Decentralizing access to Ethereum utilizing Ethereum's Portal Networks

Kolby ML (Moroz Liebl)

Software Engineer, Ethereum Foundation

Origins of Ethereum

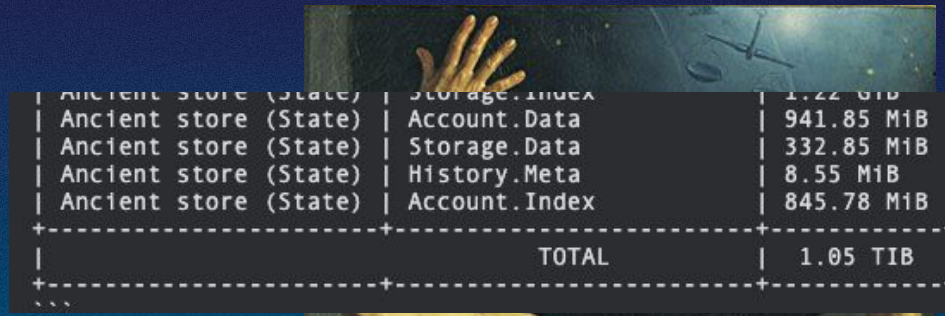
- Introduced the programmable blockchain the “world computer”
- Ethereum has an emphasis on decentralization, censorship resistance

How did people access Ethereum

- Running full nodes
  - Provides access to the Ethereum Protocol via JSON-RPC
  - Long running process which takes time to synchronize

Full nodes became inaccessible

- Requirements for users
  - Knowledge
  - Time
    - Syncing node
    - IT maintenance
  - Hardware/Cost
    - A 4 TB hard drive is expected to be required in 6 months



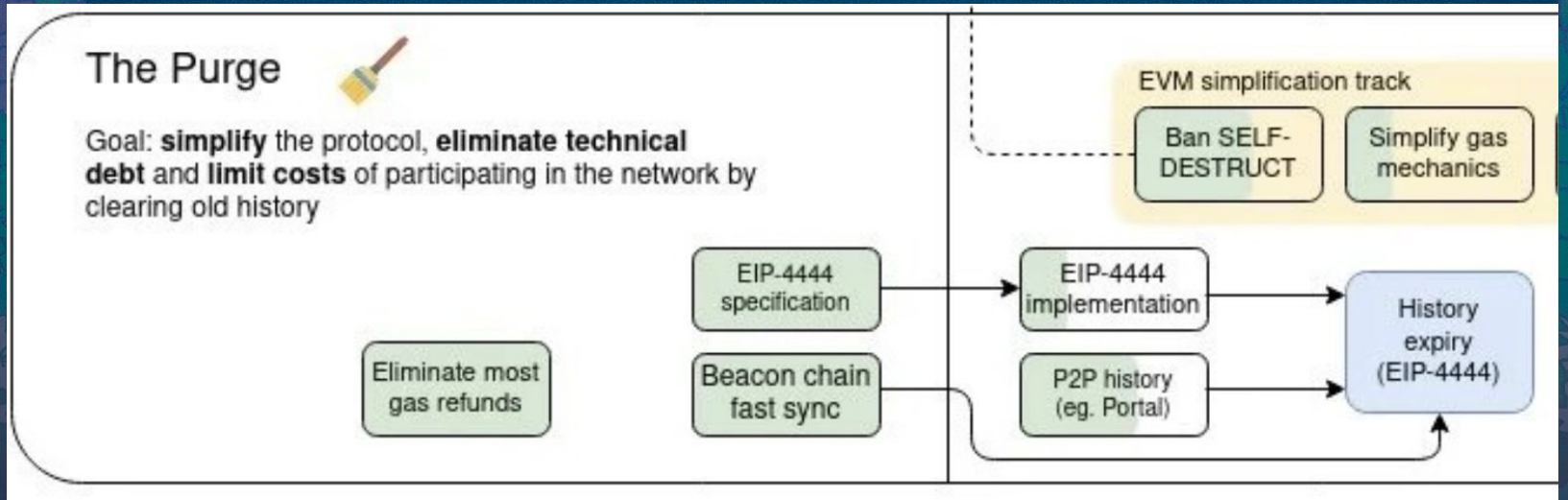
Ancient store (State)	Account.Data	941.85 MiB	90000
Ancient store (State)	Storage.Data	332.85 MiB	90000
Ancient store (State)	History.Meta	8.55 MiB	90000
Ancient store (State)	Account.Index	845.78 MiB	90000
TOTAL		1.05 TiB	

Centralized providers

- Centralized providers removed the barriers to access Ethereum
- Came with the consequence of being centralized

EIP-4444's - History Expiry

- An EIP to increase decentralization and accessibility by lowering the costs to participate in Ethereum by removing the requirement for nodes to store the full history
- That doesn't mean full nodes can't obtain the history it is just they are not responsible for the whole



How do we achieve EIP-4444s

Enter the Portal Network

What is Portal

- Instead of depending on full nodes to provide all data access, the Portal Network distributes this responsibility among numerous small, independent nodes.

