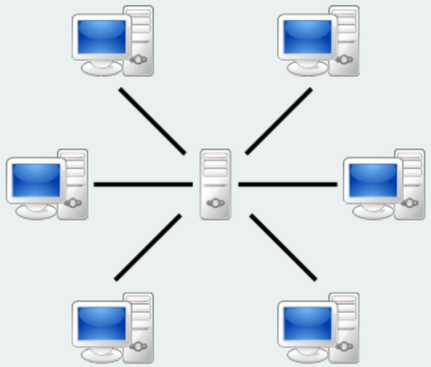
ethereum
vienna

General Introduction

ethereum

Goals

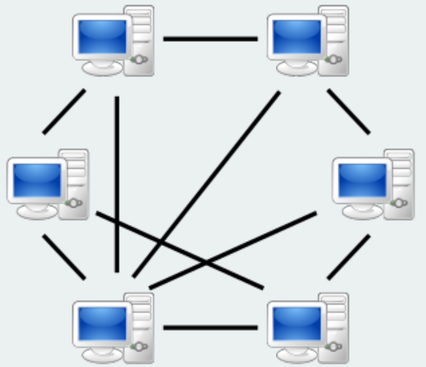


Decentralisation of applications and systems

Removing the role and power of central points

Take away control from service operators

Reduce trust requirements between parties



Why decentralise?

Data cannot just disappear

Data can only be modified by certain rules*

- provides audit trail
- protects system state from manipulation

Censorship resistant

Server cannot freeze funds

* most of the time

ethereum

Project

Platform for decentralised applications (**DApps**)

Ethereum (Blockchain)
Consensus Layer

Whisper
Messaging and Broadcasting

Swarm / IPFS (Content System)
Data publication and distribution

DApps

Escrow Standard UI Wallet

Crowdfunding Weifund

Insurance etherisc

Prediction Markets Augur / Gnosis

Registries ENS

Marketplace Safemarket

Decentralised Autonomous Organisations (DAO)

Stablecoins MakerDAO

ethereum

blockchain

Blockchain

Public record of all transactions

Can be stored, processed and validated by every node

Necessary to give transactions an ordering

Unanimous agreement on the ordering is critical

=> Different order might yield different results

This enables **global consensus** over the current state of Ethereum and its DApps

ethereum

Enterprise Alliance

Enterprise

Public Blockchain

- Public Ledger
- Anyone can participate
- Proof of Work
- Expensive
- Global consensus
- Rollbacks by mining majority

Enterprise Blockchain

- Private Ledger
- Access restricted
- PBFT
- Cheaper
- Local consensus
- Rollbacks by node majority

Blockchain

Account based System

identified by a 160 bit address

has a balance of Ether / Wei

2 types of accounts

"Accounts" (external)

Contracts (internal)

Accounts

user controlled account

controlled by a private key

can

- send ether
- receive ether
- interact with smart contracts

0x1350cf34d093953ce0d2803648da8f3b6a84de77	100
0xd5f9d8d94886e70b06e474c3fb14fd43e2f23970	2500
0xd2963cd505c94dbf3bc663bdd2321bd3000204bb	23290
0xd2963cd505c94dbf3bc663bdd2321bd3000204bb	123809
...	...

ethereum

Contracts

can do the same things as accounts

no private key

```
contract Coin {

    event Transfer(address indexed from, address indexed to);

    mapping (address => uint) public balances;

    function() {
        balances[msg.sender] = 10;
    }

    function Send(address to, uint amount) {
        if(balances[msg.sender] >= amount) {
            balances[msg.sender] -= amount;
            balances[to] += amount;
        }
    }
}
```

