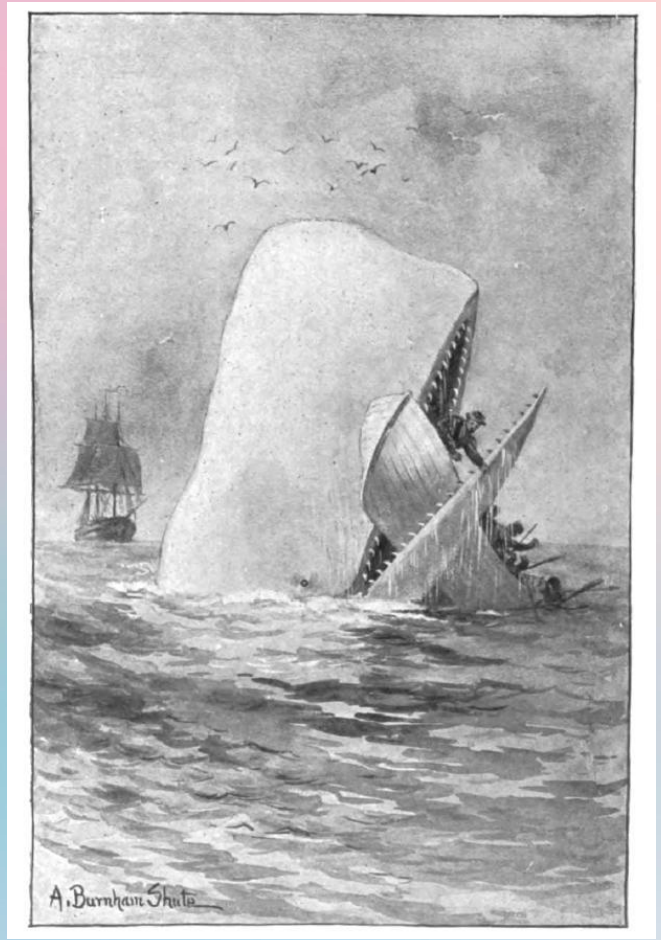
The Portal Network

Lightweight protocol access for everyone

Piper Merriam
EF

What is the Portal Network?


A. Burnham Shute

The Portal Network(s)

Beacon Light Client	State Network	Transaction Gossip
Beacon Chain Light Protocol data	Account and Contract Storage	Lightweight Mempool

History Network	Canonical Txn Index
Headers Block Bodies Receipts	TxHash > Hash, Index

Design Goals

Users

(as opposed to protocol)

decentralized

Portal Client

Full Node

light

heavy

Infura

centralized

Light

(Resource Constrained Devices)

