Today Verkle + Tomorrow ZK = Everything Stateless, Everything Lightclient

Jason Chaskin, Gajinder Singh

Part 1

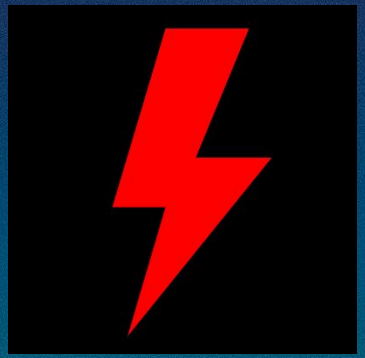
Benefits of Stateless Clients

Part 2

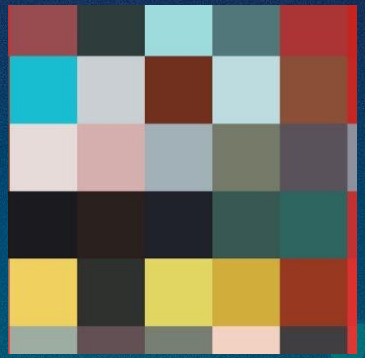
End Game of ZK Ethereum

Thank You Stateless Team

Guillaume Ballet



Ignacio Hagopian



Josh Rudolf

And many more people!

Benefits of Running a Node

Trustless Verification

Direct access to blockchain data, ensuring every piece of information is accurate by consensus rules

Privacy

Your read and write requests remain private, shielded from any RPC provider

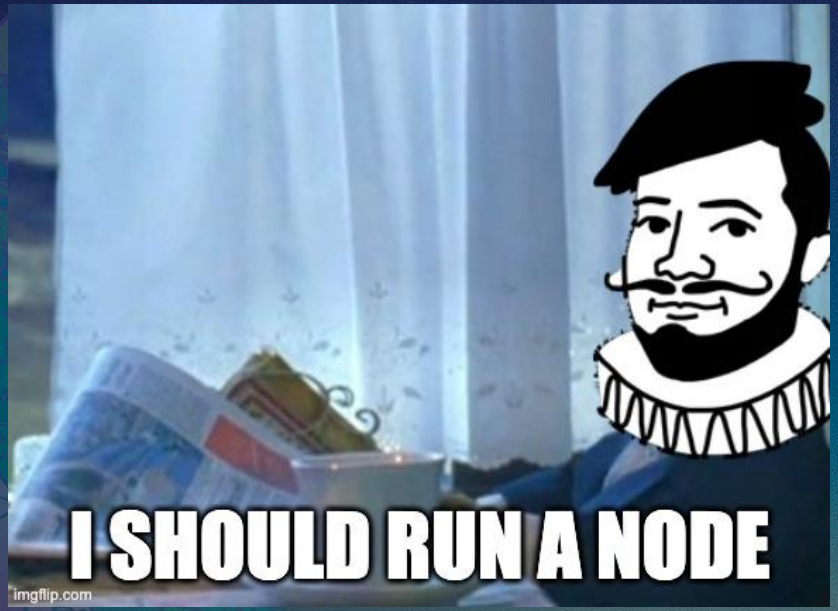
Trustless Transaction Broadcasting

Broadcast your own transactions without relying on third-party nodes

Strengthening Decentralization

Act as a watchdog, with the power to catch any attempts to break rules

The Boy Who Wanted to Run a Node



chaskin.eth 8mo

Member

I want to run a full node but am not sure about the hardware to buy. Ik there's a lot of resources but they are overwhelming for a hardware newbie like me. I'm looking for someone to hop on a call and guide me thru the purchasing options, where to buy them, and how to set it up

Reward: 0.022 ETH or 50 USDC

@bountybot

Step 1: Buy The Hardware

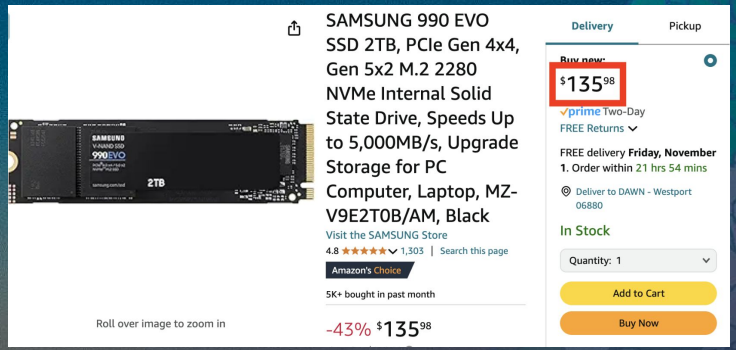
Computer ($600)

