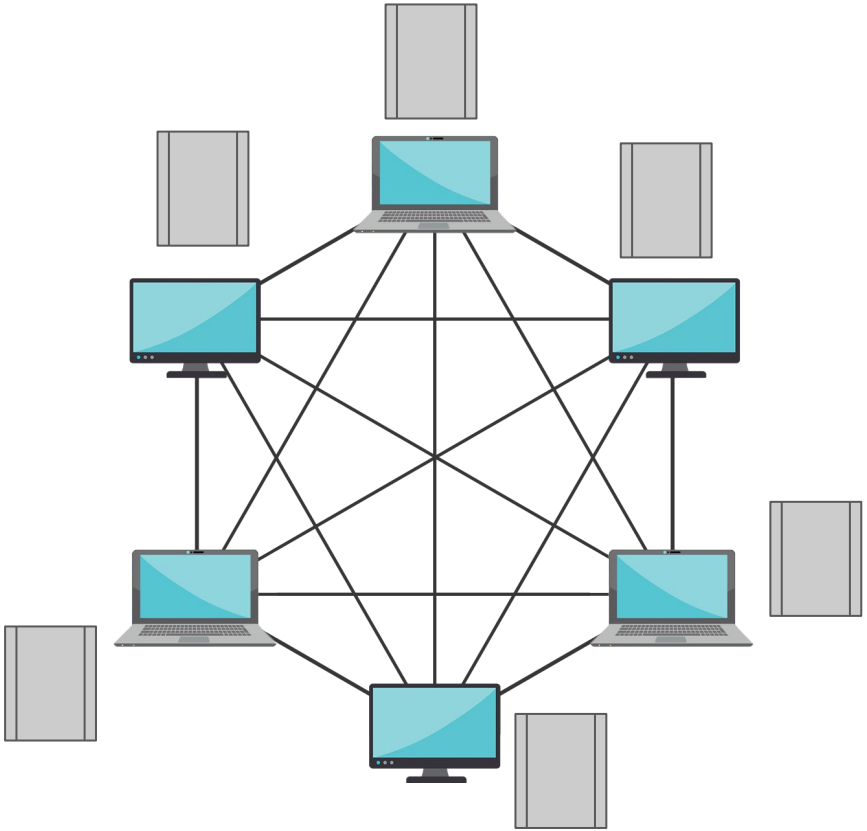
0x6b175474e89094c44da98b954eedeac495271d0f