What makes Ethereum clients heavy

EVM Execution

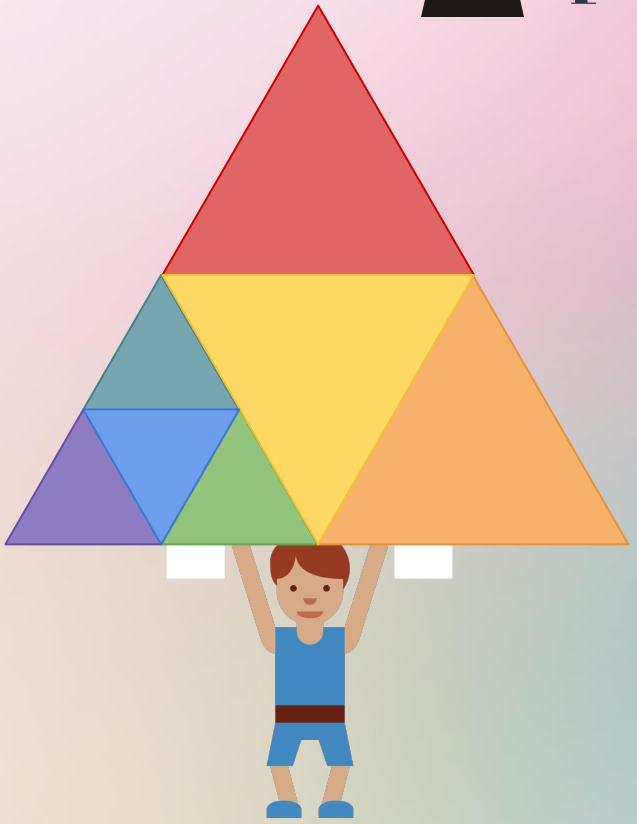
Transaction Pool

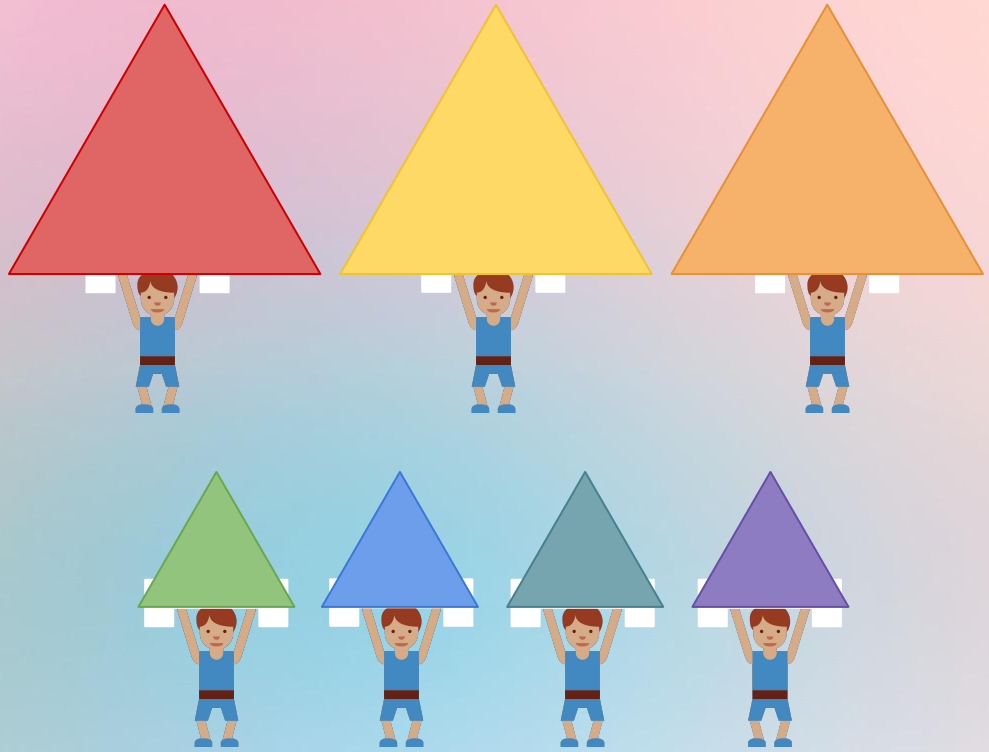
250GB of Chain History

50GB of Canonical Indices

175GB of State

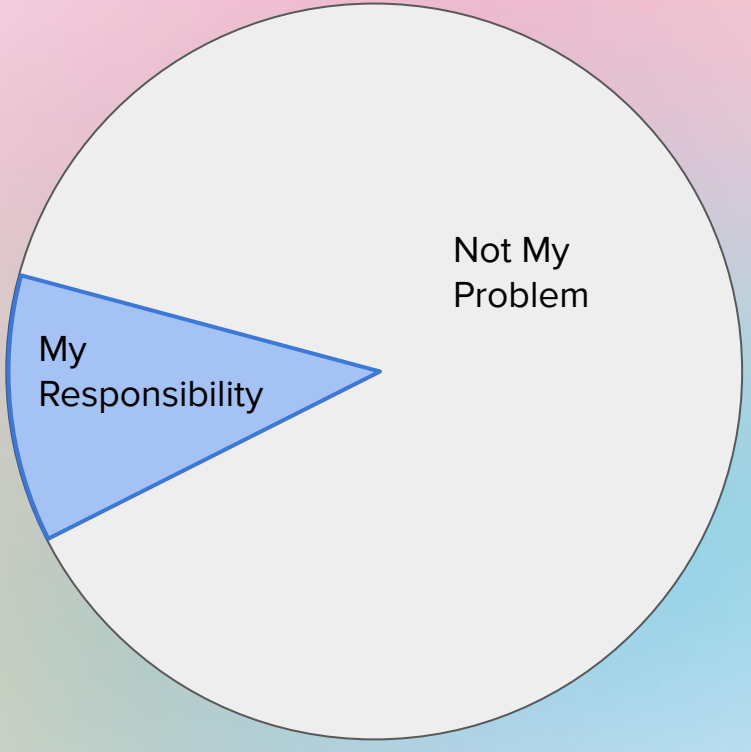
Spreading things out





No “height” restrictions

Contribute What You Can



No Sync

(Acceptable UX)

