Quorum

University of California, Berkeley 2017

Motivation

Use cases

Ethereum as a starting point

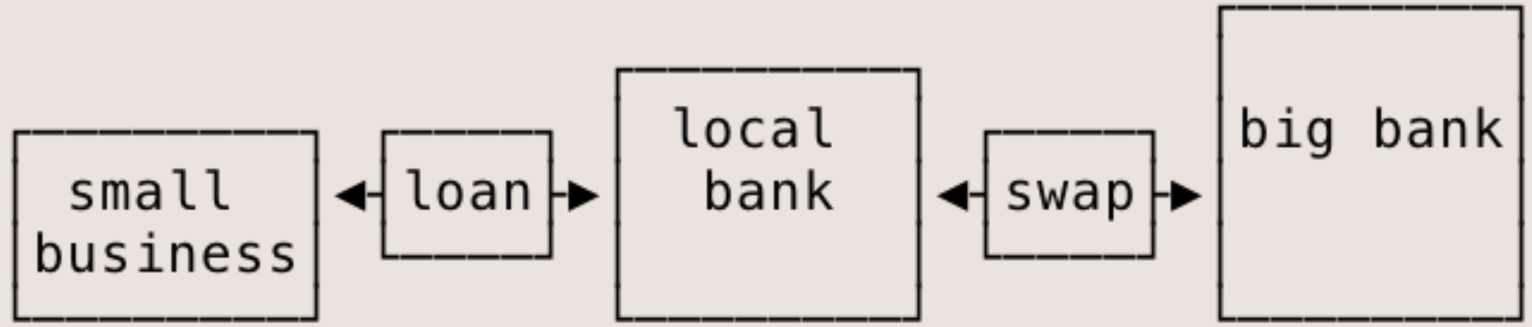
Ethereum as a starting point

Use cases

Motivation

Interbank Information Network

Credit Default Swap



Ethereum as a starting point

Use cases

Motivation

Distributed Database	Public Blockchain
closed, single operator	open, multiple operators
trust among nodes	trustless, censorship resistant
fast, capable of strong consistency	slow, eventual consistency
store of mutable state	log of state transitions

Motivation

Use cases

Ethereum as a starting point

Distributed Database	...	Public Blockchain
closed, single operator	multiple known operators	open, multiple operators
trust among nodes	accountability	trustless, censorship resistant
fast, capable of strong consistency	strong, not eventual consistency	slow, eventual consistency
store of mutable state	log of state transitions	log of state transitions

Also—

Confidential transactions

Real-world governance (tech *and* law)