Runs Ethereum software, must stay on

Storage Upgrade ($136)

2TB needed for blockchain data

Total Cost = $736



Step 2: Set Up The Node

Choose Software Clients

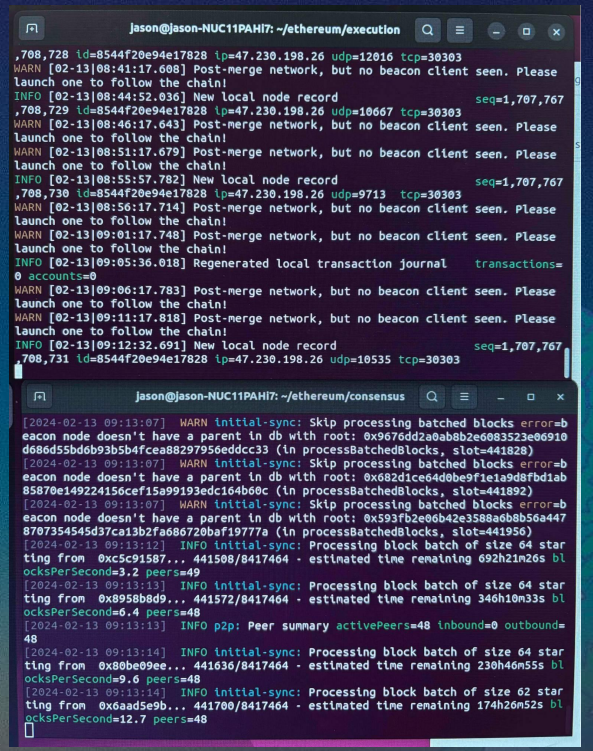
Select execution (Nethermind) and consensus client (Lighthouse)

AI Setup Help

Use ChatGPT to guide through command-line setup steps

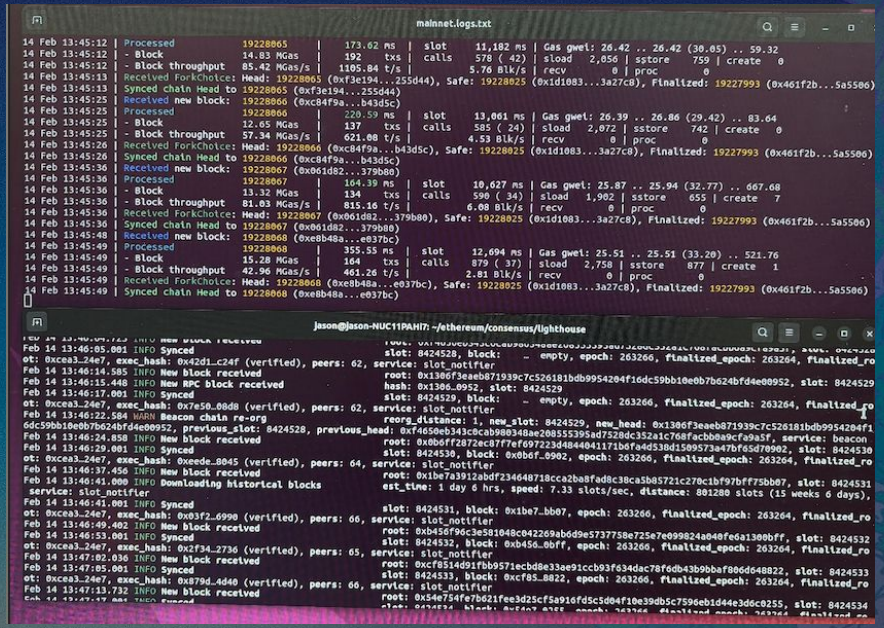
Start Syncing 🎉

Node begins syncing—this takes time, so I leave it running



Step 3: 24+ Hours Later… Node is Live

24 hours later, I’m synced and processing the latest blocks!



