TEA: Trusted Execution & Attestation

Elevating Decentralized Trusted Computing to a T

www.teaproject.org

The missions

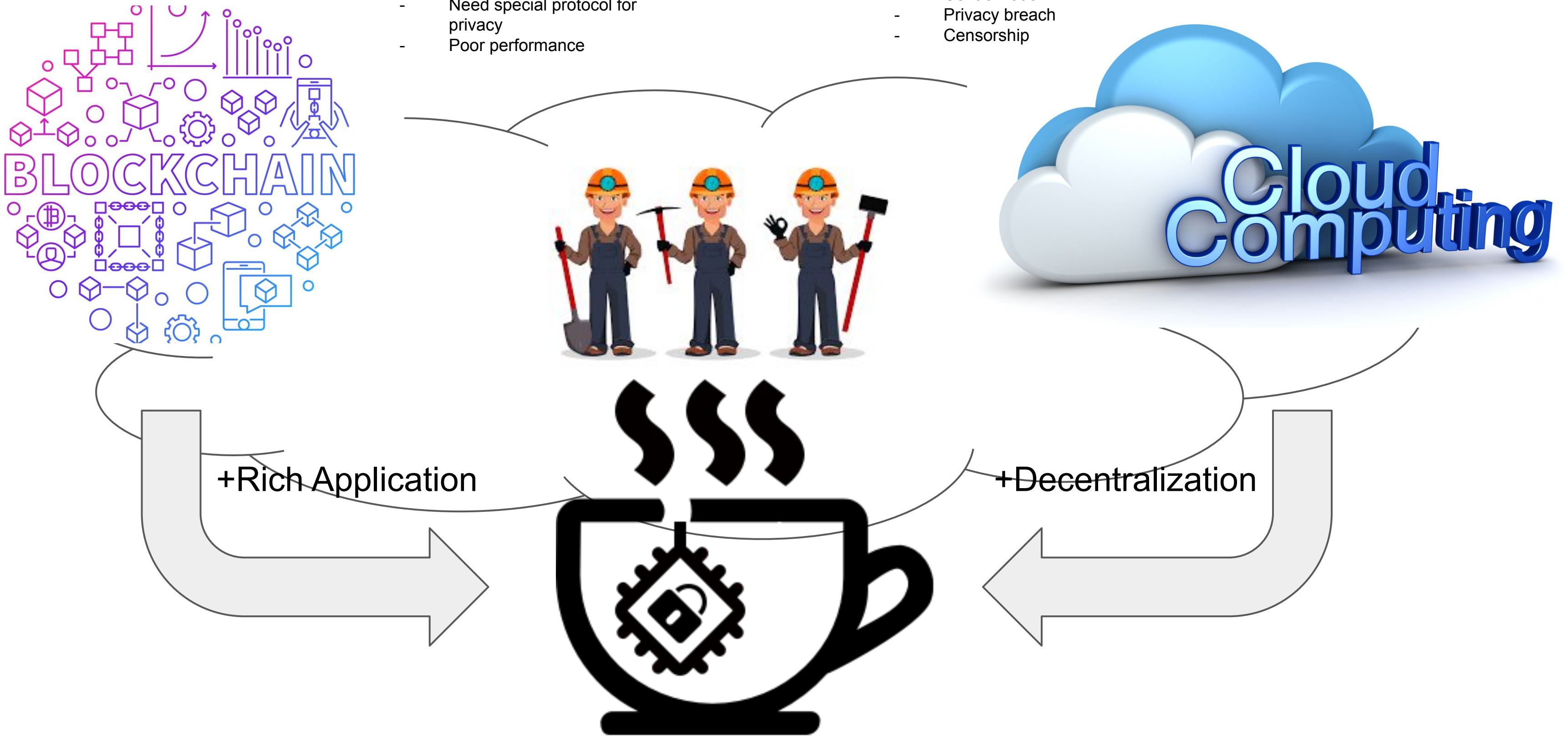
- A decentralized trusted cloud computing DAO, miners run the nodes
- A programming model that cloud apps can run decentralized
- A 2-layers consensus that blockchain dapps can be rich in UX
- Trusted and secure computation environment that protects privacy
- Censorship resistance

Decentralized but

- Hardly run rich applications
- Need special protocol for privacy
- Poor performance

Run rich application with high performance but

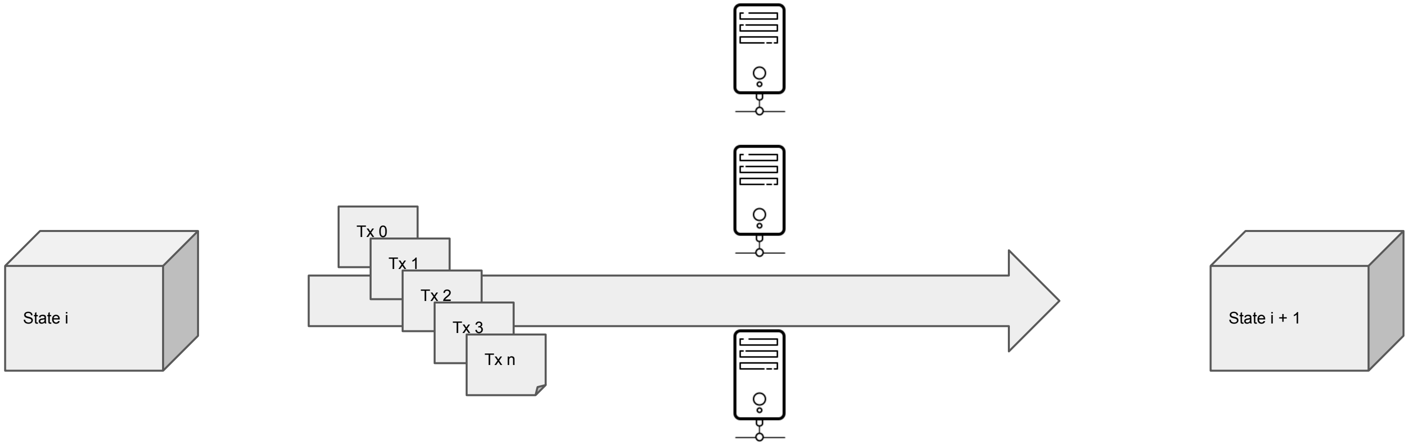
- Centralization
- Privacy breach
- Censorship