Enterprise deployment & support

Simple Privacy

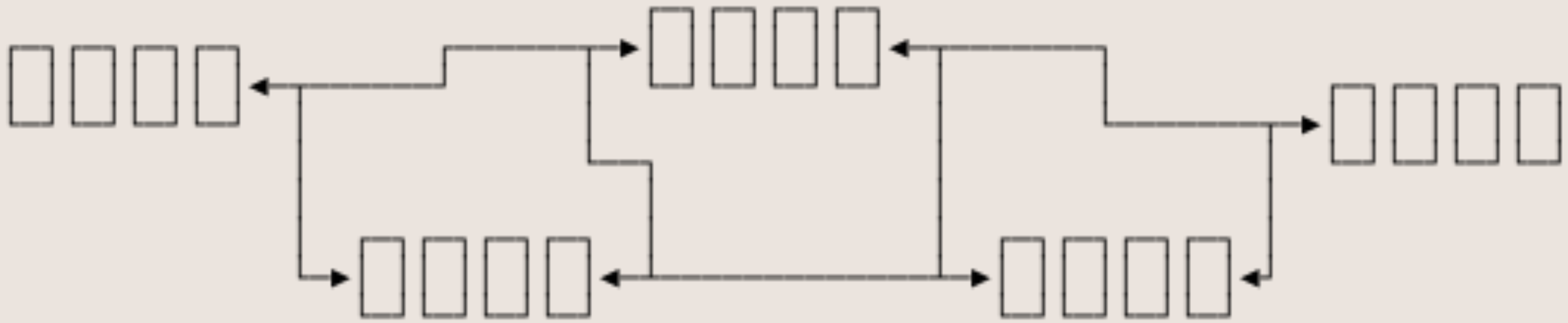
Two separate state trees

Public State
Private State

Simple Privacy

Constellation

Ethereum network



Constellation

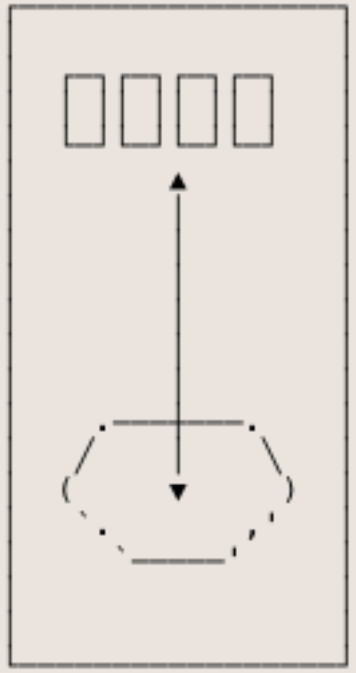
Simple Privacy

One node

Constellation

Simple Privacy

With a private enclave

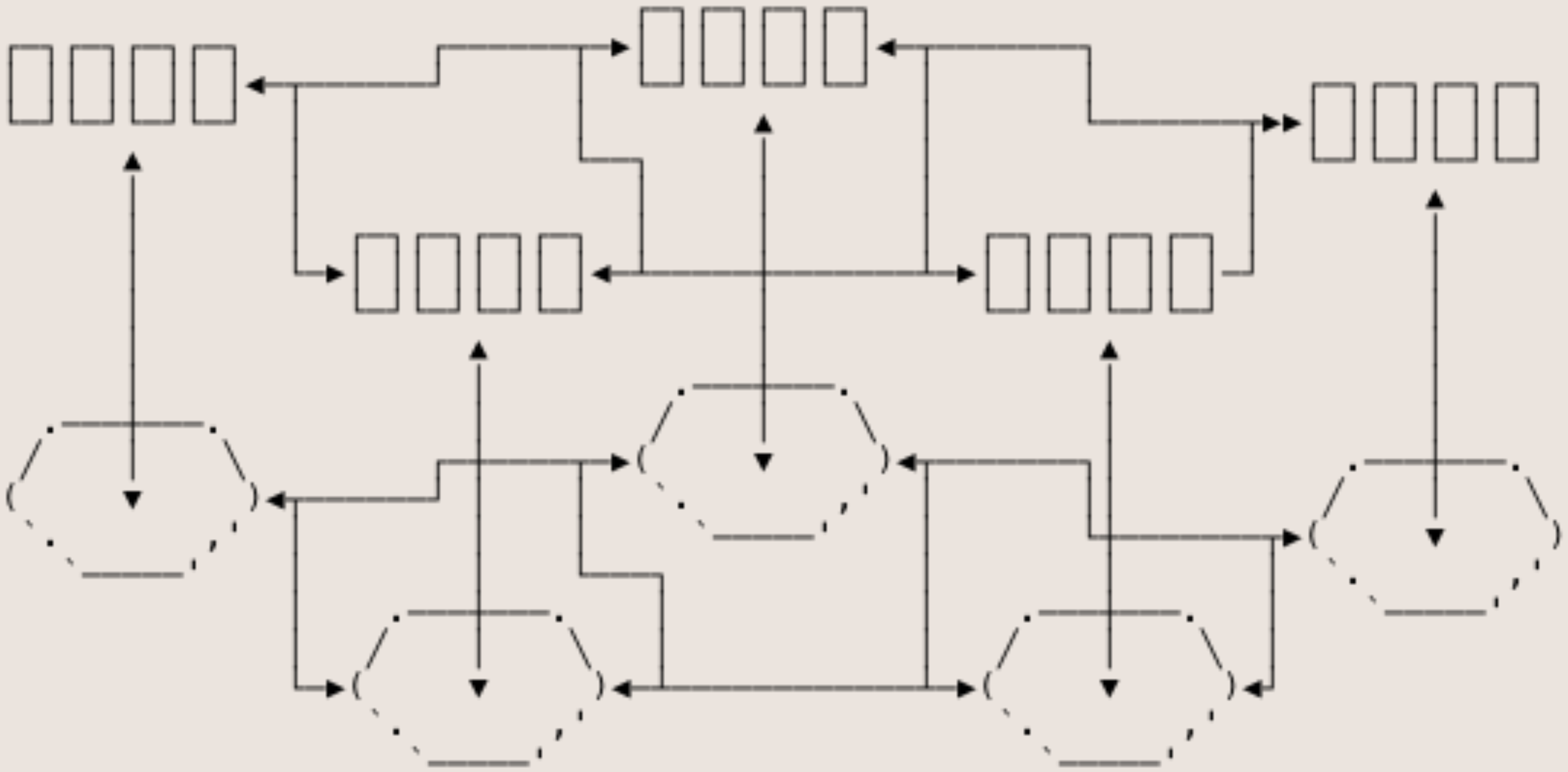


Constellation

Simple Privacy

Quorum network

Peer-to-peer encrypted message exchange



Simple Privacy

