How the TEA Project Solves a Central Web3 Problem

The movement of developers to Web3 hasn't yet created an adequate architecture for devs to deploy fully decentralized Web3 dApps. Currently, the lack of a development framework for dApps has acted as a bottleneck for widespread deployment of Web3 dApps. That's where the TEA Project comes in. Just as Ruby on Rails ushered in the Web 2.0 era by allowing full-scale internet applications to be built rapidly, the TEA Project aims to provide a similar development framework for Web3 that allows developers to quickly get their dApps up and running.

Unfortunately, developers too often must cobble together hybrid dApps as they're often met with a blockchain layer that’s too slow for running rich dApps. Which raises a question many users might have about Web3:

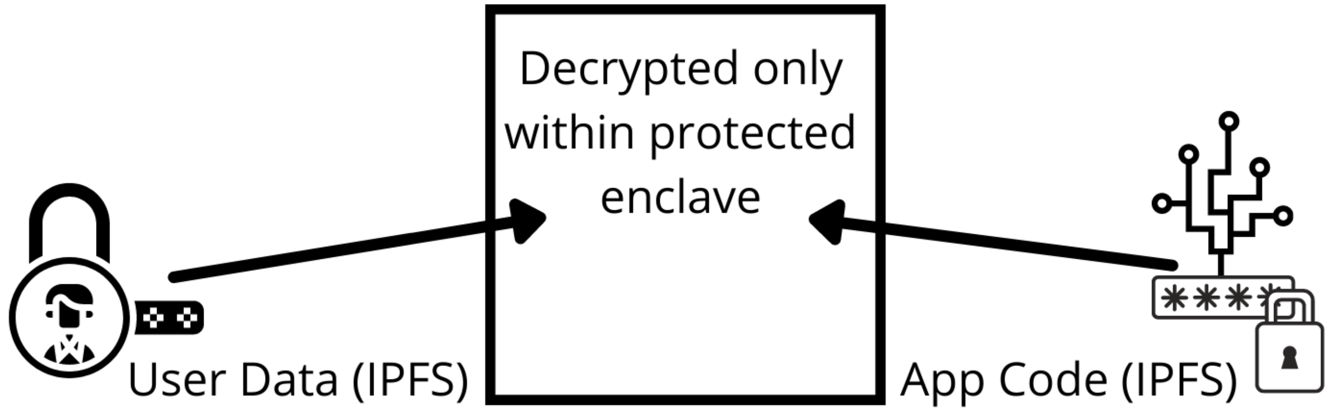
> Are smart contracts a necessary component of Web3?

The answer from the TEA Project's perspective is no, they're not. Smart contracts have two major limitations: they're limited in the type of algorithms they can execute, and they require consensus. The TEA Project doesn't use smart contracts as part of its execution layer for dApps. Smart contracts have their applications in areas such as DeFi, but they can't possibly be the basis of rich dApps that rival the speed and functionality of current centralized cloud computing apps.

And while traditional cloud apps achieve their speed at the high cost of centralization, the TEA Project's dApps (known as TApps) run full speed while still remaining decentralized. TEA Project TApps combine decentralization, speed, and security through its two-layer blockchain design.

- The layer-1 blockchain keeps trust data on the mining nodes running on layer-2. This layer deals with Byzantine fault and ensures that all layer-2 nodes are validated as trustworthy.
- TEA Project's layer-2 can ignore Byzantine faults as the decentralized mining nodes on layer-2 have already gained trust from layer-1. Each layer-2 mining node has a TPM-protected enclave where the encrypted app code interacts with encrypted customer data where neither the miners or the developers have access to the customer's data.

Layer-2 Node



Decrypted only within protected enclave

User Data (IPFS)

App Code (IPFS)

Layer-1

- Keeps the trust certificates for nodes running on layer-2.
- Stores the results of public attestation for layer-2 nodes.