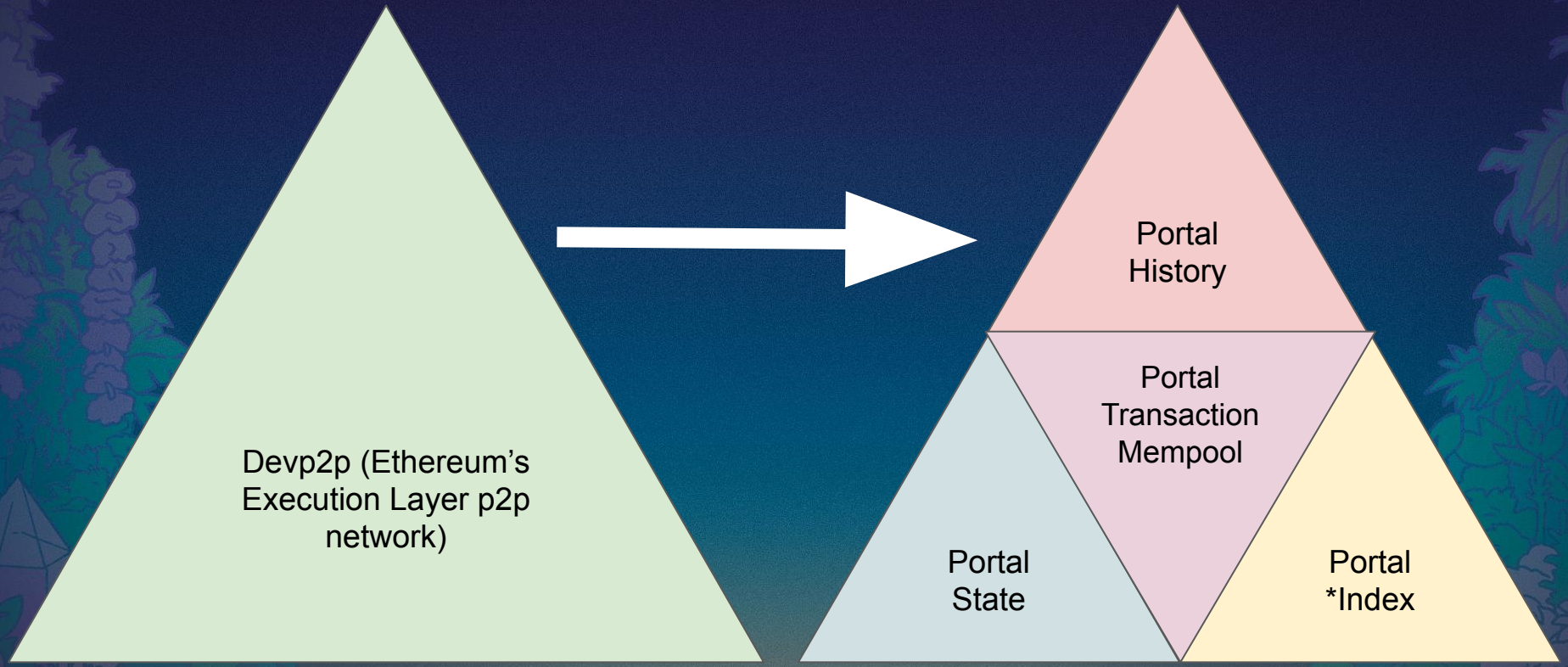
Portal is sustainable

- Portal nodes are both servers of the data, but also consumers.
- Portal's storage requirements are configurable by the user, so it will never use too much space!
- Light CPU usage

But how does it work?

- Portal uses a fundamentally different style of P2P Network then Ethereum Today
- A Distributed Hash Table (DHT) is a decentralized storage and lookup system that enables data to be distributed across a network.
- Each piece of content is addressed by a unique identifier and stored on multiple nodes for speed and redundancy
- DHT's make it possible to find content in log2(nodes on the network) look ups
- Each node stores a slice of content depending on the network they are apart of

Instead of 1 P2P network for Execution…



What does each network do

Portal History	Portal State	Portal *Index	Portal Transaction Mempool
- Headers - Bodies - receipts	- Account State (account trie/storage trie)	- Transaction to block hash indexes	- A distributed mempool for sending transactions