What Happened While the Node Was Syncing

Connect to Peers

Client connects to other Ethereum nodes to start data sharing

Download Blockchain History

Receives block headers and bodies, storing all historical transactions on disk

Load Current State

Downloads active account balances, contracts, etc. **This takes time due to frequent disk lookups and updates**

State Verification

Confirms the state root hash matches the latest block header, ensuring accuracy

Sync Complete

After ~24 hours, the node is fully synced and ready to participate in the network

What Happens During Block Verification?

0 - 3 Seconds
Block Proposal and Propagation

- Proposer assembles and broadcasts a new block
- Nodes receive the block within seconds

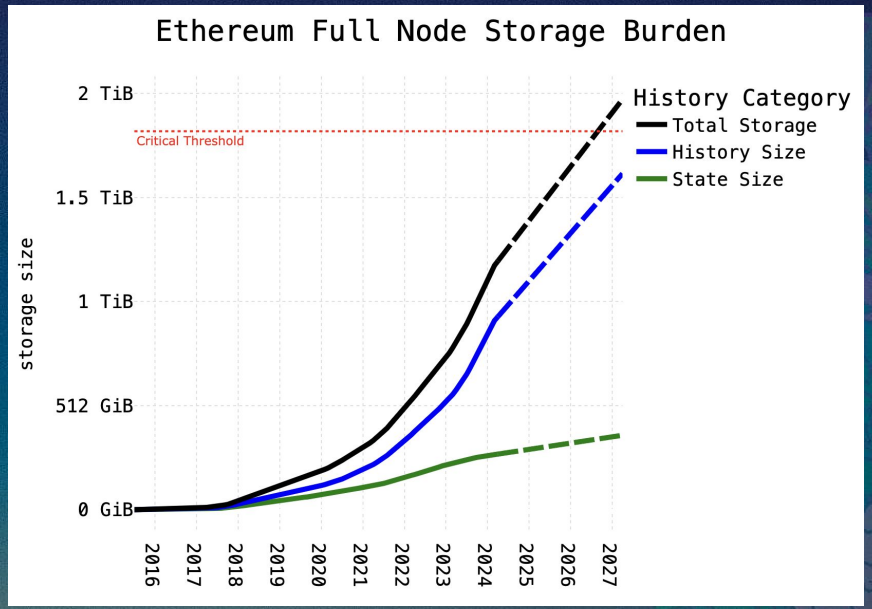
1 - 4 Seconds
Transaction and Consensus Checks

- Verify signatures, transaction validity, and proposer eligibility
- Access and update state database (balances, contracts)
- **State Growth Bottleneck: Slower lookups and updates as the state database expands**
- Processing time per block: ~300-600ms

4 - 12 Seconds
Consensus Voting

- Consensus voting stuff between all the staking nodes that we don't need to worry about

State Growth Bottleneck



Running a Node in a Stateless World

No need for new hardware—just use your current computer

Choose and install your Ethereum client software

Near instantly go live, syncing and verifying with the network!

Block Validation with Stateless Nodes

Builder

Creates a new block with transactions and generates a compact Verkle proof witness for validation

Witness

Pre-state data with a proof of correctness (e.g., Alice's and Bob's balances)

Proofs are compact < 2,000 KB

Stateless Node

Verifies transactions using witness to confirm state accuracy

Slaying the State Growth Bottleneck

3-10x Block Verification Efficiency Boost

Yes, that means a potential 3-10x gas limit increase for scaling L1

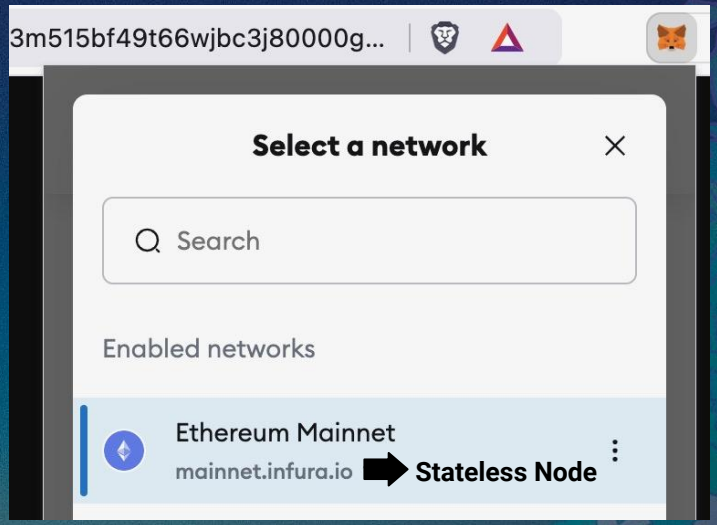
Achieved by eliminating costly storage lookups and updates

