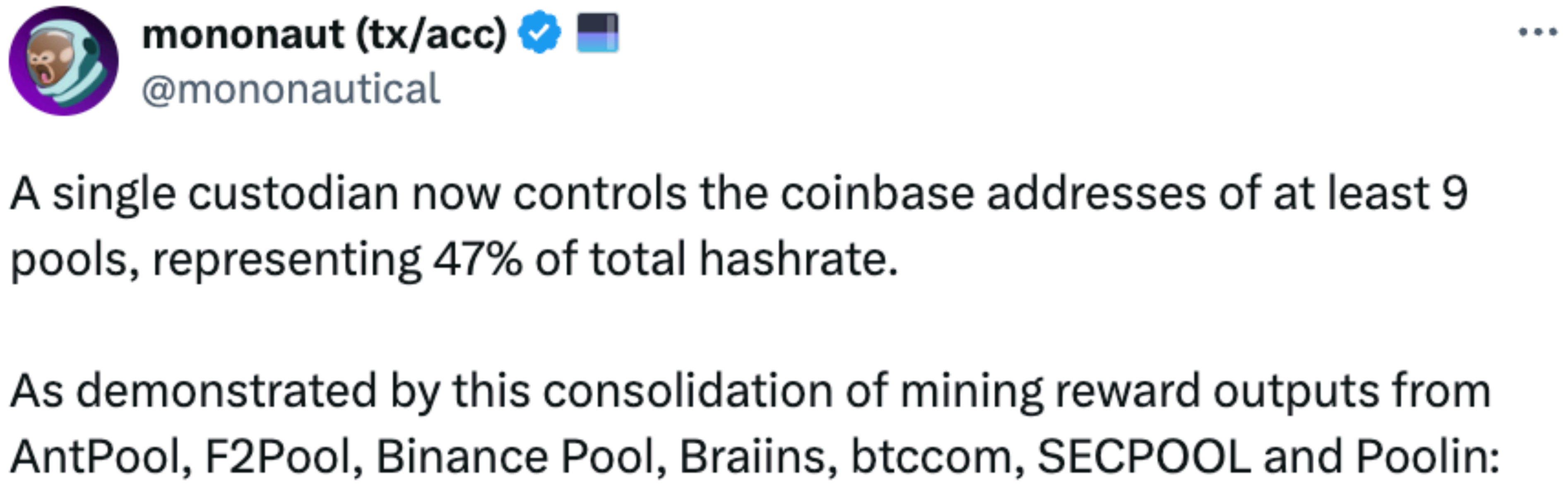
Hashpools

A New Kind of Mining Pool Powered by Ecash

vnprc / Oct 2024

The State of Bitcoin Mining

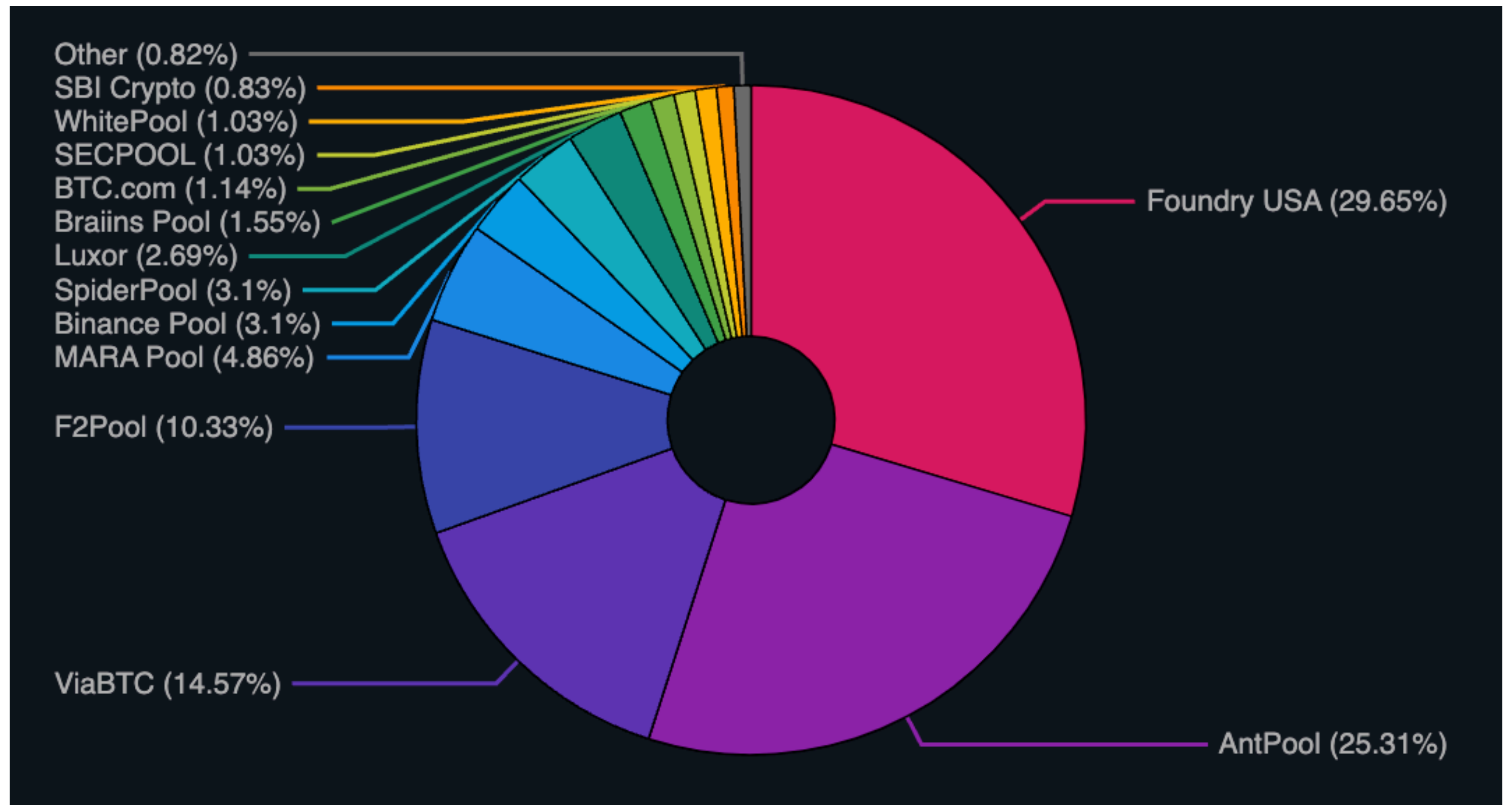
- 80% of blocks mined by 4 pools
- 47% of mining rewards by hashrate flow to the same custodian
- 37.6% of hashrate is mining on the same block templates
- Bitcoin mining is under significant pressure to centralize