Creating a Private Contract

```
var simple = checkingAccountContract.new(42, {
  from: web3.eth.accounts[0],
  data: bytecode,
  gas: 300000,
});
```

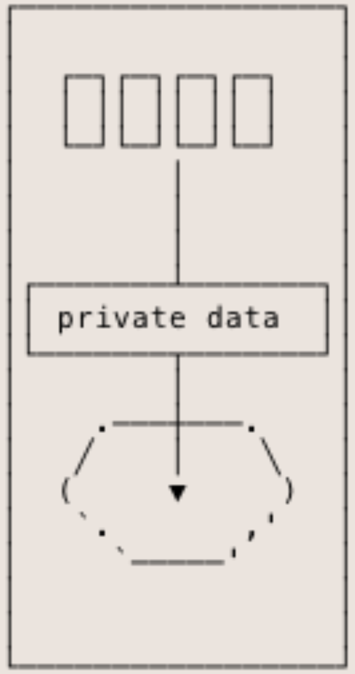
Creating a Private Contract

Simple Privacy

```
var simple = checkingAccountContract.new(42, {
  from: web3.eth.accounts[0],
  data: bytecode,
  gas: 300000,

  privateFor: ["ROAZBWtSacxXQrOe3FGAqJDyJjFePR5ce4TSIzmJ0Bc="]
  //           <-               public key                 ->
});
```