Full archive node but you know like distributed ™©

Portal is fully provable

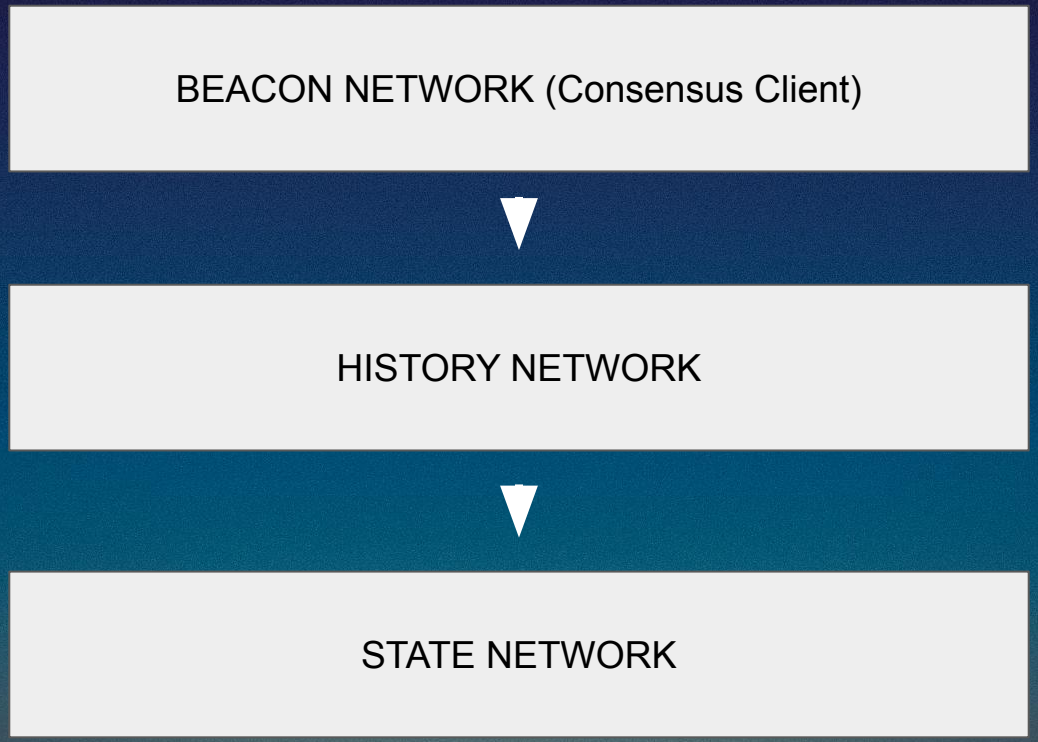
Portal Beacon Network

- Light weight consensus layer access, based off the consensus layer light client protocol, but optimized for Portal's usecases.

Ethereum Consensus layer client

- Used by full nodes today

Validation


BEACON NETWORK (Consensus Client)
HISTORY NETWORK
STATE NETWORK

How do Portal clients initially get the data?

- Through Bridges which are full nodes
  - Long term full nodes are expected to gossip this data to the network the same way blocks are today

Incentives?

- Financial incentives are difficult
  - Leads to race to bottom
  - Easy to game
  - There are other projects trying this approach
  - Wouldn't everyone prefer a free peer-2-peer model?
- Ethereum's Protocol has inherent value
- Portal gives access with to Ethereum Protocol with cheap hardware

Use Cases

Lighter Full Nodes with EIP 4444s while maintaining the UX

One click nearly instant sync time archival node access

Replacement for centralized providers

- Fully decentralized lightweight wallets

What’s the catch?

- Portal is a trade off between latency and storage
- Nothing is as fast as having it on disk
- But do we need instant access to the fully history?

For the first time, you can choose how you participate in the Ethereum Protocol.

Who is building Portal Clients?

OPEN SOURCE

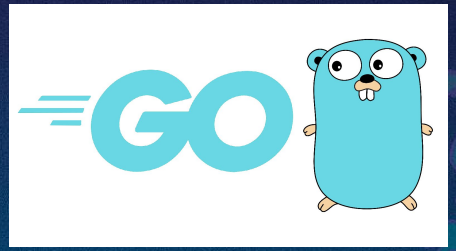
EPF

status

NETHERMIND®

YOU?