controlled by code instead

gets executed when somebody sends to or calls the contract

has persistent storage

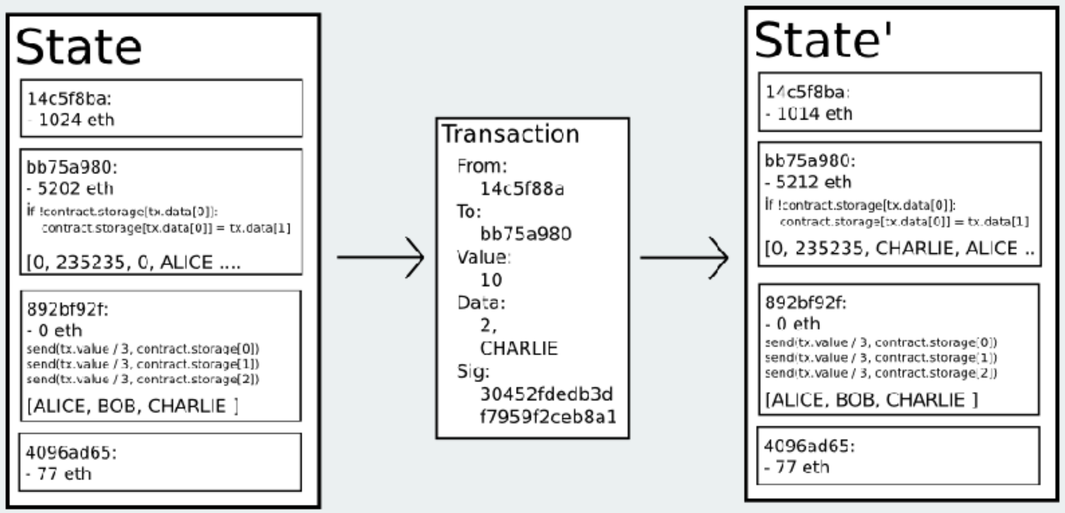
Blockchain

Transaction

Signed by a private key (external account)

Transitions from one state to the next

Can transfer ether, call contract functions, etc.



Gas

Gas

- Used for transaction fees
- Sender “buys” gas at a **sender-specified gasprice**
- Every computational step has a fixed **gas cost**
- Remaining gas **sent back** to sender
- If gas runs out (**"out of gas"**)
  - the state changes **revert** (including any ether transfers)
  - but miner keeps the gas payment

ethereum

Gas

Gasprice

Associated gas cost for some action is constant

But the price of ether is not

Gasprice can be a scale factor against ether price

=> but there is also a lower bound due to block reward

Ether goes up -> Gasprice goes down

Ether goes down -> Gasprice goes up

Gas

Example

Bob sends a transaction with **100000** gas at a gas price of **0.000001** ether

Thus he pays at most **0.1** ether as a fee (the product)

If the transaction only ends up using **32400** gas he only pays **0.0324** ether

The remaining **0.0676** are refunded to Bob

ethereum

Blockchain

Blockchain gives transactions an order

Transactions are grouped together into blocks (~15s apart in time)

Order is important:

- Double spend (no unspent outputs, but balance might become 0)
- 2 transactions interacting with the same contract
- Different order -> Potentially different outcome