+Rich Application

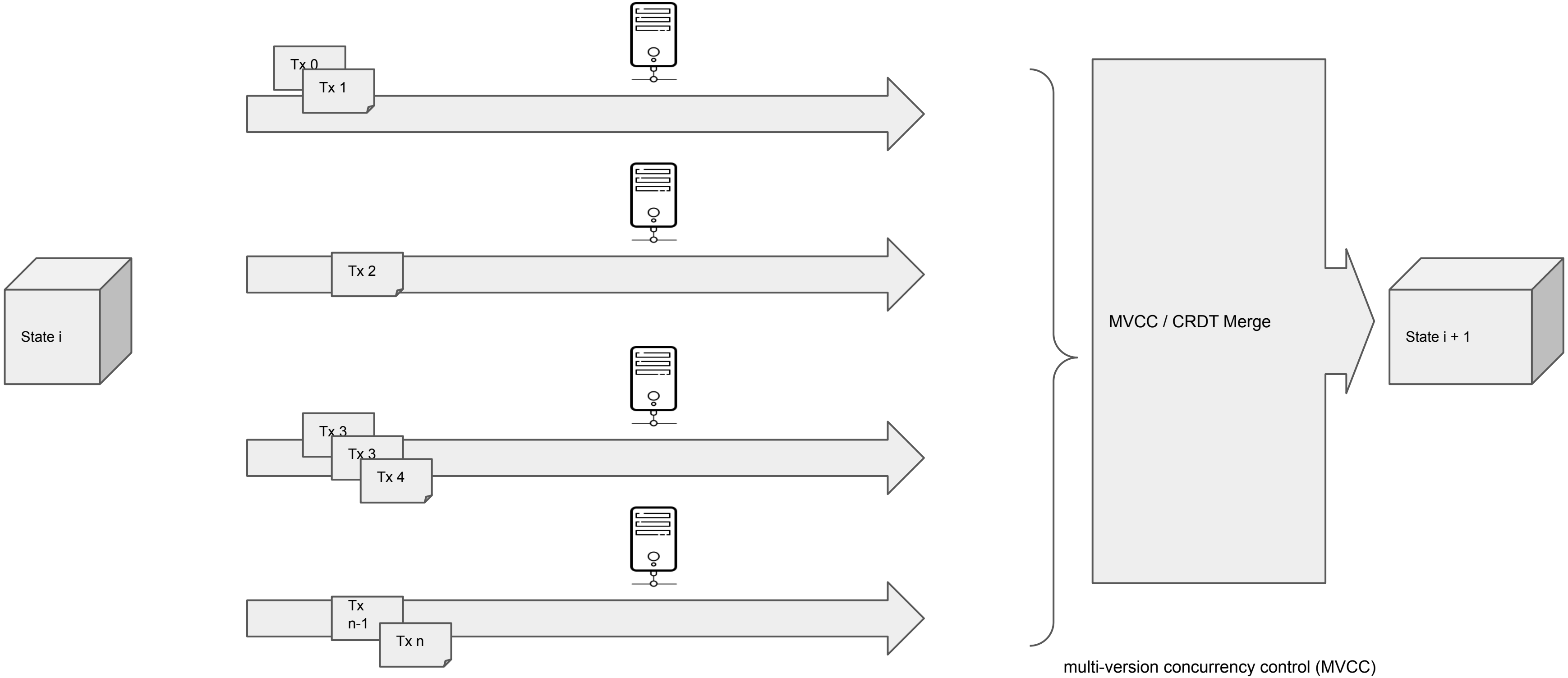
+Decentralization

Existing blockchains...

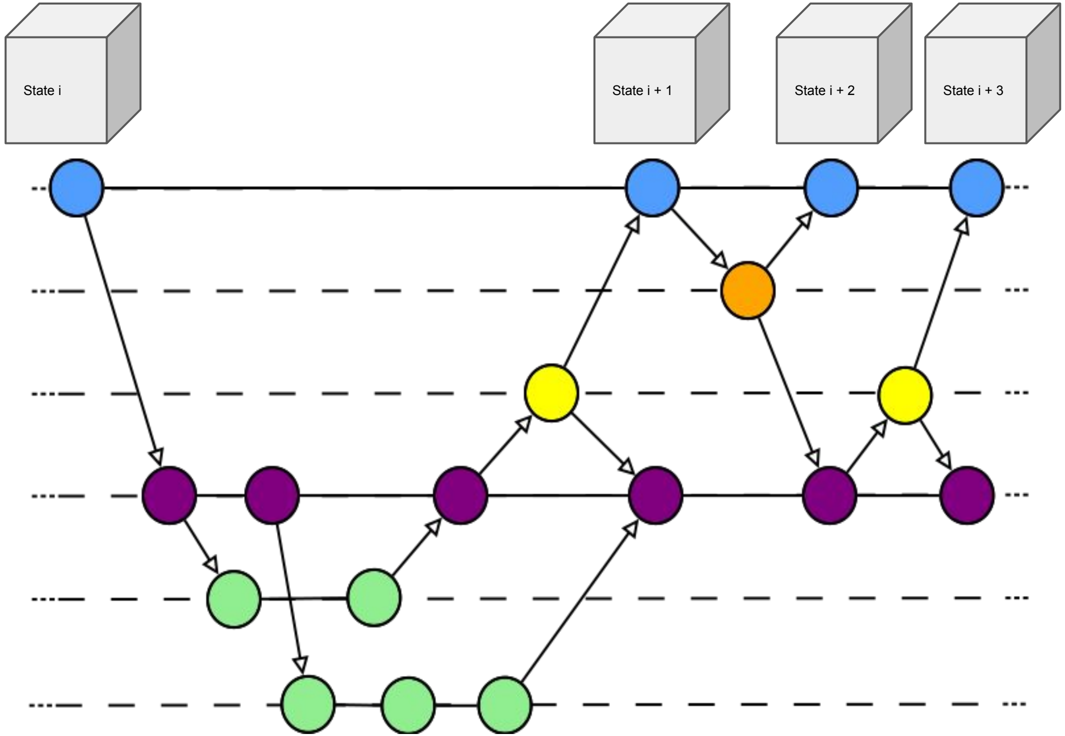


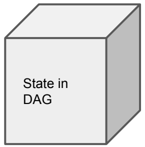
No matter how many nodes are running in a typical blockchain system, due to the existing consensus, it works as if a single slow computer processing all transactions one by one

Can we run blockchain as if it is a distributed cloud?