NIM

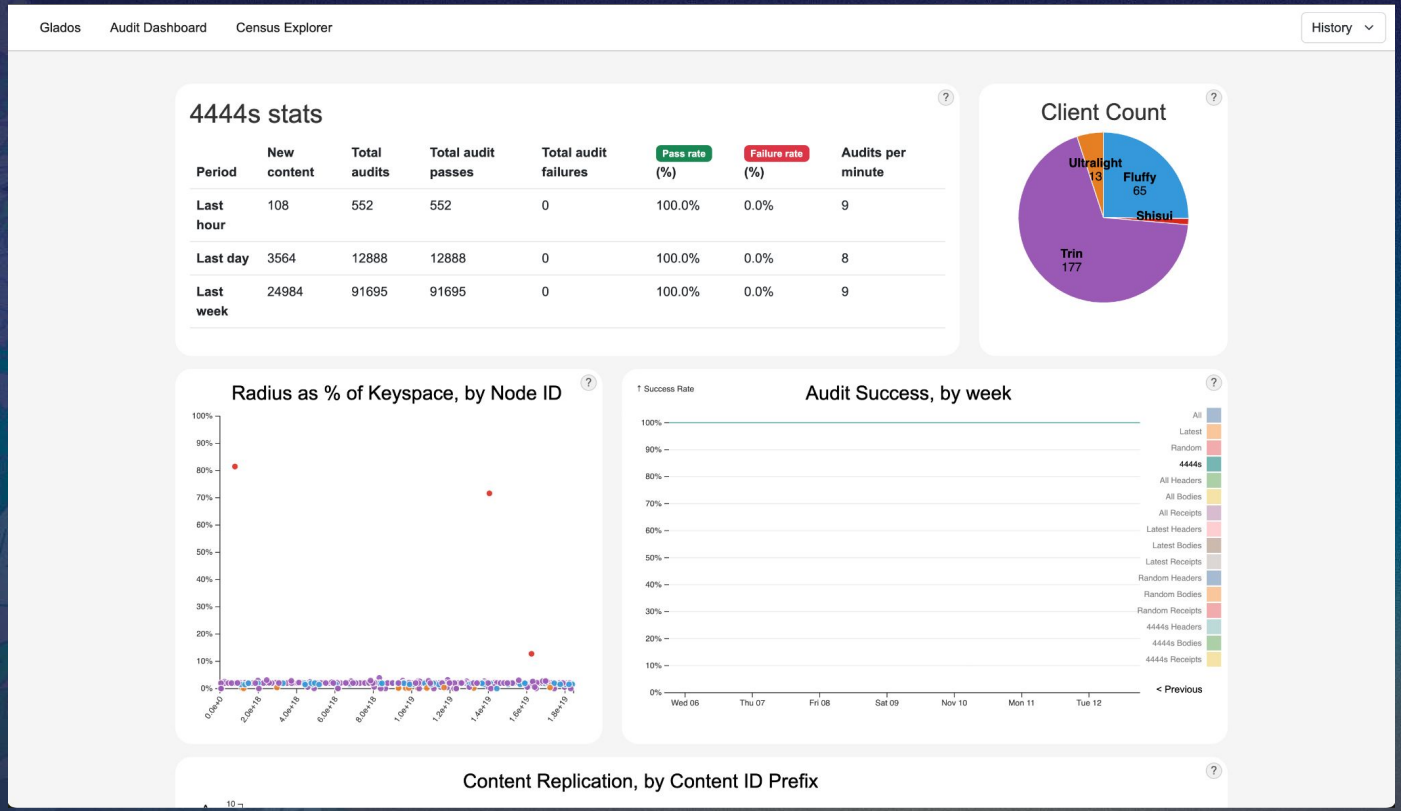

GO


JS

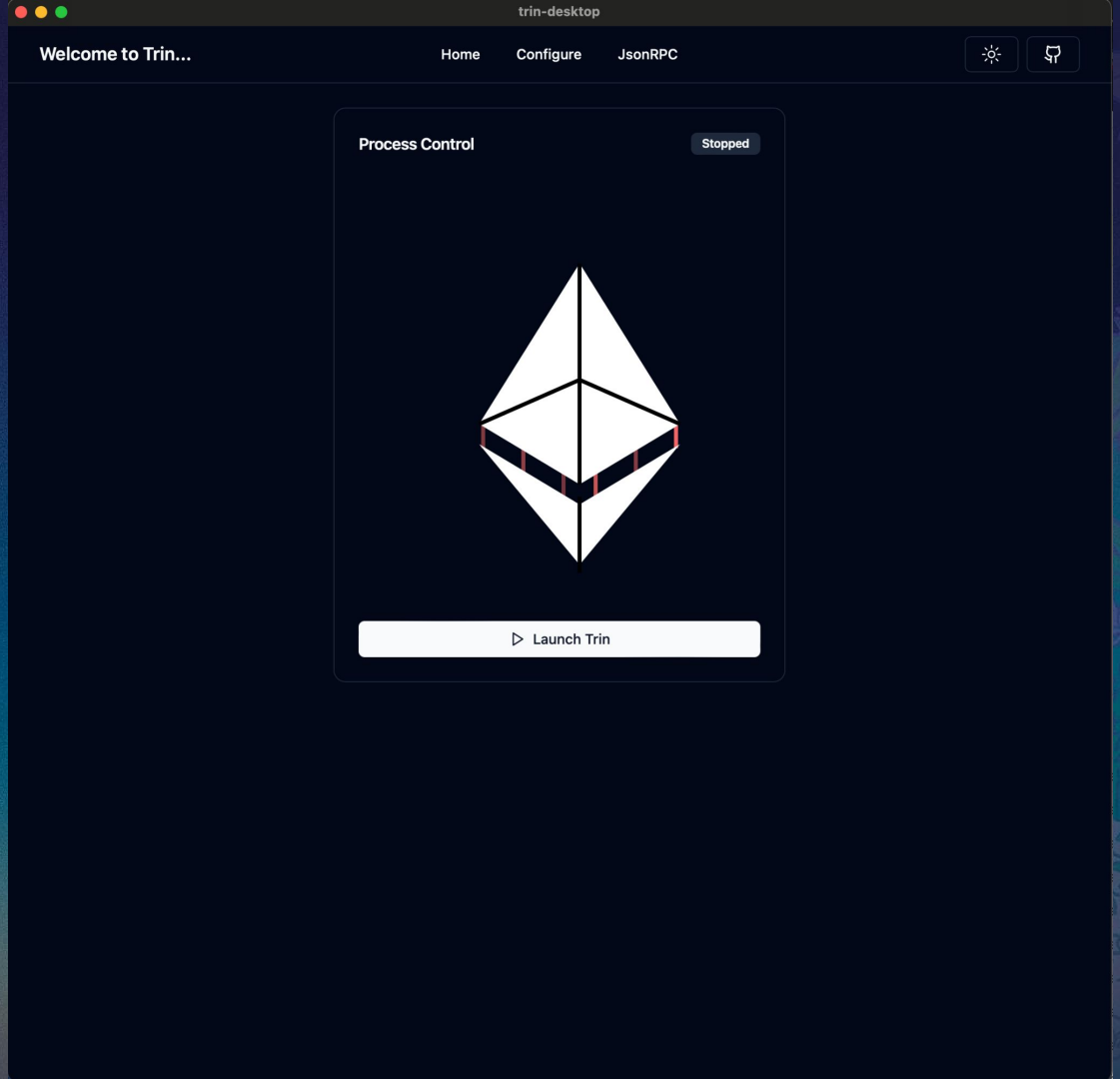

Java