The trust certification kept by layer-1 on all layer-2 nodes is how the TEA Project keeps the apps decentralized without sacrificing speed. Rich applications can run at cloud computing performance and scale because the TEA Project's layer-2 no longer has to deal with Byzantine faults which requires blockchains to continually inspect nodes for signs of bad actors. Circumventing slow processing speeds inherent with BFT is an endemic problem of decentralized systems. TEA's layer-1 blockchain still uses Byzantine fault tolerance. The longer times needed to reach consensus on TEA's layer-1 won't interfere with the faster speeds required on its layer-2. The TEA network only needs to use the Proof of Trust (PoT) data on layer-1 to reach a consensus on the trustability of nodes on layer-2.

In the TEA Project, dApps run WebAssembly (Wasm) code on layer-2 within hardware-protected enclaves. TEA's use of Wasm means developers can use the programming language they're accustomed to using, making it much more straightforward for developers to deploy apps to Web3 through the TEA Project. Additionally, developers accustomed to the front-end / back-end / database tiers of the cloud computing tech stack will also find them present in the TEA Project. TApps follow the same three-tier architecture prevalent in cloud computing (presentation tier, app tier, and database tier), but TEA Project TApps run fully decentralized.

Applications can run full speed on the TEA Project's layer-2 because, from the app's point of view, it looks just like a normal three-tier cloud computing architecture. And developers themselves don't need any special knowledge about blockchain or consensus algorithms to get the decentralization and security benefits inherent in the TEA Project's 2-layer blockchain. They just need to set WebAssembly as their code compilation target while still using their preferred programming languages.

The TEA Project's Three Roots of Trust

Traditional blockchains have two roots of trust: consensus and cryptography. But their reliance on consensus makes it impossible for these blockchains to scale and meet the needs of rich applications. The TEA Project has the usual blockchain roots of trust (consensus and cryptography) as part of its layer-1. But since our layer-2 isn't a blockchain and doesn't rely on consensus, we add two more roots of trust.

1. A hardware root of trust. One of TEA's roots of trust comes from the miners' hardware TPM security chips that generate PoT data stored on TEA's layer-1 blockchain. This setup allows our layer-1 to poll the layer-2 nodes' trust data in order to reach consensus on the trustworthiness of TEA's layer-2 mining nodes.

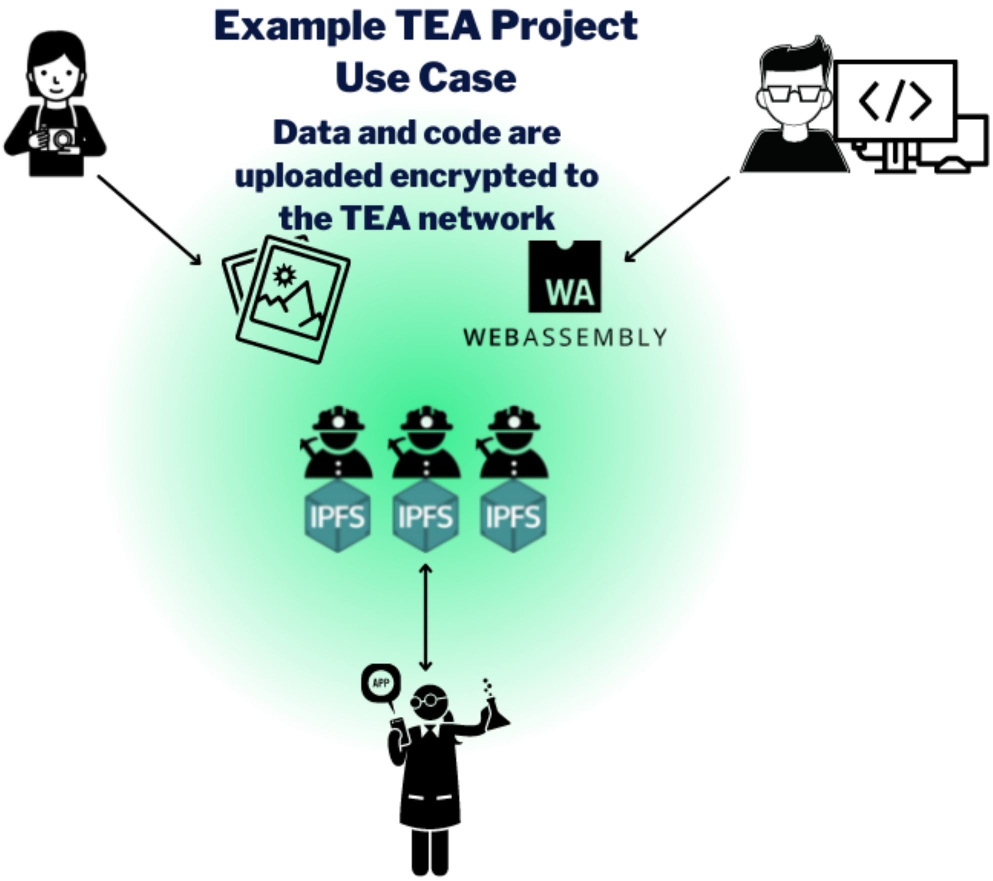
2. Time as a root of trust. The current state of every dApp running on the TEA network is maintained through a state machine that orders transactions using Proof of Time. Using time as a root of trust allows dApps to query the current state through the nearest node that works like a database tier from the app's perspective.