Syncing is bad UX

Hours of Syncing

175GB of State 1 TON

250GB of Chain History 1 TON

50GB of Canonical Indices 1 TON

Elimination of Syncing

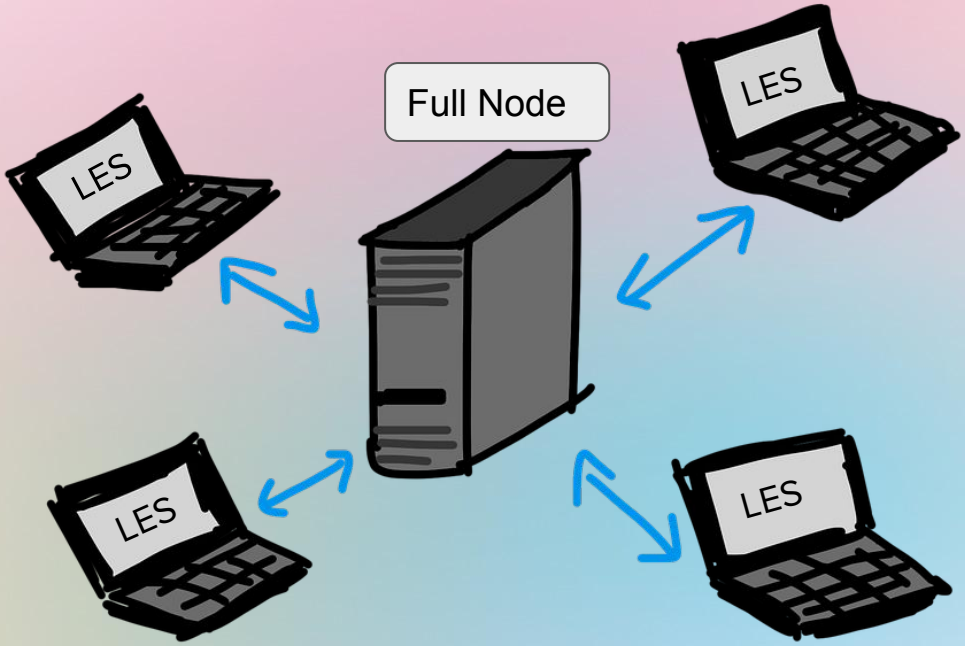
Acquire Tip of Header Chain

Scalable

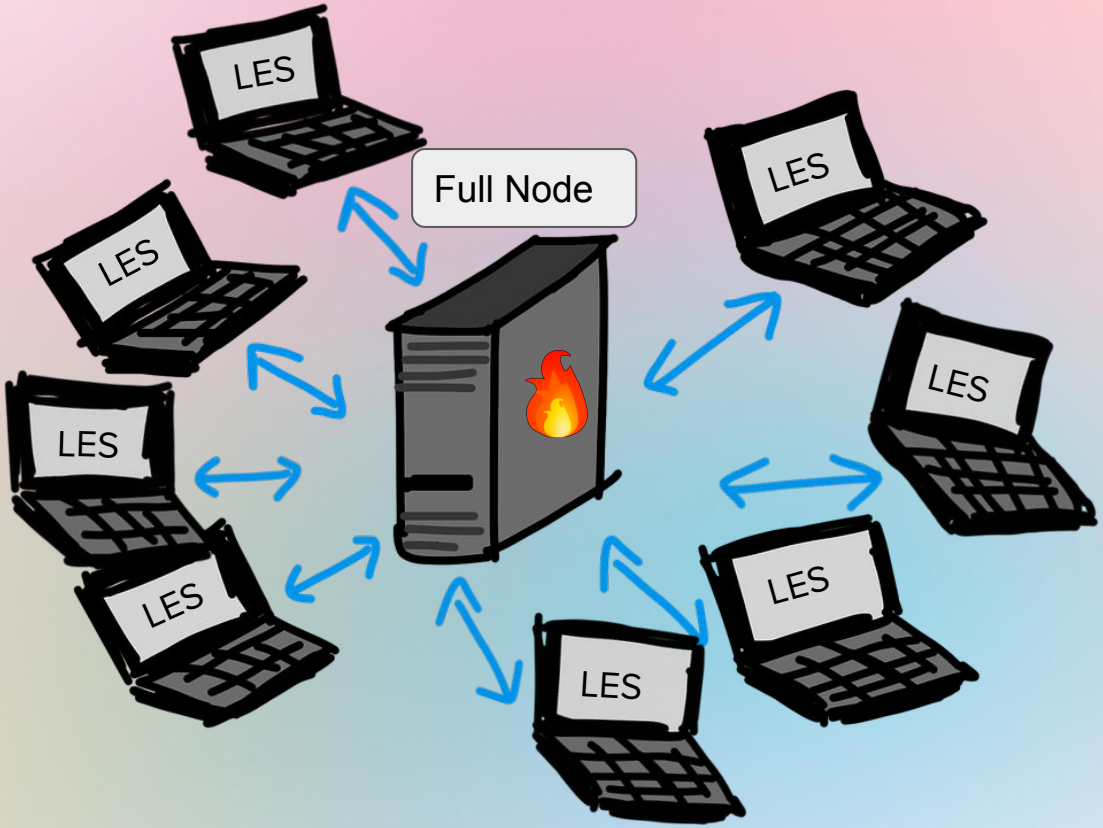
(Millions of participants)