Portal History is Ready for Developers

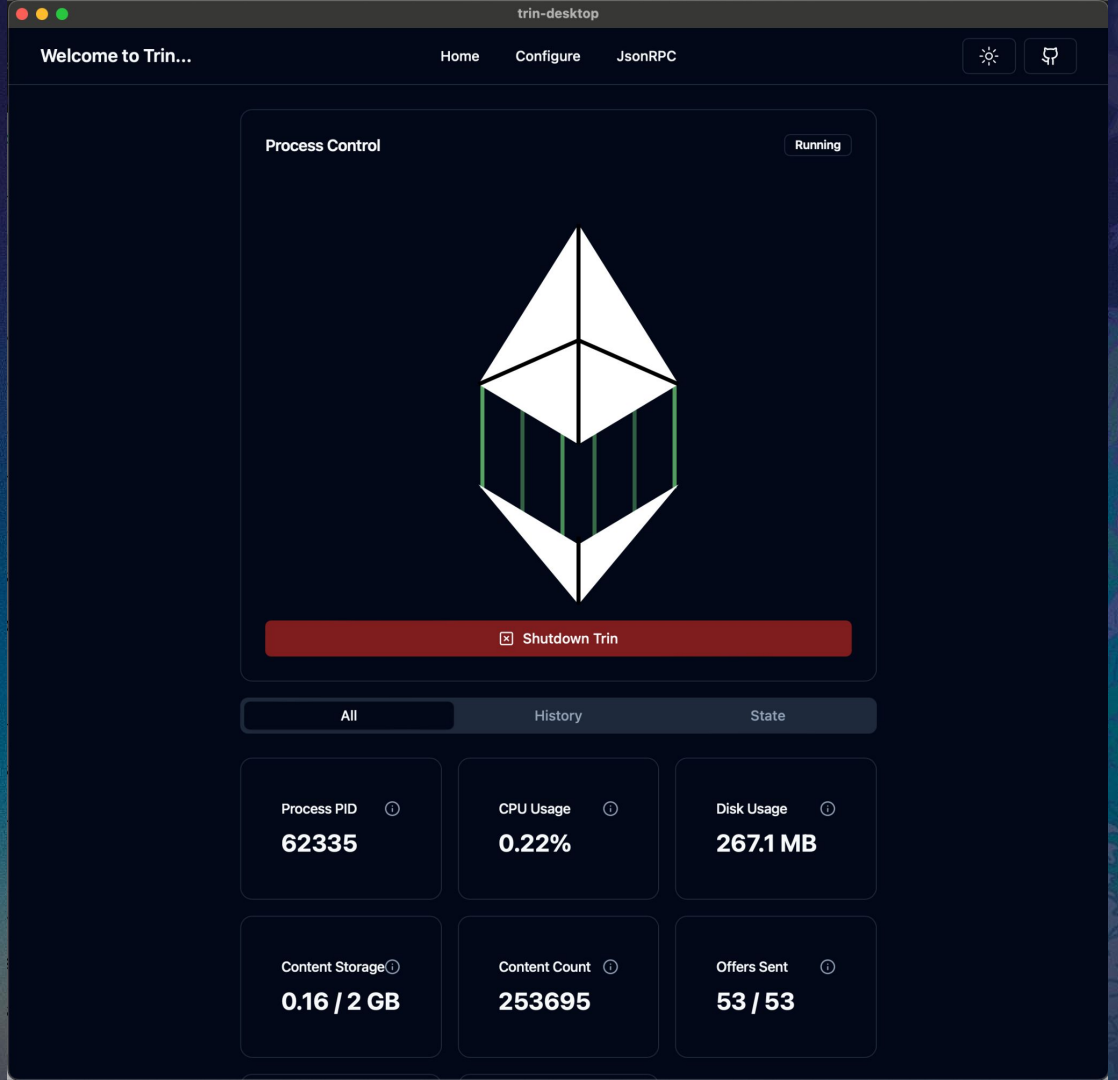
We have the stats: glados.ethdevops.io