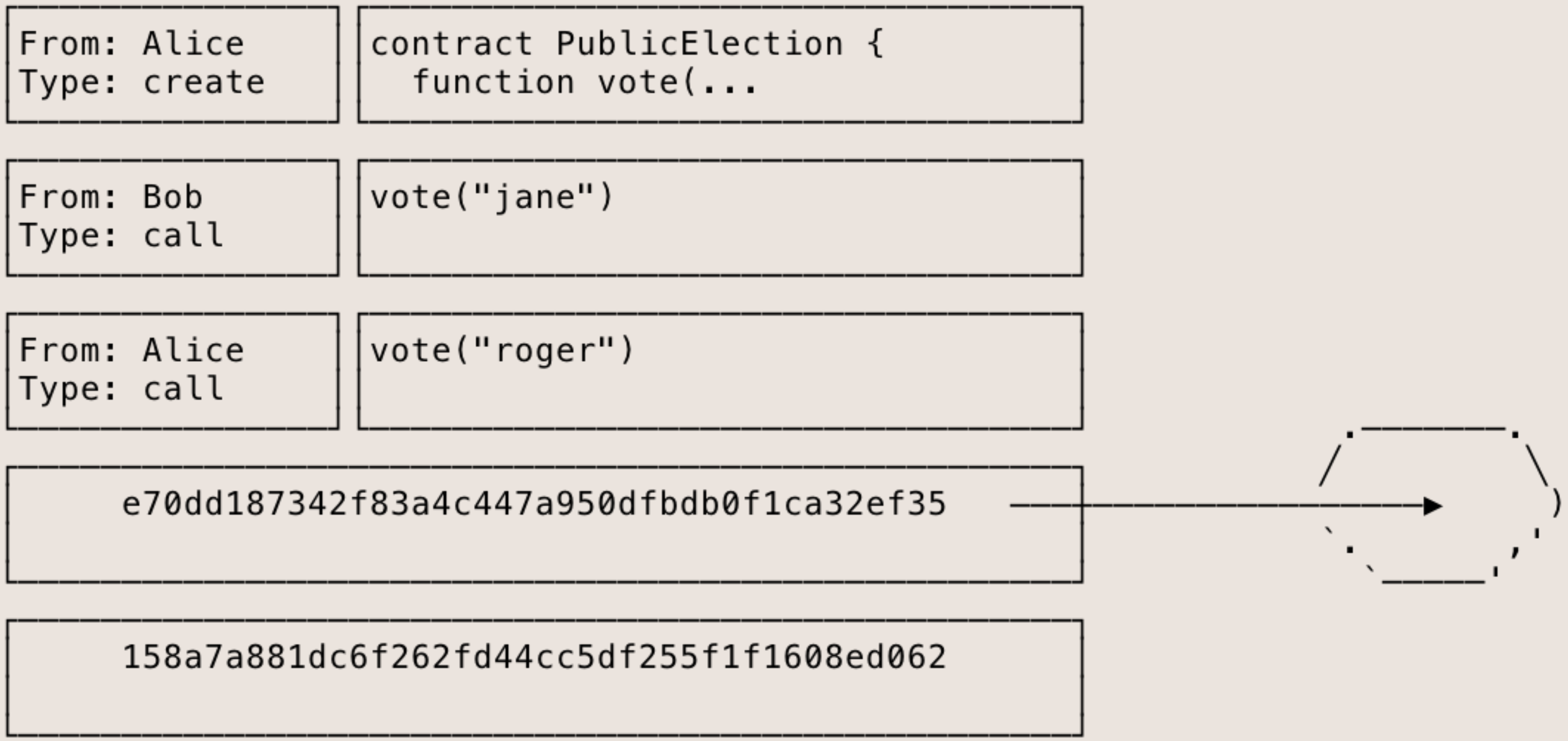
Creating a Private Contract

Simple Privacy



Conseus with Private State

Simple Privacy



```
From: Alice
Type: create
```

```
contract PublicElection {
  function vote(...
```

```
From: Bob
Type: call
```

```
vote("jane")
```

```
From: Alice
Type: call
```

```
vote("roger")
```

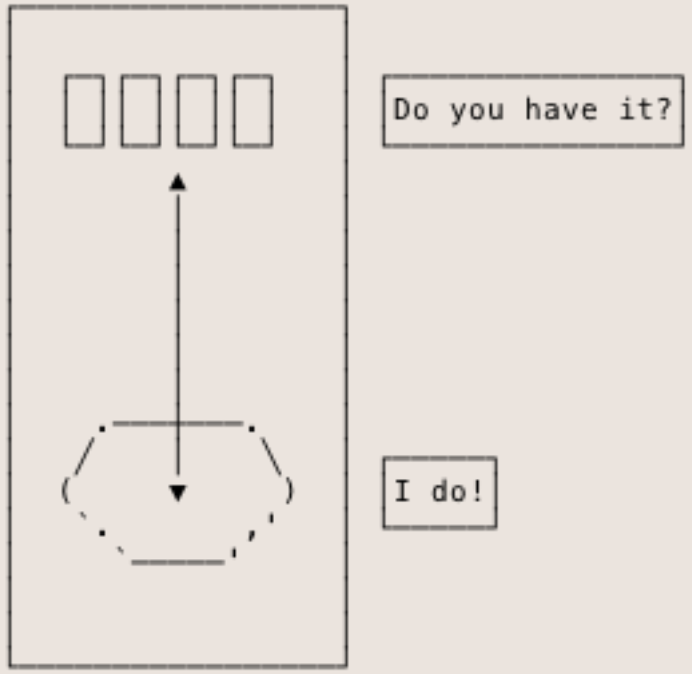
```
e70dd187342f83a4c447a950dfbdb0f1ca32ef35
```

```
158a7a881dc6f262fd44cc5df255f1f1608ed062
```