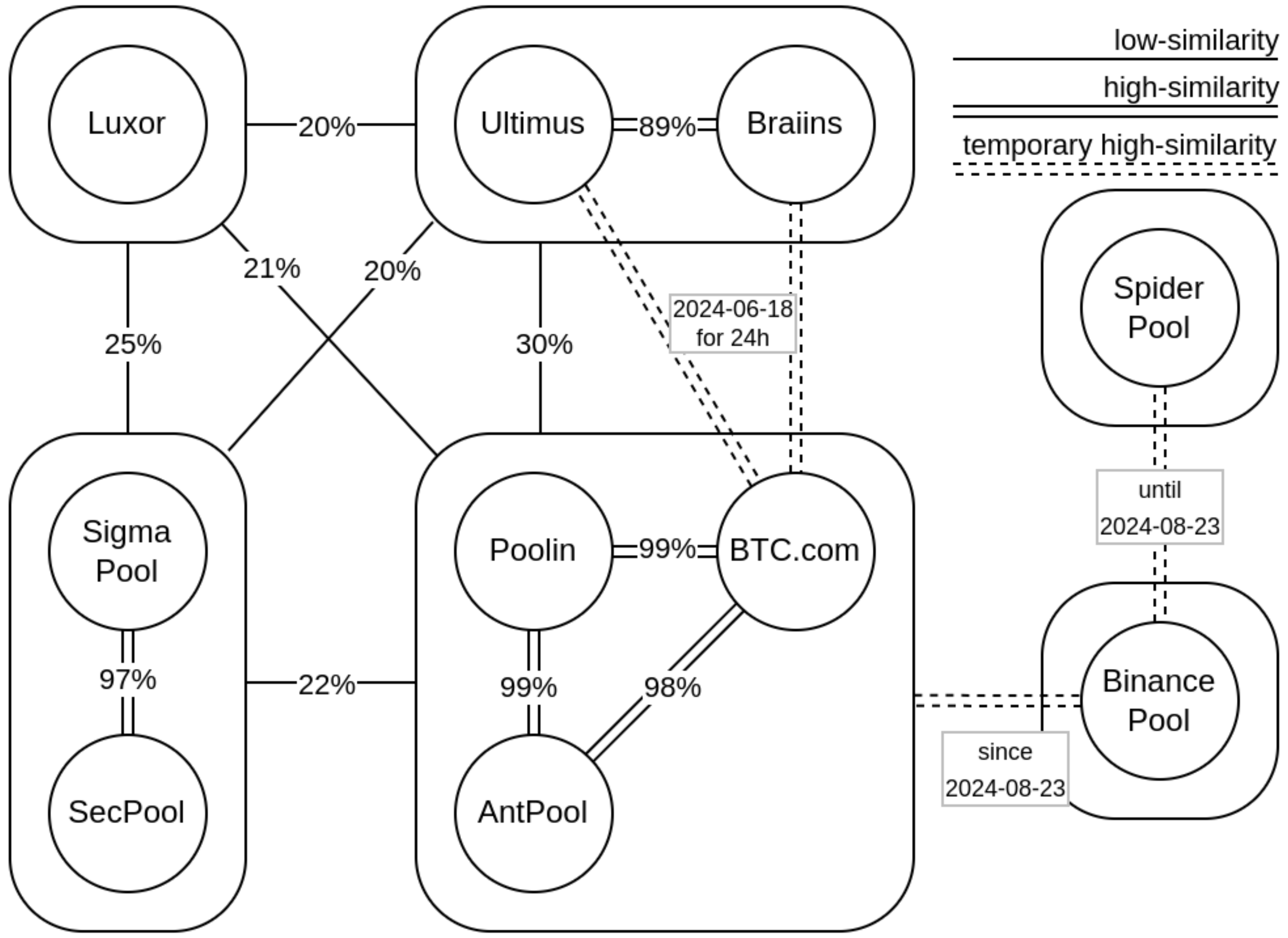
mononaut (tx/acc) @mononautical

A single custodian now controls the coinbase addresses of at least 9 pools, representing 47% of total hashrate.

As demonstrated by this consolidation of mining reward outputs from AntPool, F2Pool, Binance Pool, Braiins, btccom, SECPOOL and Poolin:



Pool Relationship Graph



Why is this a Problem?

- Pools control block template production
  - They can censor transactions
- Pools control all newly mined bitcoin
  - UTXOs with no history are a pristine privacy asset
  - Privacy properties are wasted in custodial mining pool wallets
- Pools have outsized influence in bitcoin consensus changes
  - Block producers ‘vote’ to activate soft forks
  - Mining pools can ‘spike the ball’ during a soft fork activation

The State of Bitcoin Software

- **Leaves : UTXOs**
  - Wallet software
    - Plurality of Options
    - Some are Free, Open Source, & Self-hostable
- **Trunk : Blockchain**
  - Node software
    - One Option: bitcoind
    - Free, Open Source, & Self-hostable
- **Roots : Miners**
  - Hardware & Firmware
    - Limited Options
    - FOSS miners are coming
    - Thank you Bitaxe & Block! 🙏
  - Mining Pools
    - Plurality of Options
    - None are FOSS & Self-hostable
    - SRI is coming

there's your problem

Mining Pools are Inefficient Markets

Because Hashrate is not Fungible

- Modern mining requires a pool account
  - The pool keeps track of all your shares and uses them to calculate payouts
  - Shares are non-transferrable
  - Payout thresholds limit the minimum practical hashrate
- Selling hashrate requires reconfiguring your miners
  - Directly or through a proxy
  - Significant technical barrier for most miners
- Few marketplaces to choose from
  - Nicehash…and that's it!
- Abundant regulatory hurdles
  - Who regulates hashrate?
    - Is it a security?
    - Is it a commodity?
    - Is it a derivative?
  - How is it taxed?
- We will not get clarity on these questions because the state is actively hostile towards bitcoin
  - Modern governments are funded by seignorage and run by cantillionaires
  - They will not provide solutions, only obstacles
- We have to solve our own problems

Cypherpunks Fix This

With Open Source Software

- Stratum v2 enables:
  - FOSS pool & proxy software
  - Block template selection
- Ecash enables:
  - Accountless mining pools
  - Tradable mining shares
- Nostr enables:
  - Decentralized identity
  - Mint discovery
- With our powers combined:
  - Private, free, & unstoppable bitcoin hashrate markets

How Does it Work?

