

# **zero gravity**

 OSDI23.pdf 526 kB

 atc20-li-chenxing.pdf 789 kB

 Ionian - whitepaper.pdf 468 kB



**ZERO GRAVITY**

Data Availability 2.0

**1000x Better Data Optimized for Onchain AI**



## Meet the team building



**Michael | CEO**  
[Bio](#)



**Ming | CTO**  
[Bio](#)



**Fan | CSSO**  
[Bio](#)



**Thomas | CBO**  
[Bio](#)



### Exceptional tech team

Proven experience designing and launching Layer 1 (\$B Conflux) and distributed storage systems (Microsoft Research)

10 gold medals in Informatics Olympiads, top academic backgrounds with 10 CS PhDs, one of world's top cryptographers

Award-winning papers in peer-reviewed tech conferences\* rapidly implemented into production

### Well-balanced business team

Track record of fundraising \$100Ms and driving \$100Ms of revenue with unicorn company garten (top YCombinator, top entrepreneur, and YPO awards)

Recruited 100s of people from Academia, Apple, Amazon, Microsoft, Bridgewater, etc. that drove industry leading metrics 80 NPS, 9/10 CSAT



\* Example: IEEE ICBC 2022 best paper award for "LMPTs: Eliminating Storage Bottlenecks for Processing Blockchain Transactions"

## Problem: Web3 Data is 1000x slower & more expensive on Ethereum

### The World Today

- Eth L1 cost of 1kb = ~\$51 (\$51M for a GB!), on L2 it's ~\$5, Sol \$0.008, ours \$0.0005
- 70% of L2 gas costs spent on storing tx data on Ethereum
- Not possible to do onchain AI and save data / dApp state via smart contracts
- Danksharding, Celestia, and others don't solve this

### End Customer Impact

- High performance Dapps can't be built onchain without storing enough data
- For performance Amazon built DynamoDB, Facebook MyRocksDB, Google Spanner, BigTable

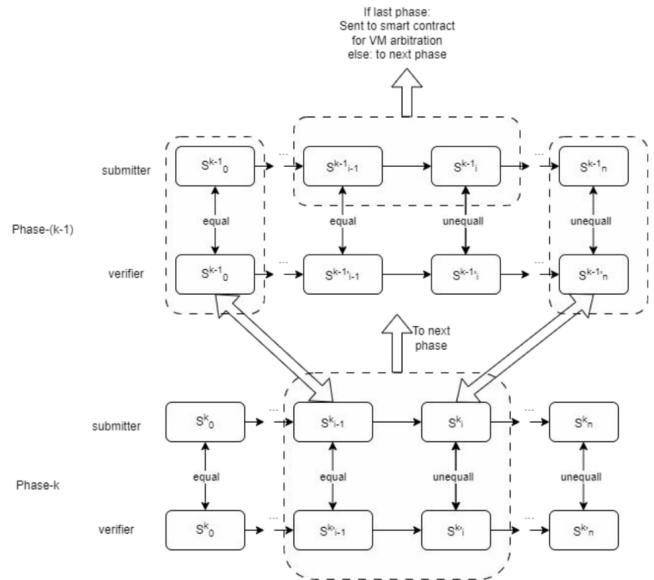
### Why is this important

- Validators won't keep up. Scaling to 10M tx/s across N rollups requires 5GB/s or 3PB/week!
- Web3 L2 starting to be on par, bringing DA on par unlocks use-cases such as onchain AI

Note: Sol and our pricing is for storing data for one month | Source: [StackExchange](#), [L2Notes](#), [Reddit](#)

---

## Use-Case: You can't deploy OPML / ZKML



OPML like [Hyperoracle](#) can run large AI models (AI model inference and training/fine-tuning on the blockchain), which have large inference traces

LLMs + metadata need to be stored (10s of GB) and inference traces and input data need to be stored.  
E.g, with 1M concurrent users, each with a inference request per minute:  
 $1M / \text{minute} \times 100 \text{ KB}^* = 1.6\text{GB/s}$  or  $144\text{TB / day!}$

=> **Requires highly performant DA and current systems won't scale**

\*assumed blended average between text, pictures, voice, documents

## Use-Case: You can't scale onchain MMO games like Poker



These dApps will require 1000x scale to save state via smart contract: e.g., player stats (money, hand, bets), game stats (difficulty / AI, epoch, flop)

E.g., 100k concurrent users and 3% saving and loading state = .5 KB/s (8 Merkle tree commits) x 3K tx/s = 120 GB/day = 3.6 TB/month

=> Need to offload state that can be customized, is high performance, scalable, and programmable

## + Many other use cases

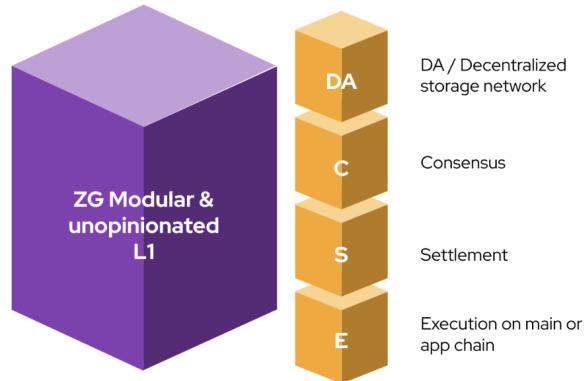
High Frequency DeFi & Onchain order books  
Data marketplaces  
Decentralized ID  
RWA (metadata like house & car titles)  
Past Insurance records  
Collaborative documents (gDocs, Notion)  
& more

## Solution: Others are simply providing data publishing. We are providing hyperscale programmable DA optimized for AI.

EigenDa, Celestia, Avail don't fully solve Ethereum's transaction storage and state explosion problem

### Our Approach unlocks 1000x DA for use cases like onchain AI

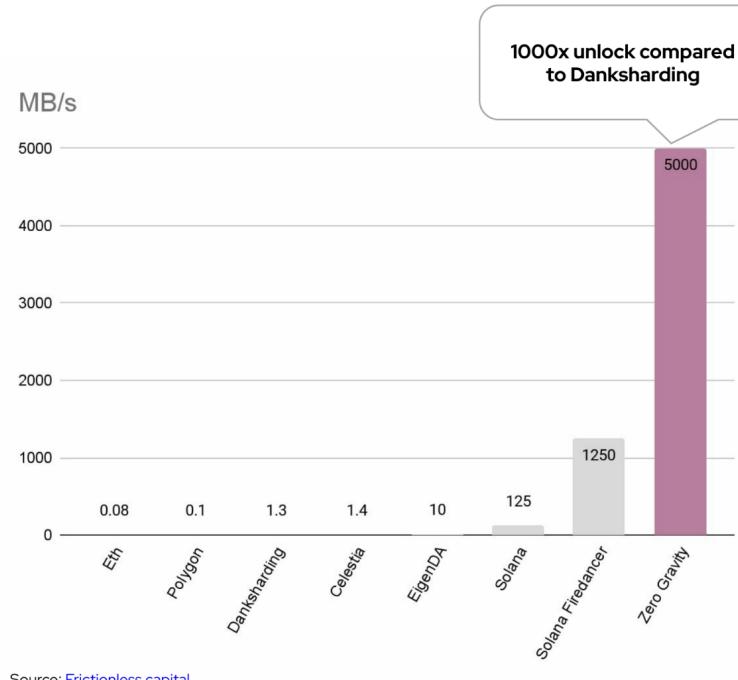
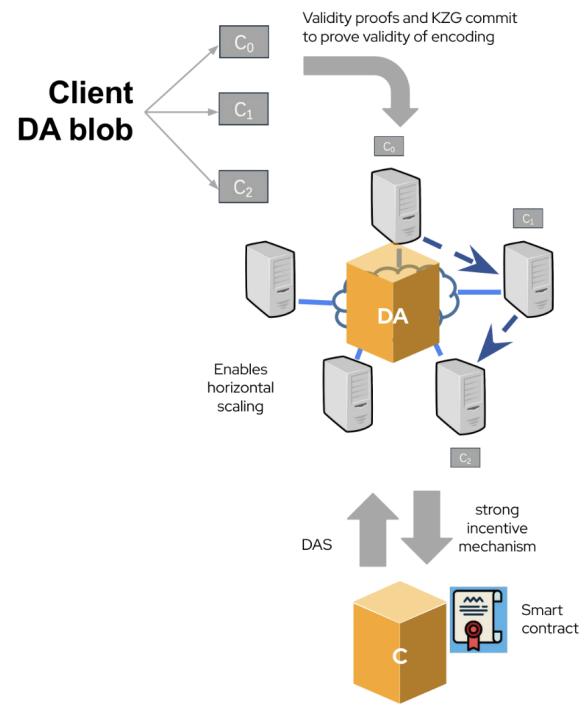
1. Highly performant, modular and unopinionated L1
2. Customizable decentralized data: All storage types found in Web2, data persistency, replication, location, etc.
3. Programmable DA: saving and loading state via smart contract
4. Extensibility: Utilize our infra to later build zk prover aggregation or decentralized shared sequencers