Conseus with Private State

Simple Privacy

IMAGE



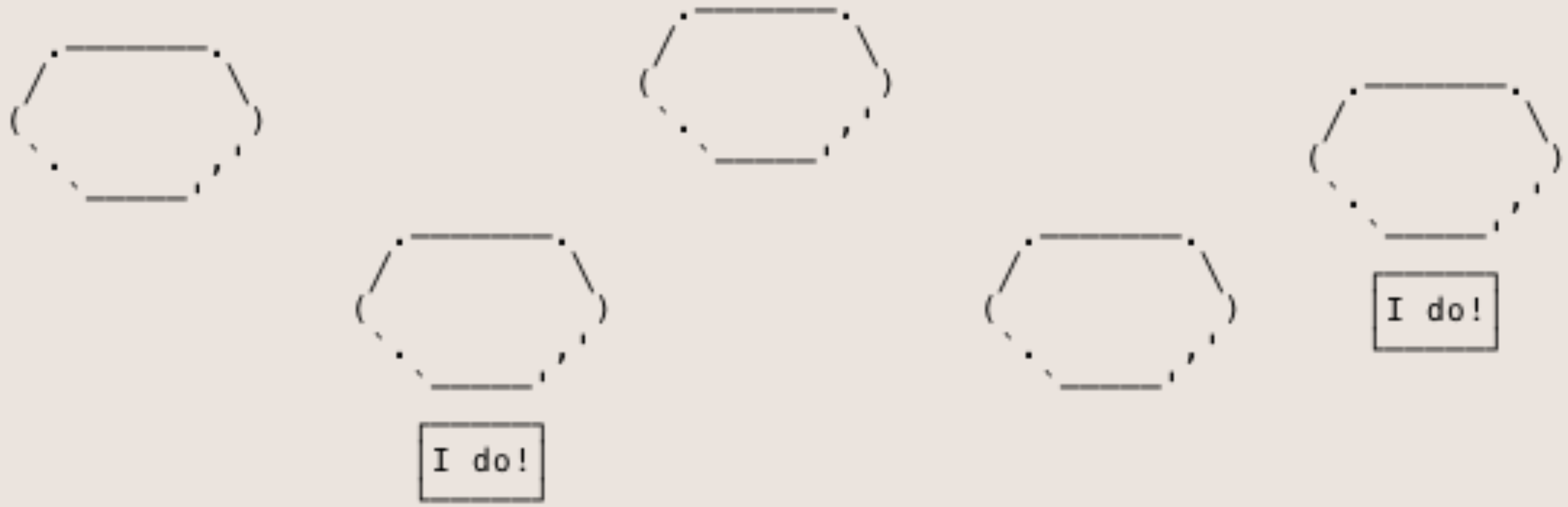
Do you have it?

I do!