Transactions
Per
Second

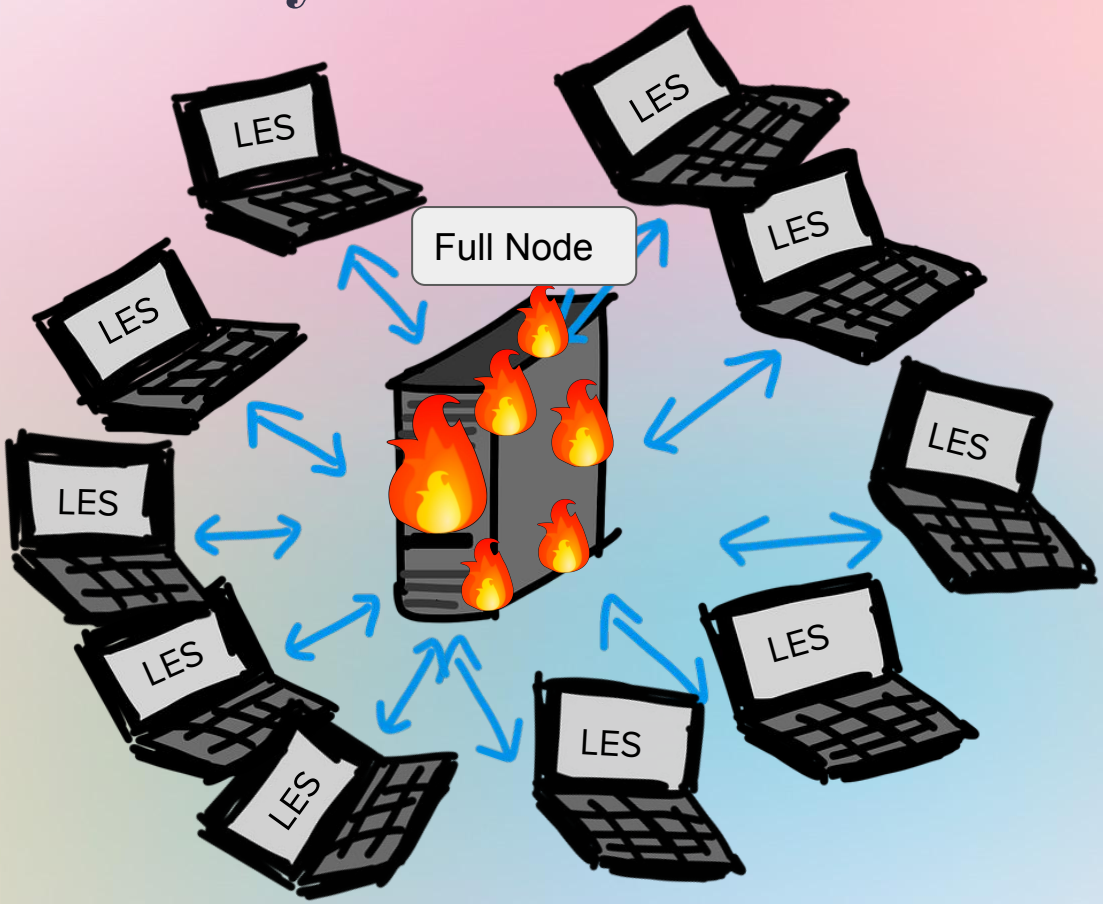
Why LES doesn't scale



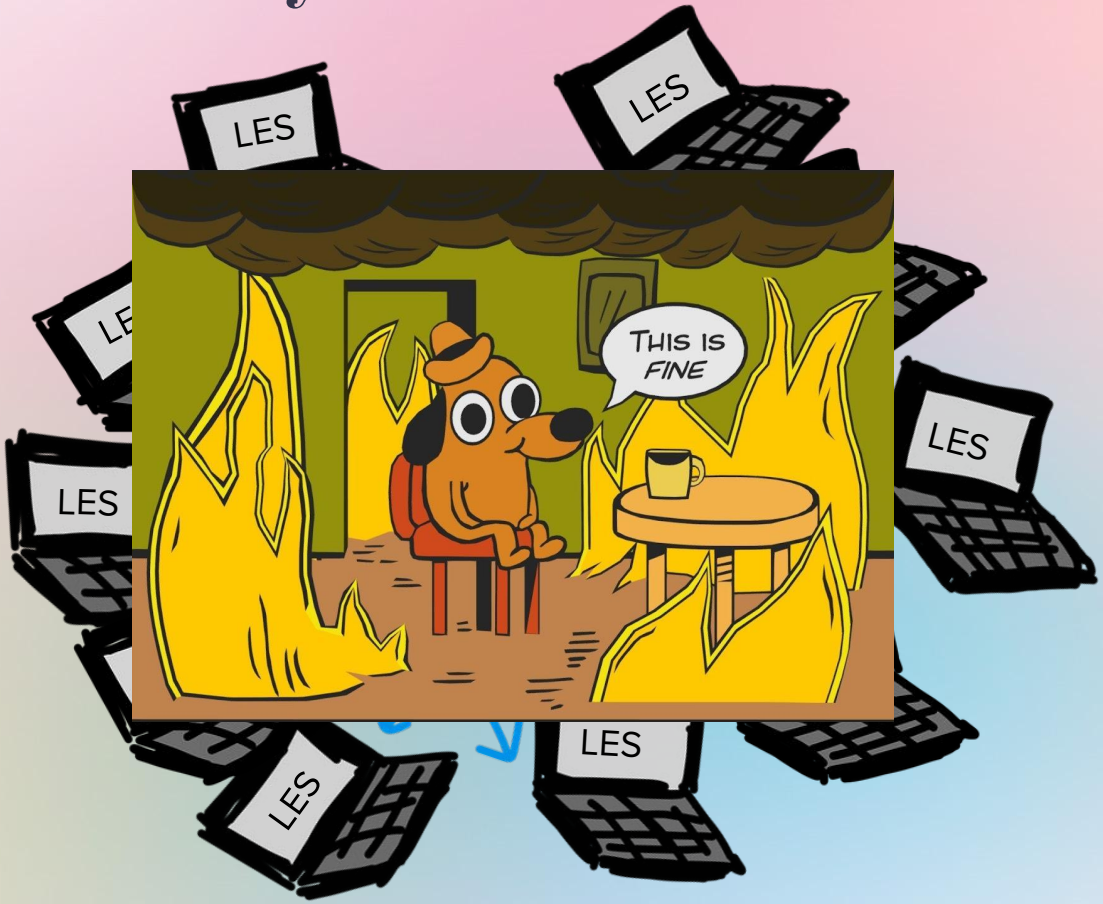
Why LES doesn't scale