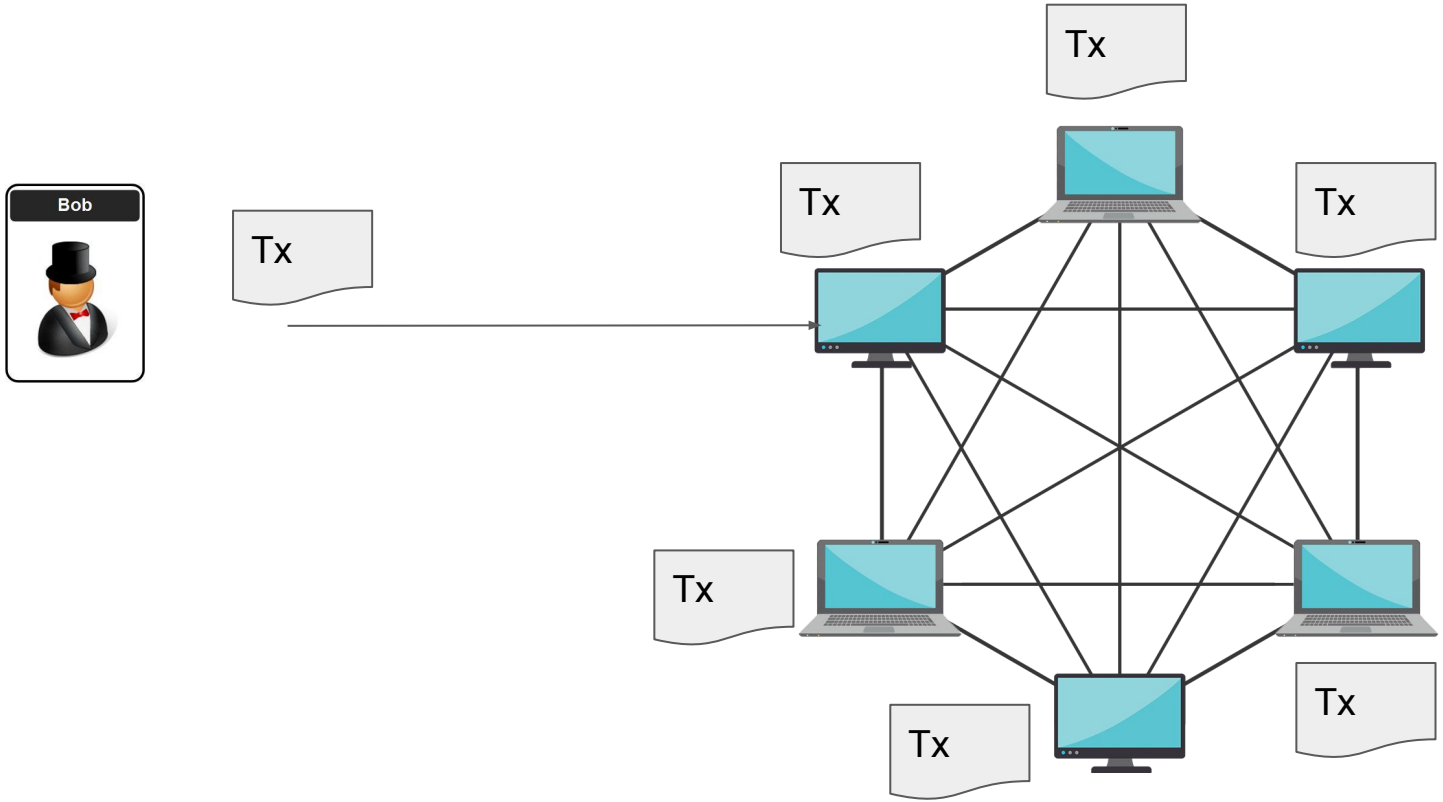
Creates smart contract and submits it to the blockchain

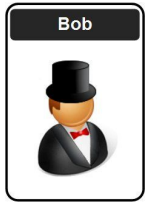


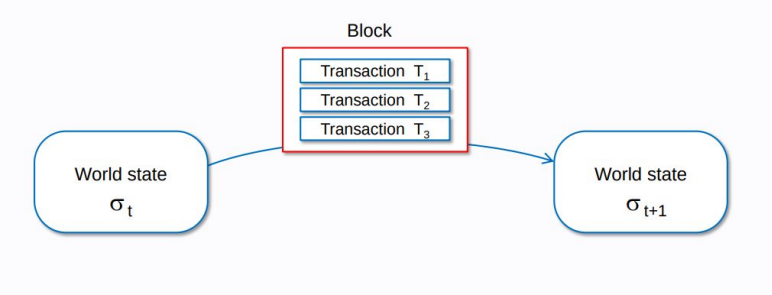
EXTROPY.IO

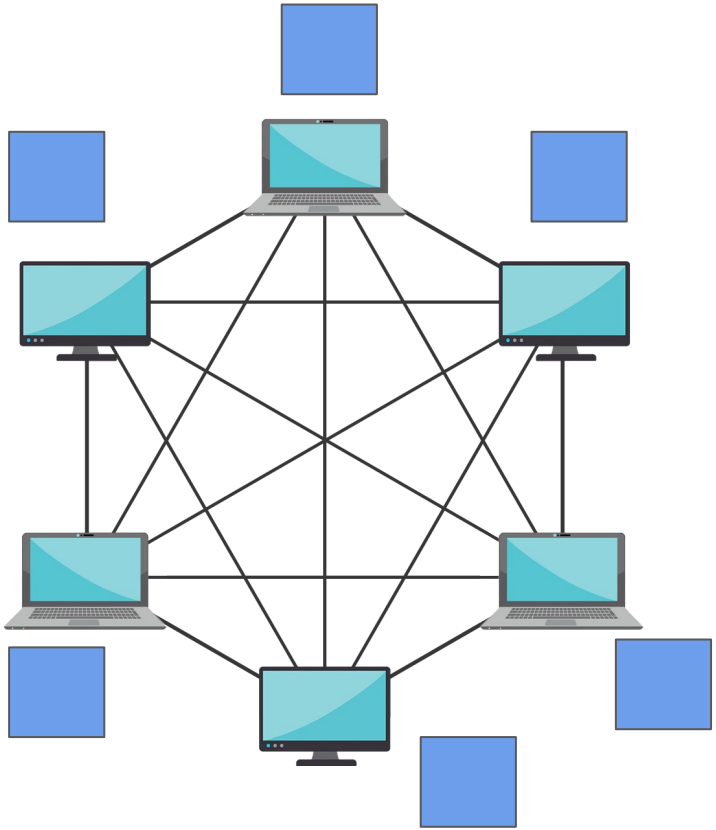
Wants to call a function in the contract