Conseus with Private State

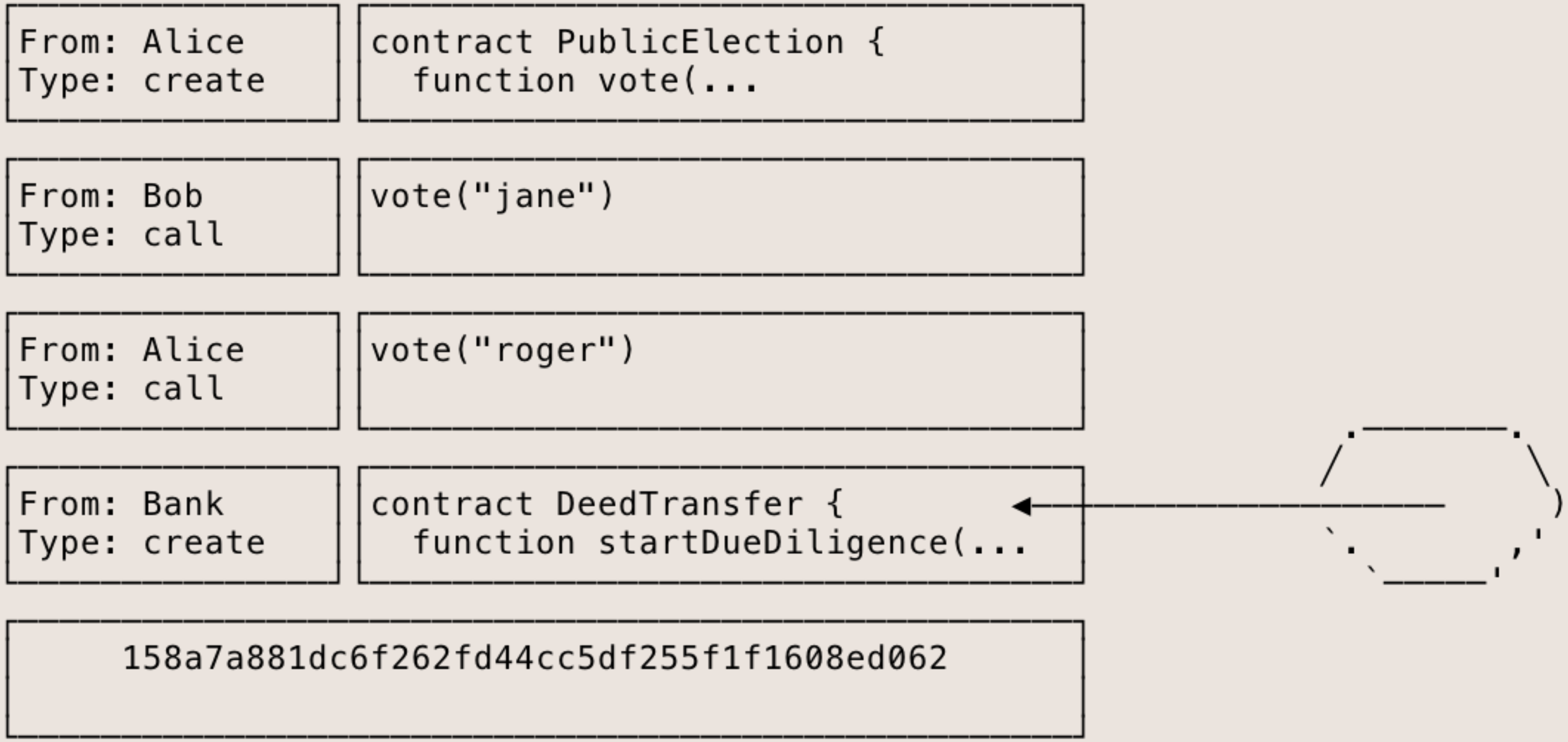
Simple Privacy

Who has this payload?



Conseus with Private State

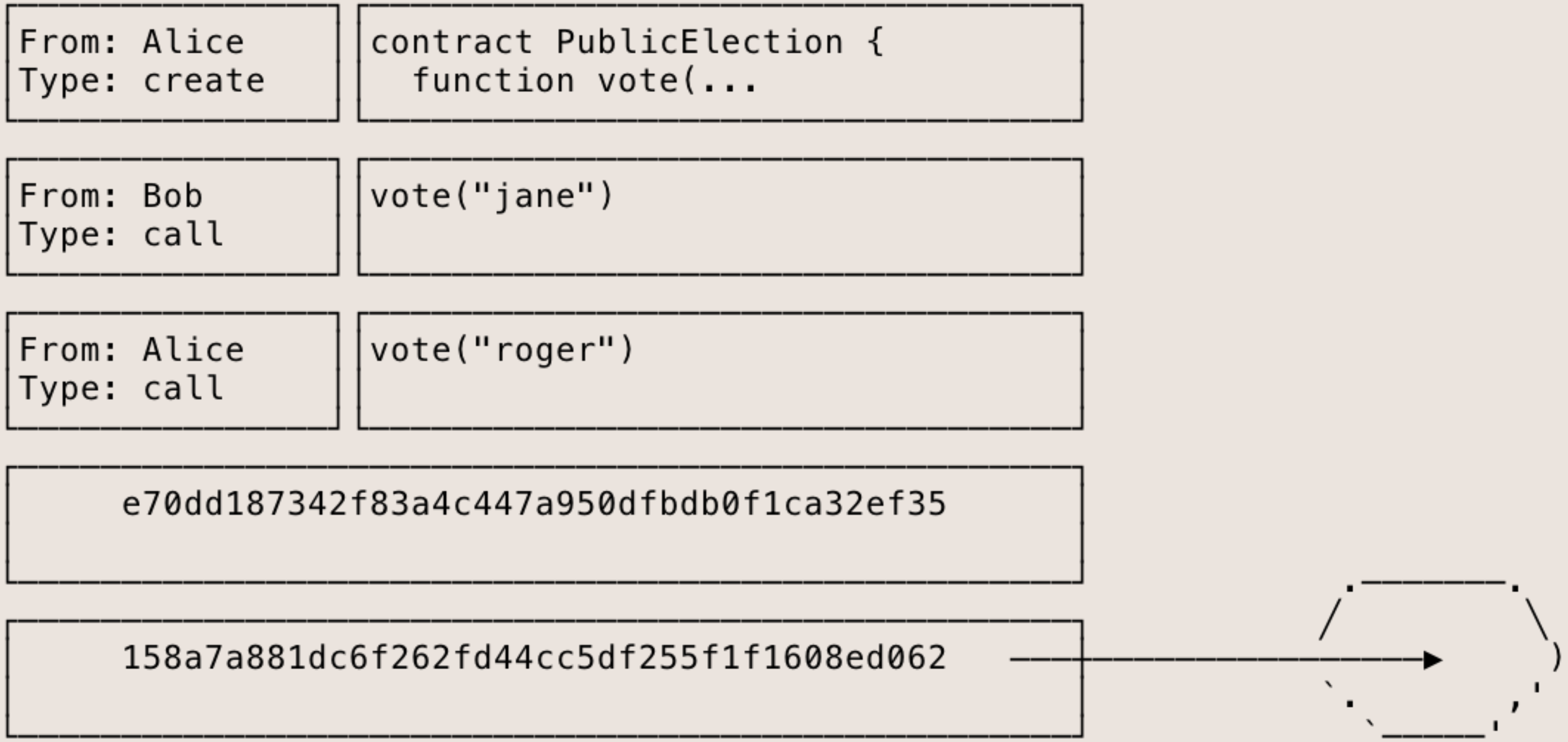
Simple Privacy



```
From: Alice     contract PublicElection {
Type: create      function vote(...

From: Bob       vote("jane")
Type: call

From: Alice     vote("roger")
Type: call

From: Bank      contract DeedTransfer {
Type: create      function startDueDiligence(...

      158a7a881dc6f262fd44cc5df255f1f1608ed062
```