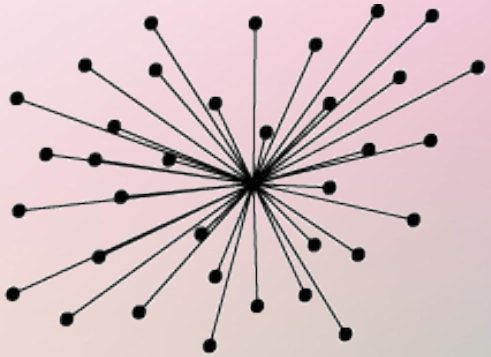
Why LES doesn't scale



Why LES doesn't scale

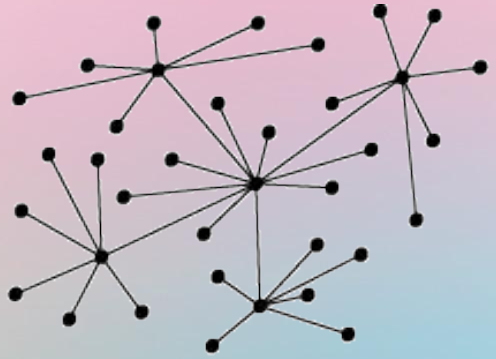


Scalable Lightweight Protocol Access



centralised

Infura



decentralised

LES

distributed

Portal

Kind of like...

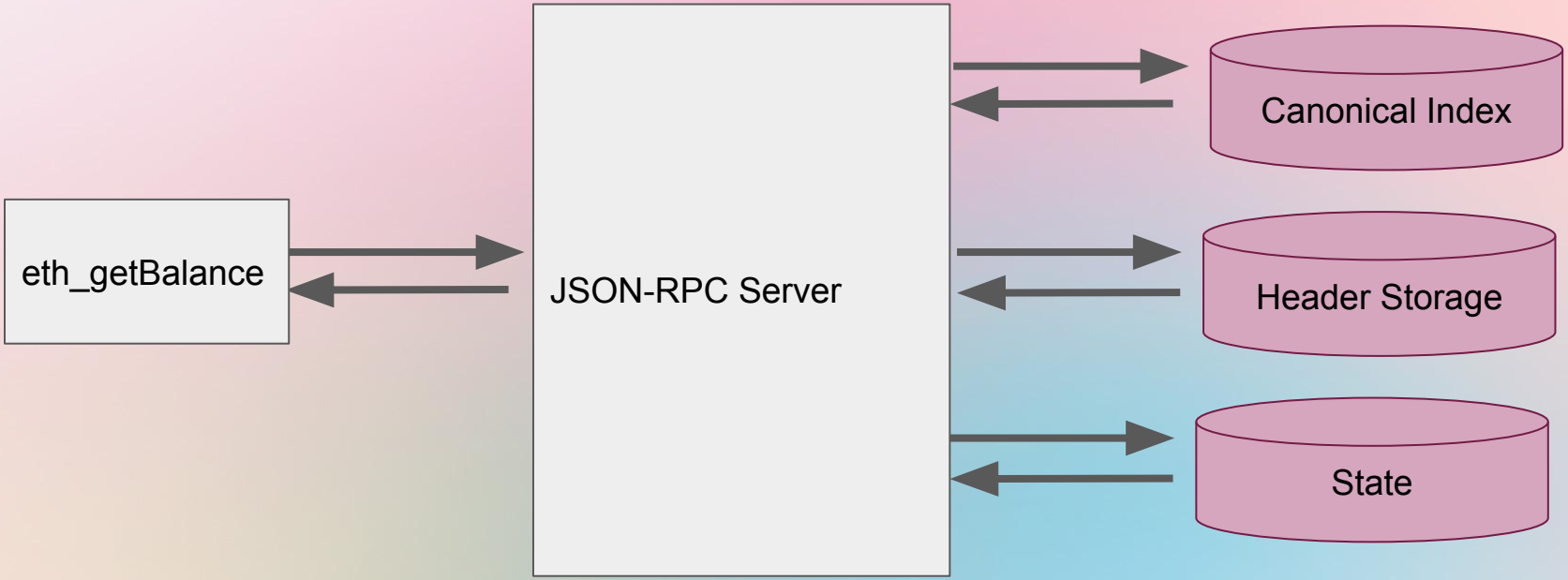
A Practical Example

The Portal Network(s)