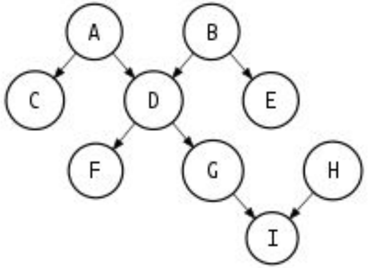
multi-version concurrency control (MVCC)
Conflict-free replica data type (CRDT)





State is actually DAG powered by IPLD

Sync, Diff, Update, Merge made easier



Full decentralized parallel execution, continue integration can make a dApp runs as rich as existing centralized cloud apps

But why did not this happen yet?

Because blockchain needs consensus to fight against trustless environment, while cloud computing requires an assumption of trust …

...What if the trust is taken care by someone else?

2

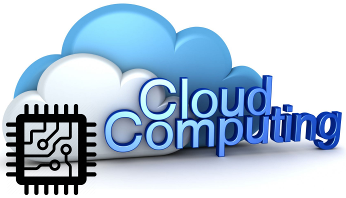
- Execute dApps without trust concern
- Eg. Agents inside a highly secured office can trust each other when work together

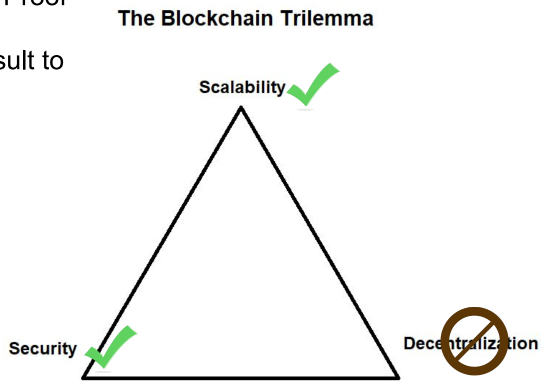
1

- Handle trust concern without running dApps code
- Eg. Security guards know nothing about business domain but check badges at the gate

- Rich Applications run on Decentralized Nodes
- Secure Logic and Data inside Hardware Protected Enclave
- Security Chip as Root of Trust (RoT) Generates Proof of Trust (PoT)
- Verify other nodes’ PoT and send verification result to Layer 1

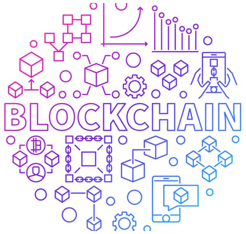
2

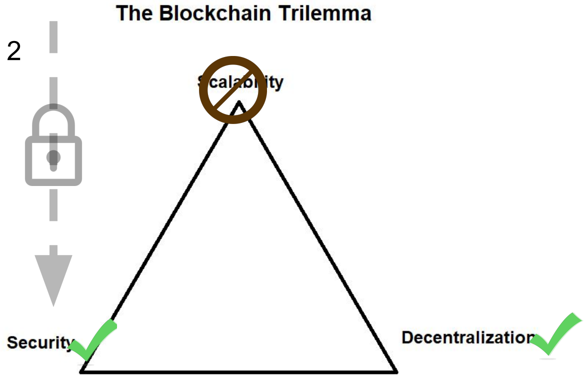




- Immutable Data Storage
- Consensus on the verification result from Layer 2
- Managing Remote Attestation
- Token Economy
- Verify block

1

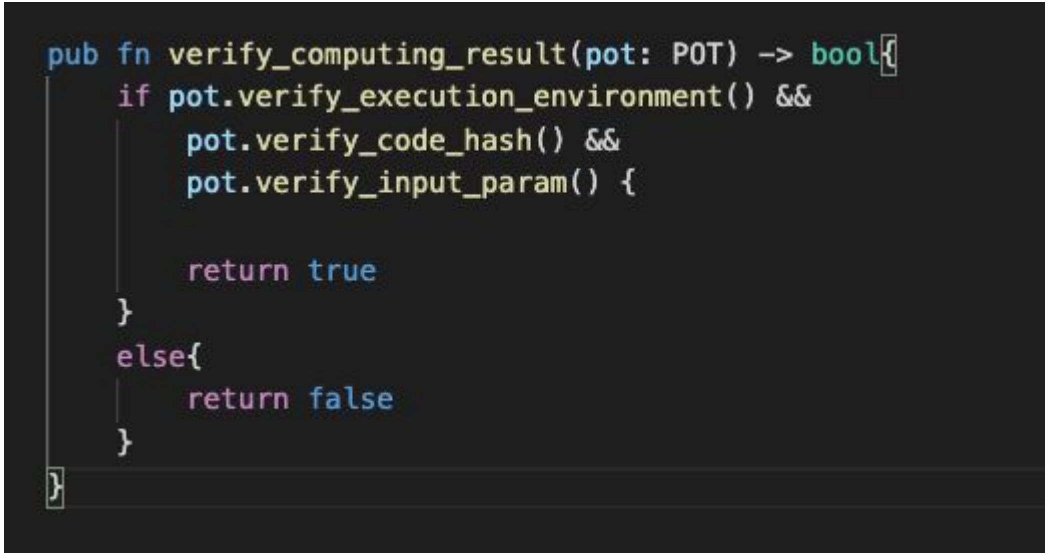




Our basic assumption is that:

If the execution environment, code, and input data are trusted,

then the execution result can be also be trusted.

