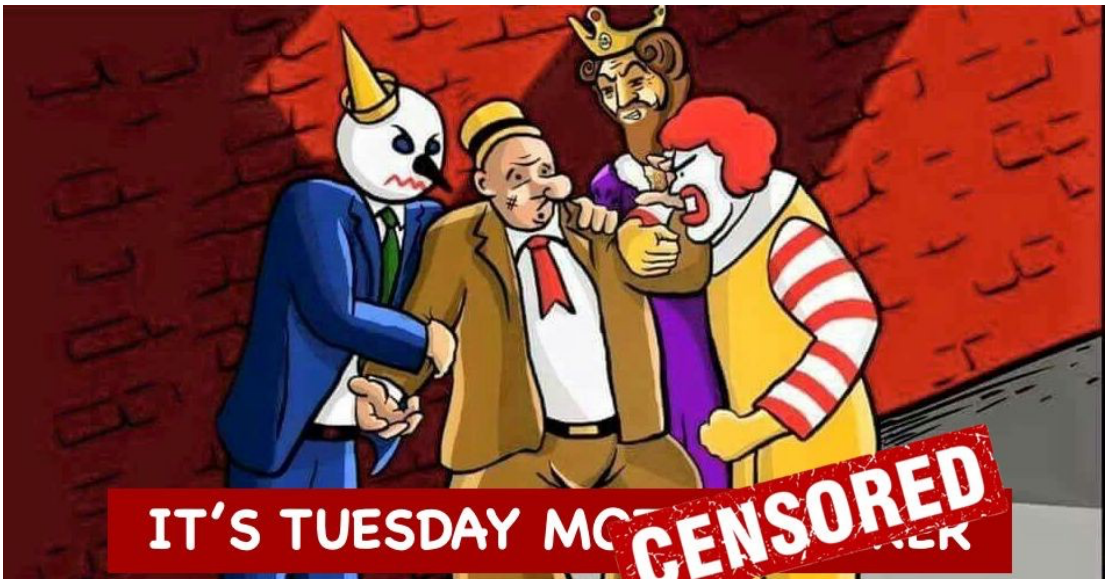
Code-is-Law, Pure DeFi, & Measuring Real Events on the Virtual Machine

"Having Ethereum's virtual machine constantly validate every contract's status based on the passage of time and all events remains prohibitively expensive and impractical."



Only a few years after decentralized finance's rollout (DeFi), its champions argue that it's ready to replace real-world companies and exchanges.

Theoretically, an array of smart contracts can be uploaded to blockchain addresses to operate as banks, brokerages, insurers, and exchanges. These smart contracts can contain immutable code moving "money" from place-to-place based on transaction requests and market outcomes.

But limitations in core blockchain make "pure" DeFi an impossibility, and platforms claiming to practice "pure" DeFi invariably have code and data strewn across local machines and their distributed ledger. Without clarity and standardization, the question of "which code is law" is unavoidable. In short, you can have either "code is law" orthodoxy or real-world applications, but you cannot have both (yet).

Blockchains are designed to maintain a record of their current state – who's transacted what, and by extension, who holds what. And in a genuinely trust-less system, theory demands that everything which might trigger, impact, or vary a blockchain's state must (or should) be validated with some kind of sufficiently distributed consensus process.

But, those DeFi protocols which are most successful at mimicking real-world businesses generally operate with high levels of centralized authority, localized

storage, and off-chain processes. Having Ethereum's virtual machine constantly validate every contract's status based on the passage of time and all events remains prohibitively expensive and impractical. Off-chain hybrid oracles are often used to trigger "autonomous" on-chain contracts. This current state-of-art results in DeFi either not delivering what's promised (in "the legend") or secretly operating Siemens machines as surrogates for the Edisons. [1] In order to practice real-time processes, consensus-based virtual machines generally require reminders and pokes -- that's why hybrid (and more 'localized) data designs have been quietly getting rolled into "pure" DeFi systems over the last several years.

Blockchain and Smart Contract Realities

Blockchains are distributed, replicated, deterministic state machines, operated and secured with elliptic curve cryptography; none of Citibank, the CBOE, or the NYSE can make this sexy claim. At their core, blockchains are shared ledgers, where Alice and Bob can transfer coins to each other without cheating, and an army of validators are rewarded for checking the validity of every submitted transfer against the ledger's current state. Transactions consistent with the current state are appended to the ledger, and they become part of the new current state.

The Ethereum blockchain launched in 2015 introduced the first scalable smart contract architecture where computer code stored at blockchain addresses could execute on the blockchain's virtual machine based on values and data available on the blockchain[2]. By design, Ethereum's base level smart contracts were limited in their ability to interact with the outside world to avoid unchecked data and other dangers relating to outside internet calls.

Practical Solutions

Computers are very good at tirelessly monitoring conditions and transmitting and validating transactions based on conditions; high-frequency equity trading, exchange traded fund arbitrage, and listed futures market-making are good examples. Each of these systems are purpose built for transactional speed in data processing and accuracy in recording.

But, most DeFi teams originally tackled real-word data applications through the same lens as the trustless transaction between "Alice and Bob"; a consensus-based framework run through a cryptographically secured fully distributed virtual machine. Further, where a data item is consequential, there's likely only one answer and probably only one place to get it. For example, why would we want to poll crypto Hodlers for the closing price of MSFT, and even crazier, waste time and machine resources validating the polling? What's necessary for transaction records is almost never the right approach for (consequential) real-world data.

Where DeFi teams have succeeded, they've generally hidden the hybrid and off-chain components necessary to actually run time-sensitive and commercially viable systems. More centralized "Siemens machines" are now necessarily buried throughout Github and "mainnet" code to make things more accurate, more robust, and practical to use.

Away from DeFi's regulatory debates, DeFi teams will get much closer to (real-world) prime time with a more holistic view of systems architecture and building for purpose.

[1] Theranos

[2] It's arguable that versions of smart contracts and scripting languages preceded the Ethereum blockchain.