Wallet Integration

Wallets run stateless clients in the background, users won't even realize they're running a node

Users interact trustlessly with the blockchain, no need to rely on centralized RPCs like Infura

Possible on lower-powered devices like browsers or maybe phones



Millions of Nodes?

“They need a little bit of storage, they don’t need much memory, they need a little bit of bandwidth, but it could be that you can run these on your phone. Ethereum may grow from 10,000 nodes to millions of stateless nodes."

“Running a node as part of your wallet software (even on your mobile phone) becomes practical.”

“I'm not sure yet if phones and smartwatches could be enabled only by Verkle, that might be a stretch.”

Why Decentralization Matters?

“The elites of your blockchain community, including pools, block explorers and hosted nodes, are probably quite well-coordinated; quite likely they're all in the same telegram channels and wechat groups. If they really want to organize a sudden change to the protocol rules to further their own interests, then they probably can.”

The End Game

“You’ll be running a node, whether you realize it or not, contributing to a truly trustless, unstoppable network.

Ethereum isn’t a tech startup, it’s a global public good for all of humanity.”

Recap

	Before Stateless Nodes	After Stateless Nodes
Storage Needed	2TB and growing 📈	Minimal* 🗃️
Verification Speed	Slower due to DB lookups 🐢	At least 3x faster 🐆
Set Up	Takes over a day 🦥	Near-instant ⚡
Browser Compatibility	No ⛔	Yes ✅
Mobile Device Compatibility	No ⛔	Potentially 🔄
Number of Verifying Nodes	Thousands 🌆	Potentially Millions 🌐

** (with history expiry)*

But wait there's more

L2 Scaling

Ethereum will scale its DA layer using DAS. The more nodes sampling blob data, the more blobs the network can handle per block

Remember how stateless clients could lead to millions of nodes...

Cross-L2 Communication

Smaller, cheaper-to-verify proofs make inter-L2 communication onchain more cost-effective, supporting seamless interactions between different L2s