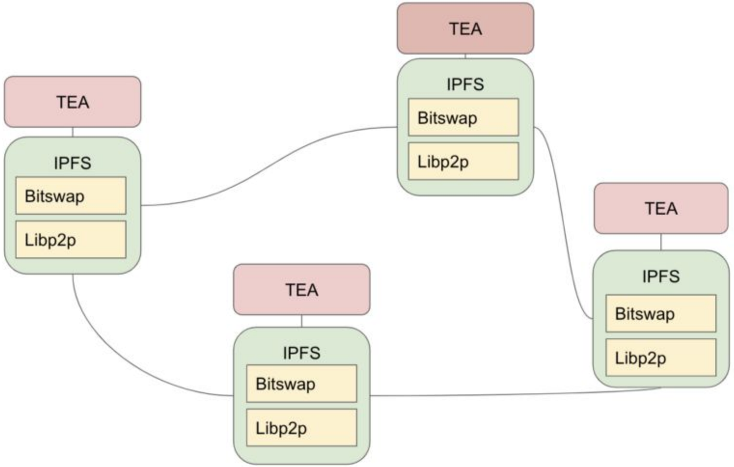


```
pub fn verify_computing_result(pot: POT) -> bool{
    if pot.verify_execution_environment() &&
        pot.verify_code_hash() &&
        pot.verify_input_param() {

        return true
    }
    else{
        return false
    }
}
```

IPFS uses *content addressing* to identify content by what's in it rather than by where it's located

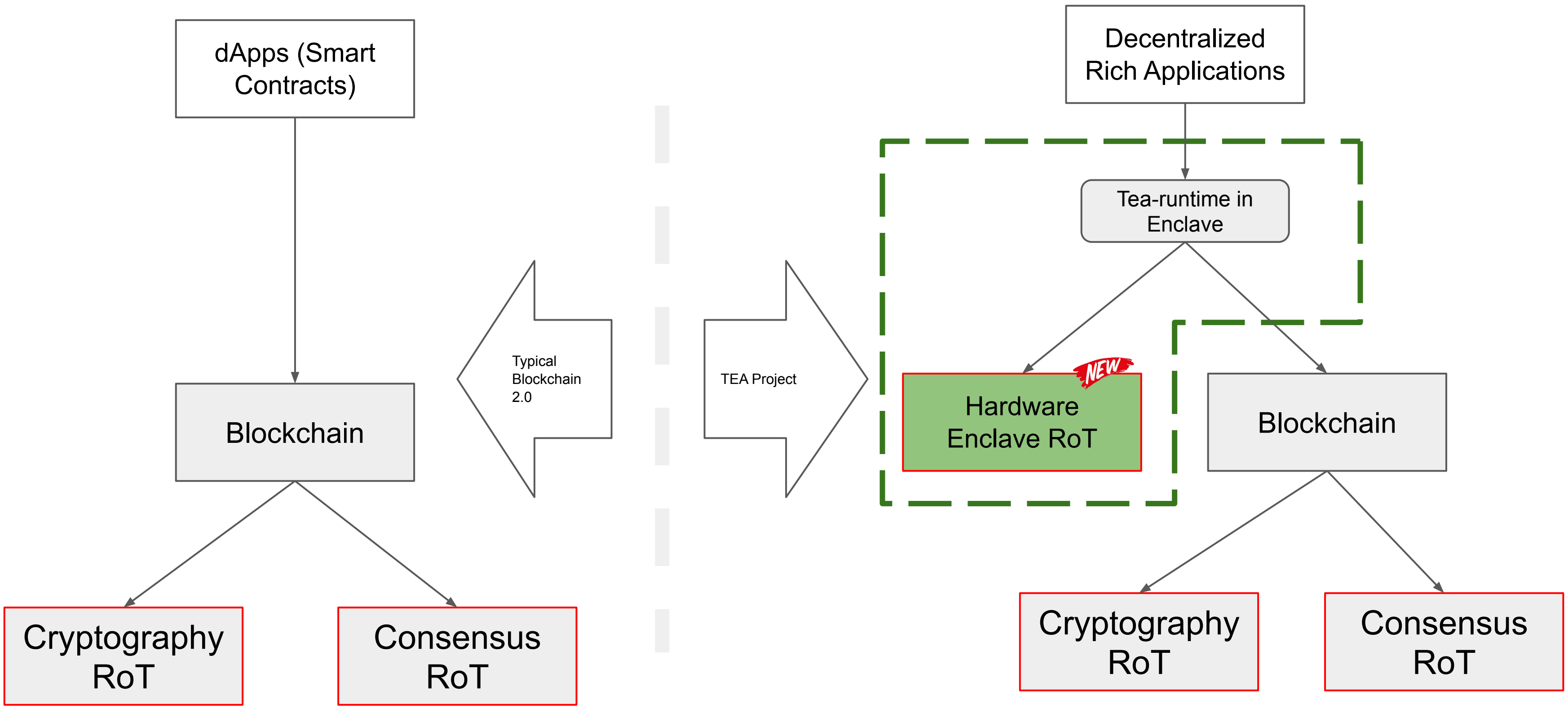
TEA uses Proof of Trust (PoT) to execute code and data by how trust is the execution environment rather than by where (or who) it's executed