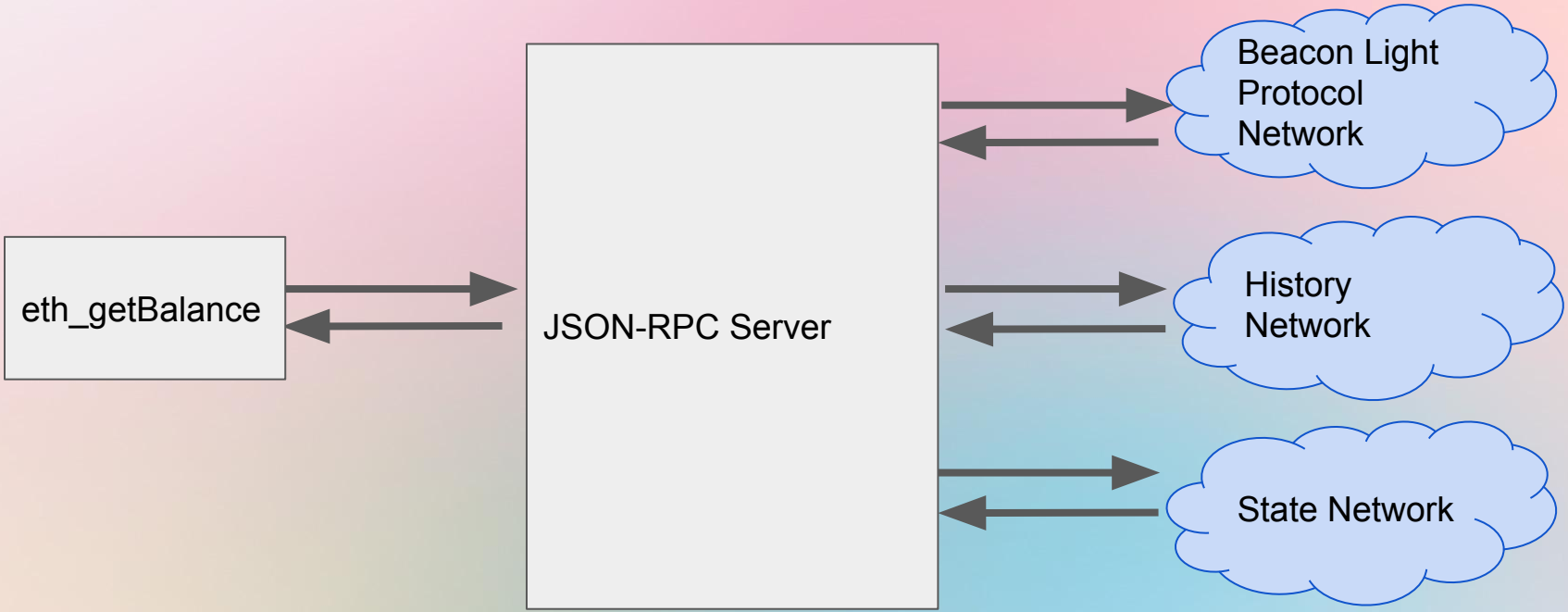
Beacon Light Client	State Network	Transaction Gossip
Beacon Chain Light Protocol data	Account and Contract Storage	Lightweight Mempool

History Network	Canonical Txn Index
Headers Block Bodies Receipts	TxHash > Hash, Index

eth_getBalance

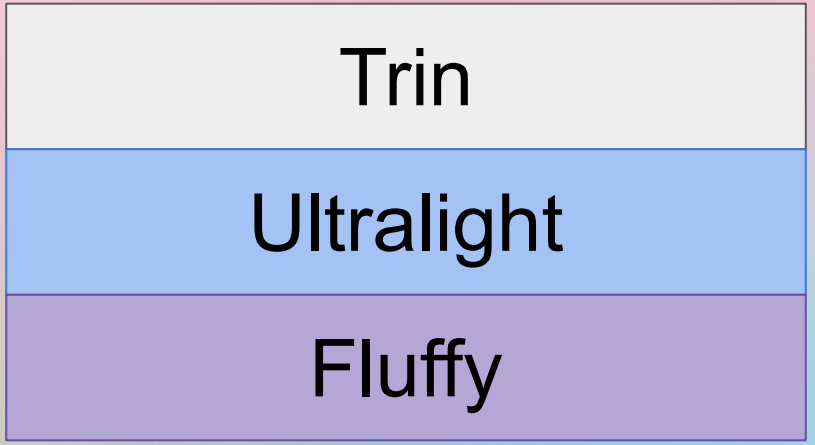


eth_getBalance



Project Status

Three Portal Client Implementations



Rough timeline

Imminent

Fully Operational History Network

Next

Beacon Light Network

And Then...

State & Transaction Gossip Networks

Fin

Questions?