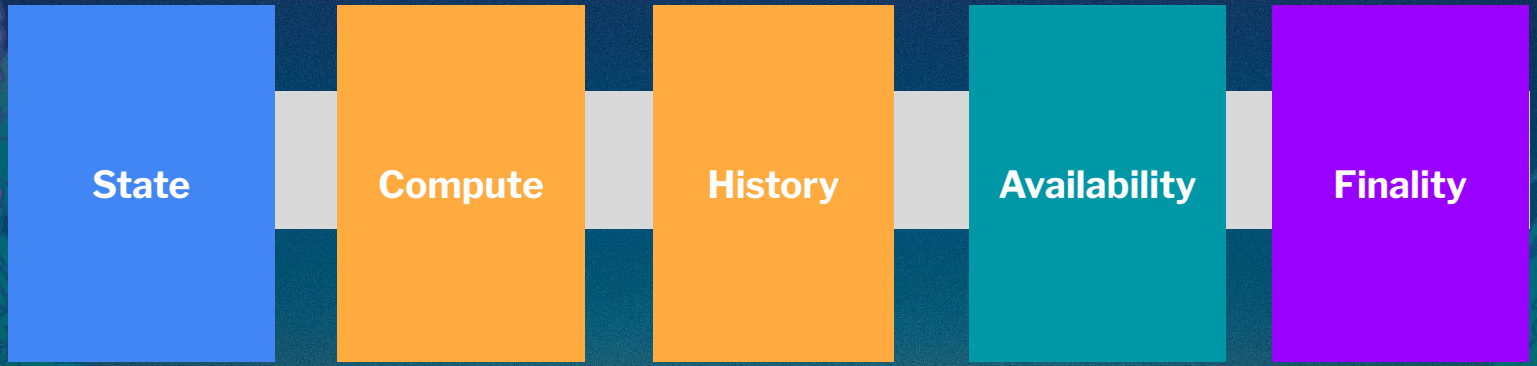
Targeted Transaction Processing

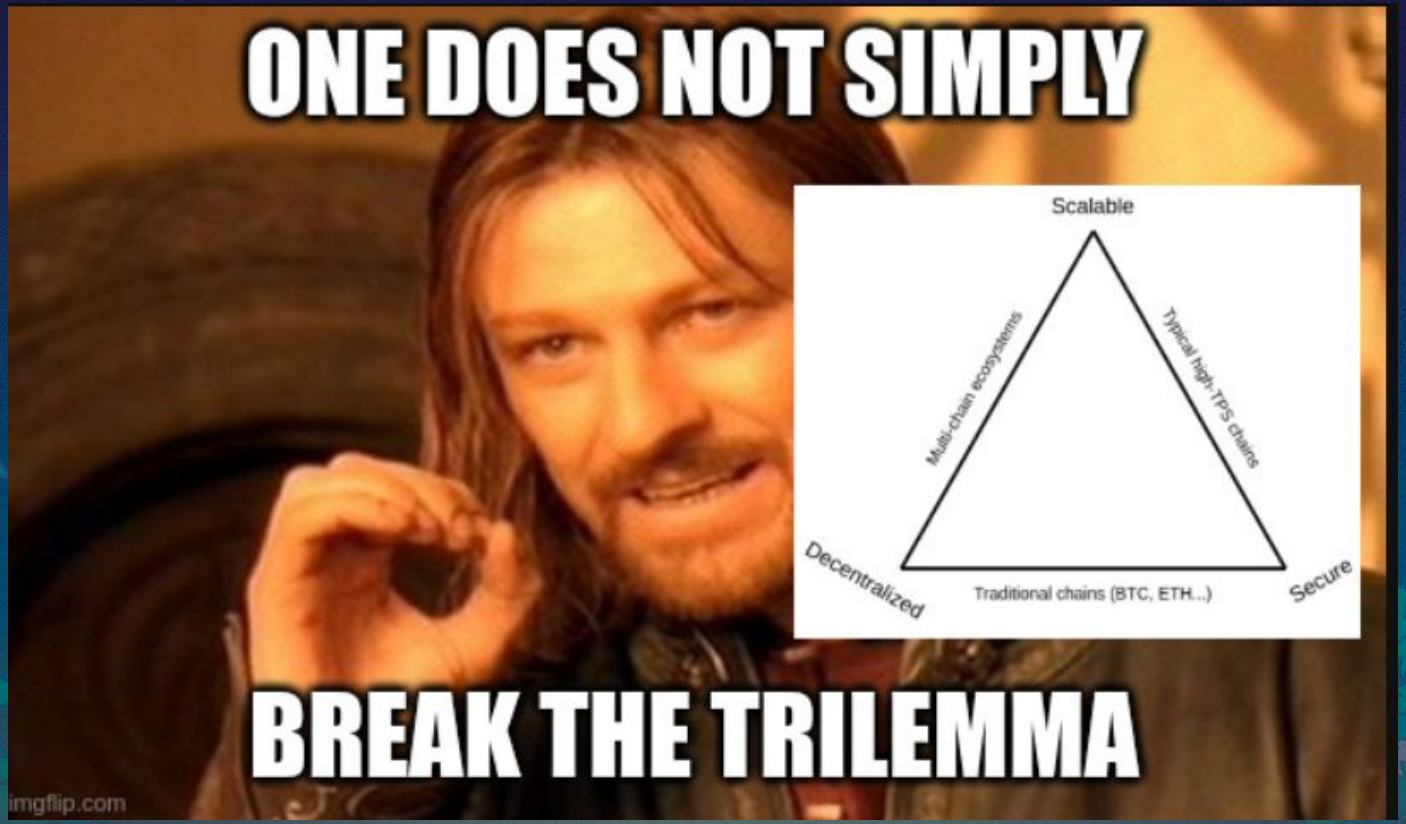
Can quickly check if a block affects specific accounts or contracts, allowing them to skip unnecessary processing and focus only on relevant transactions