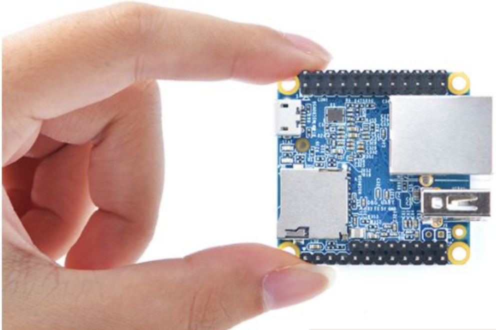


IPFS

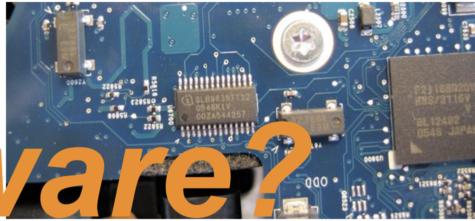
Inter Planetary Function as a Service

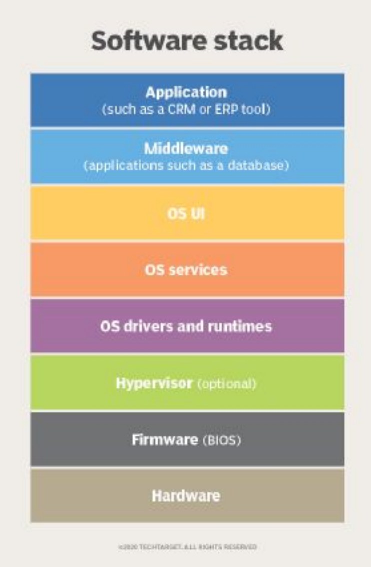
Dependencies of RoT

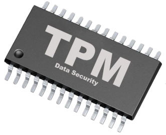




WHY Hardware?




Software stack
Application (such as a CRM or ERP tool)
Middleware (applications such as a database)
OS UI
OS services
OS drivers and runtimes
Hypervisor (optional)
Firmware (BIOS)
Hardware
©2020 TECHTARGET. ALL RIGHTS RESERVED



Cryptography options?
SMPC, FHE, ZKP

Hardware options?
CPU-based TEE, TCG-based TPM

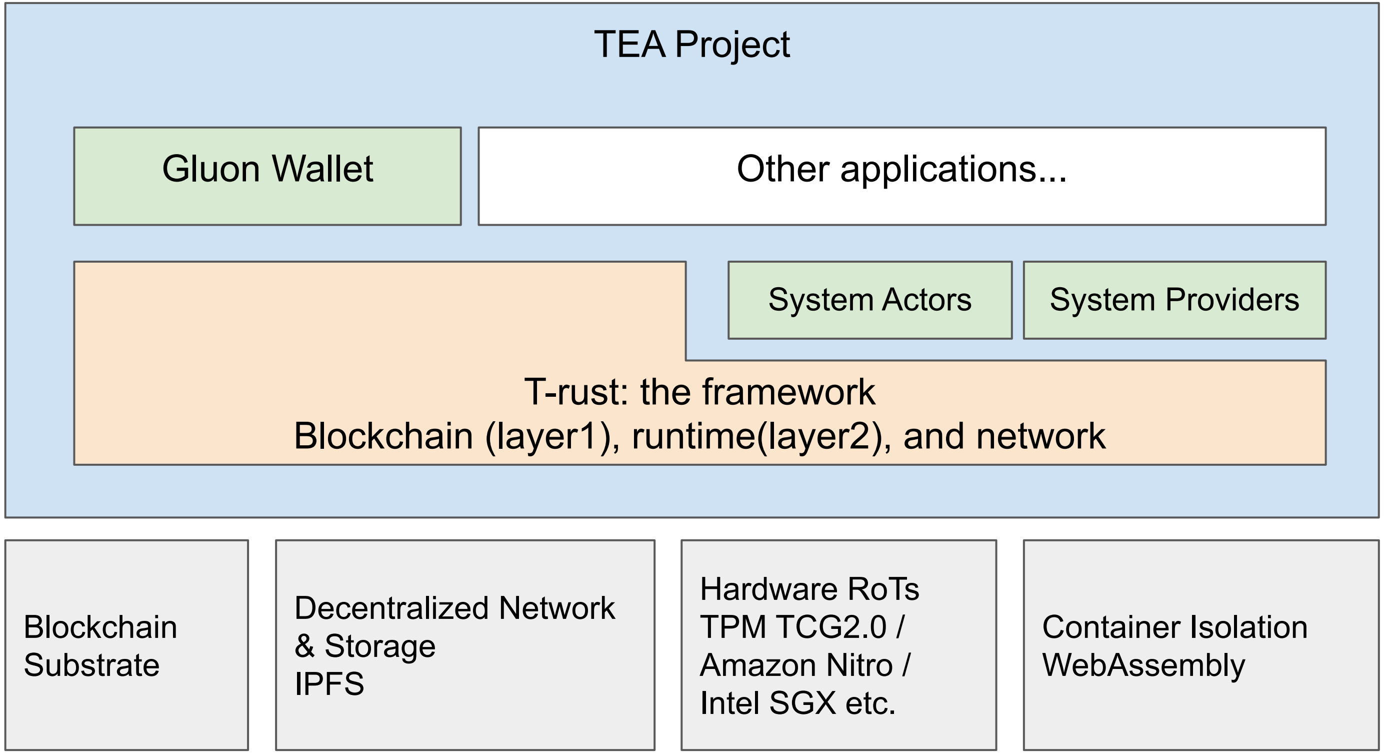
	TEA Support	Technology	RoT Verification	Cloud IaaS 4 Rent?
Google Cloud / MS Azure Confidential Computing	On Roadmap	CPU Based (AMD/Intel)	Centralized Cloud	Y
TEE SGX/SEV/TrustZone	On Roadmap	CPU Based	Centralized by CPU manufacturer	N
Amazon Nitro NEW	In Development	TPM Based(?)	Centralized Cloud	Y
Trusted Computing (TPM)	Software Simulator Completed	TPM Based	Decentralized	N

Increase cost of attacks

- Verifiable Randomness
- Distributed computing
- Zero knowledge
- Hardware protection
- Cost of hardware life cycle
- Diversity technical stack
- Blockchain based penalty and incentive

Decrease profit of successful attacks

- Decentralized storage
- Partial data
- Busy / idle ratio
- Phishing tasks



TEA Project

Gluon Wallet

Other applications...