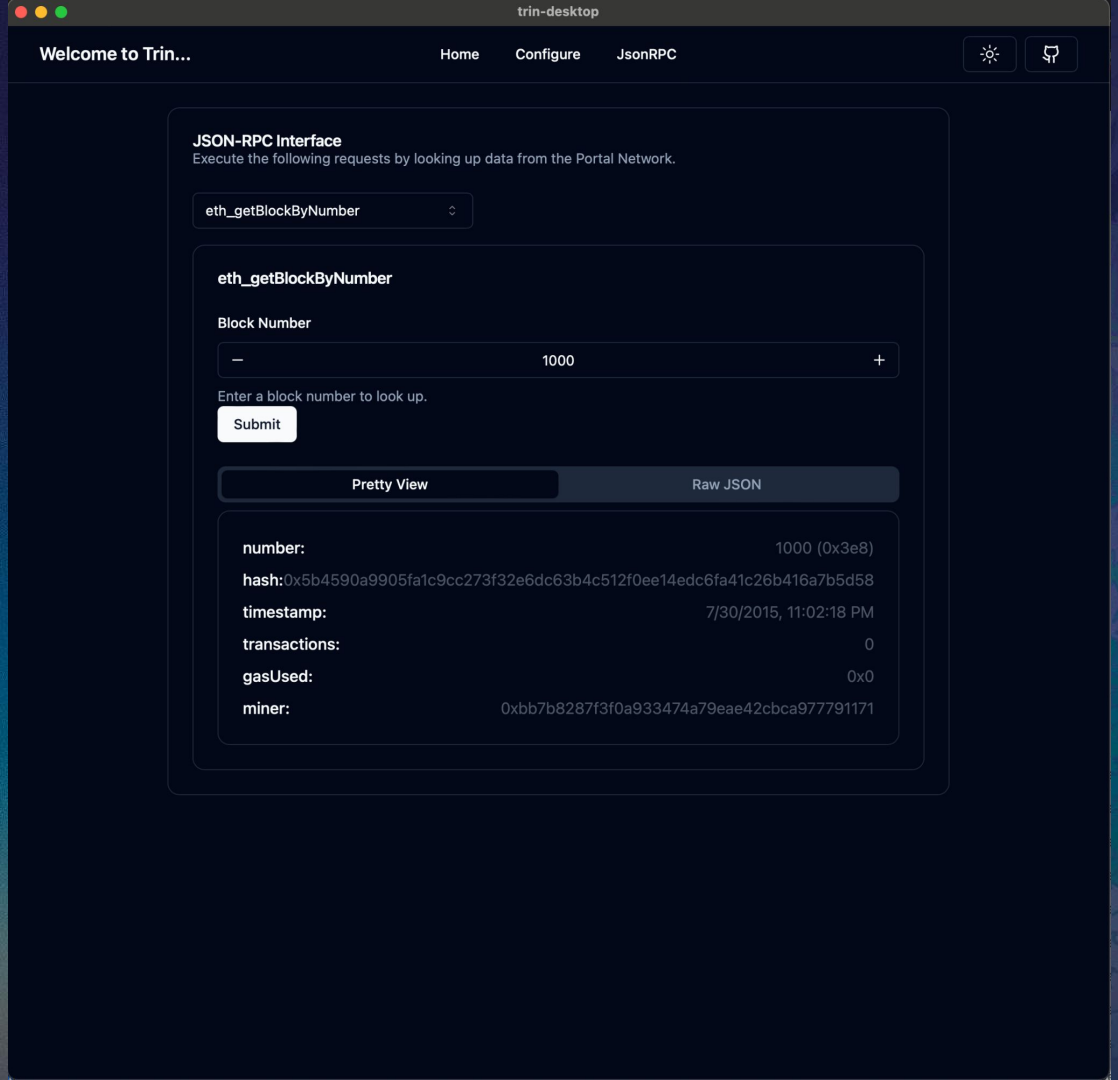
Trin Desktop



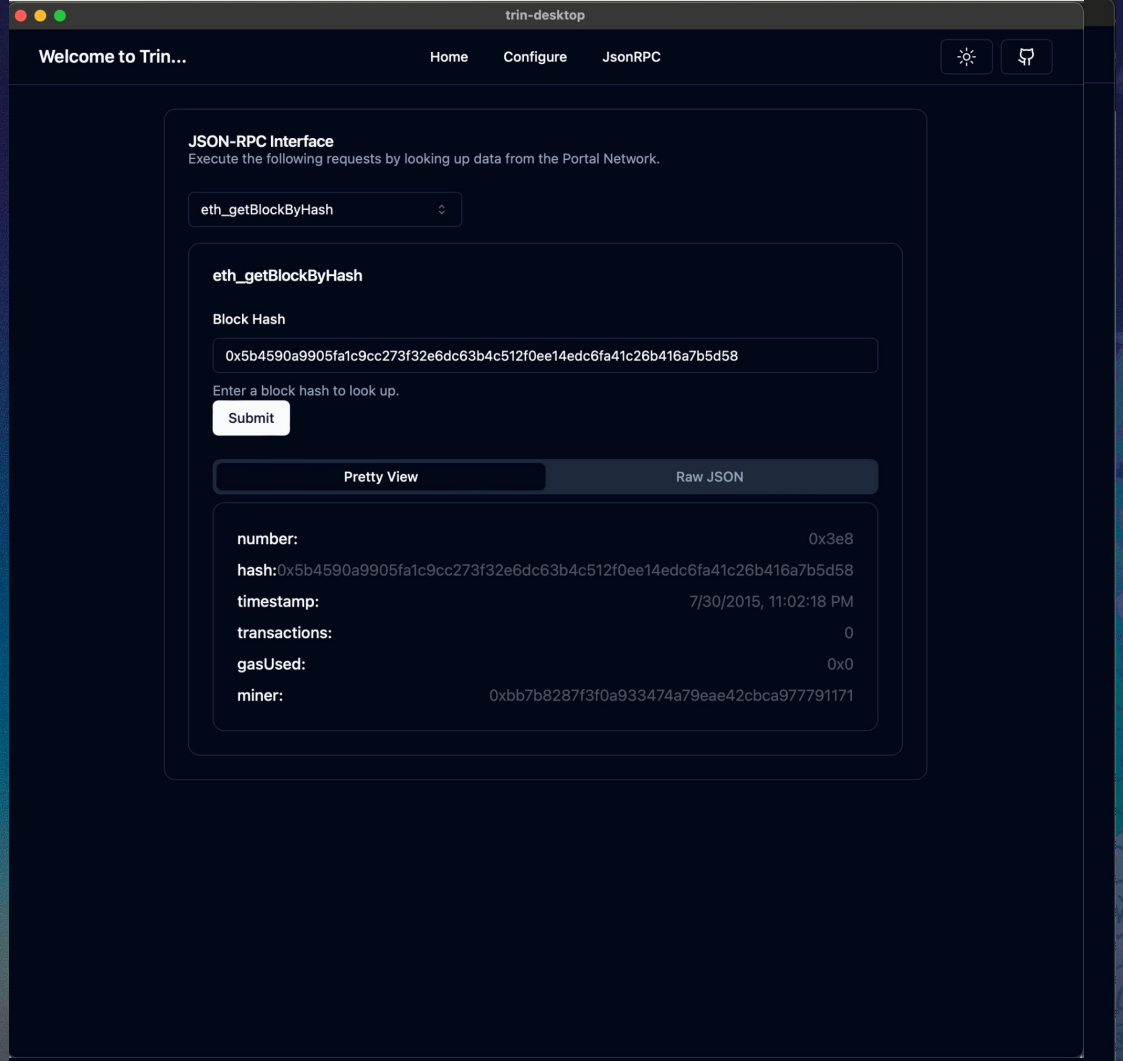
Trin Desktop Running Trin



Trin Desktop
eth_getBlockByNumber



Trin Desktop eth_getBlockByHash



Trin Desktop
eth_getBalance

trin-desktop

Welcome to Trin...