Part 2 – Back to the (ZK) future

©g11tech

End Game of ZK ETHEREUM




ONE DOES NOT SIMPLY
Scalable
Multi-chain ecosystems
Typical high-TPS chains
Decentralized
Traditional chains (BTC, ETH...)
Secure
BREAK THE TRILEMMA
imgflip.com

Shattering the TRILEMMA

State

Less/sharded state in L1 chain

- although currently only 15-20% (~200GB) of actual storage
  - but we want Giga Gas!
- L2s sort of sharded state
  - ZK rollups with state transition proofs
- Engineering solutions to shard, slice and prove the state

Statelessness

- (almost) complete removal of state without compromising security

Source: ETHDubai 24 - Anatomy of a stateless client by Guillaume Ballet

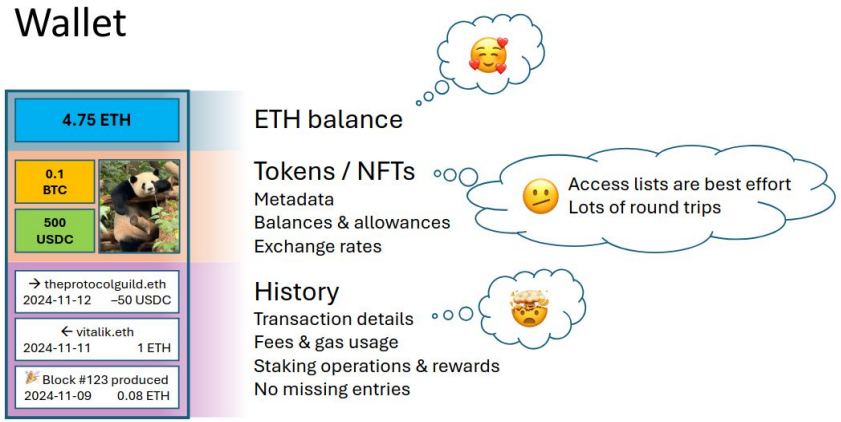
Shattering the TRILEMMA

History

Need ETH logs

Less/Sharded History

- EIP 4444 gives some mitigation
- ZK friendly Accumulators and filters for rapid global indexing and lookup
- Engineering solutions/products to shard, slice and prove the history



Source: https://fusaka-light.box/slides.pdf

Shattering the TRILEMMA

Compute

- Distributed block building
  - Composable Multi proposer Bundles
  - ZK-EVM
  - Verifiable State transitions
- Enable huge smart contracts
  - Verkle/stateless unified tree
    - provable code chunks
    - On demand loading

Availability

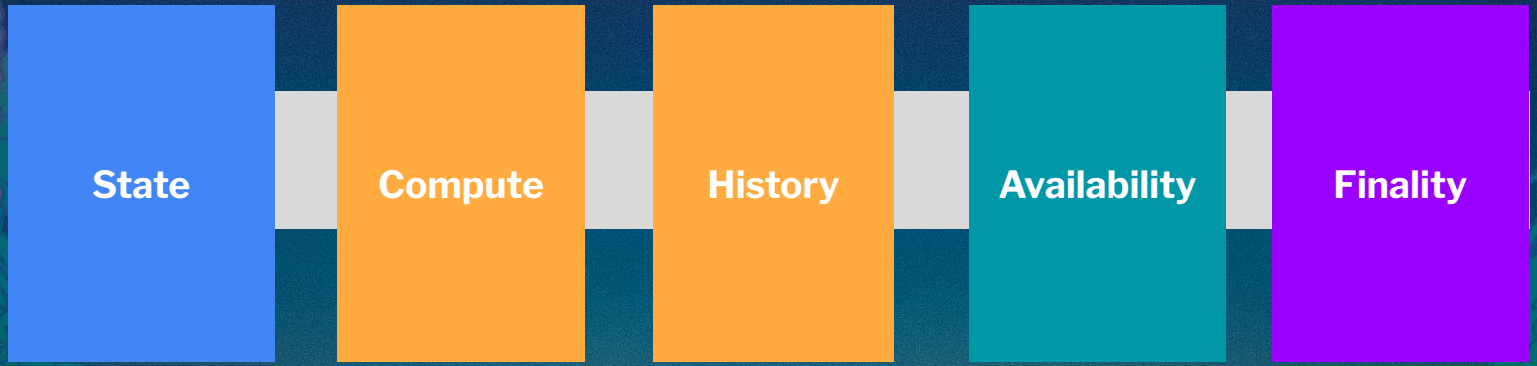
- Transient data (blobs)
- Sharded blobs (peerDAS)
  - Jointly verify data availability
- Sharded transaction mempool

