Transactions on Blockchain

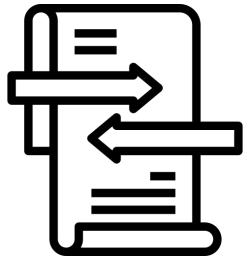
Components of Blockchain

- A peer-to-peer network connecting participants and propagating transactions.
- A set of consensus rules, governing what contributes a transaction and what makes for a valid state transition.
- Messages, in the form of transactions, representing state transitions.
- A state machine that processes transactions according to the consensus rule.
- A chain of cryptographically secured blocks that acts as a journal of all the verified and accepted transactions.

Transactions

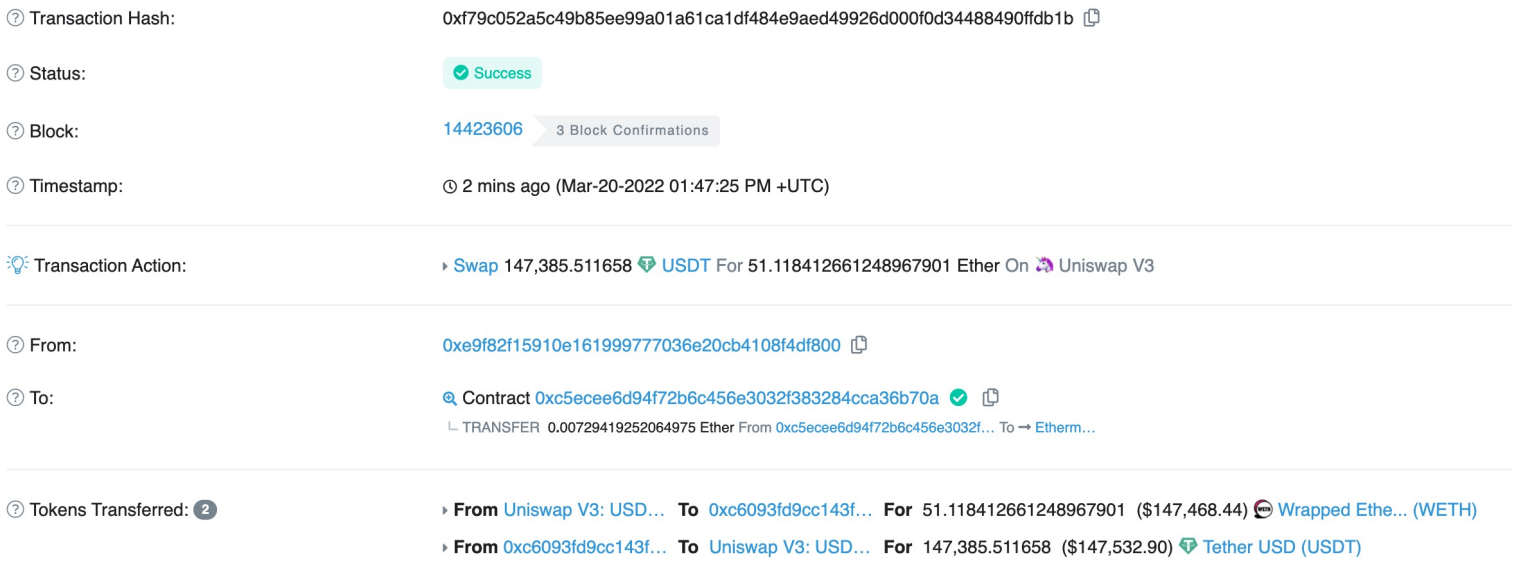
- Basic components of blockchain
- Carrier of all on-chain activities

We would just focus on Ethereum transactions in the following



Samples of Transaction

- https://etherscan.io/tx/0xf79c052a5c49b85ee99a01a61ca1df484e9aed49926d000f0d34488490ffdb1b



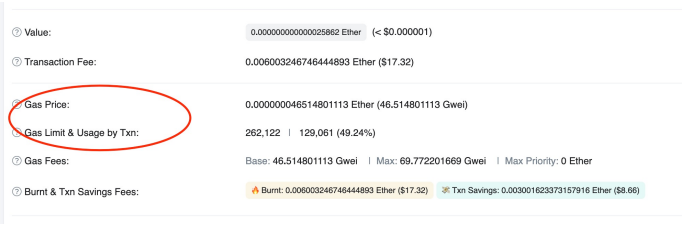
Structure of a Transaction

gas price & gas limit

Value:	0.000000000000025862 Ether (< $0.000001)
Transaction Fee:	0.006003246746444893 Ether ($17.32)
Gas Price:	0.000000046514801113 Ether (46.514801113 Gwei)
Gas Limit & Usage by Txn:	262,122 | 129,061 (49.24%)
Gas Fees:	Base: 46.514801113 Gwei | Max: 69.772201669 Gwei | Max Priority: 0 Ether
Burnt & Txn Savings Fees:	Burnt: 0.006003246746444893 Ether ($17.32) Txn Savings: 0.003001623373157916 Ether ($8.66)