Not only are TEA Project dApps fully decentralized, but they can run full speed (thanks to the layer-2 nodes' protected execution enclaves) and can query a database layer (thanks to the layer-2 nodes' GPS modules).

How The TEA Project Looks in Use

The TEA Project is looking to disrupt the world of cloud computing by enabling decentralized apps to run on the blockchain at cloud computing speeds. Because we no longer have the cloud servers of web 2.0, we must incentivize miners to provide the underlying infrastructure. IPFS is the actual storage infrastructure for storing both encrypted app and client data, which miners will host within their protected enclaves when their node is selected for a task.

Let's look at the economic incentives for a hypothetical TApp running on the TEA network:



Example TEA Project Use Case

Data and code are uploaded encrypted to the TEA network

In this example, the end client, an AI researcher, uses the developer's Tensorflow image recognition app to analyze the photographer's pictures.

- The **developer** uploads their Tensorflow image recognition analysis TApp to IPFS.
- A **photographer** uploads their pictures to IPFS.
- A **researcher** wants to use the **developer's** TApp with the **photographer's** pictures.
- A random **miner** is selected to host the Tensorflow TApp and the pictures within the protected enclave of their mining machine.

In economic terms, the **researcher** pays to use the TApp. The payment flows through the **miner** who provided the node to perform the secure and private calculation, the **photographer** who is paid for the use of their pictures, and the **developer** who is paid for the usage of their code. Every transaction enriches a wide variety of participants in the TEA ecosystem and helps sustain and incentivize further usage.

The TEA Project is able to provide all participants a decentralized computation network that is both secure and private. In practical terms, that means the developer who wrote the code cannot breach the photographer's image. And the miner who is randomly selected to host it in their enclave cannot access either the photographer's data or the developer's code. And the researcher is only able to receive the result of the app's calculation and isn't able to access the photographer's data outside of the enclave. The security and privacy that the TEA Project's decentralized computing network is able to provide has wide range of emerging use cases like residential IoT data and patient health records.

Because the TEA Project is able to provide the underlying trust, there's no longer any trust required between all four parties.

The TEA Project's Two Tokens: TEA & CML

The TEA Project uses a dual token setup to incentivize participation among miners, developers, investors, and consumers in the TEA ecosystem.