**We have a technical moat via our academic research**

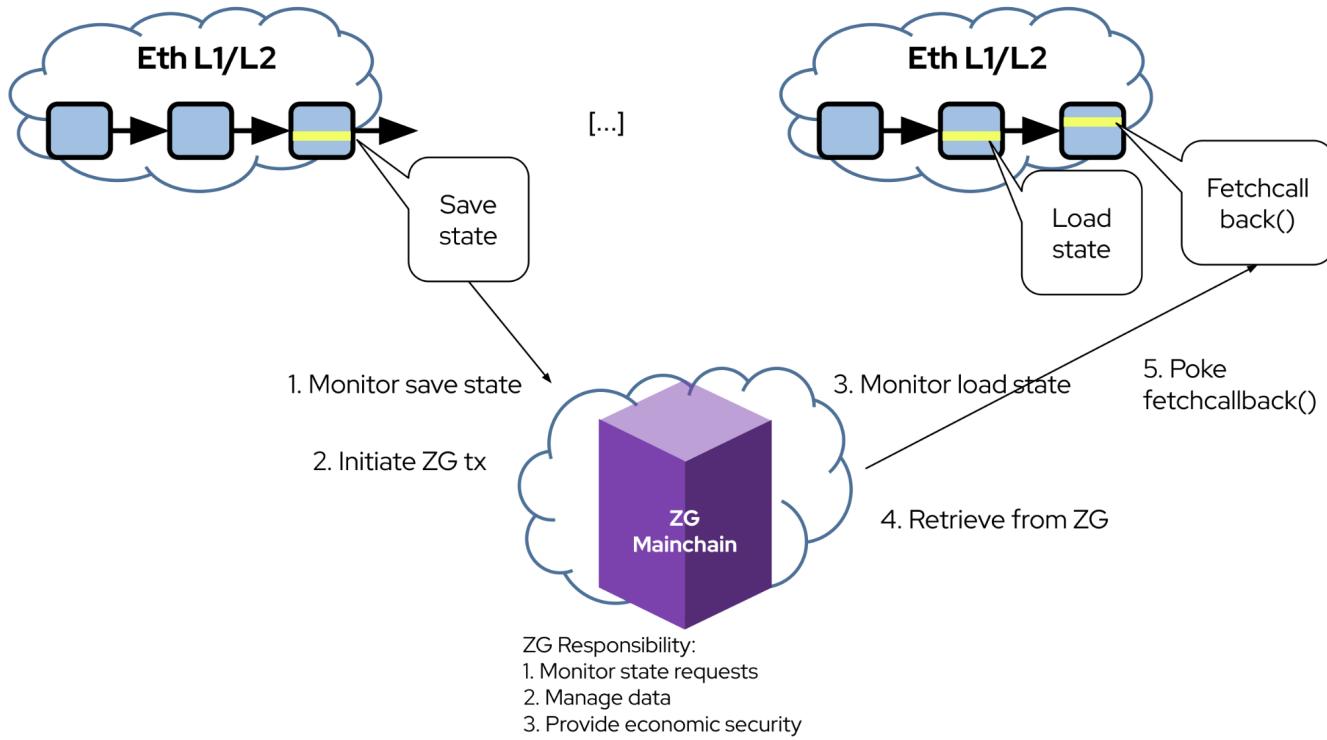
We have an edge on L1 performance, AI, distributed storage, security, and more  
Papers won various awards like IEEE ICBC 2022 best paper

## The fastest and lowest cost data propagation for AI and more

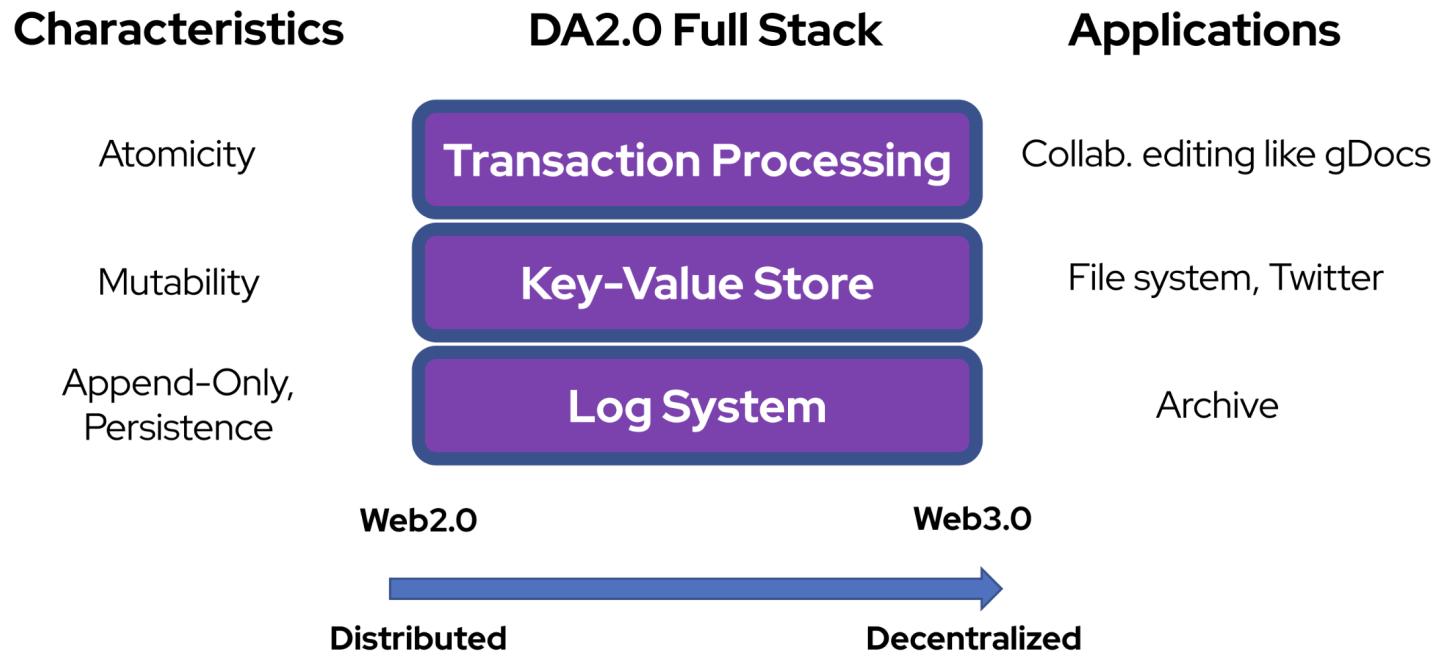


Source: [Frictionless capital](#)

## The first fully customizable & programmable DA, DA 2.0



## Ability to port more Web2 applications to Web3



## DA 2.0 Summary: The most performant and AI optimized DA

	Celestia	EigenDA	Avail	ZG
Performance	Optimized for onchain AI	○	○	○
Security	High DPR (high node bandwidth, low latency)	○	○	○
Low cost (storage, retrieval)	○	●	○	○
Validator & Proposer separation (censorship resist.)	○	●	○	○
Decentralization	●	●	●	○
Economic security (slashing)	●	●	●	○
Programmability	Storage location & replication	○	○	○
Data persistence	○	○	○	○
Smart contract state management	○	○	○	○