Structure of a Transaction

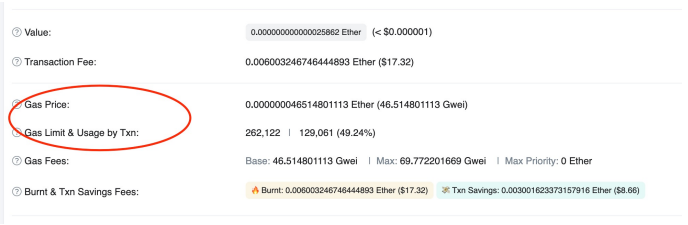
gas price & gas limit



- gas price: the amount of ether that sender is willing to pay for each unit of gas, *the higher, the more sooner to get executed.*
- gas limit: the maximum amount of gas that sender is willing to afford for this transaction. *If less than required, then the tx would be reverted.*

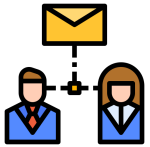
Structure of a Transaction

gas price & gas limit



- Gas War?
- What is the most appropriate price?
- https://etherscan.io/gastracker

Structure of a Transaction



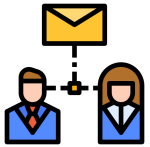
~~sender(from)~~ & recipient(to)

From:	0x1020dddf28736eaec77b1b30fff096b5dfad0e37
Interacted With (To):	Contract 0x1111111254fb6c44bac0bed2854e76f90643097d (1inch v4: Router)

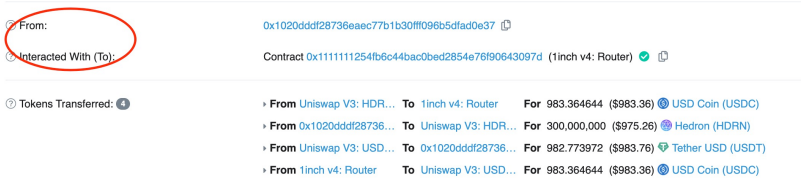
Tokens Transferred: 4

- From Uniswap V3: HDR… To 1inch v4: Router For 983.364644 ($983.36) USD Coin (USDC)
- From 0x1020dddf28736… To Uniswap V3: HDR… For 300,000,000 ($975.26) Hedron (HDRN)
- From Uniswap V3: USD… To 0x1020dddf28736… For 982.773972 ($983.76) Tether USD (USDT)
- From 1inch v4: Router To Uniswap V3: USD… For 983.364644 ($983.36) USD Coin (USDC)

Structure of a Transaction



sender(from) & recipient(to)



- ~~from: the one who proposed this transaction, ***must be an EOA***~~
- to: the destination address, can be an EOA or smart contracts

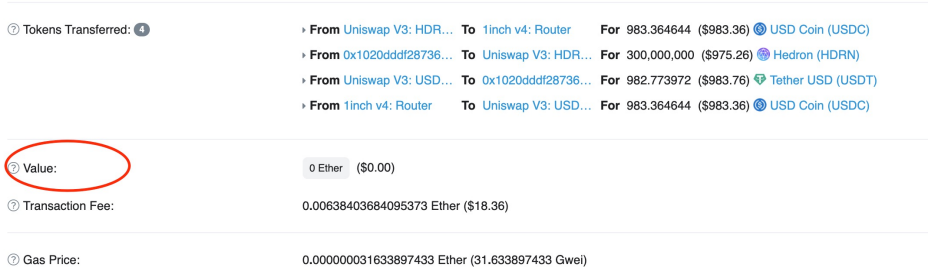
Structure of a Transaction

value

Tokens Transferred: 4	From Uniswap V3: HDR… To 1inch v4: Router For 983.364644 ($983.36) USD Coin (USDC) From 0x1020dddf28736… To Uniswap V3: HDR… For 300,000,000 ($975.26) Hedron (HDRN) From Uniswap V3: USD… To 0x1020dddf28736… For 982.773972 ($983.76) Tether USD (USDT) From 1inch v4: Router To Uniswap V3: USD… For 983.364644 ($983.36) USD Coin (USDC)
Value:	0 Ether ($0.00)
Transaction Fee:	0.00638403684095373 Ether ($18.36)
Gas Price:	0.000000031633897433 Ether (31.633897433 Gwei)

Structure of a Transaction

value



- value: the amount of ether(native token) to send to the destination, must be non-negative

Why can be zero? Can it be zero in Bitcoin Network?

Structure of a Transaction

data: *the most essential part*

Gas Price:	0.000000031633897433 Ether (31.633897433 Gwei)
Gas Limit & Usage by Txn:	279,495 | 201,810 (72.21%)
Gas Fees:	Base: 29.633897433 Gwei | Max: 49.862397266 Gwei | Max Priority: 2 Gwei
Burnt & Txn Savings Fees:	🔥 Burnt: 0.00598041684095373 Ether ($17.20) 💸 Txn Savings: 0.00367869355129773 Ether ($10.58)
Others:	Txn Type: 2 (EIP-1559) Nonce: 1479 Position: 72
Input Data:	