System Actors

System Providers

T-rust: the framework
Blockchain (layer1), runtime(layer2), and network

Blockchain
Substrate

Decentralized Network
& Storage
IPFS

Hardware RoTs
TPM TCG2.0 /
Amazon Nitro /
Intel SGX etc.

Container Isolation
WebAssembly

Alice: I took a picture of a wild animal. I do not know what it is, but I know it has value to others. I hope to get paid from the usage of my picture but technically have my copyright protected

Bob: I wrote a Tensorflow image recognition model and compile to webassembly as an application. It can determine what is on a picture. I hope to get paid by the usage of my application, but no one can steal my code.

Charlie: I am an IPFS miner. I recently add a TEA module to my mining machine. I am now a IPFS + TEA miner so that I can min both Filecoin and TEA token. How could my clients trust me and send me sensitive information to compute?

Dave: I am a naturalist and scientist, I have research fund on looking for wild animal behavior by analysing pictures. I cannot analyze every pictures manually, I can pay to run the AI algorithm on wild animal pictures from other paid photographers.

Stable coin: TEA

- Utility token. Stable coin pegging to computing cost not fiat.
- Used as Gas, unlimited supply
- No genesis block supply. Every TEA needs to be mined
- Born from public service rewards burnt by DAO when recycle
- Stake to Camellia for revenue sharing

NFT: Camellia

- TEA Node can be activated when a Camellia associate with it
- Camellias are owned by miners
- Miners buy new Camellia seeds through a Birth Control Bidding
- Camellia has its life cycle. DAO generate it with birth control. DAO burns it when die.
- Camellia has technical stack used in diversity control
- Investors can stake to a Camellia for revenue sharing

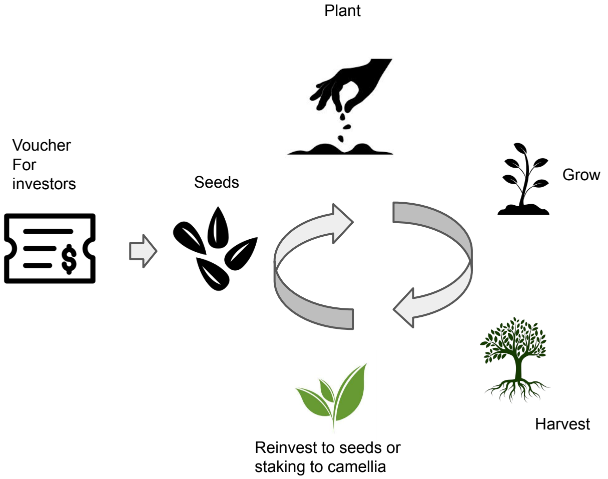
Early Stage Miner Economy: FOMO mining

Prior to the maturity of Web 3 Rich dApps ecosystem, mining economy is necessary to keep economy running and.

TEA's carefully designed token economy creates NFT scarcity during mining. The scarcity encourage miners to reinvest their harvest back instead of selling.

Eg. FIL, Chia…

TEA doesn't require GPU, ASIC or Hard drive. It requires NFT and cheap secure hardware.



TEA can be used

1. Pay for computing services
2. Buy Camellia to be a farmer
3. Stake to others Camellia as an investor, revenue share

TEA born and burn

1. Issued by public service
2. Premining is a special public service
3. DAO burns TEA when bidder pay for Camellia seeds
4. Slash due to illegal behavior

TEA Scarcity

1. Limited public service bonus in every block
2. Shareholders premining TEA need to be locked and release gradually
3. Staking lock
4. Burn and slash

Camellia can be used

1. Plant to mining machine. It is soul
2. Credit loan. Future income as collateral
3. Investment (seed->grow->revenue) Nursery?

Camellia seed to RIP

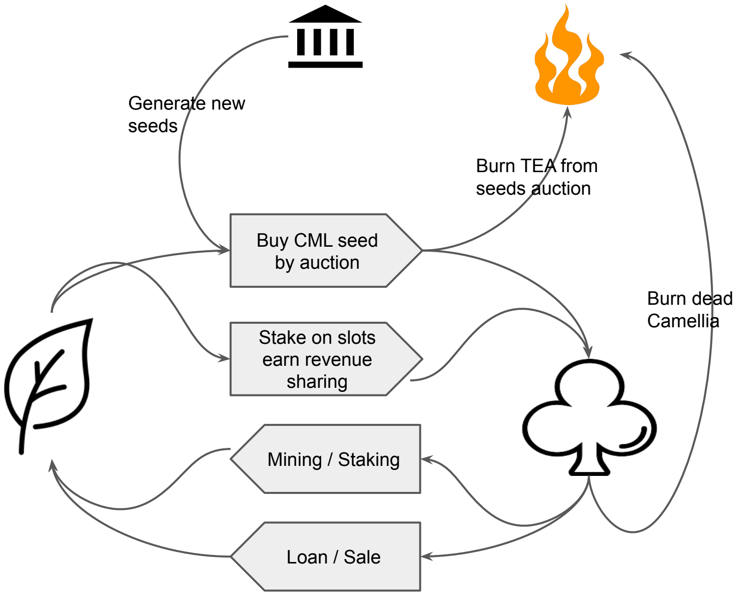
1. DAO generate seeds based on auction price. Maintain a reasonable scarcity
2. Limited lifetime create additional scarcity
3. Grow and aging vary profitability
4. Diversify and future evolution

Premining

1. Fast initial TEA distribution to early investors
2. Special Camellia during this period
3. Early investors agree on that small portion of premined TEA in free mode. Others either staking or buying seeds only

Resilience to Pump & Dump

1. Control the size of private sales
2. IDO at DEX rather than CEX
3. Reserve fund to market making
4. Maintain the scarcity of free TEA by locking staking, seed
5. Fast development to the real use case to consume TEA token