The Details

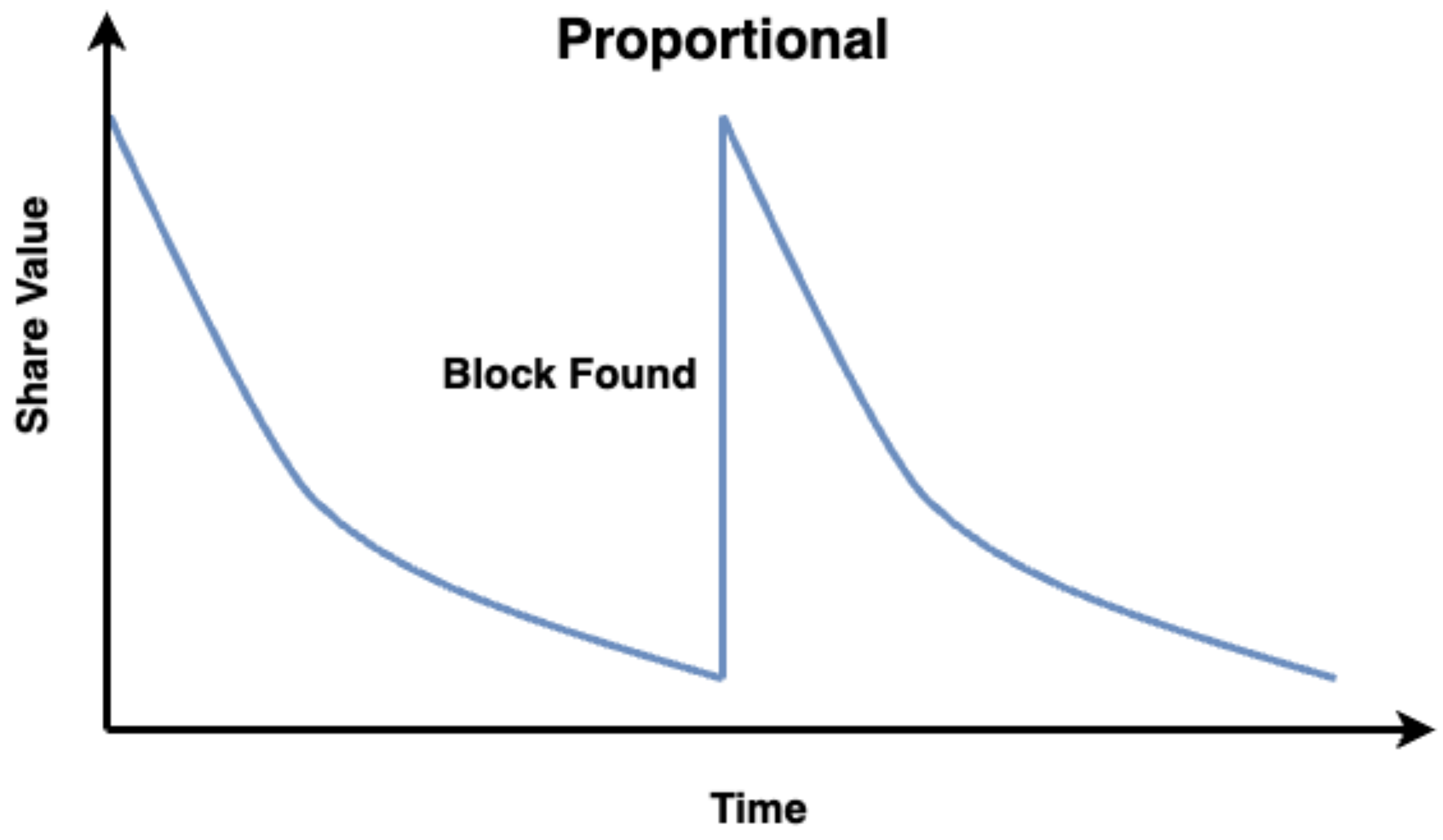
- Pool issues an ecash token for each mining share
    - Let’s call it ehash
- Use ecash epochs to approximate the age of shares
    - Can’t expire individual ehash tokens
    - But we can use key rotation to create time buckets & expire those
- Publish Verification Proofs for each ecash epoch and each block found
    - Ecash proof of liabilities
    - Hash proofs: block templates, headers, & nonces
- Use PPLNS payouts