ethereum

Whisper / Swarm
Mist

ethereum

Whisper

Decentralised Messaging

Messages can be filtered by topics

Very flexible

- Messages can be encrypted
- Messages can be signed
- Broadcast

PoW for spam protection and priority

Not designed for real time communication

ethereum

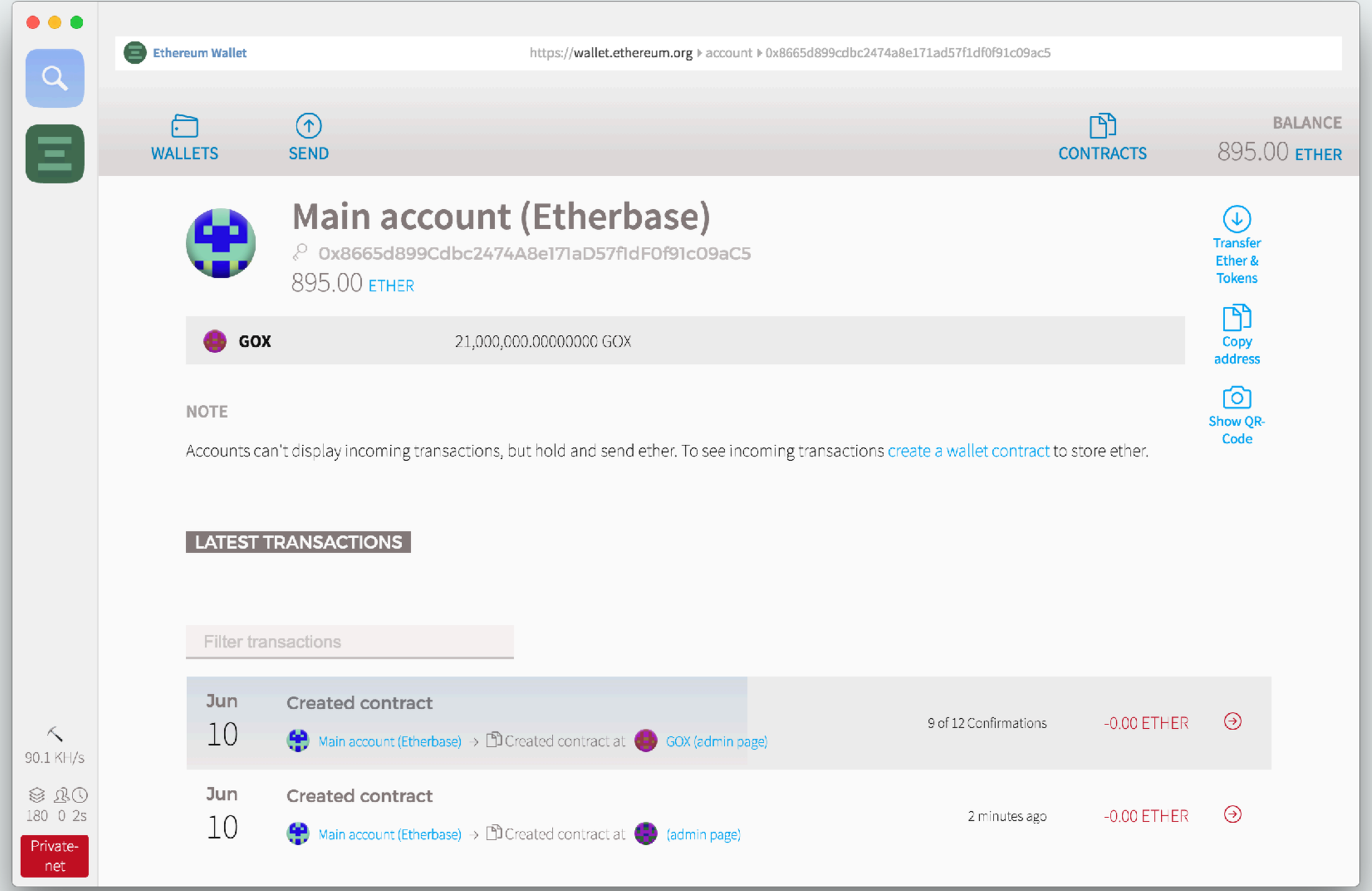
Swarm

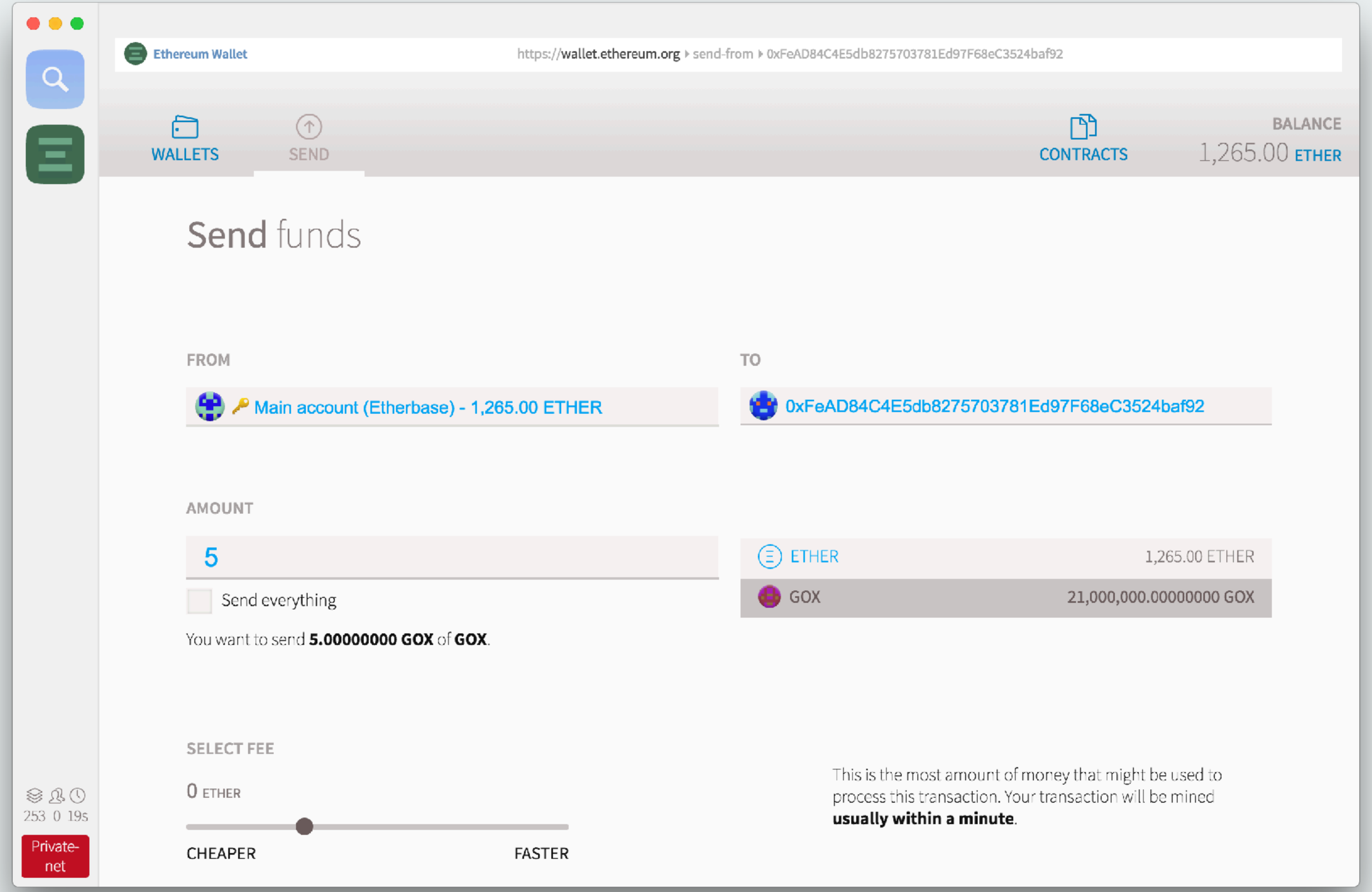
Swarm

- Reverse Hash-table
- Distributed chunk store
- Low-latency
- Incentivation model for storage

now ships with Mist

incentive layer still missing


Ethereum Wallet
https://wallet.ethereum.org ▸ account ▸ 0x8665d899cdbc2474a8e171ad57f1df0f91c09ac5
WALLETS
SEND
CONTRACTS
BALANCE
895.00 ETHER
Main account (Etherbase)
0x8665d899Cdbc2474A8e171aD57f1dF0f91c09aC5
895.00 ETHER
Transfer Ether & Tokens
Copy address
Show QR-Code
GOX
21,000,000.00000000 GOX
NOTE
Accounts can't display incoming transactions, but hold and send ether. To see incoming transactions create a wallet contract to store ether.
LATEST TRANSACTIONS
Filter transactions
Jun 10
Created contract
Main account (Etherbase) › Created contract at GOX (admin page)
9 of 12 Confirmations
-0.00 ETHER
Jun 10
Created contract
Main account (Etherbase) › Created contract at (admin page)
2 minutes ago
-0.00 ETHER
90.1 KH/s
180 0 2s
Private-net