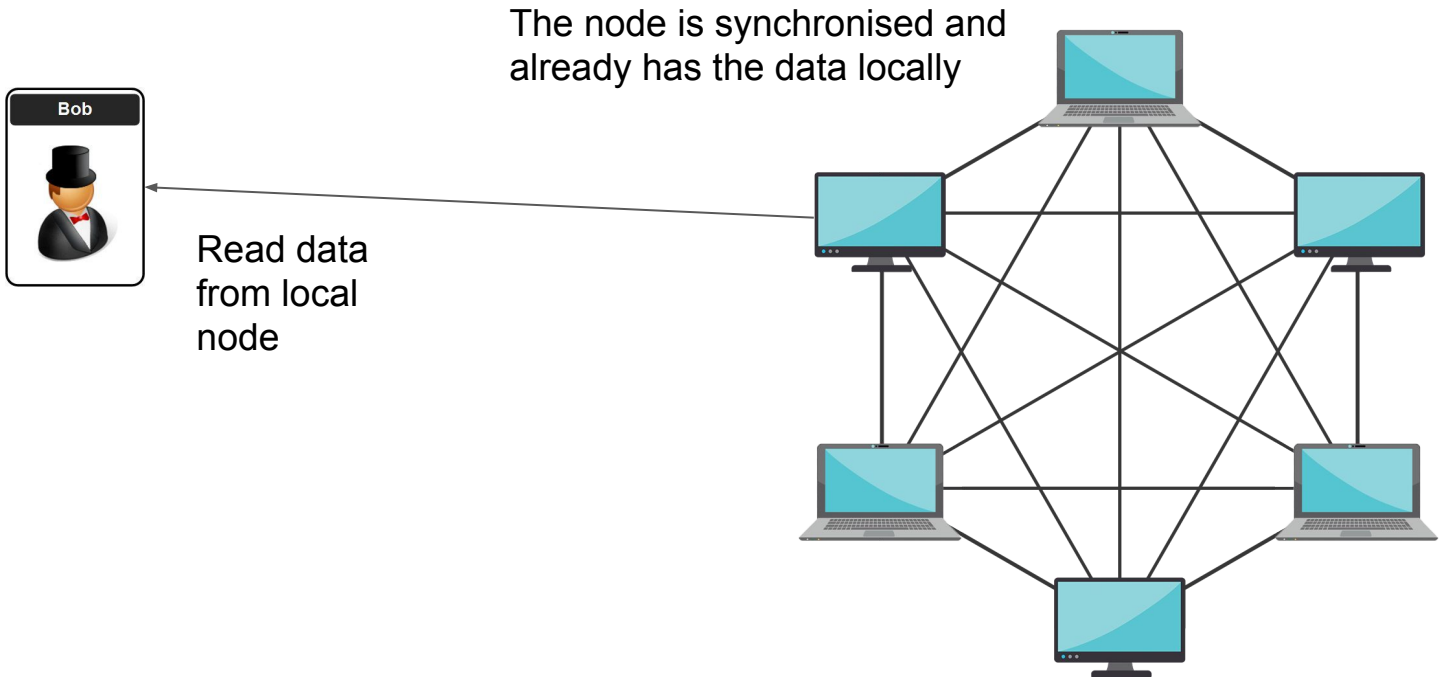

Bob
Tx
Tx
Tx
Tx
Tx
Tx
Tx
Tx


Bob


Block
Transaction T_1
Transaction T_2
Transaction T_3
World state
σ_t
World state
σ_{t+1}



View Function



Restrictions

- Gas - Amount of Computation
- Determinism
- Oracles - Getting data from the internet

EXTROPY.IO

Gas

"Gas is essential to the Ethereum network.