# WEB3 IS A \$100T+ MC OPPORTUNITY

with ZG building a foundational building block



=



\$10B+ in annual revenue

\$1T Today

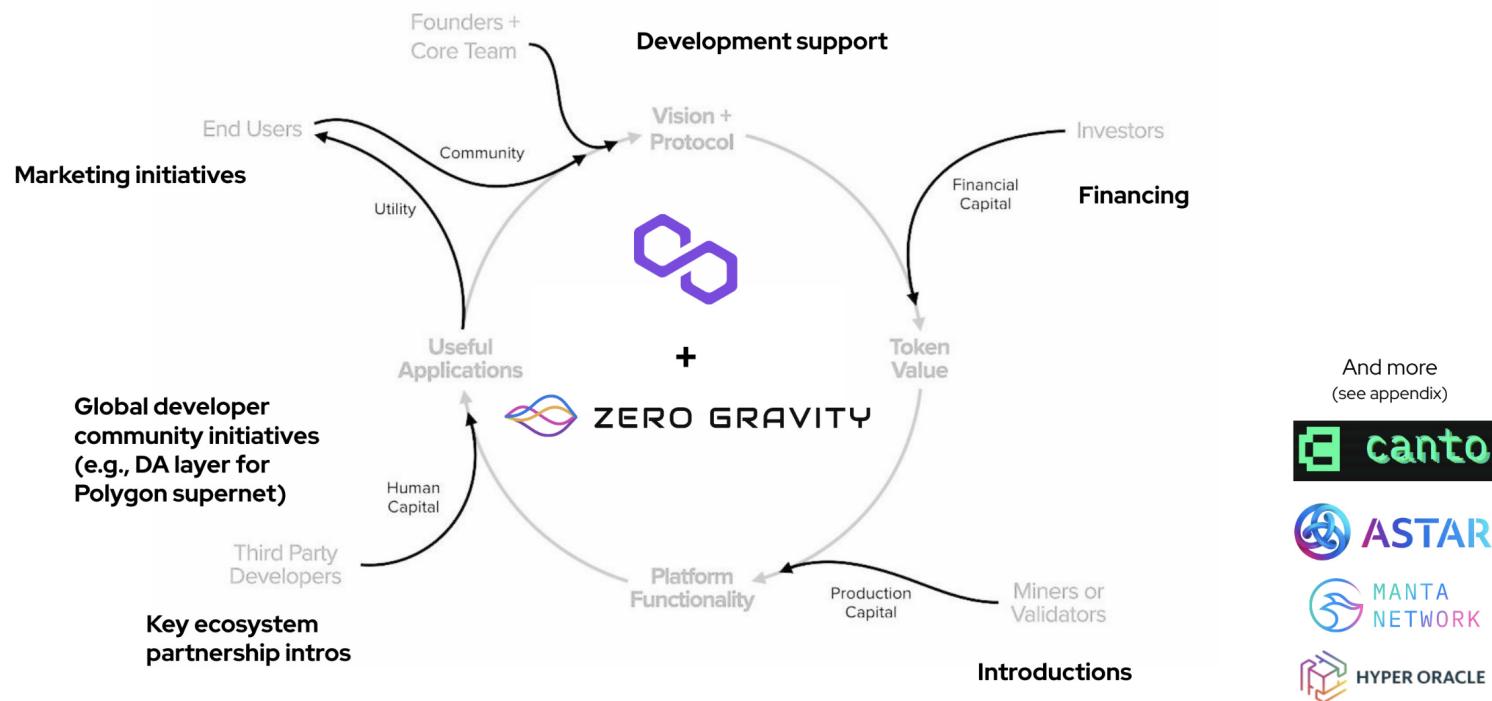
BTC

Eth & rest

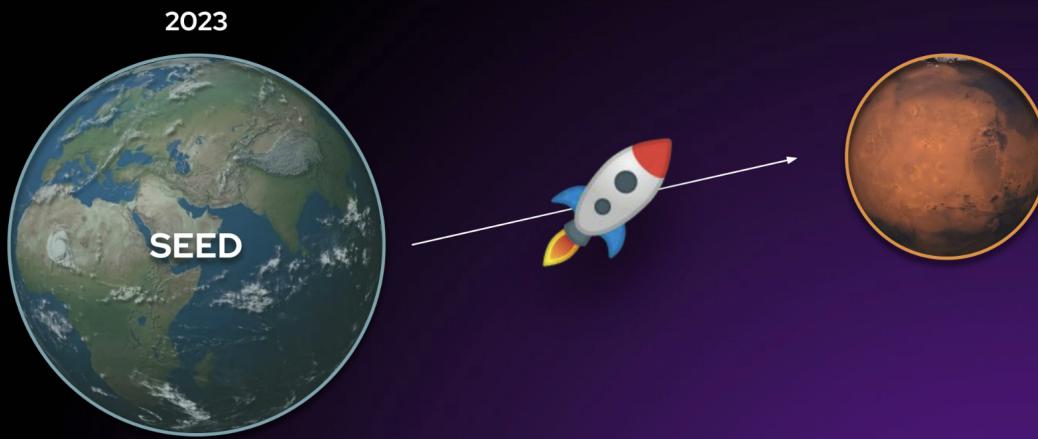
Customers: Rollups & L1s  
Biz model: 10% transaction fees, \$0.0005 / kb / mth  
= \$8B rev. at full capacity w/o Web3 growth factor  
+ Token (security, governance, fees for storage, transactions)

Note: It's potentially a bigger opportunity if all forms of money including loans, derivatives, real estate become tokenized. WEF estimates \$867T are ripe for disruption by Web3. At 60% CAGR \$100T in market cap reached in 10 years.  
Source: [Marketwatch](#), [McKinsey](#), [World Economic Forum](#), [Silicon Angle](#)

## First ecosystem integration with Polygon to bootstrap network effects



## Funding to escape gravity & build



70% **Engineering & Product**

20% **SG&A**

10% **Misc**

**Our vision is to develop the most modular and free infra**





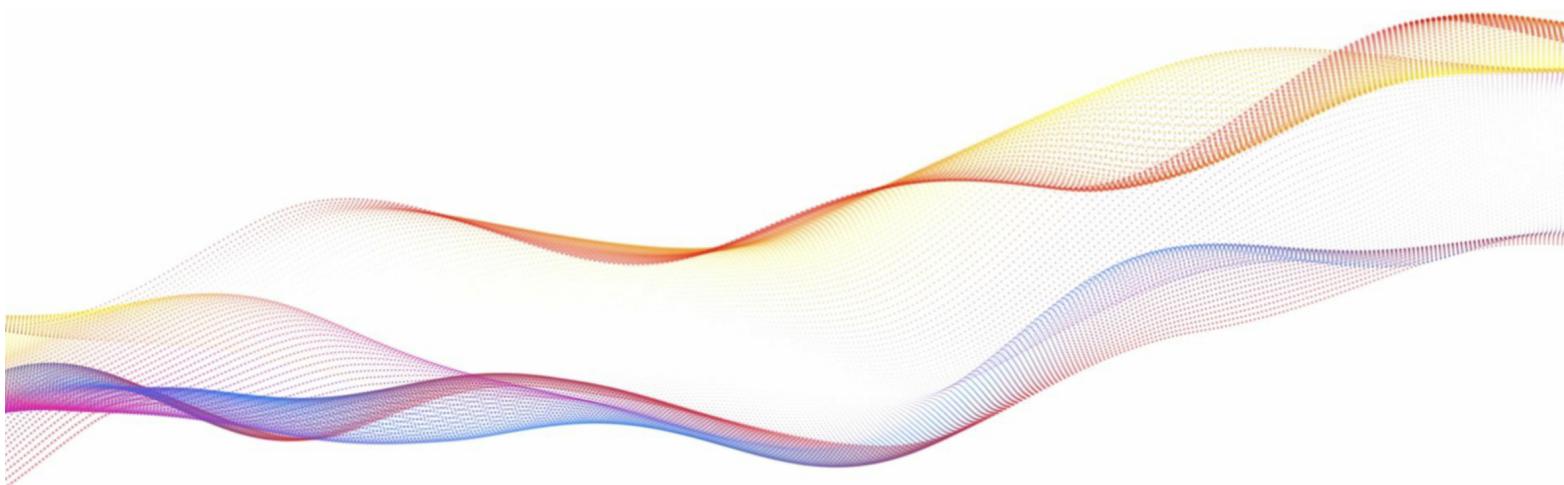
Join us to make AI a public  
good and Web3 friction free