Generate new seeds

Burn TEA from seeds auction

Buy CML seed by auction

Stake on slots earn revenue sharing

Burn dead Camellia

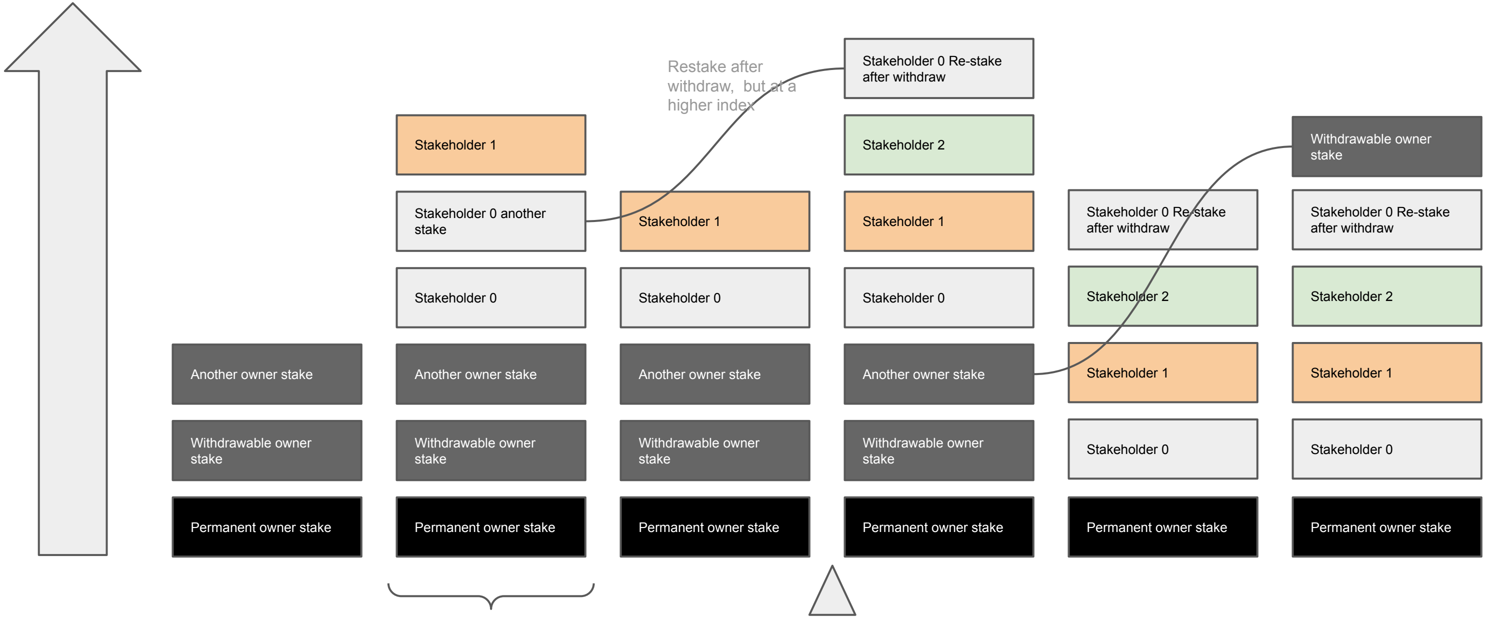
Mining / Staking

Loan / Sale

TEA Staking and Camellia Life Cycle

Revenue_share = this_slot_weight X total_revenue / total_slots_weight
this_slot_weight = (this_slot_index + 1).sqrt() - this_slot_index.sqrt()

Slot_index
Weight reduced



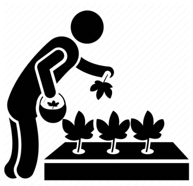
Restake after withdraw, but at a higher index

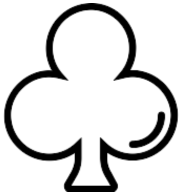
			Stakeholder 0 Re-stake after withdraw		
	Stakeholder 1		Stakeholder 2		Withdrawable owner stake
	Stakeholder 0 another stake	Stakeholder 1	Stakeholder 1	Stakeholder 0 Re-stake after withdraw	Stakeholder 0 Re-stake after withdraw
	Stakeholder 0	Stakeholder 0	Stakeholder 0	Stakeholder 2	Stakeholder 2
Another owner stake	Another owner stake	Another owner stake	Another owner stake	Stakeholder 1	Stakeholder 1
Withdrawable owner stake	Withdrawable owner stake	Withdrawable owner stake	Withdrawable owner stake	Stakeholder 0	Stakeholder 0
Permanent owner stake	Permanent owner stake	Permanent owner stake	Permanent owner stake	Permanent owner stake	Permanent owner stake

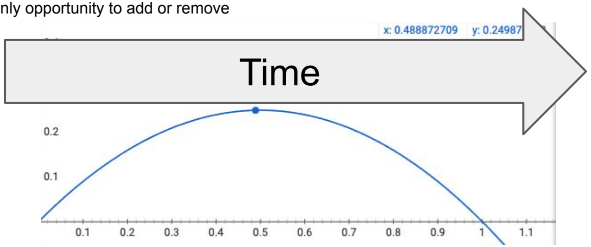
Withdraw Permanent owner stake when die

Every slot is 1000 blocks long

Adjust_time_window every 1000 blocks
This is the only opportunity to add or remove stakes







Core team and milestones

Kevin G. Zhang

- Founder of ELK Insight LLC
- CTO of iHealth Labs US
- https://www.linkedin.com/in/kevingzhang/