It is the fuel that allows it to operate,

in the same way that a car needs gasoline to run."

https://ethereum.org/en/developers/docs/gas/

Gas

Every transaction + computation on Ethereum requires paying a fee.
In Ethereum, this fee is called **gas.**

- **Gas fees** are collected by miners.
- **Gas fees** are paid in ether / ETH, as part of the transaction.

Transaction fee = total gas used * gas price paid (per unit of gas)

Gas Amount / Cost ⇒ Number of units of gas required to perform an operation.

Gas Price ⇒ Amount of ethers you are willing to pay per unit of gas

How Ethereum Miners choose which pending transactions to include in a block?

⇒ by selecting those that offer to pay a higher gas price.

<u>Conclusion:</u>

Offer a higher gas price
=
incentivize miners to include your transaction in a block
=
Get your transaction processed faster

EXTROPY.IO

New address detected! Click here to add to your address book.

Asset: ETH
Balance: 0.240643 ETH

Amount: 0.234637 ETH
$421.97 USD
Max

Transaction Fee:

Slow	Average	Fast
0.00525 ETH	0.00601 ETH	0.00661 ETH
$9.44	$10.80	$11.90

Advanced Options

New Transaction Fee
0.006006 ETH

Gas Price (GWEI): 286
Gas Limit: 21000

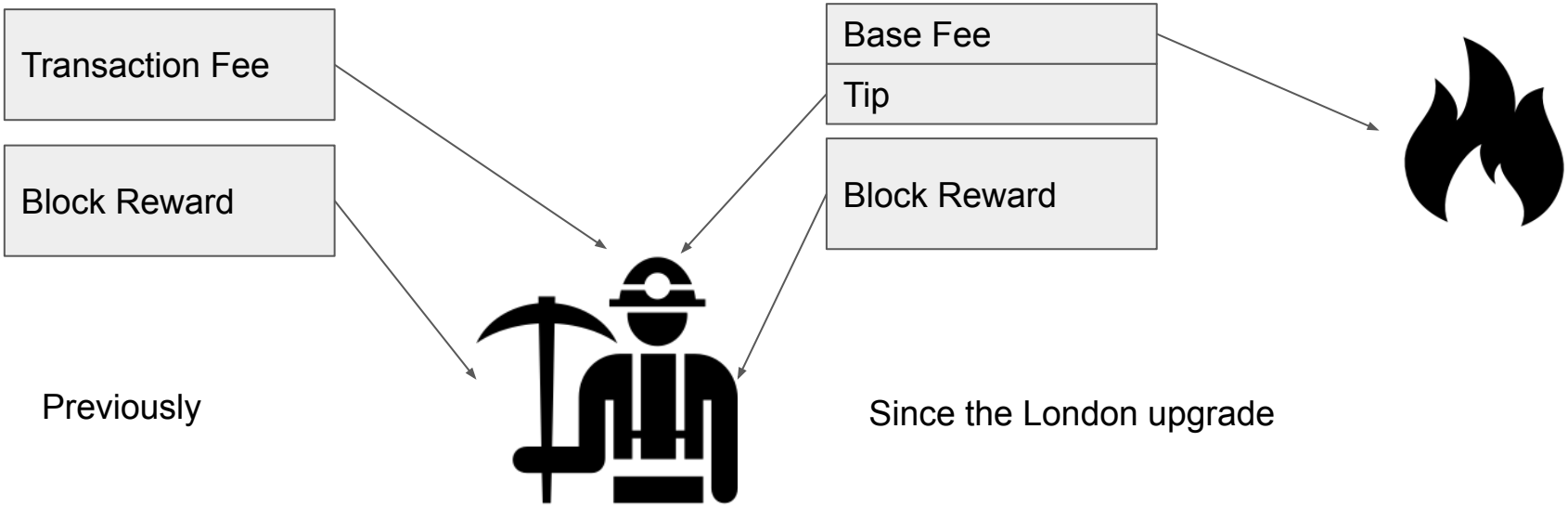
Send Amount	0.234637 ETH
Transaction Fee	0.006006 ETH
New Total	0.240643 ETH $432.70

EIP 1559

The gas fees are now split into

- Base fee
- Tip

The base fee depends on how full a block is
The tip is optional



Oracles

= bridge between the blockchain and the real world.

