Not sure if this is the best place to ask questions, but this is something that bugged me for a while. Assume we have a network like Bitcoin, but without a native token. That is, block headers stay basically the same, but transactions are just bulk, arbitrary data rather than outputs. Miners include transactions to blocks by sorting them by hash (PoW) or other incentives. In other words, such network would be merely an ever-growing, resilient log of data.

Now, imagine a group of people agree on a single function compute_dapp_state(txs : Vec) -> Bytes

which takes a log of buffers, filters it looking for a specific format, interprets those buffers as transactions and returns a final state. As an example, suppose I develop a crypto_bunnies()

function that 1. filters transactions in the format I,want to sell bunny , signed:

, 2. parses it, 3. checks signatures, 4. updates a state, 5. returns it.

As long that group of people keep the same copy of crypto_bunnies()

and have access to the token-less blockchain, couldn’t they essentially emulate smart-contracts? That is, to interact with the contract, they just need to make a transaction on the token-less chain with the format expected by its defining function, i.e., crypto_bunnies()

. Others users would be automatically notified of it, since they all access the same chain of logs. In other words, computation would be 100% offline; full nodes would not perform any computation at all, they’d only aggregate events. Then, users of a smart contract would agree with the same deterministic function to compute the application’s state based on those events.

Surely, such network would have major drawbacks. For example, since the merkle-patricia root of the state of a block ins’t included on its header, full nodes can’t produce short proofs of selected states; in fact, chances are full nodes don’t even compute such states. Also, not having a gas mechanism means each contract would need to ensure its computational cost is a linear function on the number of transactions, otherwise it’d easily become too expensive. On the other hands, there would be no more such a thing as a Crypto-Kitties like DApp DDOS’ing the network, since the consensus-relevant part of the system doesn’t perform any computation.

Nether less, is any fundamental flaw on this understanding that a token-less, data-only blockchain coupled with a reducer function like that would essentially implement smart contracts? Is there any obvious attack that could be performed on such design, that is somehow prevented on Ethereum?