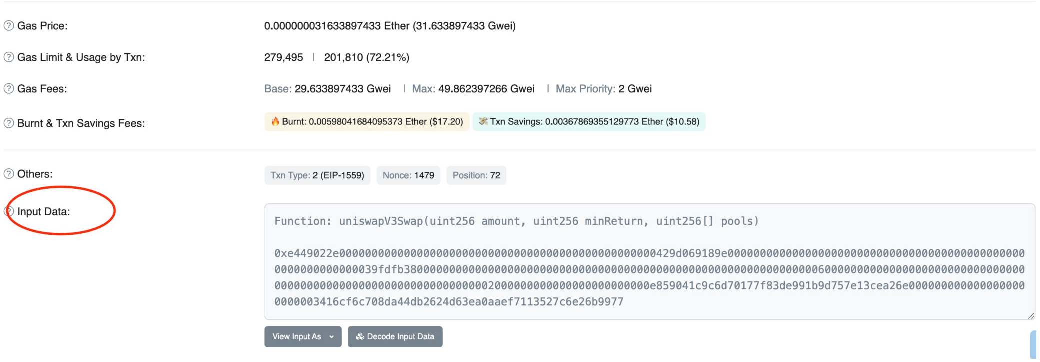
```
Function: uniswapV3Swap(uint256 amount, uint256 minReturn, uint256[] pools)

0xe449022e00000000000000000000000000000000000000000000000000000429d069189e000000000000000000000000000000000000000000
0000000000000039fdfb3800000000000000000000000000000000000000000000000000000000000000600000000000000000000000000000000
00000000000000000000000000000000200000000000000000000000e859041c9c6d70177f83de991b9d757e13cea26e000000000000000000
0000003416cf6c708da44db2624d63ea0aaef7113527c6e26b9977
```

View Input As | Decode Input Data

Structure of a Transaction

data



➢ data: transaction info inside

https://etherscan.io/tx/0x62cb94a4797eb00101eb4974c93a3c53f53686b9774cf60c0cf20f53a01356e6

```
Function: uniswapV3Swap(uint256 amount, uint256 minReturn, uint256[] pools)

0xe449022e0000000000000000000000000000000000000000000000000429d069189e0000000000000000000000000000000000000000000000
0000000000000039fdfb3800000000000000000000000000000000000000000000000000000000000000600000000000000000000000000000000
000000000000000000000000000000000200000000000000000000000e859041c9c6d70177f83de991b9d757e13cea26e000000000000000000
0000003416cf6c708da44db2624d63ea0aaef7113527c6e26b9977
```

More about value & data

- Transactions can have both value and data, only value, only data, or neither value nor data.
- If transfer value to some non-payable smart contracts, then the transaction will be reverted.
- For contract invocation, data is a hex-serialized encoding of
  - A function selector: the first 4 bytes of the Keccak-256 hash of the function's prototype, to identify which function you wish to invoke.
  - The function arguments
  - Padded to 32 bytes finally

e.g. (coding in Python)

To call the function withdraw(uint256)

```
function_part = w3.sha3(text='withdraw(uint256)')
[output]: HexBytes('0x2e1a7d4d13322e7b96f9a57413e1525c250fb7a9021cf91d1540d5b69f16a
function_selector = function_part[: 4]
[output]: HexBytes('0x2e1a7d4d')
```

The first 4 bytes of the hash are '0x2e1a7d4d'. That's our "function selector" value.

```
withdraw_amount = w3.toWei(0.01, 'ether')
withdraw_amount_hex = w3.toHex(withdraw_amount)
[output]: '0x2386f26fc10000'
```

Add the function selector to the amount(padded to 32 bytes), we get:

2e1a7d4d00000000000000000000000000000000000000000000002386f26fc10000

Structure of a Transaction

nonce: a sequence number, issued by the originating EOA, used to prevent message replay. Starting from zero.

v, r, s: the three components of an ECDSA digital signature of the originating EOA

Structure of a Transaction

- *All we need*: ***gas price, gas limit, to, data, value, nonce, and (v, r, s)***
- Most internal representations and user interface visualization embellish this with additional information, which can be derived from indicators above(e.g. address)

Learn Using EtherScan

Users can use Etherscan to:

- Calculate Ethereum gas fees with the Etherscan gas tracker
- Lookup and verify smart contracts
- View the crypto assets held in or associated with a public wallet address
- Observe live transactions taking place on the Ethereum blockchain
- Lookup a single transaction made from any Ethereum wallet
- Discover smart contracts with verified source code
- Interact with smart contracts directly

Construct tx using Python

- ❑ Native token transfer
- ❑ Interact with smart contracts (https://remix.ethereum.org/)

Take Rinkeby testnet as example:

- Etherscan URL: https://rinkeby.etherscan.io/
- Facet: https://faucet.rinkeby.io/

- Codes today: https://github.com/zhshang1221/BootCampTech-Data/tree/master/01_tx
- Mastering Ethereum Chap 6.
  https://github.com/ethereumbook/ethereumbook/blob/develop/06transactions.asciidocf
- Using EtherScan
  https://cointelegraph.com/news/what-is-etherscan-and-how-does-it-work
- Ethereum Official Website
  https://ethereum.org/en/
- Web3.py document
  https://web3py.readthedocs.io/en/stable/