= on-chain API

Data feed that connect Ethereum to off-chain, real-world information *(eg: weather forecast, crypto prices, etc...)*

Examples:

- Predictions market: settle ETH payments based on outside the blockchain events (like “who will be the next US president?”)

FOAM

augur

Chainlink

oraclize

Oracles - Practical Examples

You borrow ETH via a Defi application.

When paying back, the Defi app needs to know what is the price per ETH to redeem.

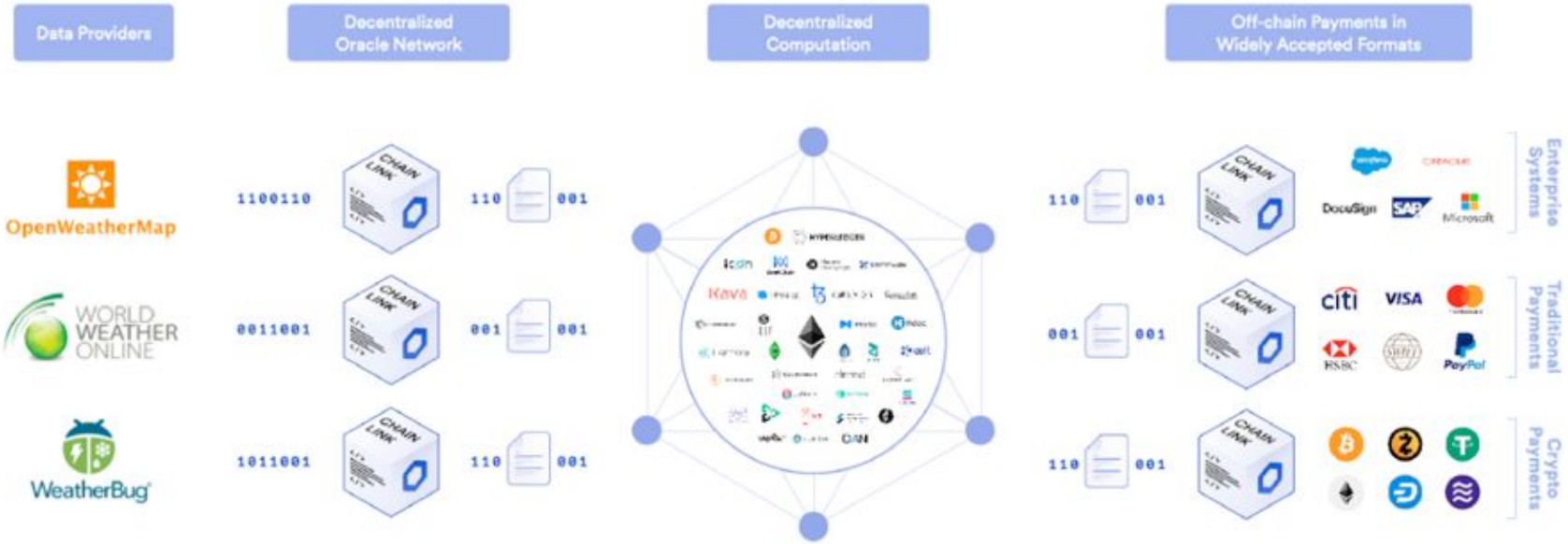
FOAM:

- Decentralized location service
- Can be used to verify shipments in supply chain
- Trigger payment automatically when a shipment has arrived.

