Smart Contracts

(Ethereum 1.0)

PoW Blockchain

- Different puzzle (ethash), ASIC-proof
- ≈12 secs between blocks
- Smart contracts

2 types of accounts:

① Normal (public + private key)
→ maintain a balance
Transaction: One payer
One recipient
One amount

② Bot account → On the blockchain
- Controlled by code
- "Smart contract"

Puzzle

Challenge: Instantaneous transactions

Alice

Bob

$1000

Charlie
- Trustworthy
- Stupid
- Can: - Send/receive texts
  - Send/receive $$

(1) Alice send book to C
(2) C sends picture to B
(3) B sends $1K to C
(4) C sends $1K to A
book to B

X C can't send/recv book or pic
X B might not be able to verify

(1) B sends $1K to C
(2) C tells A $1K sent
(3) A sends book to B
(4) B tells C to release $ to Alice
(5) C does it

X 1-sided. B feels good
A can be cheated

B pays half
A sends book
B verifies, pays remaining

X 1-sided, Alice can steal half.

0. A sends $1K to C
1. B sends $2K to C
2. C tells A $1K sent
3. A sends book to B
4. B tells C to release $ to Alice
5. C sends $1K to B, $2K to A

Ethereum Accts

Both types
- Have addresses
- Receive transactions
- Have a balance
- (Send transactions)

Distinction: control
- Normal acct: Have private key
- Smart contract: Code

Auction

Seller
- Put up good
- Specify deadline
- Reserve price

Buyers
- Submit bids
- Highest bid wins.

Auction S.C.

Highest bid amt
Highest bidder

ctor:
- Good being sold → transfer ownership
- Deadline → contract var
- Reserve price ☆

place_bid:
- Anyone can run
- Payable, ~~any amount~~

~~Check deadline~~ Check amount > highest bid
If highest_bidder != 0: refund them
Set highest_bidder and highest_bid.

end_auction:
- ~~Seller only~~
- No payment

Check reserve price met (otherwise refund)
Check deadline passed
Pay seller
Tranfer good buyer

Next time: gas

↳ fuel to move transactions

What does a miner do?

most expensive

- Solve proof of work
- Share new block to others
- Get txns from pool and create the block
- Verify txns

Bitcoin

variable ← - Run S.C. code. Ethereum

How do BTC fees work?

- Σ inputs − Σ outputs
- Decided by senders

How do they decide

- Look at recent transactions
- More or less depending on prioritization
- Per byte size of txn

What if BTC/USD price suddenly goes up?

(How will txn fees change?)

- Pending txns are paying more!
- Fees *should* go down
  - Block size doesn't change, so depends on txn demands.
  - Could also go up based on higher demand.

Gas pricing

History: Type 0: old model, simpler gas
Type 2 ("London")

- Gas to run txn itself
  - Normal: 21000
  - S.C.: Price for each executed op.
- How much Gwei per gas?
  - Base price → determined by network
  - Prioritization fee → chosen by submitter

Base price

→ Set algorithmically to keep blocks ≈ 50% full (Price matching txn demand)

→ *Burned* (no one gets it)

Prioritization fee

- Decided by sender
- Miner gets it.

Gas: fees to make txns happen

Smart contract code

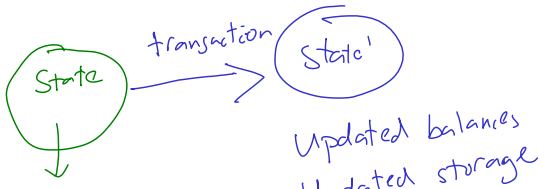
Interface

Compiled to ABI + Bytecode

- Vyper (Python-like)
  - declare all types
  - @payable @external ...
  - not all types available (strings, floats)
- Solidity (javascript-like)
- others...

Ethereum Virtual Machine

Big state machine



Account balances
data storage

Updated balances
Updated storage

Gas depends on cost to make the transition

EVM Bytecode

- Few hundred instructions ("Opcodes")
- Each has a price
- Stack-based virtual machine
  - ↳ temporary