Home Configure JsonRPC

JSON-RPC Interface
Execute the following requests by looking up data from the Portal Network.

eth_getBalance

eth_getBalance

Address

0x9D0F347e826b7dCeaAd279060a35c0061eCf334b

Enter an address to check the balance.
Balance will be checked at block number 1000000.

Submit

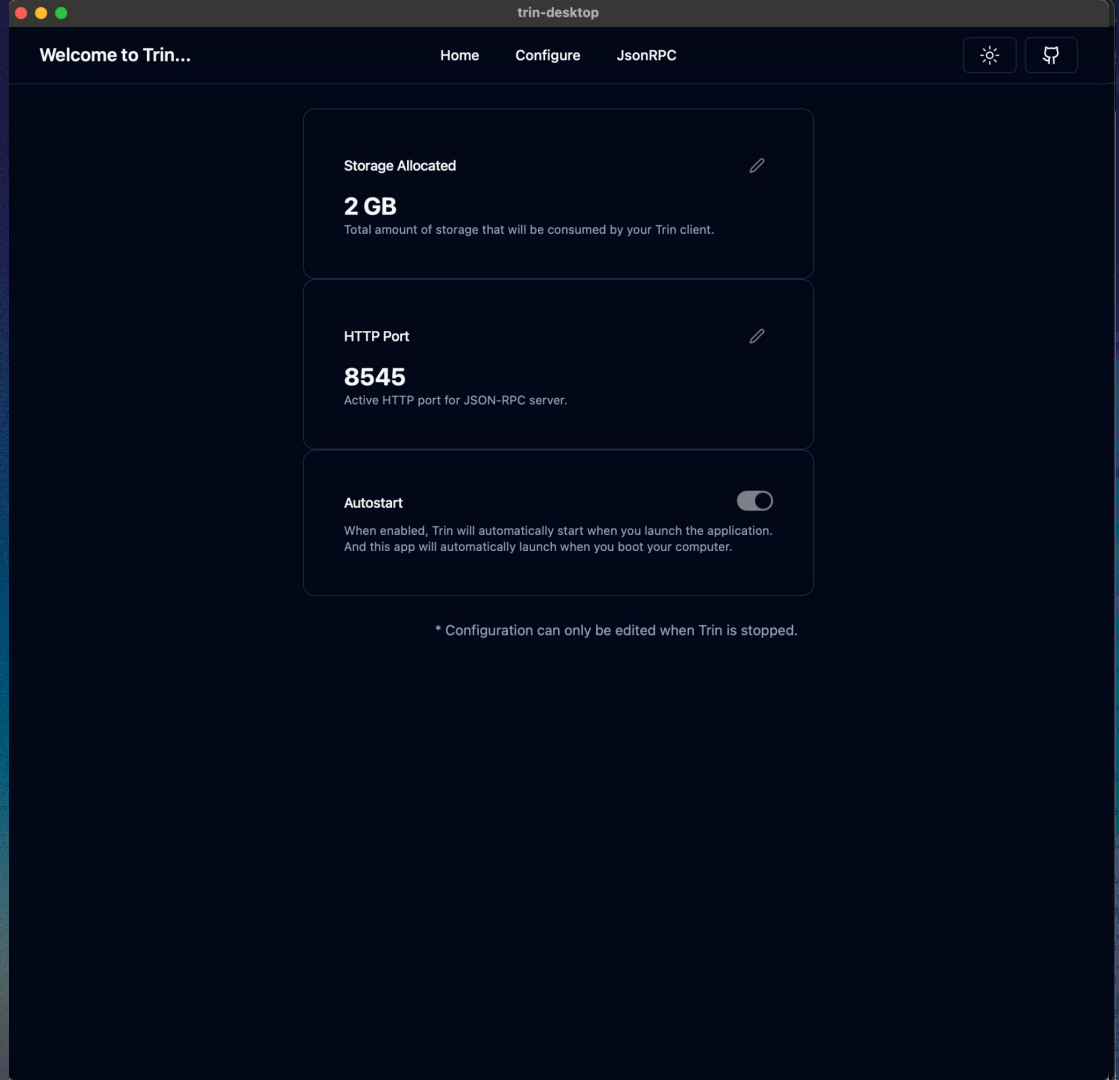
Account Balance:

4000 ETH

4e+21 GWEI

Raw: 0xd8d726b7177a800000

Trin Desktop Configure



Now, anybody can participate in the Ethereum Protocol.

Where to find out more

Website: https://www.ethportal.net/

Discord: https://discord.gg/5bWJ2tkns9

Twitter: @ethportalnet

Specs: https://github.com/ethereum/portal-network-specs

Discord Invite


Thank you!
Kolby ML (Moroz Liebl)
Software Engineer, Ethereum Foundation
kolby.morozliebl@ethereum.org
@Kolby_ML