Conseus with Private State

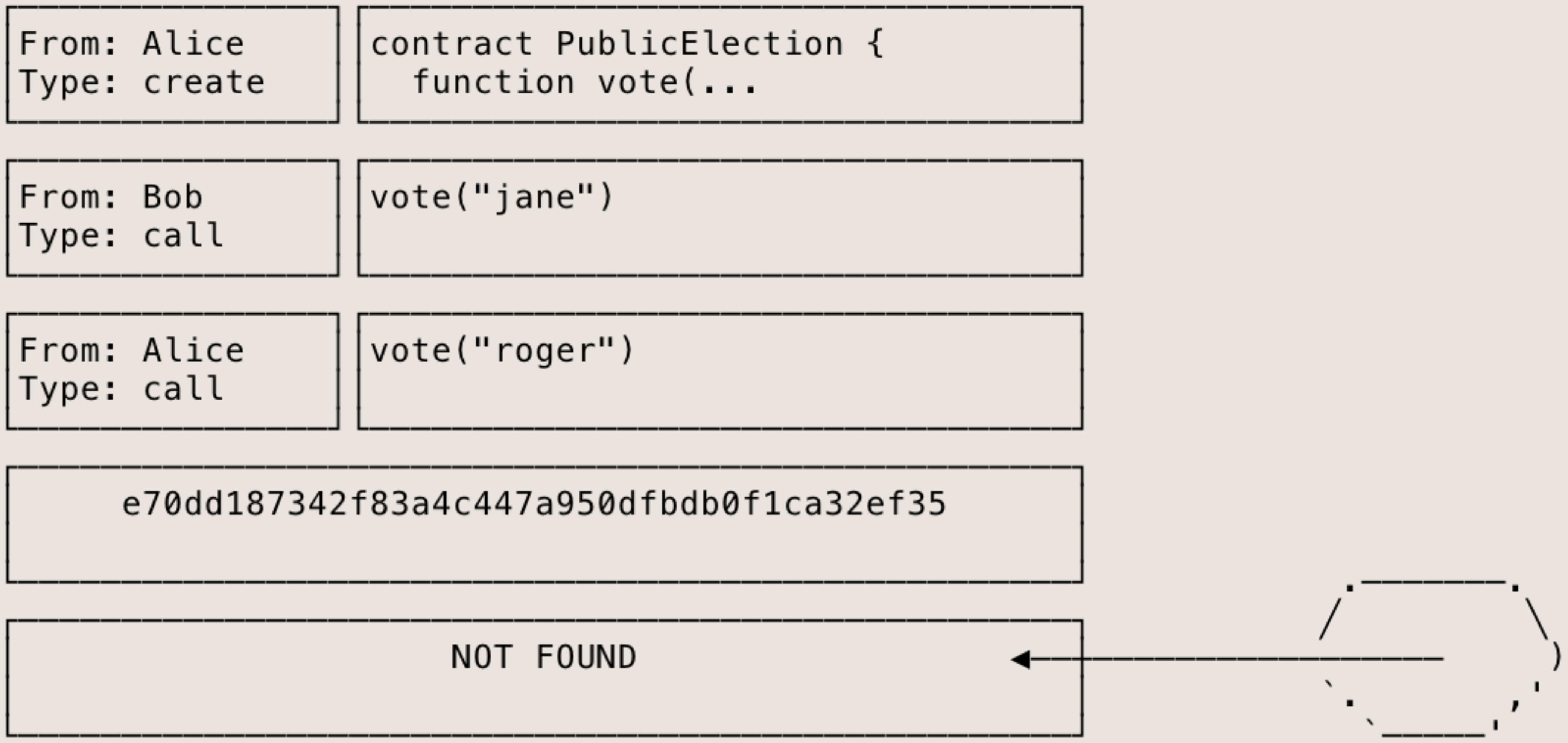
Simple Privacy



```
From: Alice     contract PublicElection {
Type: create      function vote(...

From: Bob       vote("jane")
Type: call

From: Alice     vote("roger")
Type: call

      e70dd187342f83a4c447a950dfbdb0f1ca32ef35

      158a7a881dc6f262fd44cc5df255f1f1608ed062
```