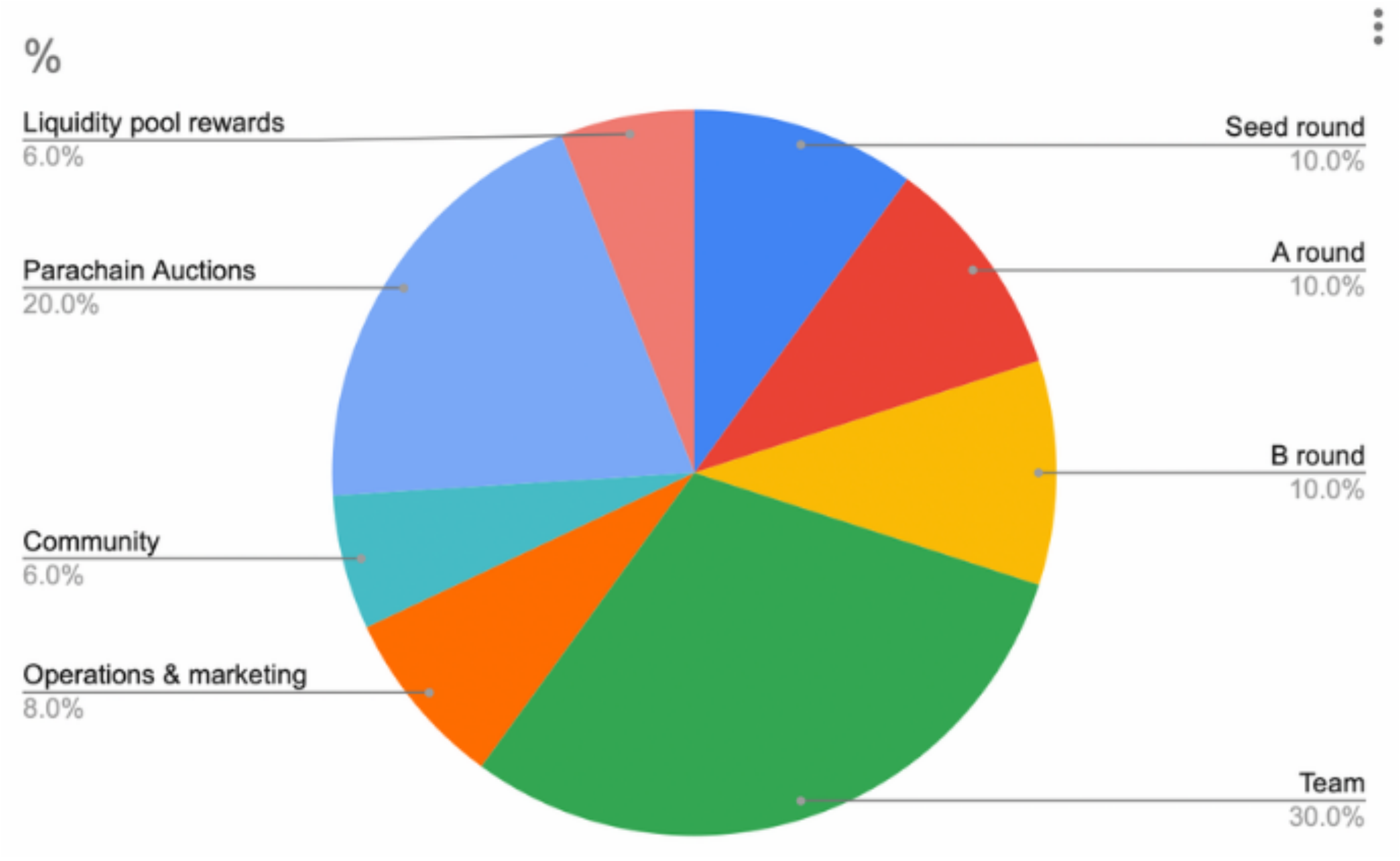
TEA

The first token, TEA, is a stablecoin utility token that will have a 100 million token pre-mine. TEA is also generated as the reward given to miners for running security validation services on the TEA network. TEA is burnt when users buy new CML tokens through DAO auctions. Its supply is therefore influenced by the supply and demand of CML tokens which miners need in order to mine TEA.

TEA has a variety of uses on the platform:

- TEA pays for gas when performing any transaction on the network.
- TEA is used by consumers to pay for TApps usage.
- For miners, TEA can be part of the reward for mining on the network.
- Users can also stake TEA to mining nodes to share in a percentage of that node's mining rewards.