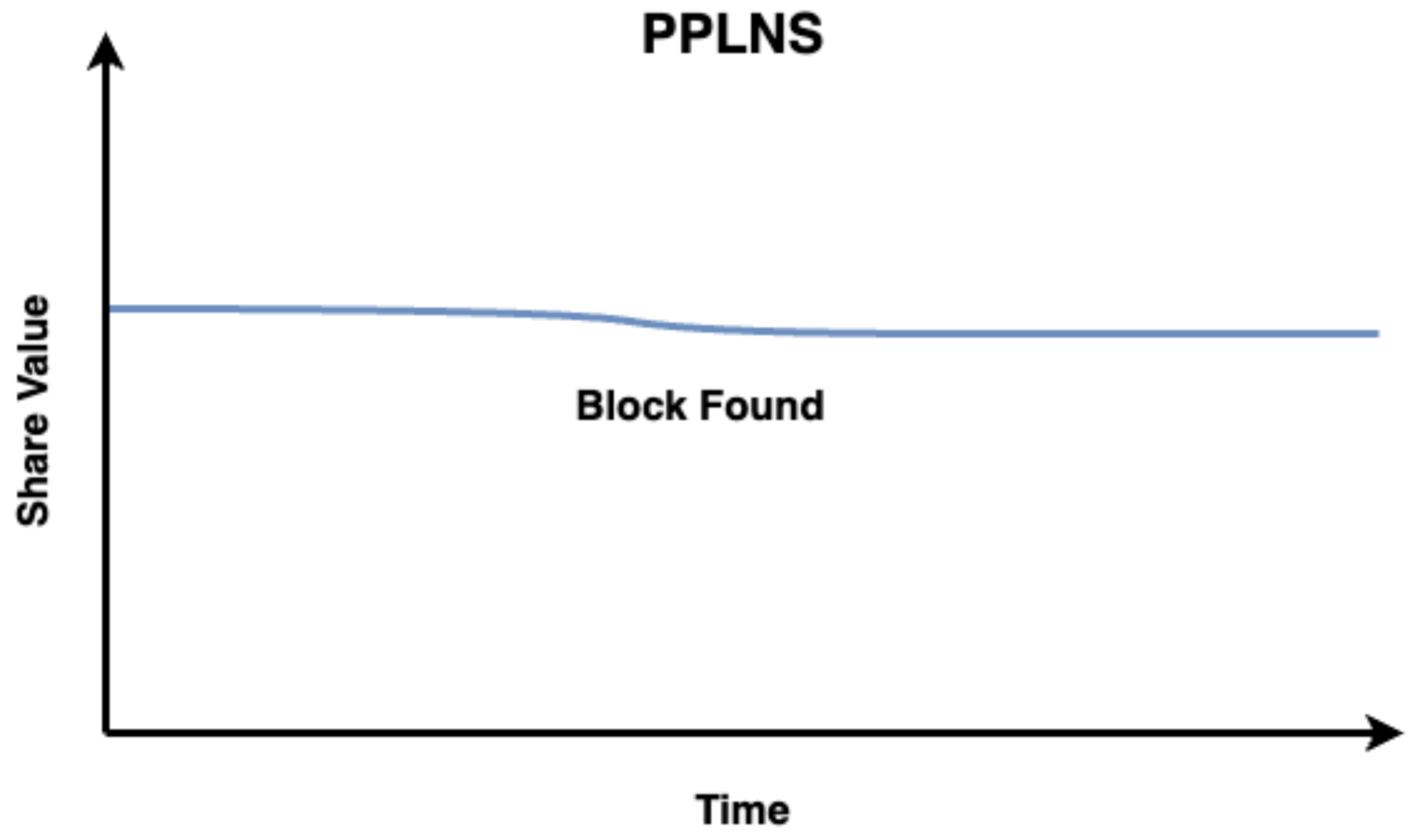
- PPLNS = Pay Per Last N Shares
    - Earliest form of pooled mining
    - Miners get paid directly from the block reward
        - Simple in concept, less simple in execution
- PPLNS features
    - Miner assumes all payout variance
    - Pool doesn’t assume any stochastic risk
    - Pool doesn’t require up front capital
- Today, all major mining pools use FPPS
    - Pro: More stable payouts for miners
    - Con: Not verifiable
    - Con: Not trustless
    - Con: Creates centralization pressure

PPLNS

How and Why

- Proportional:
  - Pay all shares since the previous block
  - Problem: share dilution—later shares are worth less than early shares
  - Result: miners arbitrage pools, aka “pool hop”, to maximize expected payout
- PPLNS
  - Amortize share payouts over a time window
  - Shares can pay out multiple times (or 0 times)
  - Share value calculated each time a block is found
  - Projected share payouts decrease over time as the window closes