NB: smart contracts oracles can also be triggered by events happening in other blockchains, like Bitcoin, Polkadot, etc…)

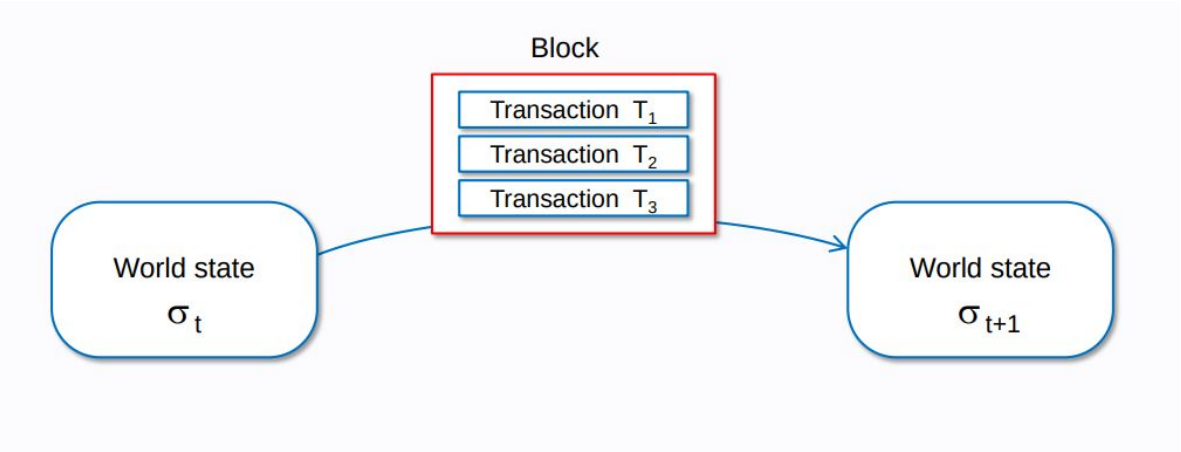
EXTROPY.IO


NYSE
Bloomberg
Market Data
101101
Events Data
101101
Chain Link
HYPERLEDGER
Chain Link
VISA
PayPal
Retail Payments
101101
SWIFT
HSBC
CHASE
Bank Payments
101101


Data Providers
Decentralized Oracle Network
Decentralized Computation
Off-chain Payments in Widely Accepted Formats
OpenWeatherMap
WORLD WEATHER ONLINE
WeatherBug
1100110
0011001
1011001
110
001
CHAIN LINK
DocuSign
SAP
Microsoft
citi
VISA
HSBC
PayPal
Enterprise Systems
Traditional Payments
Crypto Payments

Chainlink

Ethereum Transactions



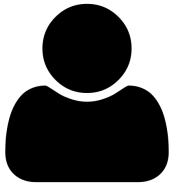
A submitted transaction includes the following information:

- `recipient` – the receiving address (if an externally-owned account, the transaction will transfer value. If a contract account, the transaction will execute the contract code)
- `signature` – the identifier of the sender. This is generated when the sender's private key signs the transaction and confirms the sender has authorised this transaction
- `value` – amount of ETH to transfer from sender to recipient (in WEI, a denomination of ETH)
- `data` – optional field to include arbitrary data
- `gasLimit` – the maximum amount of gas units that can be consumed by the transaction. Units of gas represent computational steps
- `maxPriorityFeePerGas` - the maximum amount of gas to be included as a tip to the miner
- `maxFeePerGas` - the maximum amount of gas willing to be paid for the transaction (inclusive of `baseFeePerGas` and `maxPriorityFeePerGas`)

2 Types of accounts in Ethereum

Externally Owned Accounts (EOA)

Controlled by people + private keys



Contract accounts (Smart Contracts)

Controlled by smart contract code + storage

Note that because a contract account does not have a private key, it cannot initiate a transaction. Only EOAs can initiate transactions, but contracts can react to transactions by calling other contracts, building complex execution paths.

The Account State

The account state comprises the following four fields:

- balance: A scalar value equal to the number of Wei owned by this address. Formally denoted σ[a]b.
- storageRoot: A 256-bit hash of the root node of a Merkle Patricia tree that encodes the storage contents of the account (a mapping between 256-bit integer values)
- codeHash: The hash of the EVM code of this account
- nonce: A scalar value equal to the number of transactions sent from this address or, in the case of accounts with associated code, the number of contract-creations made by this account.