The 100 million pre-mined tokens will be allocated with different vesting schedules (18 months for the investment rounds, 22 months for the community and team). The pre-mine allocation is as follows:



CML

The TEA Project's second token, CML, is an NFT with a life cycle determined randomly via algorithm. CML, also known as Camellia, has some unique properties and uses on the TEA Project platform:

- A TEA mining node can only be activated by associating a CML NFT with it. CML functions as a mining license and a credit record. Note that mining TEA doesn't require a GPU or an ASIC. It only requires the Camellia NFT and cheap secure hardware such as an RPi with a GPS module and a TPM chip.
- Miners buy new Camellia seeds through open bidding, with the TEA exchanged for them burned by the TEA Project DAO.
- Investors can stake their defrosted CML to mining machines for revenue sharing.
- Each CML has a lifespan associated with it (approximately 2 years). Its lifespan and other attributes such as productivity have variations that are generated algorithmically.
- Each mining machine's characteristics (e.g. its RAM and whether it has a graphics card) will be recorded within the CML nft.
- CML is issued by the DAO at a rate to regulate supply and ensure enough uptake by miners and investors to avoid flooding the market. The DAO also burns CML when it dies.

There are three types of CML seeds:

- **A CML** when planted into mining nodes provides the infrastructure for our layer-2 state machine as well as the TApps' database layer.
- **B CML** when planted into mining nodes host the TApps on the layer-2 network.
- **C CML** can be planted into mining nodes to perform public service tasks like remote attestation. C CML is primarily used for users who want to use TApps to interact with their own private data.

Although each CML has a lifespan, they can remain frozen past its defrost date for an indefinite amount of time. Investors who have participated in funding rounds in exchange for CML seeds can therefore keep their seeds frozen if they don't wish to mine with them.

Each TApp Can Have Its Own Token

The TEA Project is designed to allow for TApp token bonding curves for TApp developers and other creators to monetize their ideas. Investors back projects by purchasing TApp tokens on its bonding curve. TApp revenue, if set by the developer, can also go directly to the TApp token bonding curve. The gains from consumer app usage are therefore shared by the app developers, the hosting miners, and the investors. Additionally, any investors or enthusiastic users of the TApp can share in its profits by investing directly in the TApp's token.

TEA Ecosystem Development

Besides early investors, we envision the TEA Project ecosystem being built through successive outreach to the following three demographic groups: miners, developers, retail investors, and end-users.

Miners Prior to the maturity of the TEA Project's Web3 Rich dApps ecosystem, a mining economy is necessary to keep the TEA economy running. TEA’s carefully designed token economy creates NFT scarcity during mining. The scarcity encourages miners to reinvest their harvested TEA back into CML instead of selling TEA. Miners are of course free to host whatever TApps they wish with free market principles guiding them towards hosting TApps that are popular and able to reward miners.

Developers After attracting miners to the platform, the TEA Project will begin outreach to developers to build TApps using the TEA dev framework. Developers upload their apps to the TApp market and wait for miners to host them. The outreach phase will feature developer tutorials, hackathons, grants, as well as emphasize the ease of compiling to the WebAssembly format used by TApps.

Retail Investors Retail investors want to enjoy capital appreciation through investment. We expect retail investors to be interested in CML just like institutional investors, but there's also another way of participating in the TEA ecosystem as a retail investor: TApp tokens. Each TApp will have its own TApp token that investors can purchase as if investing in a stock. Each TApp token represents a share in the TApp and gives the holder dividends in the form of more TApp tokens issued to its holders whenever someone uses the TApp. As these tokens are issued on a bonding curve, the earliest TApp token investors will enjoy the largest amount of price appreciation.

End-users Concurrent to attracting developers to the platform, the TEA Project will actively market emerging TApps to consumers. We imagine the TEA Project being an ideal ecosystem for existing cloud apps looking to migrate and benefit from decentralization and data privacy. We look forward to the types of decentralized apps that will flourish on the TEA Project network. We also want to play an active part in welcoming consumers looking to make the leap from the centralized web 2.0 world to the TEA Project and Web3.

The TEA Project's Technical Details

TEA = Trusted Execution and Attestation

Any system with a high degree of anonymity and decentralization needs trust. The TEA Project leverages existing hardware security modules to achieve a secure computing environment that's both trustable and scalable. The TEA Project ecosystem achieves trust through the self-interested actions of the miners on its network. TEA miners are economically rewarded for running nodes that, through remote attestation, ensures that any TEA mining nodes hosting apps on the network haven't been tampered with. These attestations query the information stored inside each mining computer's hardware security module, and the reports are stored directly on the blockchain.