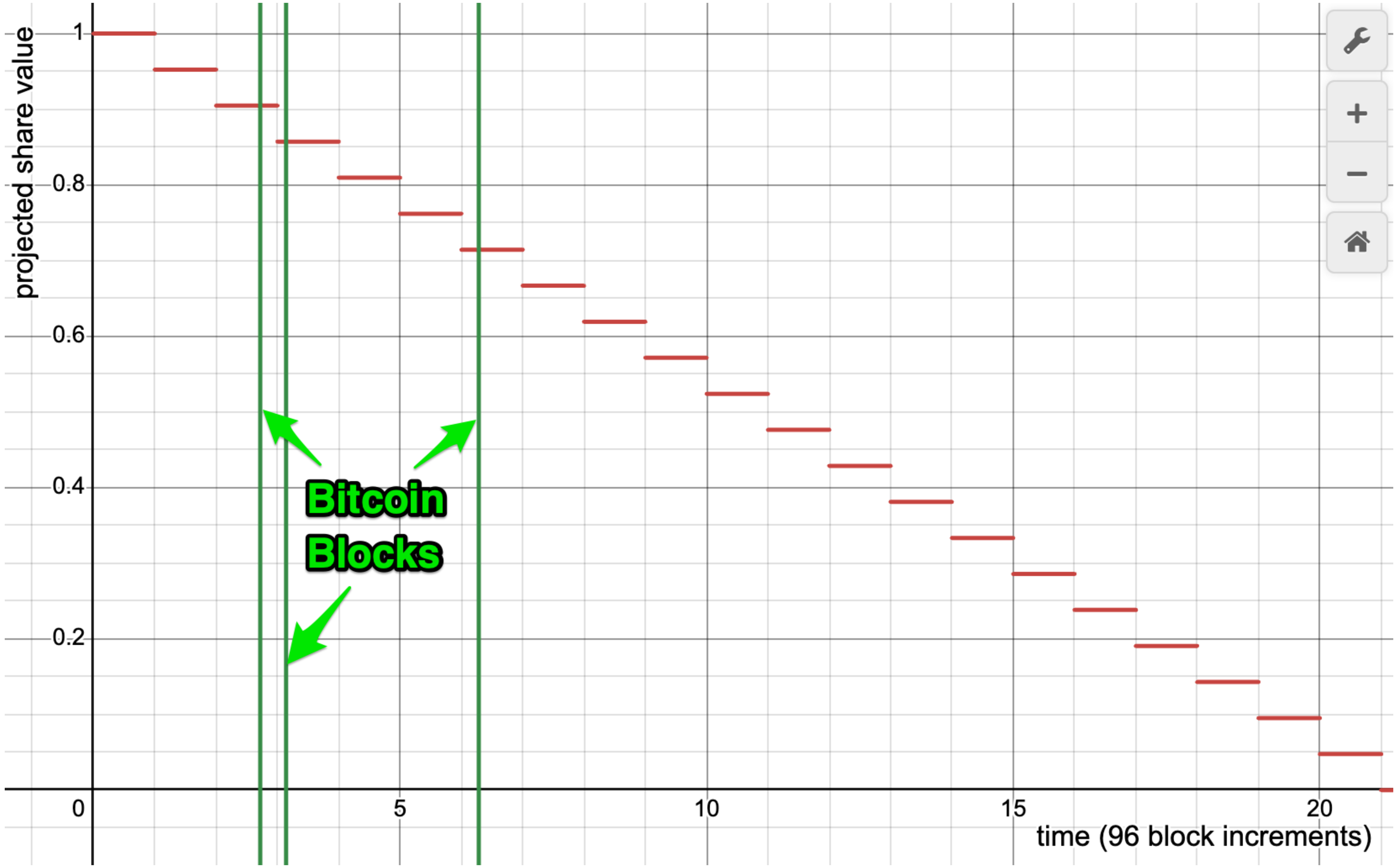




Hashpool Share Value

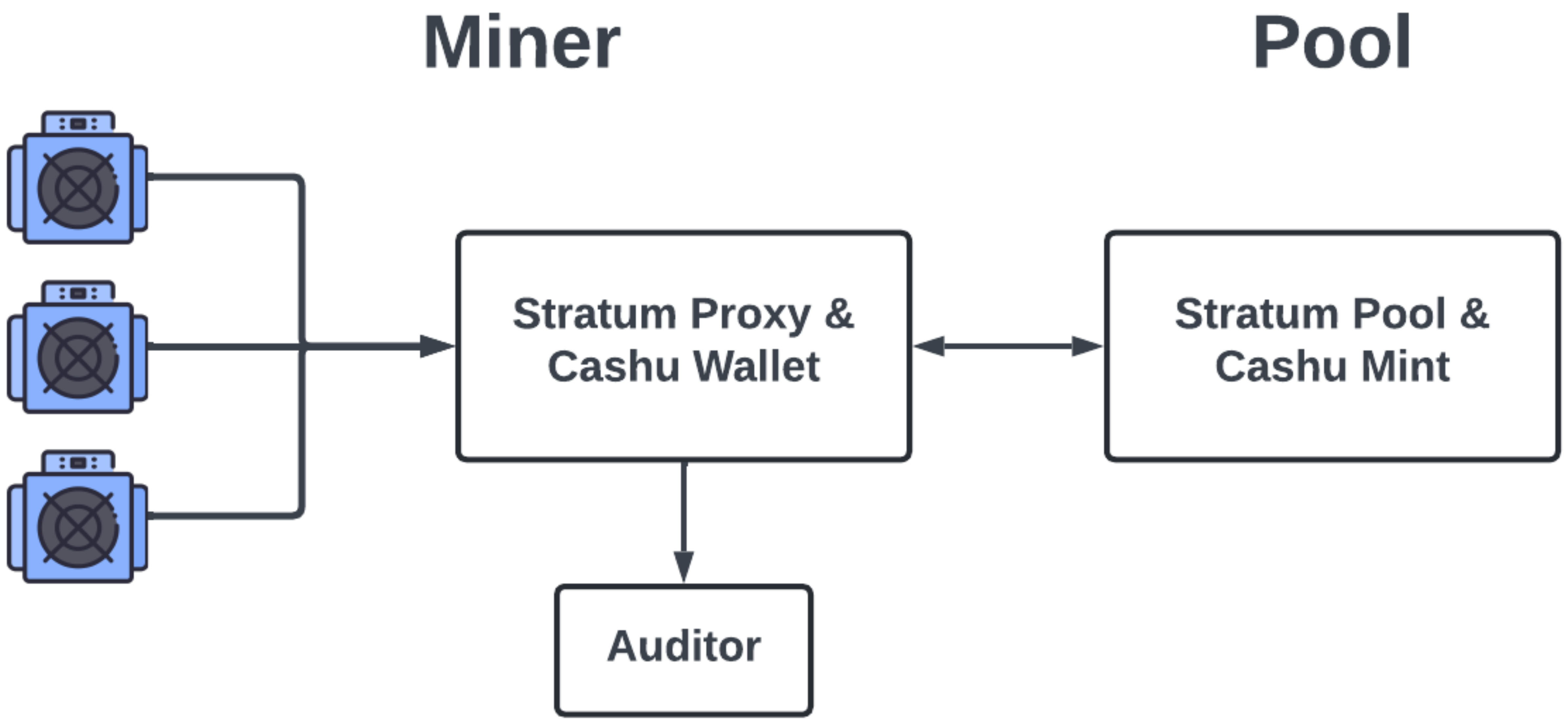
Example Payout Schedule

- Time periods approximate share order
  - Enforced by mint keyset rotation
- Projected share payout value decays over time as the share window closes
- Shares are paid for each block the pool finds in the maturity period
- Keyset ID is committed to in the coinbase
  - Prevents miners from submitting shares to multiple pools
  - After rotation, commit to the new keyset in a new block template



Hashpool Architecture

- Miner side:
  - Sv2 translator proxy
    - Built in cashu wallet
  - Sv2 extension: Auditor role
- Pool side:
  - Sv2 pool role
    - Built in cashu mint