**Key partnerships | Private Testnet | Strong team**



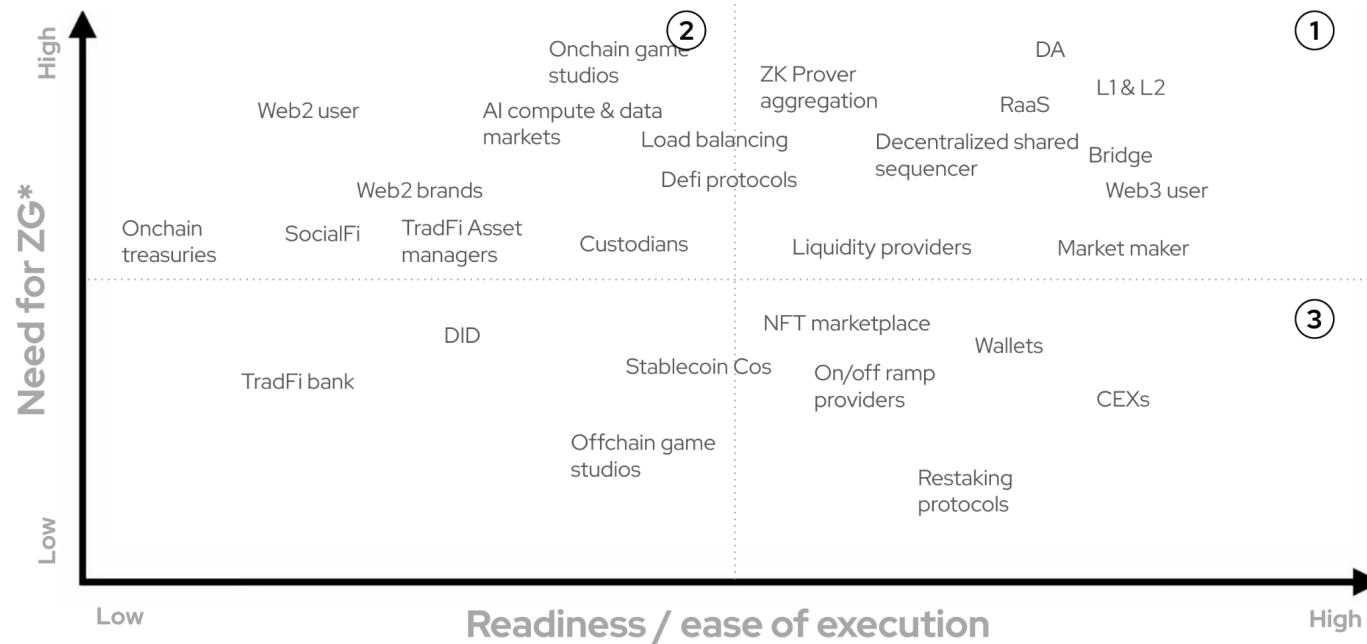


## Appendix



## GTM Details

## We'll build our ecosystem sequentially



\*ZG excels when there are high performance, security, and data needs

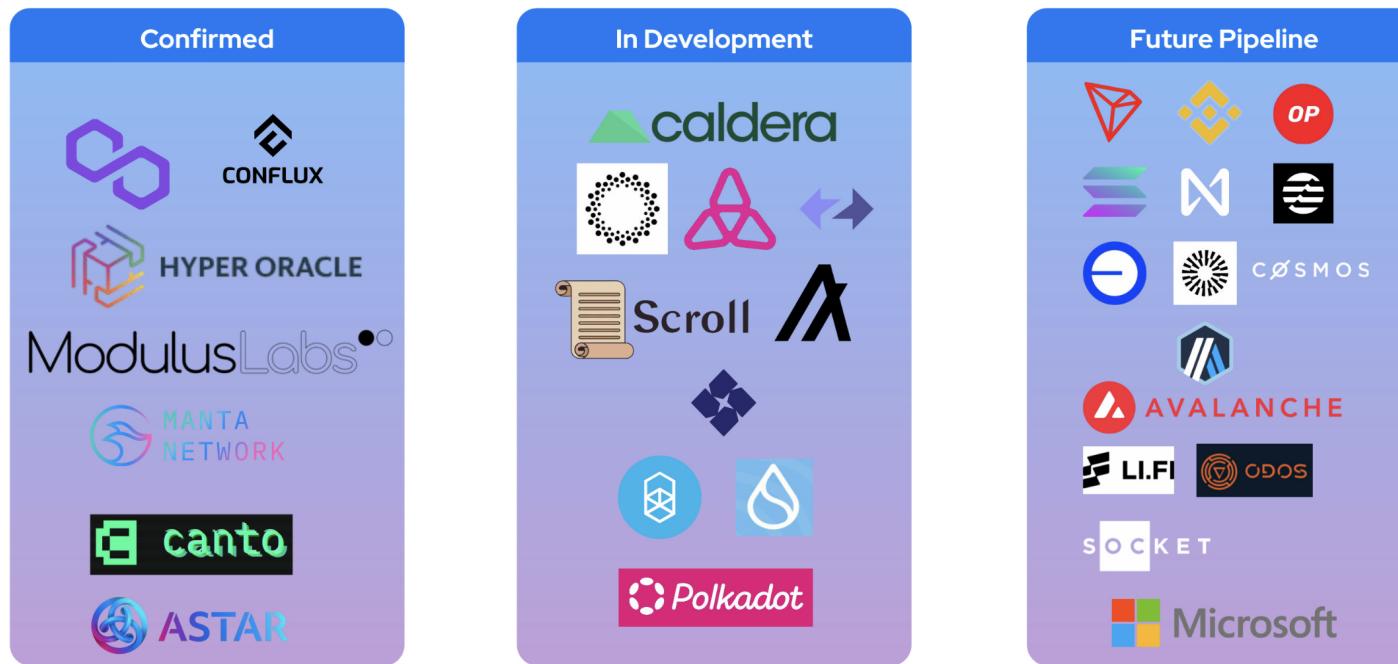
## GTM strategy focus on L2, L1, Rollups



## Detailed GTM - three phases

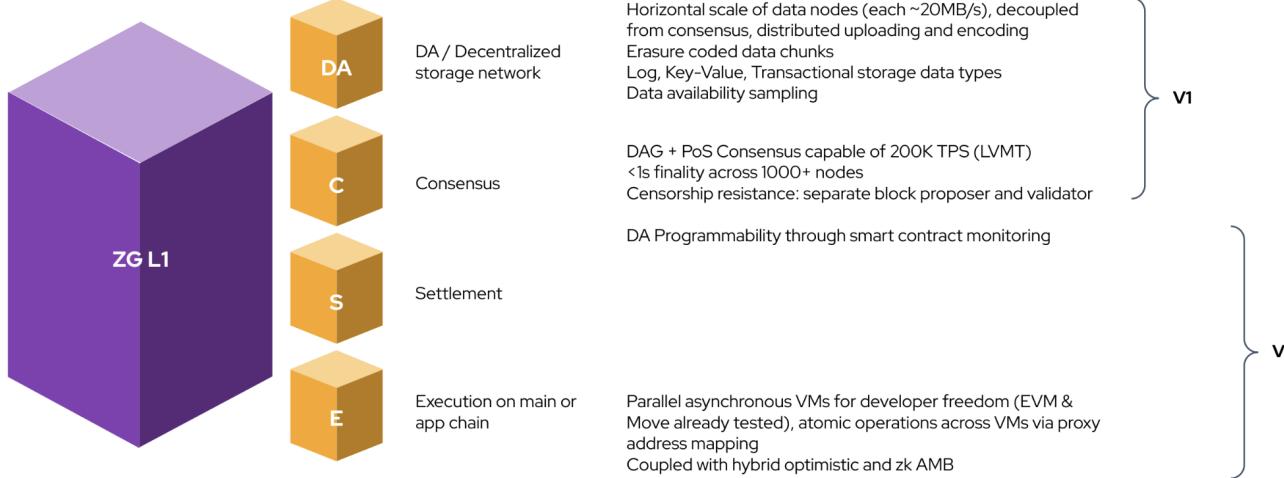
	Phase 1 (2023 - 2024)					Phase 2 (2025)		Phase 3 (2026+)	
Infra	Storage & L1 private testnet	AMB	DA	ZK prover aggregator or sequencer	Dec. shared sequencer	Public Testnet	Public Mainnet with subspaces (ZKEVM, Move...)	DID + KYC / AML / CFT	
DApps	AI	Onchain games		HiFi Trading + order books Data market place		Custodian Solutions Credit	NFT	Equity & Bonds RWA	
BD	L1s & L2s (Polygon, Eth, Sui, Fantom, Arb...)	Wallets & Bridge aggregators (Li.fi, Metamask...)		Node ops (Infura, Ankr, Alchemy, Google..)		Dex (Uni, Pancake, Sushi, GMX, SNX, Gains, Aave...)	Hedge funds: Point72, Citadel, DE Shaw, Millennium, Blackrock, 2sigma, Tiger, Renaissance, Bridgewater, etc.		
Community	Build social channels (Zealy quests)	Hackathons	Tokenomics & airdrop design (more assets transferred more gains)		Series A PR + Token Launch CEX acquisition		Airdrops MM campaigns	NFT: Opensea, MagicEden, Rarible, Top Shot... IBD: Goldman, JPM, MS, BAML, Citi, UBS, etc.	
Community	Utilize Conflux networks & PR	Speaking & Side events (Consensus, Devcon, Eth, EthCC, 2049...)						Ecosystem fund	

## Partnership Pipeline



## Layer 1 & Decentralized Data Storage Details

## High level architecture for 1000x unlock (each module can be used on its own or combined)



Current state: Alpha private testnet for decentralized storage & consensus

Remaining work for public testnet usable by any L1, L2, or Rollup: DA V1 functionality + finalizing consensus

Why L1?