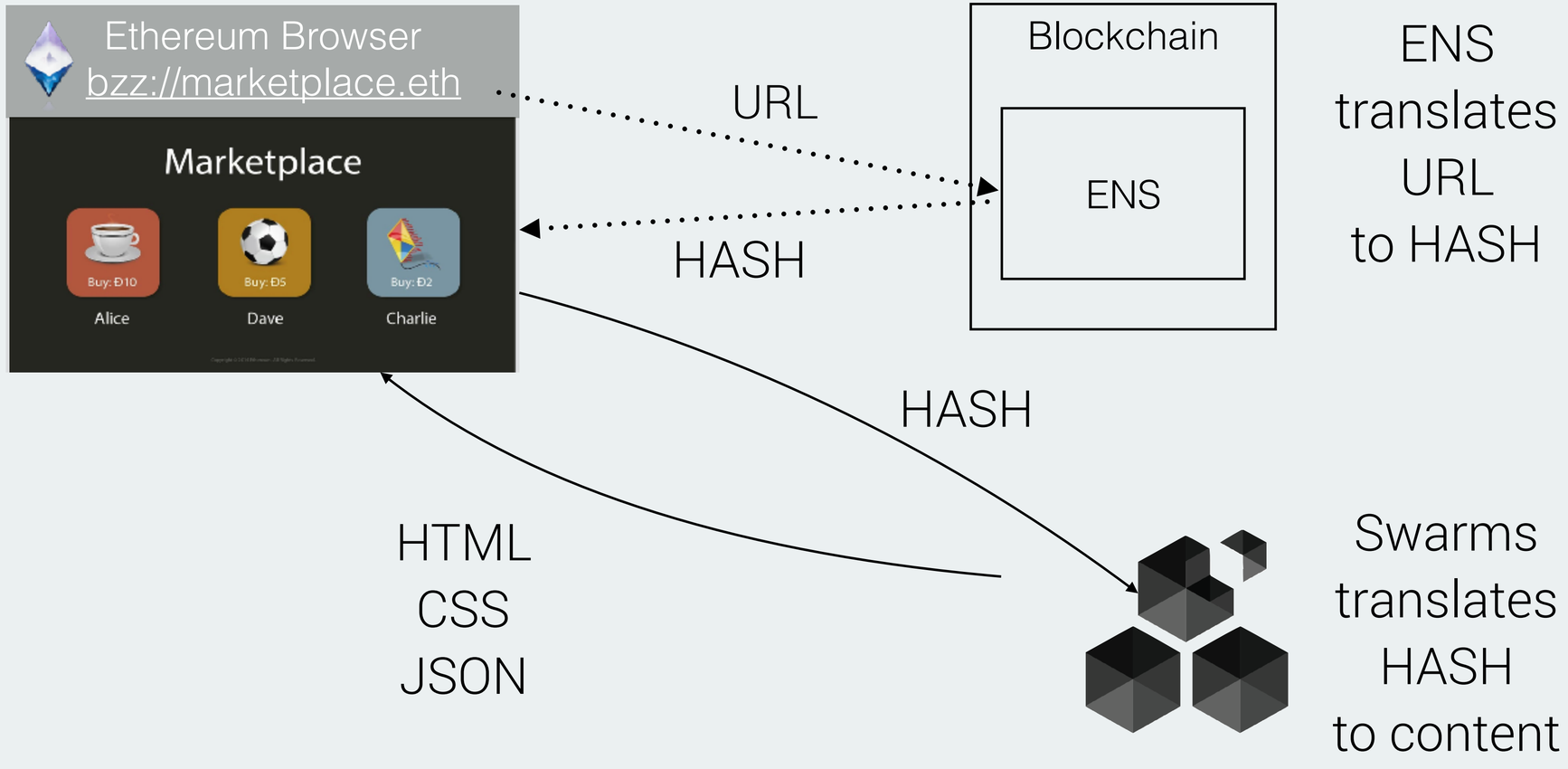

Ethereum Wallet
https://wallet.ethereum.org ▸ send-from ▸ 0xFeAD84C4E5db8275703781Ed97F68eC3524baf92
WALLETS
SEND
CONTRACTS
BALANCE
1,265.00 ETHER
Send funds
FROM
Main account (Etherbase) - 1,265.00 ETHER
TO
0xFeAD84C4E5db8275703781Ed97F68eC3524baf92
AMOUNT
5
Send everything
You want to send 5.00000000 GOX of GOX.
ETHER 1,265.00 ETHER
GOX 21,000,000.00000000 GOX
SELECT FEE
0 ETHER
CHEAPER
FASTER
This is the most amount of money that might be used to process this transaction. Your transaction will be mined usually within a minute.
253 0 19s
Private-net

Marketplace DApp

(Badly designed) Example

ethereum Marketplace

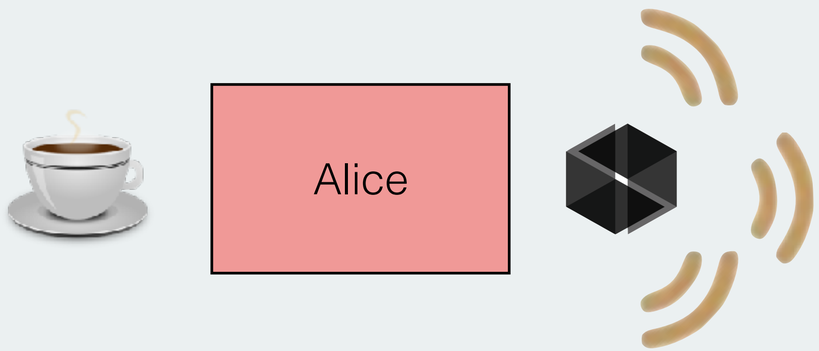
User enters URL



ethereum Marketplace

Alice wants to sell a cup for 10 ETH

Whisper Broadcast
"I want to sell a cup for 10 ETH"