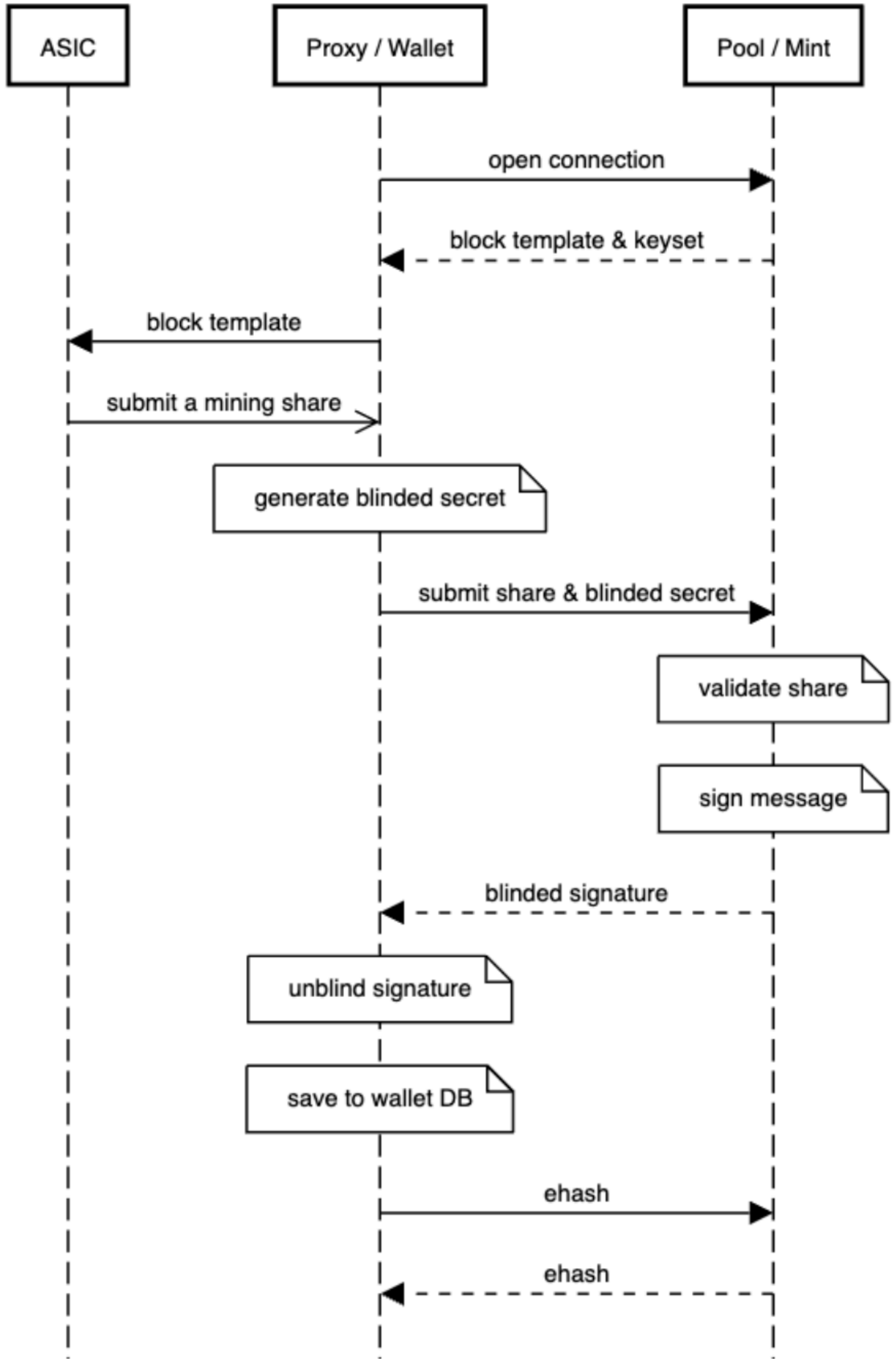


How Does It Work?

Hashpool has 3 Critical Flows:

1. **Issuance**
    - Pool accepts a mining share and returns an ehash token
    - Miner stores ehash in wallet
    - Miner saves all data: ecash proofs and mining shares
    - While the share payout window is open the pool performs ehash swaps
        - Swaps are needed to limit data storage & bandwidth spikes
            - “Roll up” small denominations into larger denominations
        - Also enables trading
        - Close ehash swaps when shares mature
            - Once their bitcoin value is fully determined

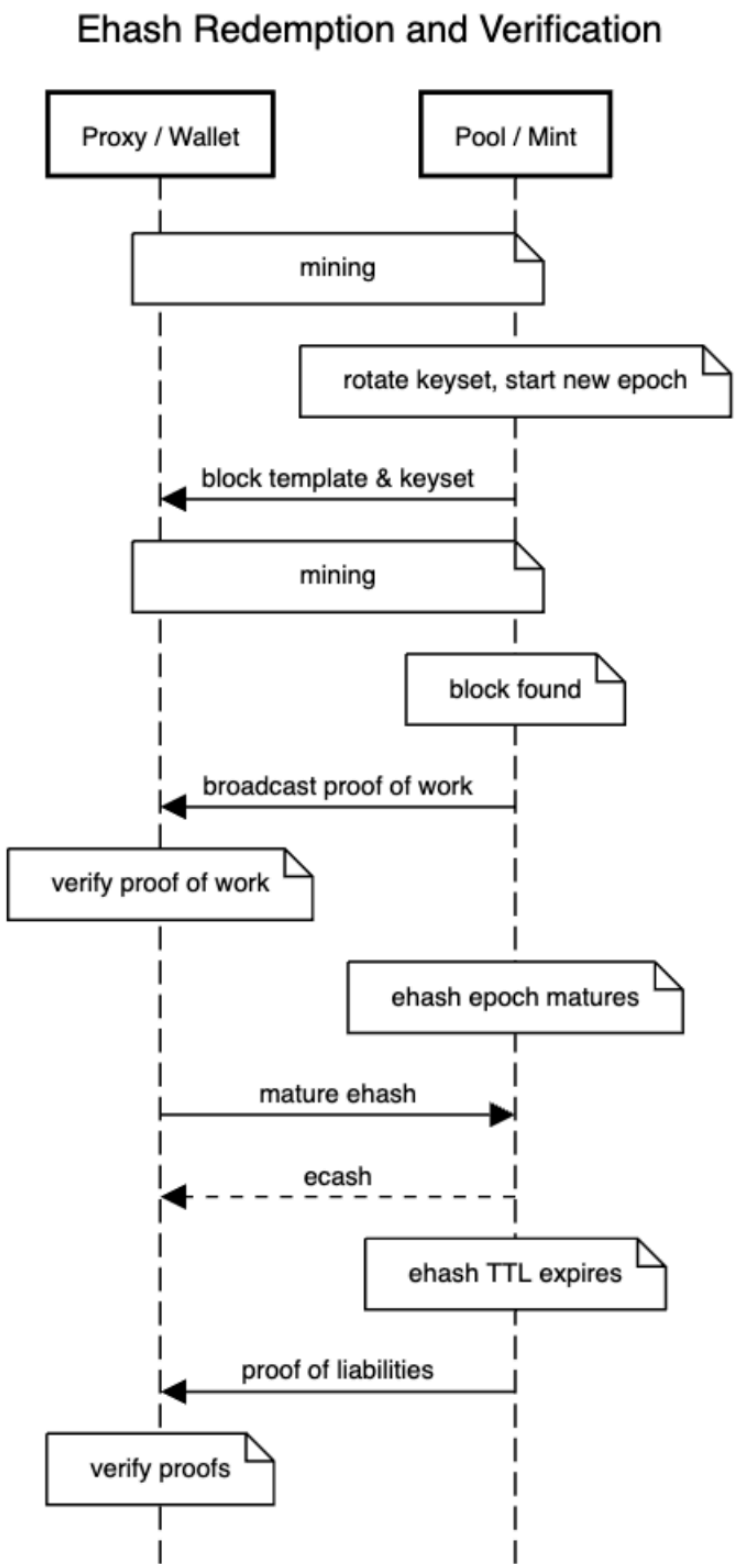
Ehash Issuance



How Does It Work?