Conseus with Private State

Simple Privacy



```
From: Alice
Type: create
```

```
contract PublicElection {
  function vote(...
```

```
From: Bob
Type: call
```

```
vote("jane")
```

```
From: Alice
Type: call
```

```
vote("roger")
```

```
e70dd187342f83a4c447a950dfbdb0f1ca32ef35
```

```
NOT FOUND
```

Creating Other Contracts

Simple Privacy

Private contracts can call other private contracts
Private contracts can also call public contracts

But…

Simple Privacy

Creating Other Contracts

DEMO

Consensus

Consensus

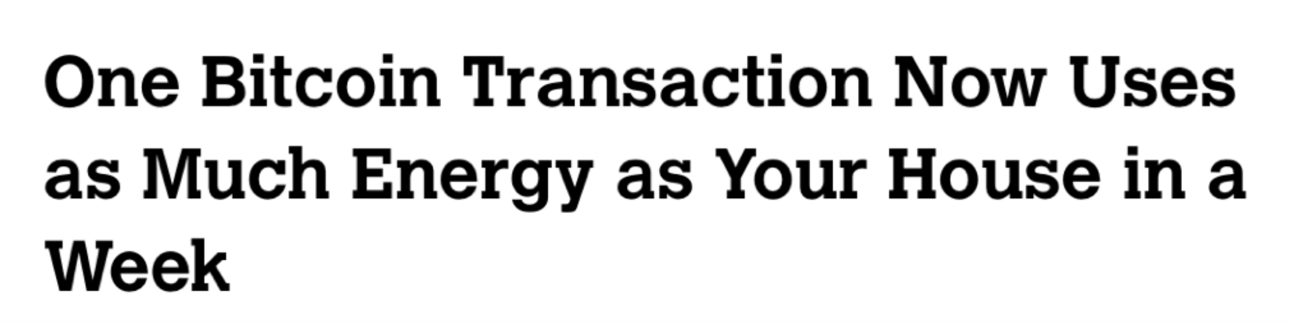
Proof of Work

Everyone is anonymous

Mutual lack of trust

Mining power as proxy for:

- Investment in the network
- How much of the vote you get



One Bitcoin Transaction Now Uses as Much Energy as Your House in a Week

Everyone is ~~anonymous~~ known

Mutual ~~lack of~~ trust

Mining is not necessary

Enterprise

Consensus

Raft

Consensus

What does a consensus mechanism do?

1	a = 1
2	b = 2
3	a = 100
4	c = 5

“Raft is a consensus algorithm that is designed to be easy to understand. It’s equivalent to Paxos in fault-tolerance and performance.”

Consensus

Raft

Formally verified protocol

We use the etcd implementation, which is written in Go and not verified, but mature

Consensus

Raft

Strenghts, Weaknesses, Limitations

Censorship

Cluster size

Throughput / latency

No forking

Cluster Size

Servers	Quorum Size (majority)	Failure Tolerance
1	1	0
2	2	0
3	2	1*
4	3	1
5	3	2*
…	…	…

Consensus

Raft

Strenghts, Weaknesses, Limitations

Throughput

Up to 1100 tx/s (ideal conditions)

0 - 50 ms latency

Ethereum + Raft

Consensus

Ethereum	Raft
~~miner~~ minter	leader
verifier	follower

Consensus

Ethereum + Raft

“Speculative Minting”

Mint every 50 ms

Raft can take arbitrarily long to confirm blocks

Consensus

Istanbul BFT/PBFT

Based on PBFT (Castro-Liskov 99)

Up to F of N fault nodes (N = 3F + 1)

Doesn’t scale to as many nodes

Censorship resistant

Consensus

New Work

The Honey Badger of BFT Protocols

- Miller, Xia, Croman, Shi, Song

Thunderella: Blockchains with Optimistic Instant Confirmation

- Pass, Shi

ZSL

```
assert(presentationEnded);
```