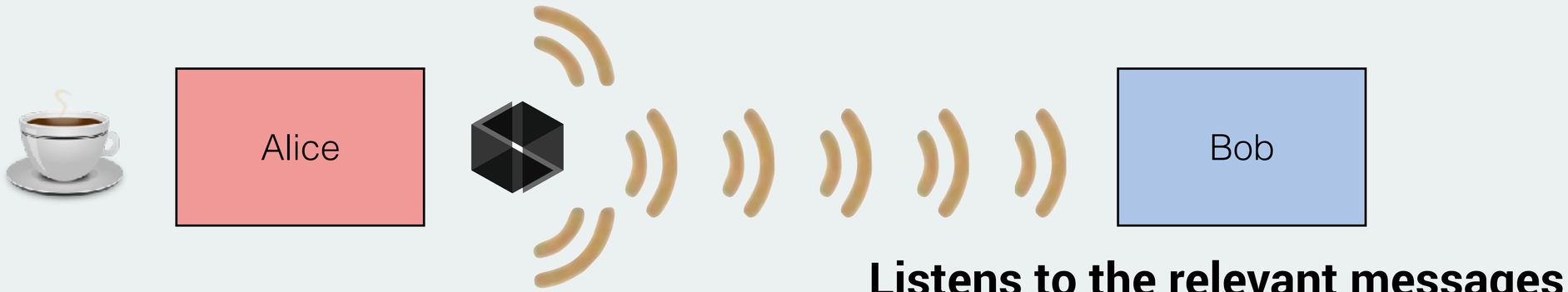


Broadcasts a Whisper message

ethereum Marketplace

Bob wants to buy cups

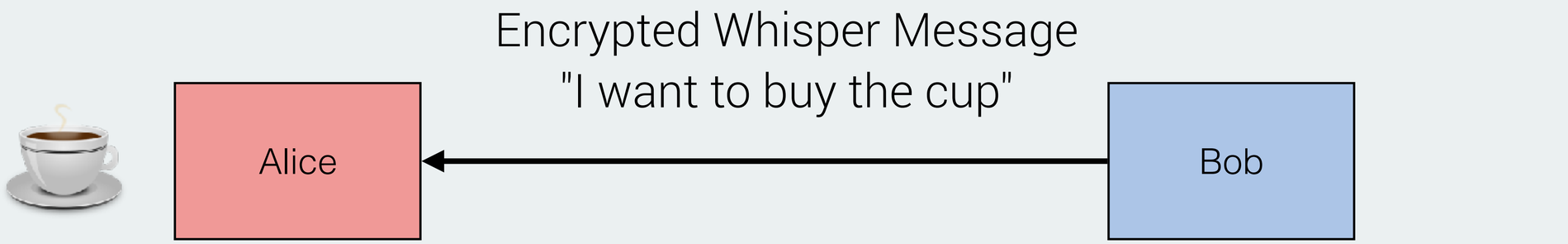
10 ETH



Listens to the relevant messages

ethereum Marketplace

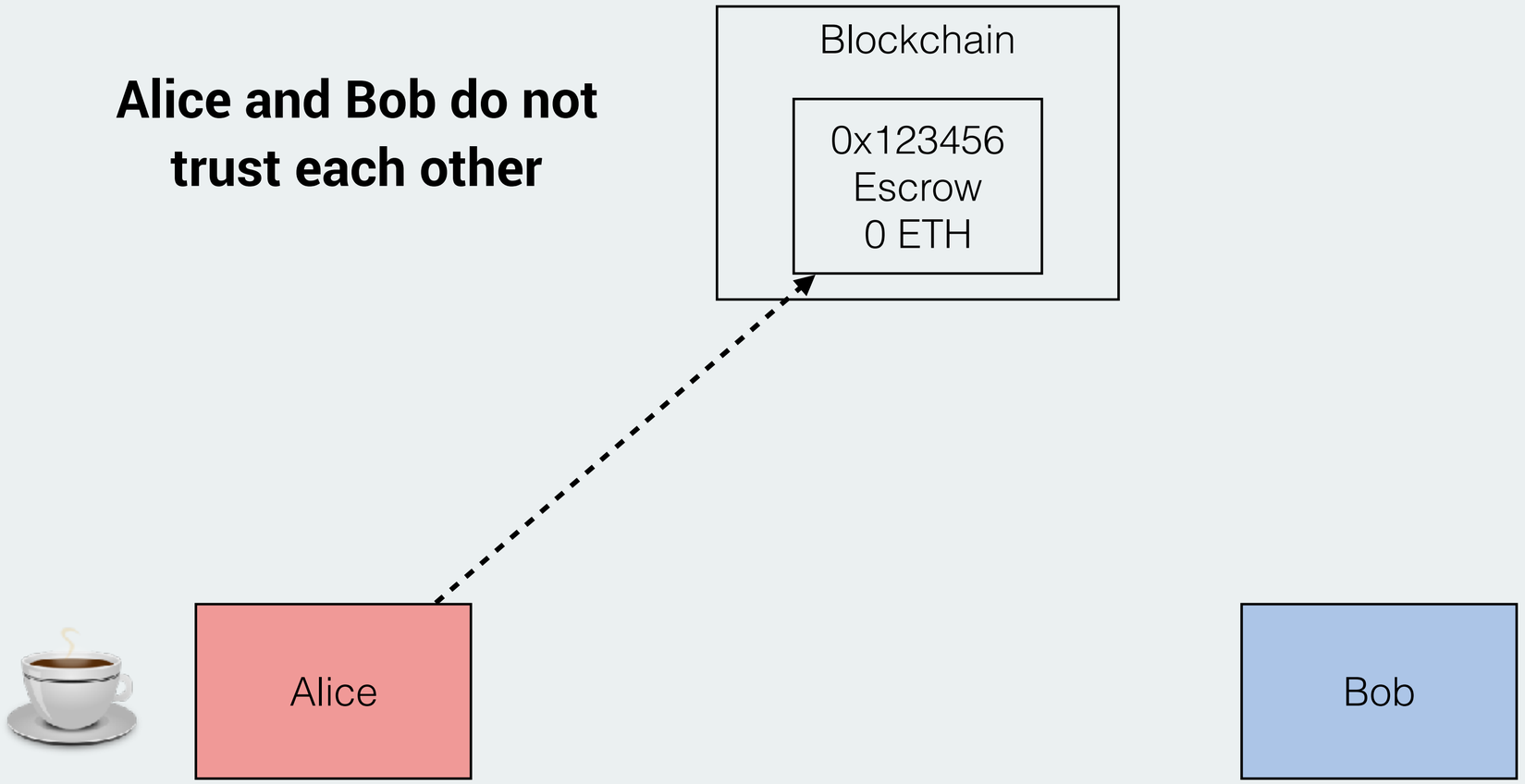
Bob sees Alice's offer and wants to buy