- For costs: EigenDa Can't keep up with performance and cost
- Extensibility: More infrastructure like decentralized shared sequencers
- More possibilities for Innovation: LVMT vs Merkle Tree
- Incentive mechanism: Storage network requires consensus protocol for better incentives + Eigenda not well designed => nodes not participate in staking mechanism (slashing all based on DDOS)

## Why Onchain AI?

By adding ML capabilities, smart contracts can be made more autonomous and dynamic, allowing them to make decisions based on real-time on-chain data, rather than just static rules. Smart contracts would be flexible and adaptable to various scenarios, including those that may not have been anticipated when the contract was initially created. In short, ML capabilities will broaden the automation, accuracy, efficiency, and flexibility of any smart contract we put on-chain. [...] In many ways, it is surprising that smart contracts do not use embedded ML models today given the prominence of ML in a majority of applications outside of web3. This absence of ML was largely due to the high computational cost of running these models on-chain."

<https://medium.com/lkxnetwork/zkml-evolving-the-intelligence-of-smart-contracts-through-zero-knowledge-cryptography-e6725412bbd1>

We address both data storage and data availability. DA is concerned with whether the data published in the latest block is available. This is distinctly different from data storage, which is concerned with storing data securely and providing guarantees that it can be accessed when needed.

## EigenDA

- **Performance bottlenecks**
  - Running full Eth node + AVS
  - Interact with the restaking smart contract on ethereum
- Incentive mechanism **discourages participation of nodes and creates inconsistent guarantees of Data Publishing and Availability**
  - The distributions of chunks are not proportional with staking distribution resulting in unfair slashing
  - Brutal slashing for failing to serve a data request, e.g., DDOS attack results in large slashing even though 99% of other commitments were met
  - Our separation of data maintenance and data publishing consensus enable us to decide who should store what based on incentives. Thus, our design allows more performant nodes to supply more bandwidth and chunks
- Economics of running validator are **not profitable**
- Not well-designed storage layer so **cannot support scenarios like OPML** where the requirements of scale and storage costs are enormous
- Data maintenance **relies on majority secure assumption making it inefficient for small data sizes**
  - Requires data being split into as many chunks as committee nodes
  - Our design separates DA consensus and data maintenance. Clients can flexibly configure the number of chunks according to the data size and maximize efficiency
- **Too much complexity**
  - Doing many things at same time, slower execution and additional risk, need to wait until Eigenlayer is done (like incentive design etc.)
  - Need to onboard AVS, uphill battle, there is misalignment on restakers (3 sided marketplace - restakers, node operators, AVS)
  - Vitalik "Don't overload Ethereum's consensus" > hurt eigenda if eigenlayer implodes (very risky, Terra projects great like Astroport but crashed)

## Celestia

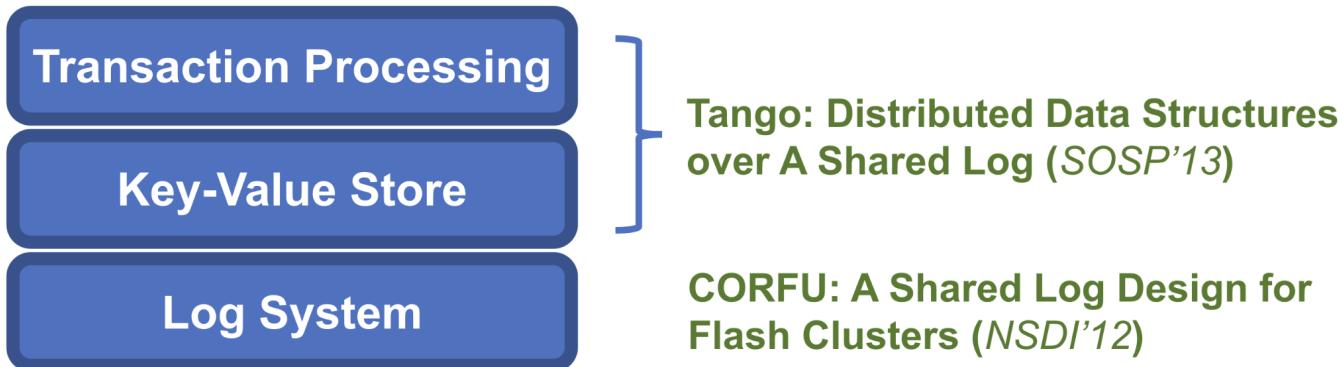
- Full blockchain consensus required > data needs to broadcast to all consensus nodes > **data fully replicated on all the nodes vs selective** = slow DPR
- Economics of running validator are **not profitable**
- Difficulty in changing old codebase, outdated design

## DA Customization

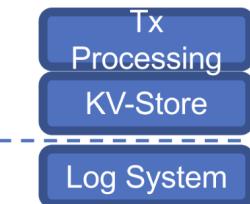
Safety / liveness	Determine how important it is for high availability of the data you are choosing to store (leads to more replication)
Token mechanics	Choose how you want token payments to work based on your data and how much stake is required for a certain node, or a rollup could choose to use dual quorum staking. Pay for DA in native token. Note: by having high throughput, congestion should be a rare phenomenon, we allow for bandwidth reservation, guaranteeing consistent costs to Rollups. A rollup could also pay for throughput on-demand (similar to DynamoDB bursts in Web2)
Fields, signatures	Choose signature and field types
Storage location & replication	Choose which countries to store in and how much of network to propagate too (few nodes, partial network, entire network)
Data persistence	Choose how long you want to store the data (1 month, 1 year, 2 years, etc) > flexible economic payment types
Smart contract state management	Ability to save and load state via smart contract on other L1s, L2s, rollups

---

## Layered Storage Design Already Exists in Web2.0

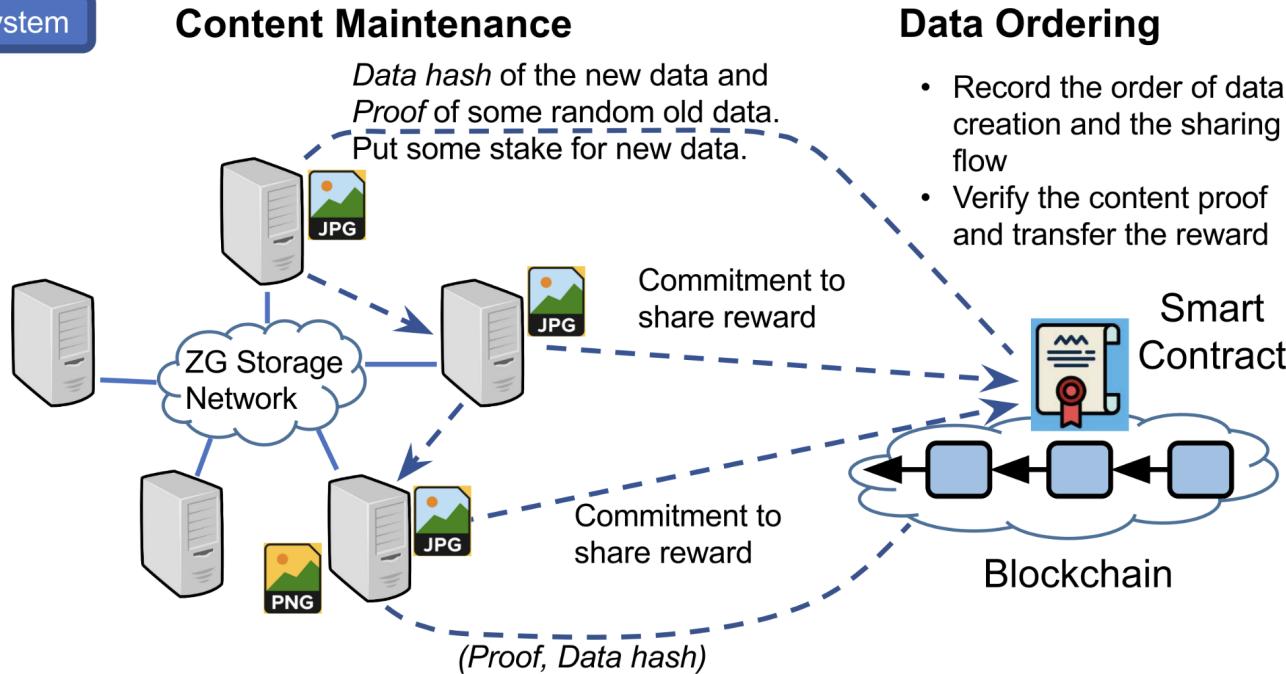


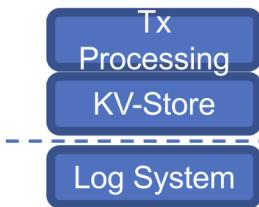
These systems were already developed and deployed for products in Microsoft and Meta



# Decentralized Log System for storage

(a modular design)