Hashpool has 3 Critical Flows:

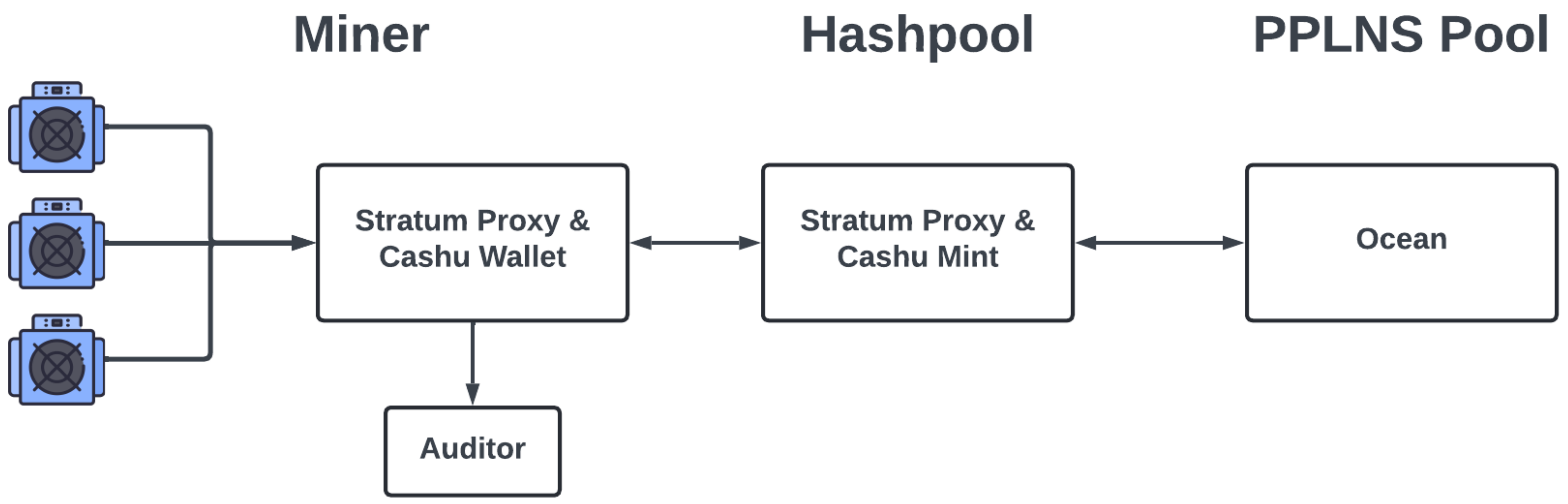
2. **Redemption**
    - Ehash is typically redeemed after maturity
        - Mature ehash has a redemption window or TTL
        - If not redeemed in time the pool/mint can claim rewards
    - Ehash can only be paid out once
        - Early redemption forfeits potential future rewards
        - Forfeited rewards distributed to other ehash holders
3. **Verification**
    - For each block the pool finds, publish all hash proofs
        - Block templates, headers, & nonce data for each share
    - Each time an ehash redemption window closes, publish all data
        - Merkle sum tree of issued and redeemed ecash tokens



Bootstrapping a Mining Pool is Really Hard

Can We Make it Easier?