Sends a private message to Alice

ethereum Marketplace

Alice and Bob do not trust each other

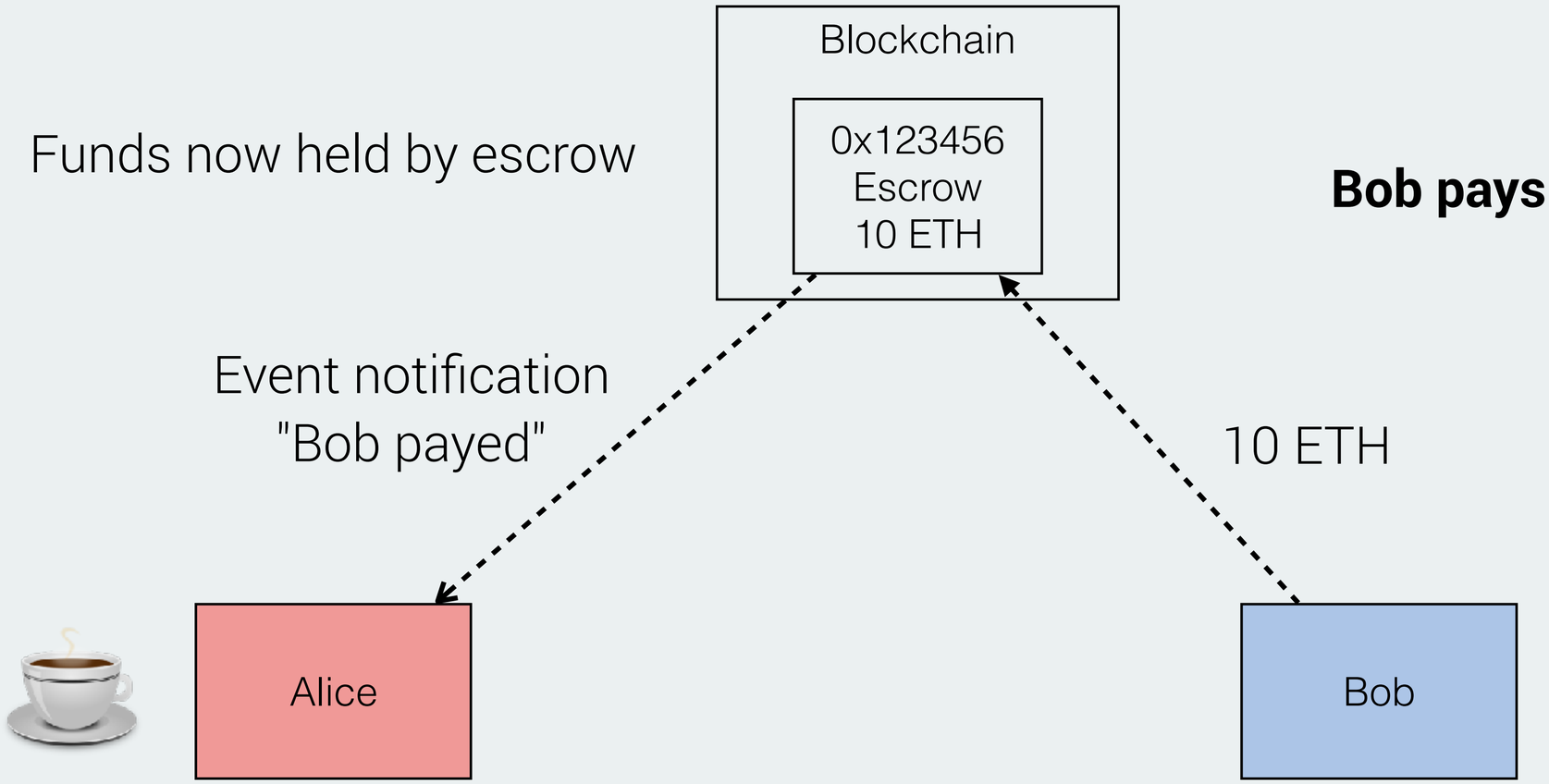


Alice creates an escrow contract

ethereum Marketplace

Alice informs Bob about the escrow

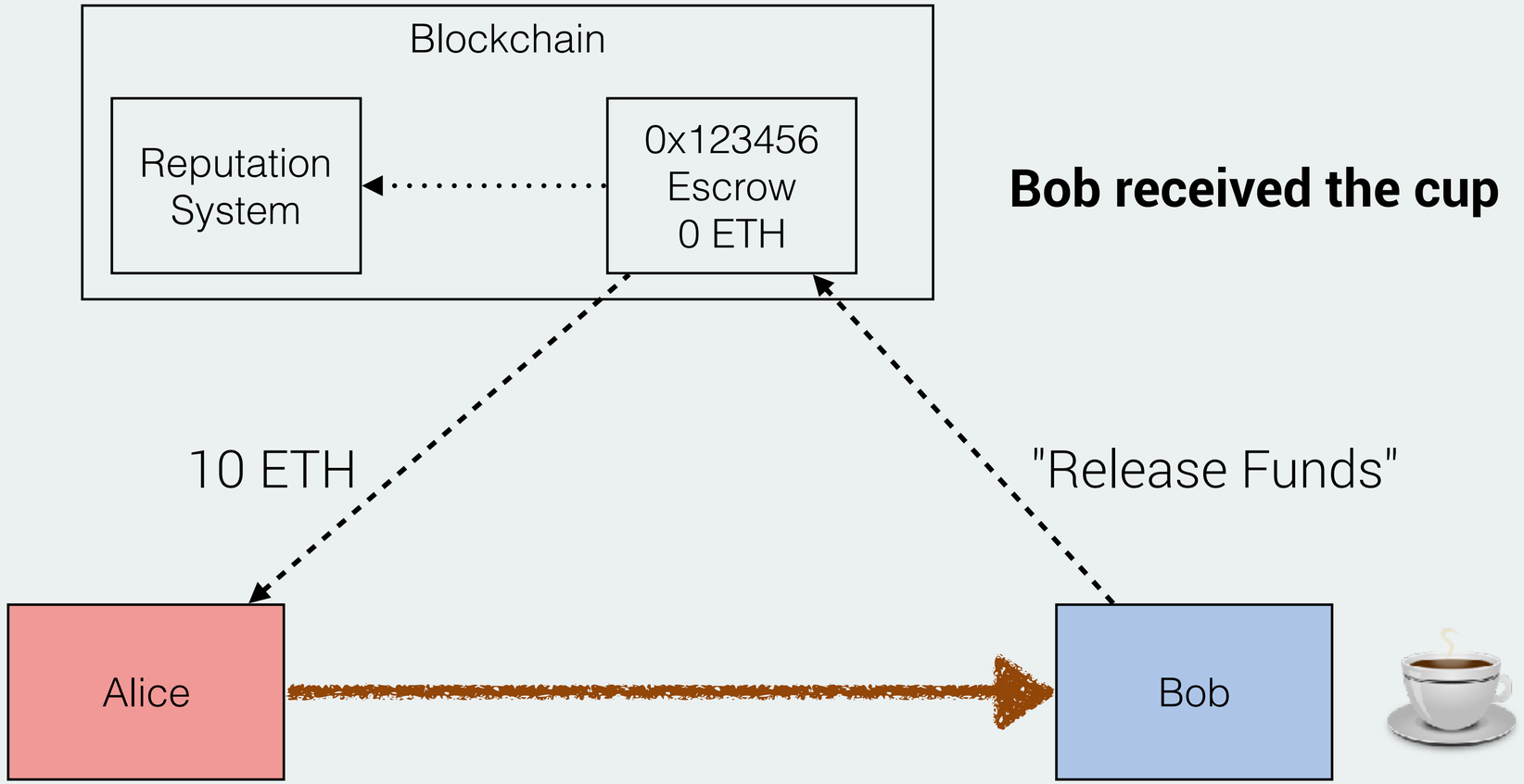
ethereum Marketplace



Alice watches the blockchain

Marketplace

ethereum Marketplace



2.0 and beyond

Abstraction

- Contract pays fee
- Other signing mechanisms

Casper

- Proof of Stake with finality
- Prediction market for blocks

Scalability

- Sharding (also offchain solutions like Raiden)

ethereum Release Process