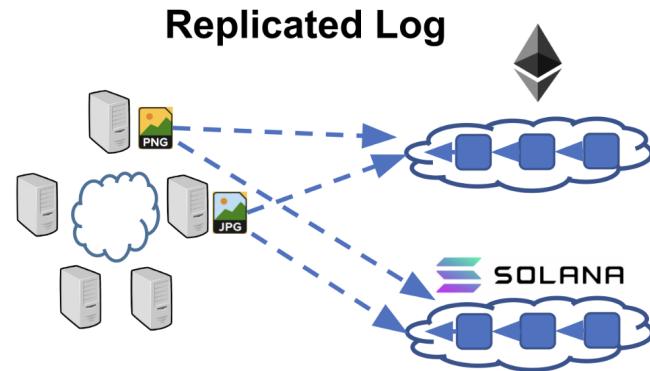




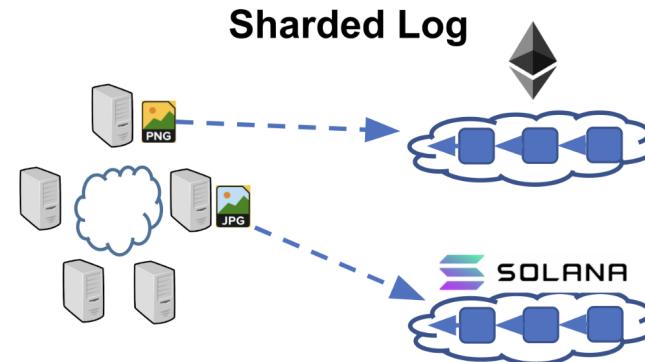
# A Decentralized Log System

(a modular design)

## Flexible log setup (possible via DAO)

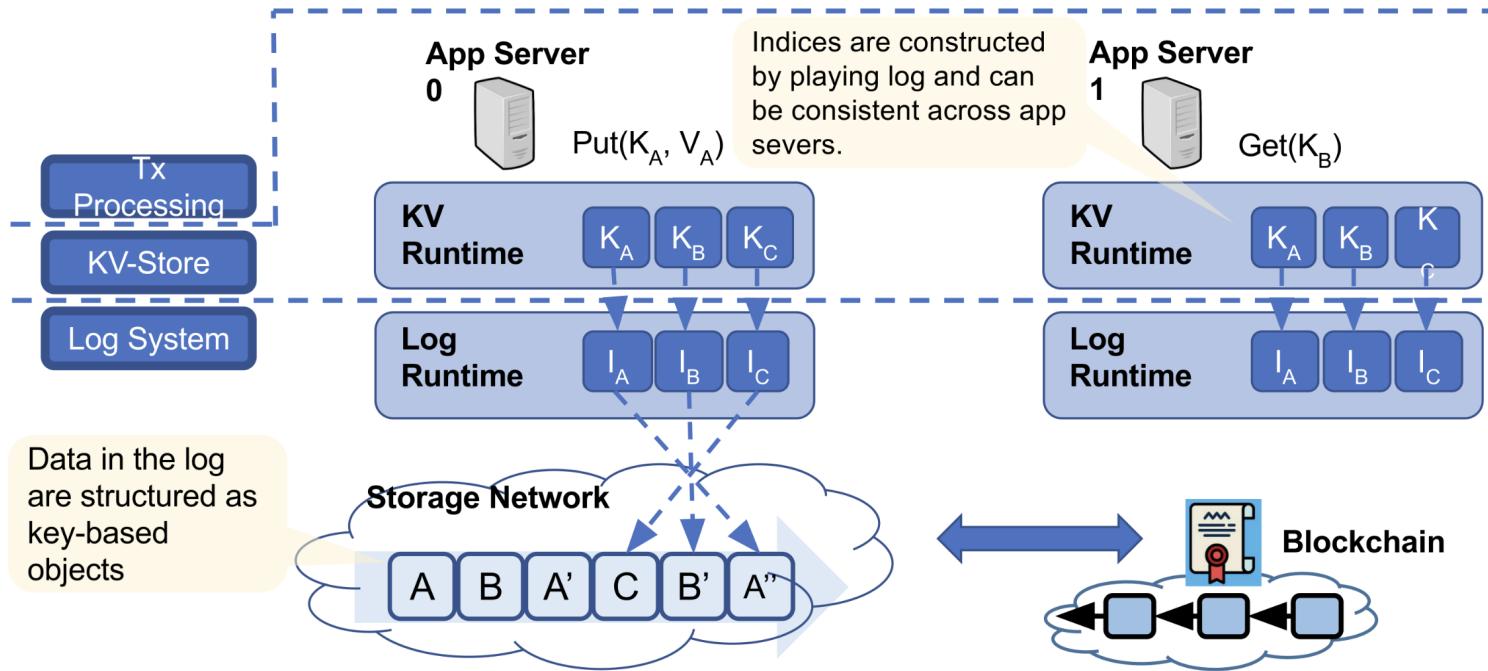


Dapps on different blockchains can share access and verification of the same data.

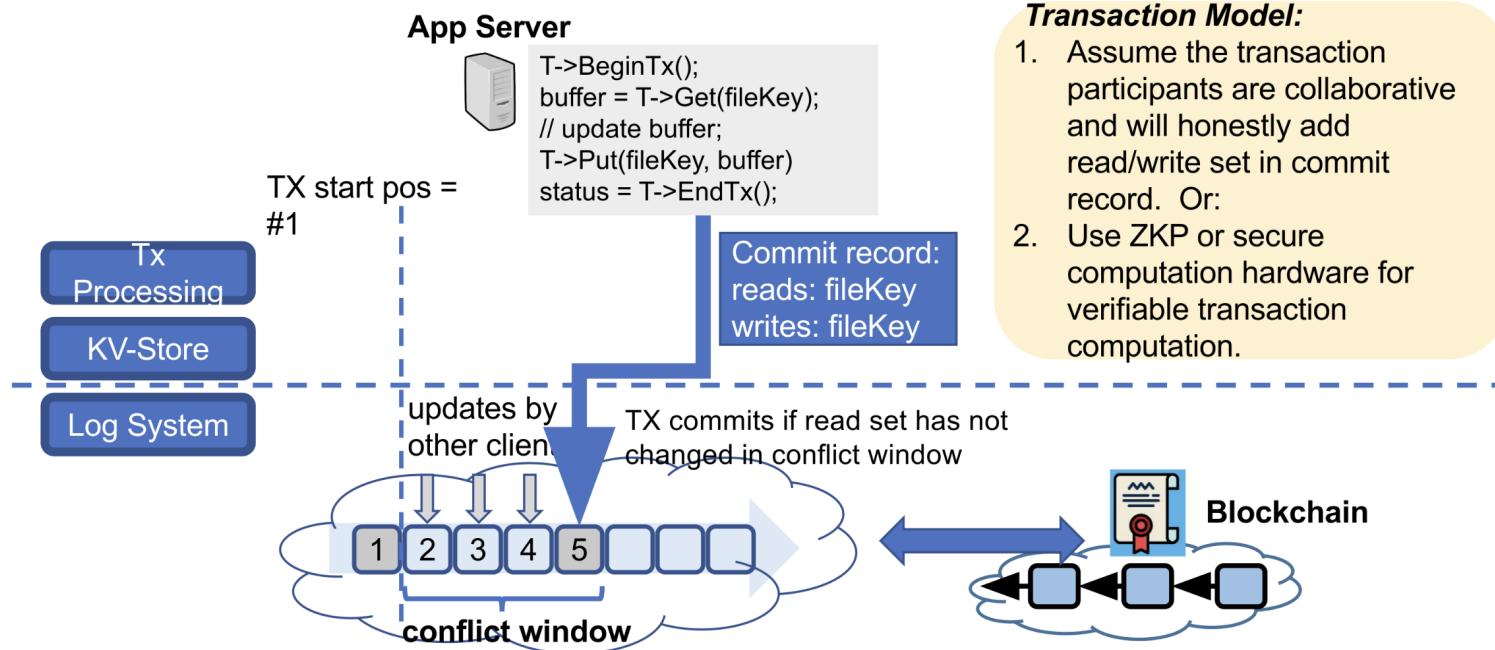


Throughput of the data commit can scale linearly with the number of chains.