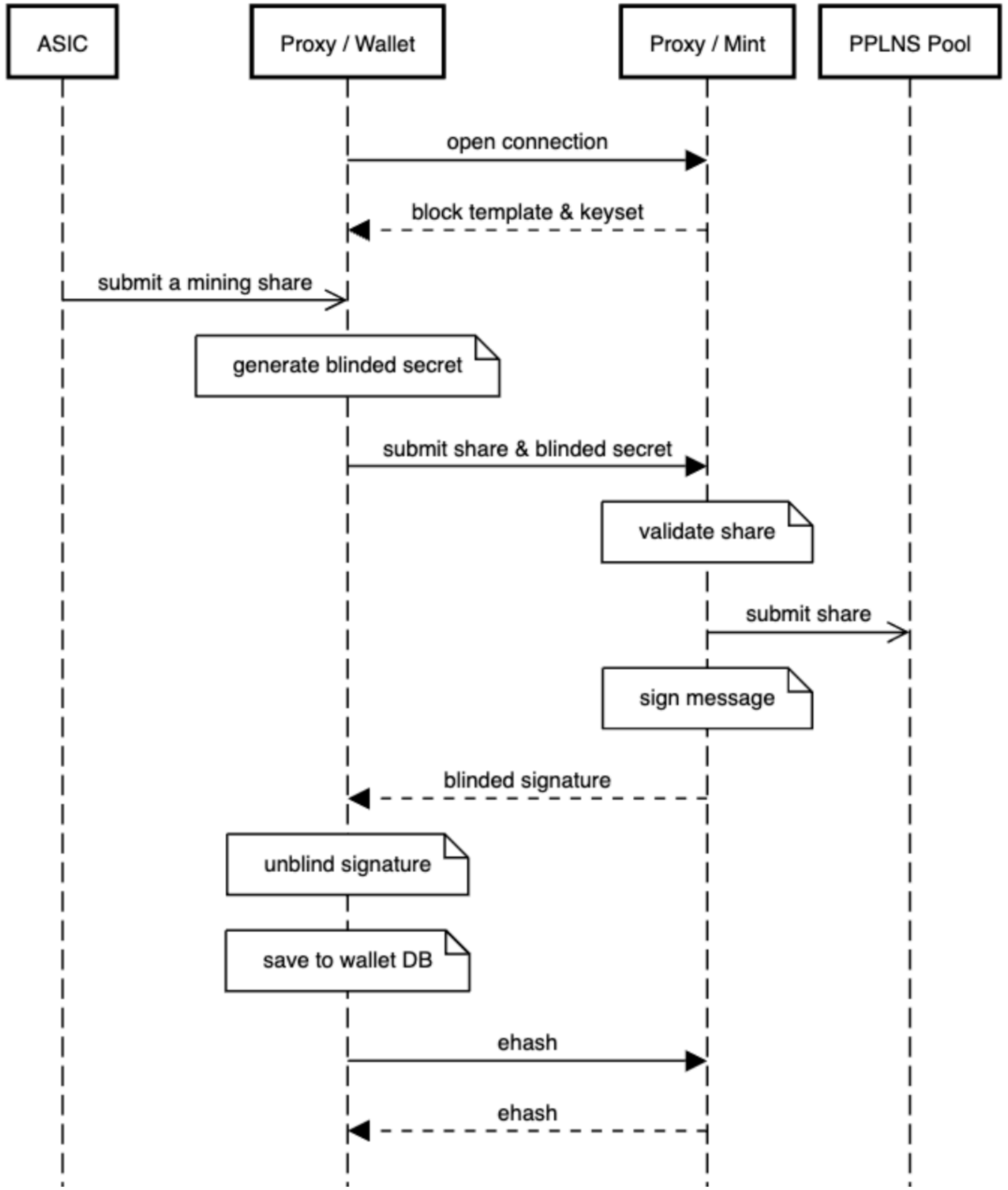
Passthrough Hashpool



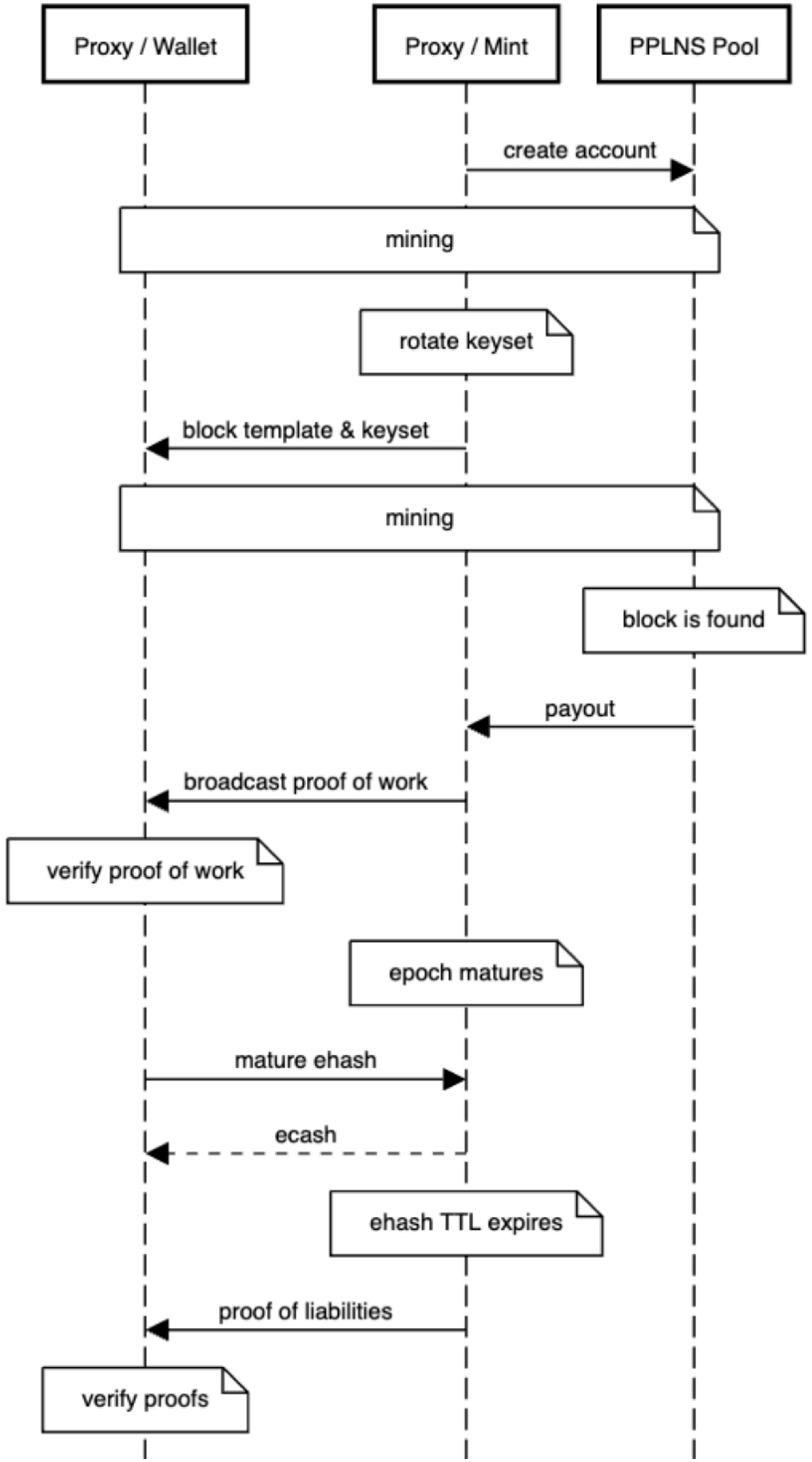
Yes.

Passthrough Hashpool

Passthrough Hashpool Issuance



Passthrough Hashpool Redemption and Verification



Use Cases

- For miners:
  - Mine privately
  - No hashrate minimum
  - Immediate liquidity, for a price
  - DIY mining pool
- For large pools:
  - Simulated FPPS
    - Pool buys ehash for a fixed price
    - Great for privacy!
  - Hashrate verification
  - Escape the mining pool monopoly trap

- For savers:
  - Buy future sats at a discount
    - Trade liquidity and risk for expected profit
  - Buy non-custodial coinbase outputs
    - Great for privacy!
    - Great for innovation!
- For all bitcoiners
  - A stronger, more decentralized, more resilient, and more secure foundation to enable the best and most free (as in speech) form of money ever invented to thrive for generations to come.

Future Work

Lots to Build!

- Payout calculations
  - Still a work in progress
  - Can we disincentivize block withholding attacks with a high difficulty share bonus?
- Ehash marketplace
  - Ehash is the first ecash asset not pegged to a currency
  - We need a market for price discovery
  - Atomic swaps
    - Ecash, lightning, stablecoins
  - Prioritize privacy & security
- Verification
  - Ecash and mining share proofs
- Coinbase payouts
  - Use NUT-11 to lock ehash to a pubkey
  - Include the pubkey as an address in the coinbase output
- Pay for template selection
  - Hashpool transaction accelerator
- Fedimint module
  - Multisig improves the security model
  - Can’t support issuance…yet
  - But we can build fedimint redemptions today

vnprc

nostr: vnprc@trianglebitdevs.org

github: github.com/vnprc

bitdevs: trianglebitdevs.org