Pareme: The Proof Of Work Blockchain

Alejandro Ochoa
www.pareme.org

Abstract. We propose Pareme, a lightweight blockchain secured by proof of work. It separates application logic from verification. Services and applications can outsource verification to Pareme. Each block in Pareme contains proof-of-work data and a custom hash chosen by the miner. No native tokens or coins exist. No public/private keys or block ownership is required. This design makes Pareme the lightest blockchain in use. Without block ownership, targeted attacks diminish, positioning Pareme as highly decentralized. Miners select the hash, representing any content they choose. We outline potential use cases in a later section.

Problem
Bitcoin has been repurposed for data storage due to its security, yet it is not designed for this. Other blockchains supporting data or smart contracts become bloated quickly. We address a core issue: how to secure content on an open platform without excessive size from mass data?

Solution
A hash uniquely maps to its content, reducing any data size to a fixed length. Storage and distribution become simple. It should be noted that content cannot be reconstructed from its hash alone. This is a feature, not a flaw. It ensures privacy and enables Pareme's key function: verification. We elaborate below.

A Technical Overview

Pareme is compact, unlike blockchains handling currency or smart contracts. Each block carries 80 bytes:

Proof Of Work Data
- **Height**: 4 bytes
- **Timestamp**: 8 bytes
- **Previous Block Hash**: 32 bytes
- **Nonce**: 4 bytes

Payload
- **Content Hash**: 32 bytes

Blocks are targeted at 10-second intervals, resulting in approximately 3 million blocks per year, or roughly 3 GB of data per decade.

Proof of Work & Difficulty Adjustment
The difficulty value validates blocks and adjusts every 2016 blocks. A miner constructs a potential block with their chosen data, then hashes it. If the block hash is less than the current difficulty value, the block is valid. If not, the miner modifies the nonce field to generate a new hash, repeating the process until a valid result is achieved.

The difficulty is calculated using the following formula:
New Difficulty = (Old Difficulty) * (Ratio)
Where **Ratio = (Average time between the last 2016 blocks in seconds) / (10 seconds)**

Additional verification steps include:

- **Height and Previous Block Checks:** These ensure the referenced previous block is valid and the height value has incremented by exactly 1.

- **Timestamp Checks:** These maintain timekeeping integrity. As users select their own timestamps across global time zones and face propagation latency, sanity checks prevent timestamps from being too far in the future or past.

Full Nodes & Pruning
By design, Pareme's chain is lightweight, with 80-byte blocks imposing negligible bandwidth demands. However, users may opt for a pruned version of the chain, where older blocks are discarded. Since each block references its predecessor, extremely old blocks are not required to verify new ones. At minimum, a node needs the last 2,016 blocks (~160 KB) to confirm a new block's difficulty and perform other verification steps.

While pruning is viable, retaining the full historical chain offers advantages. Certain use cases, detailed later, may require analyzing the frequency of a specific hash across the entire chain. Nodes interested in such applications would thus benefit from maintaining the complete record. For basic verification and mining, however, a pruned node suffices.

A Theoretical Overview

Mining Motivation
Pareme provides no native tokens or rewards. Instead, users mine to embed a chosen hash into the chain, serving their own purposes and creating an inherent incentive. Third-party applications can leverage this lightweight chain, requiring on-chain verification for access. To participate in such platforms, users must mine their hashes, either directly with their own hardware or by paying others for hash power. A clear reason to do this may be straightforward bot protection. The chain grows only as users perceive value in securing their hashes.

Security Analysis
Without rewards, Pareme relies on the personal incentives of each user. An attacker seeking to alter a hash must outpace honest users in proof of work (See the Bitcoin whitepaper to understand Proof Of Work security). Since the attacker does not receive block rewards directly, their attack itself will be very costly to complete and increases as the proof-of-work race continues. The chain's lightweight design lowers operating costs, encouraging participation. Compared to chains like Bitcoin, the likelihood of solo mining a block is dramatically increased due to the fast block times.

Comparisons to Existing Systems
Bitcoin stores data via its transactions, bloating its chain for a purpose it does not fit—many systems exploit its security to embed hashes, swelling it further. Ethereum and similar platforms carry smart contracts, adding complexity and size. Pareme strips these away, using proof of work only for hash verification. Its 80-byte blocks contrast with Bitcoin's megabyte-scale blocks, offering a leaner alternative for content validation. Pareme alleviates this burden on Bitcoin, providing a new location for hash-based systems.

Use Cases

Access to Third Party Systems
As previously noted, third-party systems can implement a verification mechanism for users by requiring their credentials to be hashed on-chain. This approach can serve as a simple defense against bots, as each connecting bots must be verified on Pareme's blockchain

Document Authenticity
Consider a document of disputed origin. Users mine the hash of the version they deem authentic. Observers tally blocks associated with each hash to identify the most trusted variant. For instance, Bitcoin users could signal their preferred Bitcoin software version by mining its hash, helping newcomers identify the crowd favorite. Observers can also compare recent blocks to the full historical record to gain insight into shifts in favorability over time.

Timestamped Records
A user hashes a file—say, a legal contract or artwork— to prove its existence at a given time. The user mines the hash into a block themselves or pays for mining power to do so. The blockchain's timestamp and hash serve as a public, tamper-proof record.

Ideology Voting
Users can embed hashes representing a set of principles or beliefs they support, effectively casting a vote for their preferred ideology in a global, decentralized space. For instance, an individual might hash a document outlining their stance on key issues—economic policies, social values, or technological priorities—and mine it into Pareme. Given different sets of ideologies, observers can tally which hashes of those sets have the most blocks, reflecting broader support. This creates a transparent, tamper-proof record of ideological alignment over time. Users could track shifts in sentiment by comparing recent blocks to historical data, offering a dynamic view of global preferences without centralized control.