# Decentralized Key-Value Store



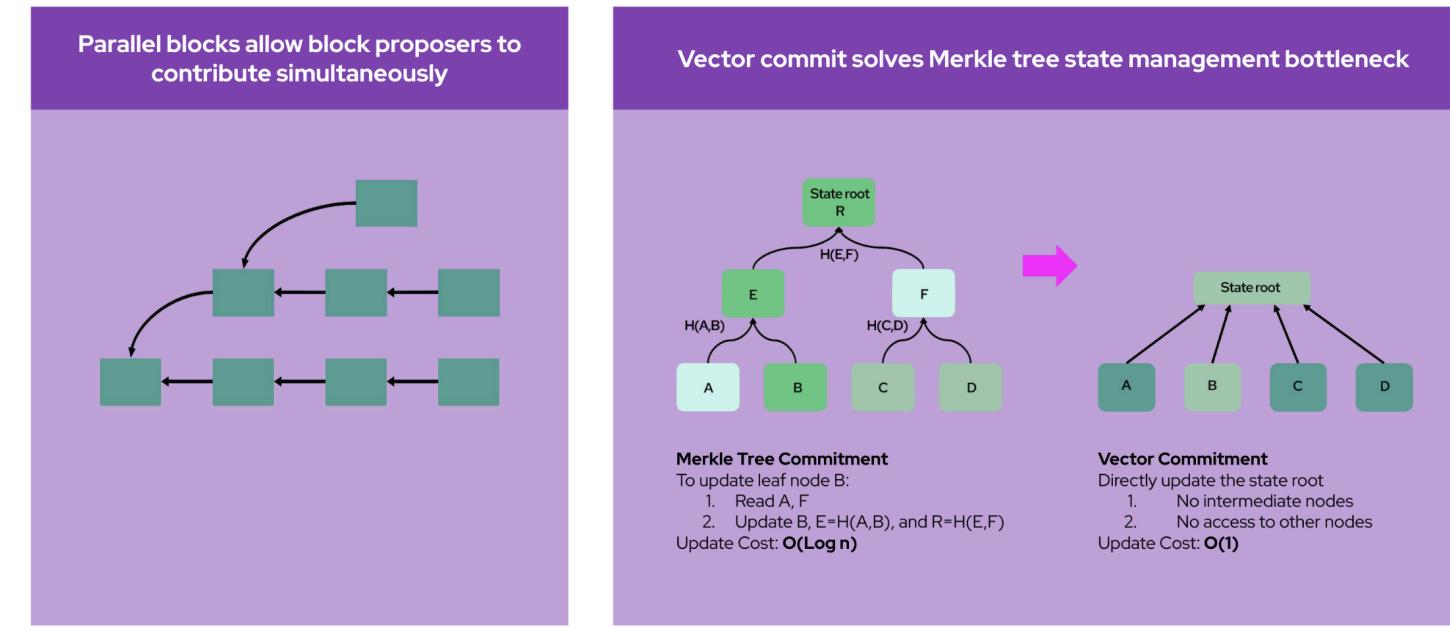
# Transactional Processing on KV Store



## Our storage system vs others

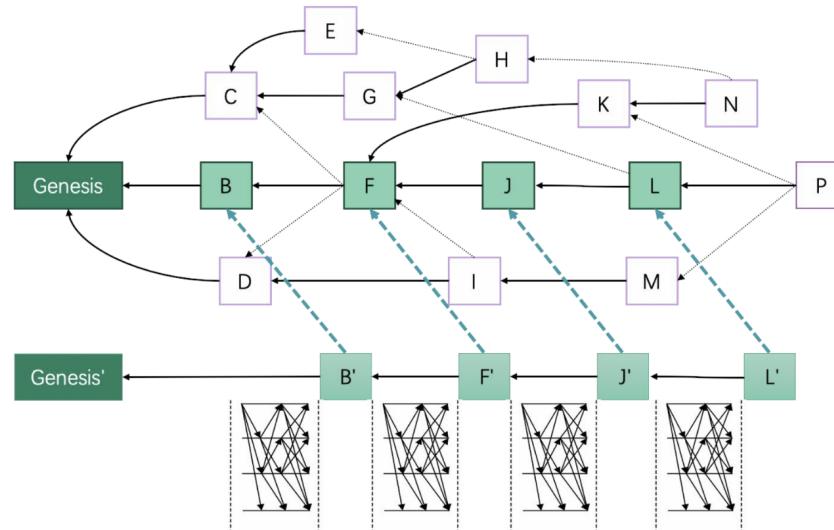
	Filecoin	Arweave	EthStorage	ZG
Data availability 1.0	○	○	○	●
Data availability 2.0	○	○	○	●
Storage types (log, KV, transactional)	○	○	●	●
Low cost (storage, retrieval)	●	●	●	●
VM execution	●	○	○	●
Validator & Proposer separation (censorship resist.)	○	●	○	●
Storage location & replication customization	●	●	●	●
Economic security (slashing)	●	○	●	●
Data persistence customization	●	●	●	●
Smart contract state management	○	○	●	●
Wallet compatibility (evm, etc.)	○	○	●	●

## 200K max TPS & unlimited via appchains, <1S finality



## Separating block proposers and validators for censorship resistance

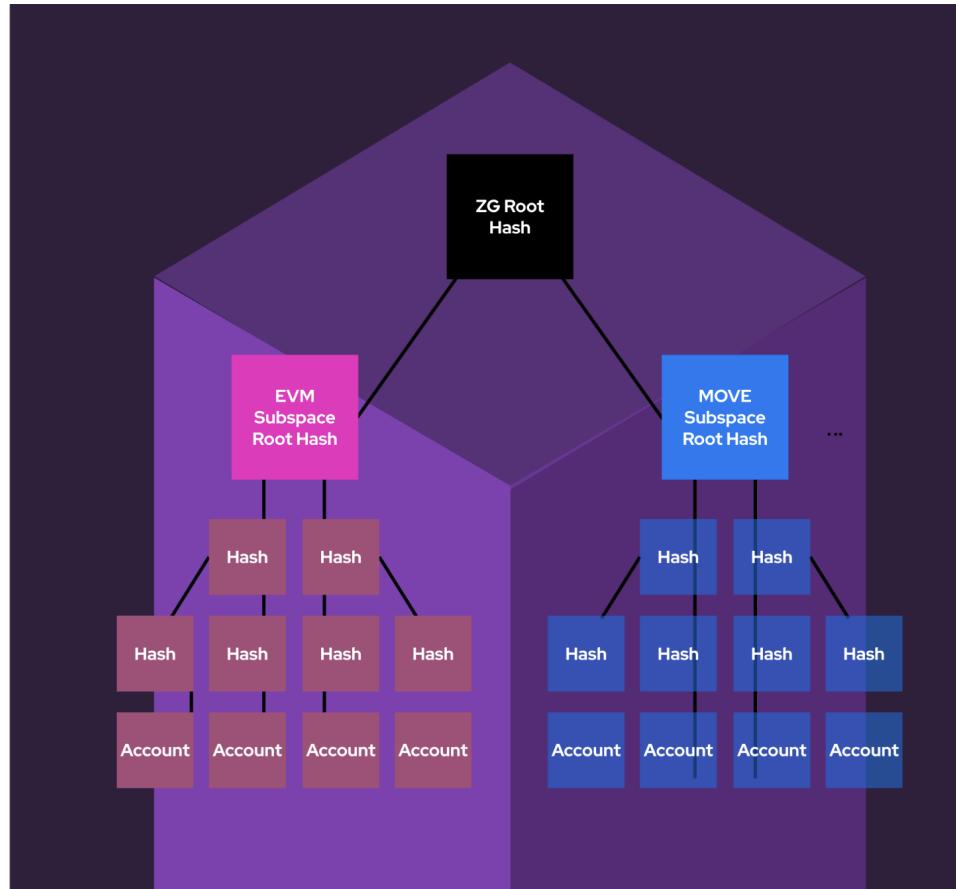
**Validator:** Robustness via add. decentralization & long range attack prevention