Finality

- SSF - Orbit!
  - Multiple committees with ZK verifiability
  - Easy ZK aggregation with Unions
- Faster slot times
  - Fast ZK verifiability

Protocol as sum of its (lightclient) parts

Composable but succinctly provable islands of



Magic Sauce - Statelessness + ZK

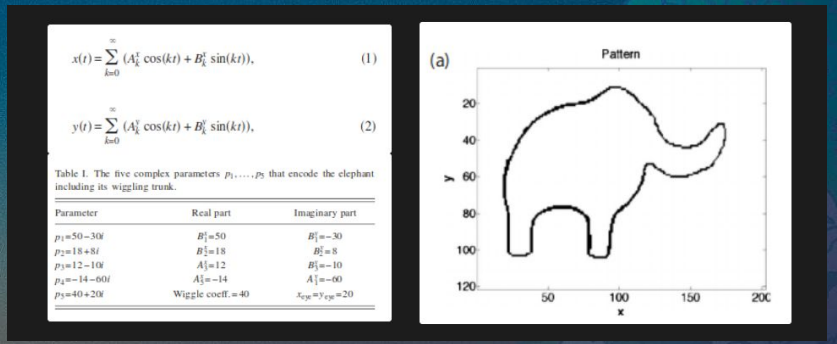
(current) Common thread: Polynomials

Polynomials are the zk friendly constructs that can "encode" high (but finite) amount of data points and commit them into small sized evaluations

- State => (key,value)
- History => (key,value)
- Computation trace/constraints => (key, value)

Generate succinct proofs (size and verifiability)

- Resource intensive
- Some large strides been made recently
  - Circle STARKS, BINIUS



$$x(t) = \sum_{k=0}^{\infty} (A_k^x \cos(kt) + B_k^x \sin(kt)), \quad (1)$$

$$y(t) = \sum_{k=0}^{\infty} (A_k^y \cos(kt) + B_k^y \sin(kt)), \quad (2)$$

Table 1. The five complex parameters $p_1, \ldots, p_5$ that encode the elephant including its wiggling trunk.

Parameter	Real part	Imaginary part
$p_1 = 50 - 30i$	$B_1^x = 50$	$B_1^y = -30$
$p_2 = 18 + 8i$	$B_2^x = 18$	$B_2^y = 8$
$p_3 = 12 - 10i$	$A_3^x = 12$	$B_3^y = -10$
$p_4 = -14 - 60i$	$A_3^x = -14$	$A_3^y = -60$
$p_5 = 40 + 20i$	Wiggle coeff. = 40	$x_{cyc} = y_{cyc} = 20$

(source: https://vitalik.eth.limo/general/2019/05/12/fft.html)

Statelessness + ZK = lightclients everywhere

- Not all participants are same
- Not all participants have to do the all/same work
  - work done by one can be used by many
  - Some centralization risk
    - mitigated by easy verifiability and highly decentralized attestation network that can be build
    - Sharded engineering solutions to help in heavy lifting
      - portal?

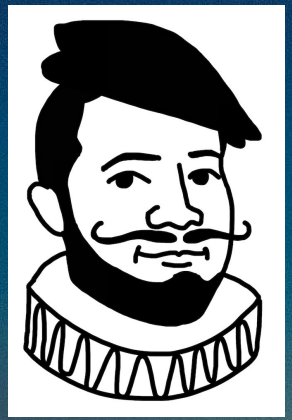
A future where no one is a “full” node/participant

Composability needs to be “protocol”-ed in

As always - truth lies somewhere in between

But lets

to an exciting (ZK) future



Thanks!

©g11tech