Zhijun (William) Zhang

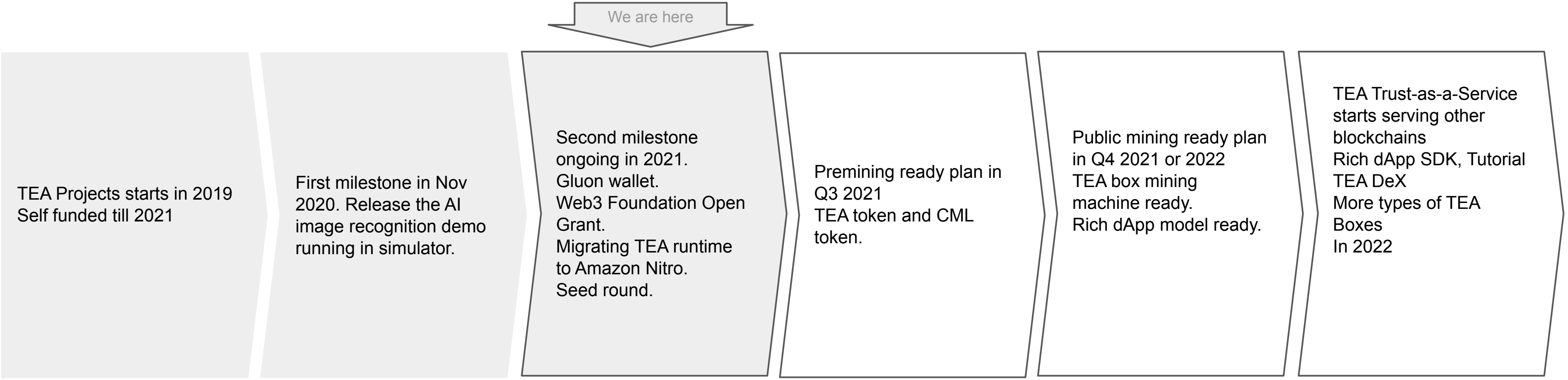
- Lead, Security Architecture at The World Bank Group
- Enterprise Security Architect at The Vanguard Group
- https://www.linkedin.com/in/zhijun/

Yang Li

- Software Engineer of iHealth Labs Singapore
- System Architect of HP Shenzhen
- https://www.linkedin.com/in/jacky-li-4039747b/

Mingzhi Yan

- Software Architect of Cloudwalk Beijing
- Lead blockchain developer of Elastos Beijing
- https://www.linkedin.com/in/mingzhi-yan-7544b9203/



We are here

TEA Projects starts in 2019
Self funded till 2021

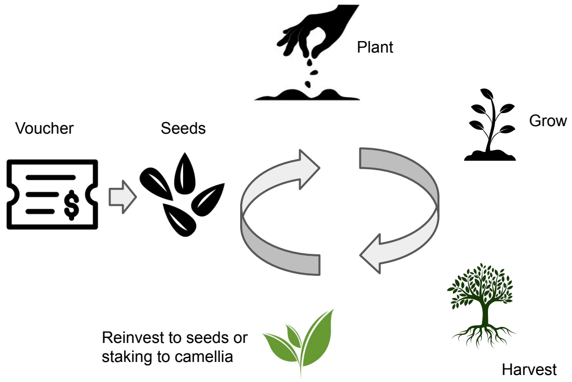
First milestone in Nov 2020. Release the AI image recognition demo running in simulator.

Second milestone ongoing in 2021.
Gluon wallet.
Web3 Foundation Open Grant.
Migrating TEA runtime to Amazon Nitro.
Seed round.

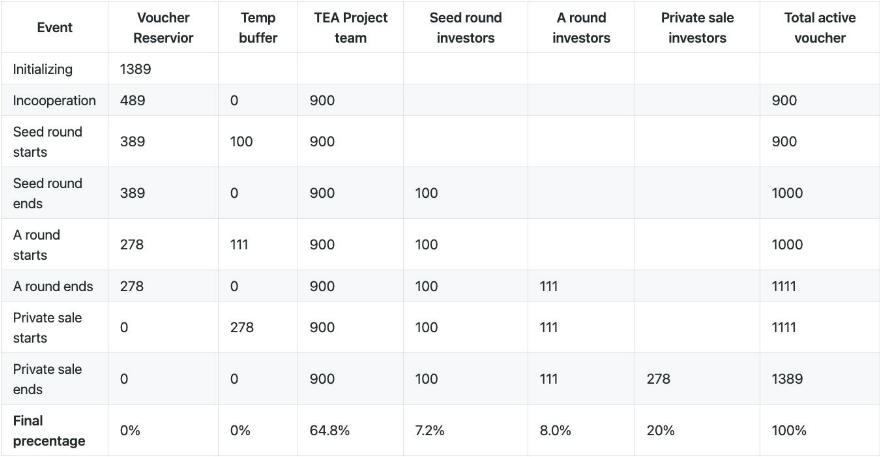
Premining ready plan in Q3 2021
TEA token and CML token.

Public mining ready plan in Q4 2021 or 2022
TEA box mining machine ready.
Rich dApp model ready.

TEA Trust-as-a-Service starts serving other blockchains
Rich dApp SDK, Tutorial
TEA DeX
More types of TEA Boxes
In 2022



Funding rounds and voucher release schedule



Event	Voucher Reservior	Temp buffer	TEA Project team	Seed round investors	A round investors	Private sale investors	Total active voucher
Initializing	1389						
Incooperation	489	0	900				900
Seed round starts	389	100	900				900
Seed round ends	389	0	900	100			1000
A round starts	278	111	900	100			1000
A round ends	278	0	900	100	111		1111
Private sale starts	0	278	900	100	111		1111
Private sale ends	0	0	900	100	111	278	1389
Final precentage	0%	0%	64.8%	7.2%	8.0%	20%	100%

Funding rounds with voucher

Investors receive temporary seeds voucher (ERC 20 like token on TEA blockchain).

Before pre-mining starts, voucher convert to random generated frozen camellia seeds (the NFT).

During pre-mining, defrosted camellia seeds are planted to TEA Nodes, and start harvest TEA tokens at an accelerated speed for short period of time.

When public mining starts, everyone can join the mining follow the same procedure without voucher.

Seeds biding->Plant->Grow->Harvest->Reinvest/Sell

Voucher conversion to frozen seeds

Voucher is fungible token, Camellia seed is non-fungible token. When converting a voucher to a seed, a random algorithm in the blockchain code creates unique properties that makes every seeds unique (NFT needs to be unique). The uniqueness controls the defrost timing, life span and other key factors. For each individual seed, the value vary but in overall the distribution is fair to everyone.

Defrost schedule for frozen seeds

Defrost 10% when premining, then 5% every month for next 18 months. Bio clock embedded into NFT of every seeds in a random manner.