**Proposer:** Extra censorship resistance and extra security via finality

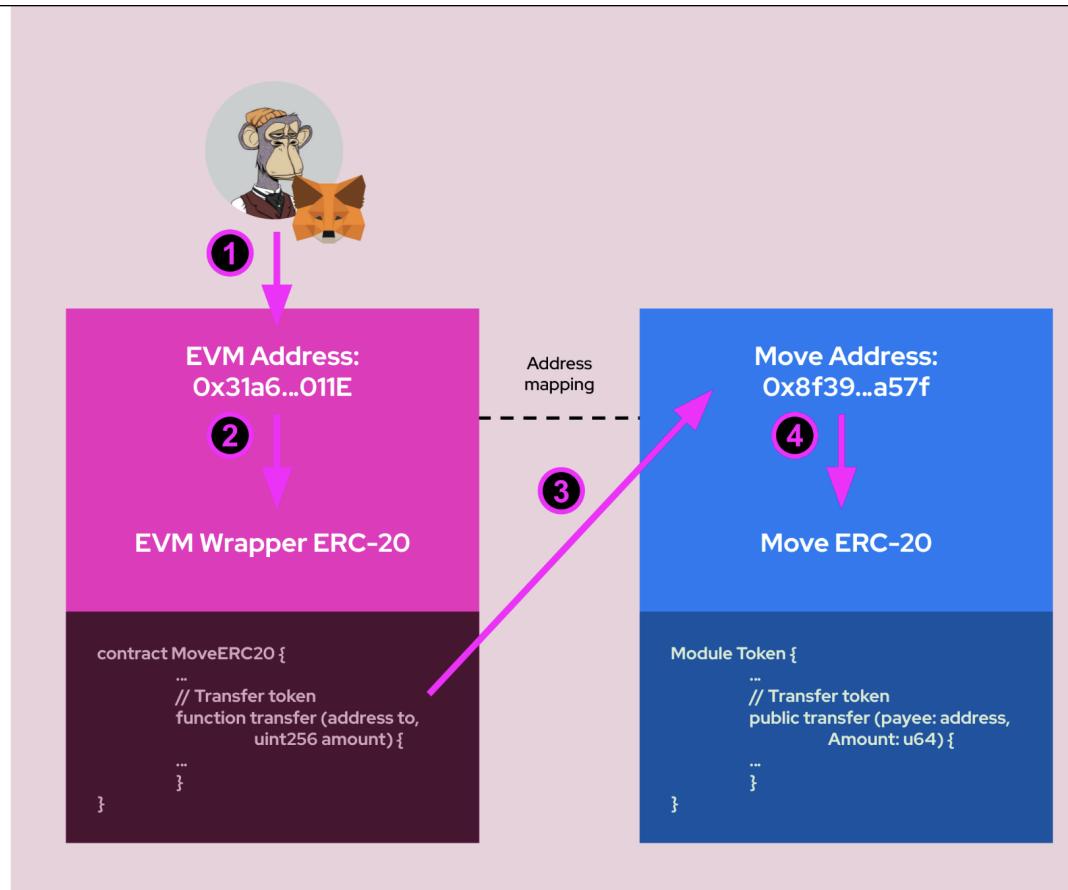
## SUBSPACES ENABLE EASY ASSET ONRAMPS VIA

1. VM State tries
2. Atomic cross-space address proxy mechanism
3. Automatic wrapper contract generation
4. X-Chain bridge functionality



# ATOMIC OPERATIONS ACROSS SUBSPACES

(Metamask example)



## Early Tokenomics Draft

# Alpha Incentive Mechanism Design: Mining Reward

➤ The mining reward assigned to a confirmed block  $B$ :

- $R(B) := R_{base} + \sum_{j=1}^n F_j + \sum_{i=1}^k R_i(B)$  where
- $R_{base}$  is the base reward of every block
- $\sum_{j=1}^n F_j$  is the total transaction fee in block  $B$
- $R_i(B)$  is the storage reward for random query  $Q_i$  in the proof,

$$R_i(B) := \begin{cases} \frac{aV(T_B - T_i) - R_i}{2}, & \text{if query } Q_i \text{ is satisfied} \\ 0, & \text{otherwise} \end{cases}$$

A miner generates  $k$  random queries to previous data chunks and the mining proceeds if a sufficient fraction (say  $k/2$ ) of these queries are proved accessible

- $T_B$  is the height of block  $B$
- $T_i$  is the height that created the data chunk queried by  $Q_i$
- $a$  is a constant defining the generation rate of storage reward
- $V$  is the unit price paid for the creation of this data chunk  $Q_i$
- $R_i$  is the accumulated storage reward for the data chunk  $Q_i$ , and after this block it will be updated to  $R_i := R_i + R_i(B)$

The storage reward is emitted from every data chunk independently, so that rare data indicates higher reward, and frequently mined cheat data generates bounded profit

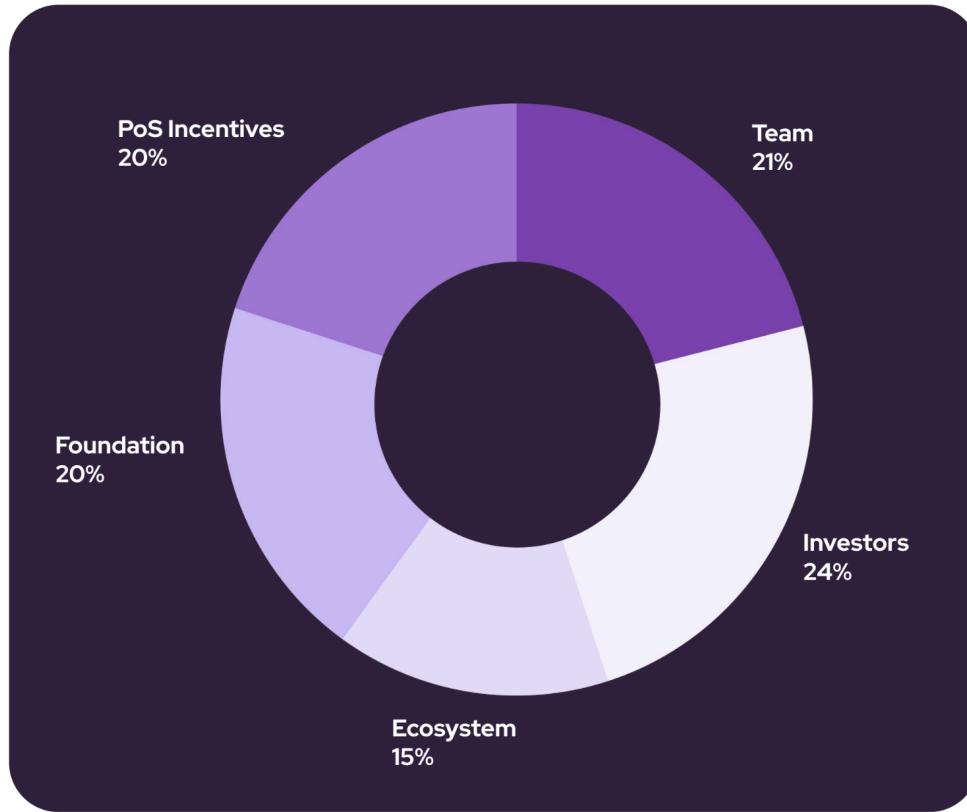
## Alpha Incentive Mechanism Design: Data Sharing

- The data sharing royalty for every query  $Q_i$  proved in block  $B$ :
  - Consider the data chunk that  $Q_i$  hits:
  - if the author of block  $B$  obtained this chunk from someone recently, i.e. the request is registered after height  $T_B - T_{royalty\ period}$
  - then the author and data provider split the storage reward for query  $Q_i$ ,  
i.e. each of them gets  $\frac{1}{2} R_i(B)$
  - otherwise the author of block  $B$  did not get this data chunk recently, he gets the full reward  $R_i(B)$
- Fairness properties:
  - Data providers are incentivized for storing and sharing data.
  - The total reward remain the same with or without data sharing royalty,  
so there is no incentive to forge sharing relations.

## Token allocation draft

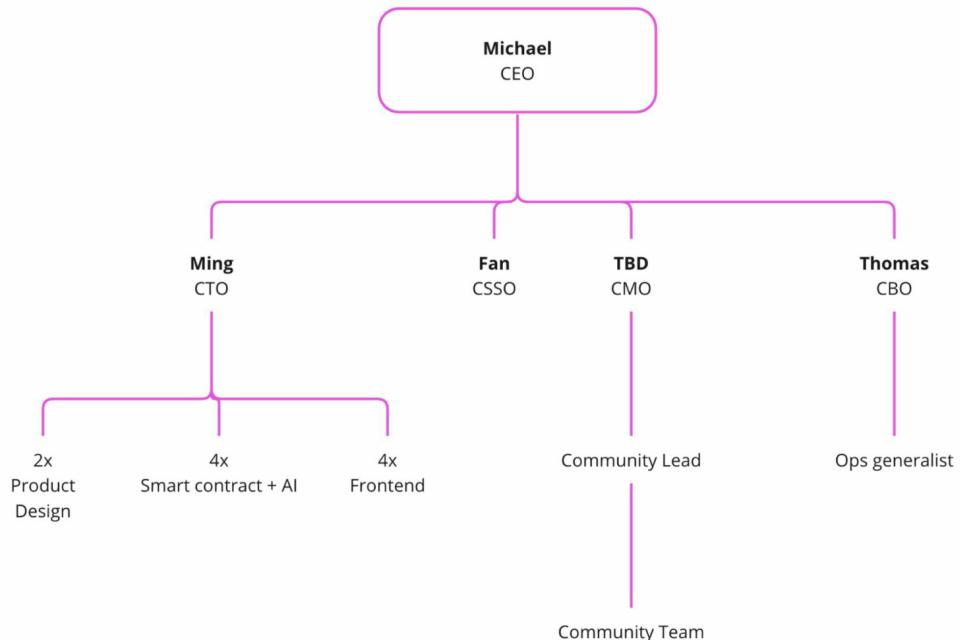
ZG token acts similar to Eth (both deflationary & inflationary fixed supply asset). 4 year vest post public mainnet.

Later incentives will come only from transaction fees.



Org Chart Draft

## ORG CHART - Min Team SEED



# ORG CHART - SERIES A