Multiple nodes will conduct remote attestations on any specific node before deciding if it's trustable. Once a node has successfully passed attestation, then these mining nodes gain two significant trusted computing capabilities:

1. All computations carried out within its hardware-protected enclave are trustable. The TEA Project's premise relative to TPM-protected mining hardware is simple: if we can trust the hardware integrity of the node, then we can trust the computational result. These hardware enclaves also ensure privacy for both the developer's code and the user's data.

2. The trustability of nodes is kept as Proof of Trust (PoT) data on the layer-1 blockchain. Because these nodes are trustable, they can run on the TEA Project's layer-2 without needing BFT (because they're already trustable.)

Let's summarize the above two points as follows:

1. **Trusted Execution** comes from protected enclaves where app logic can run while being protected by hardware security modules. All TEA Project TApps run inside of these protected enclaves. Nobody (which includes the app developer and the miners) has any control of the apps nor can they extract any data from the running enclaves.

2. **Trusted Attestation** refers to the process whereby the network nodes run reports on each other to ensure mining hardware running on the TEA network hasn't been tampered with. Once trusted status is attained, these nodes can run on TEA's layer-2 at cloud computing speeds without having to worry about (slow) BFT-consensus.

Transaction Consensus Using Time

We say that the TEA Project has three roots of trust: hardware, blockchain, and Proof of Time. We've already seen how hardware security modules and blockchain can create a trustable computing environment through remote attestation and trusted enclaves. This provides a scalable and trustable application execution tier for TApps, but we also need a data tier to track TApp state changes and transaction accounting.

The TEA Project has two state categories:

1. A strong-consistency state machine based on Proof of Time.

The first state category based on Proof of Time is for transactions requiring strong consistency, which would govern transactions involving funds and accounting. Using the time stamps from navigation satellites under watch of hardware attestation, our strong consistency state machine can achieve continuous state updates at a small synchronization cost. Its most crucial task is ensuring that the transaction sequence is consistent across all replicas. We do this by relying on the accurate time provided by the atomic clock of GPS satellites.

The reported time is recognized under the supervision of trusted TPM chips and used as the basis for the final ranking of all replicas. It's not necessary for all nodes to periodically reach a consensus on the latest block. But to ensure that most replicas can be synchronized to a consistent state, the

TEA Project's state machine requires a short waiting queue due to network latency. Since time is stable in our universe, it follows that each replica can achieve strong consistency using non-BFT consensus as the possibility of Byzantine faults has already been handled by the layer-1 blockchain.

2. An eventually consistent CRDT database built on OrbitDB that can be used ad hoc by TApps.

The other state category is a CRDT database that allows for short-term inconsistencies in the business logic of TApps. The TEA Project uses OrbitDB databases built on top of IPFS for these transactions. CRDT stands for conflict-free replication data type, which allows conflict-free mergers between different replications before ultimately achieving network-wide consistency. In fact, the business logic of most apps can tolerate short-term inconsistencies to achieve both decentralization and efficiency. A typical example of a traditional cloud app that deals with this issue gracefully is Google Docs.

Using time as a source of truth allows the TEA Project to sync the state machine between all of our layer-2 nodes without resorting to complicated consensus algorithms. This state machine also includes an embedded SQL-instance as detailed above which allows more complex apps to be built on the TEA platform.

Putting It All Together

The TEA Project aims to become an ideal onboarding point for developers looking to deploy to Web3. As an infrastructure provider for Web3, we designed our ecosystem to look very familiar to developers who will appreciate the straightforward 3-tier architecture that's similar to cloud computing.

- The **front-end** for TEA decentralized applications uses IPFS.
- The **server layer** handles dev code compiled to WASM, meaning developers can use their existing programming languages and set WASM as the output target.
- The **database layer** is handled according to the developer's needs - NoSQL data is sent to IPFS as governed by type-B CML mining

nodes, while relationship data is stored in GlueSQL as governed by type-A CML mining nodes.

We look forward to making the TEA Project the preferred development platform for developers looking to enter Web3. We believe the simplicity of our tech stack will be inviting to developers who won’t have to learn new languages or new blockchain concepts. They can enjoy the benefits of decentralization and data privacy that Web3 offers while deploying to a familiar 3-tier architecture using a mature development framework built for Web3.