Newsatoshi: A Pure-SHA-2 Shielded Electronic Cash System for Humans and Autonomous Machines

Newsatoshi Contributors
contributors@newsatoshi.org
https://newsatoshi.org
November 27, 2026 · Version 2.2 (Canonical Mainnet v2, nsM2)

Abstract. A purely peer-to-peer version of electronic cash that shields every balance by default, withstands quantum computers without elliptic curves or novel hash assumptions, and supports machine-speed streaming payments would allow both humans and autonomous software agents to transact directly without relying on banks, corporate stablecoin issuers, or trusted custodians. Bitcoin proved that proof-of-work orders history without a central party, yet left balances public, ownership guarded by elliptic curves, verification costs growing linearly with history, and long-run security dependent on fees alone. We present Newsatoshi (LOVE) Mainnet v2 (nsM2), a minimalist Layer-1 monetary protocol in which every coin is guarded solely by FIPS 180-4 SHA-2 and SHA-384 inside transparent Quartic Goldilocks STARK proofs (192-bit classical and 100–104-bit conservative QROM post-quantum soundness), mechanized across all 249 spend-circuit constraints in zero-dependency Lean 4. Blocks are ordered by Bitcoin-compatible proof-of-work merge-mined directly by existing Bitcoin hardware (AuxPoW chainid = 20051) under a staged Day-1 anti-instamine ASERT difficulty ramp, paired with a permanent tail emission floor and a 50% homeostatic base-fee burn. Constant-memory Utreexo frontiers (≤1 KB), work-proportional FlyClient MMRs, authenticated post-quantum P2P transport, seller-signed O(1) PayWord streaming channels (~0.01 ms per payment), and a Priorism–FEP epistemic stake lock (PriorLock) complete a minimal settlement floor engineered to scale without ever restarting genesis.

1. Introduction

Commerce on the Internet has relied almost exclusively on financial institutions serving as trusted third parties to process electronic payments [1]. While this model works adequately for most human retail transactions, it breaks down completely for a new class of economic participant: autonomous software agents negotiating compute, API calls, datasets, and multi-step contracts at machine speed. Software agents cannot pass know-your-customer checks, open bank accounts, or litigate disputes in court.

Corporate dollar stablecoins running on public blockchains solve bank onboarding only by introducing two fatal vulnerabilities. First, they are permissioned and freezable: a single issuer blacklist transaction can immobilize an entire autonomous treasury overnight. Second, they are completely transparent: on a public ledger, every competitor can observe an organization's or agent swarm's suppliers, unit costs, margins, and runway in real time. Bitcoin solves censorship resistance and credible neutrality, yet leaves three structural pressures unresolved: elliptic-curve signatures vulnerable to future quantum computers running Shor's algorithm, a fully transparent UTXO graph, and a block subsidy that decays toward zero while fees remain volatile [2].

What is needed is an electronic payment system based on cryptographic proof instead of trust, in which every balance is private by default, no elliptic curve or novel unvetted hash guards anyone's coins, Bitcoin's existing mining hardware secures block ordering without duplicate energy expenditure, and sub-cent machine payments stream at microsecond speed off-chain while settling cleanly onto a radically small base layer. We document the live architecture of Newsatoshi Mainnet v2 (nsM2, release v1.2.1), anchored to Bitcoin block #977,738 with zero premine.

2. Notes and Shielded Privacy

We define an electronic coin as an opaque *note* carrying a value $v < 2^{60}$ base units (1 LOVE = 10^8 base units), an owner's shielded address a_{pk}, and 160 random bits ρ. Following the shielded-pool paradigm introduced by Zerocash [3], every primitive in Newsatoshi is reduced exclusively to raw FIPS 180-4 `SHA256Compress` (denoted C) with four-byte domain separation tags:

```
a_pk = C("nsAD", sk, 0) // 256-bit address from secret sk
cm = C("nsCM", a_pk, v_hi, v_lo, rho) // commitment in depth-32 tree
nf = C("nsNF", sk, pos, rho, 0) // unique nullifier revealed on spend
node = C(left, right) // depth-32 SHA-256 note Merkle tree
```

There are no public keys and no elliptic-curve signatures guarding notes. When a note at position *pos* is spent, the spender publishes its nullifier *nf*, two new output commitments, a public fee, and a zero-knowledge STARK proving that: (1) each spent note exists in the note tree under a historical root, (2) the spender knows *sk* matching a_{pk}, (3) each nullifier is derived honestly, (4) output notes are well-formed and bounded below 2^{60}, and (5) input values equal output values plus public fee and optional deposit.

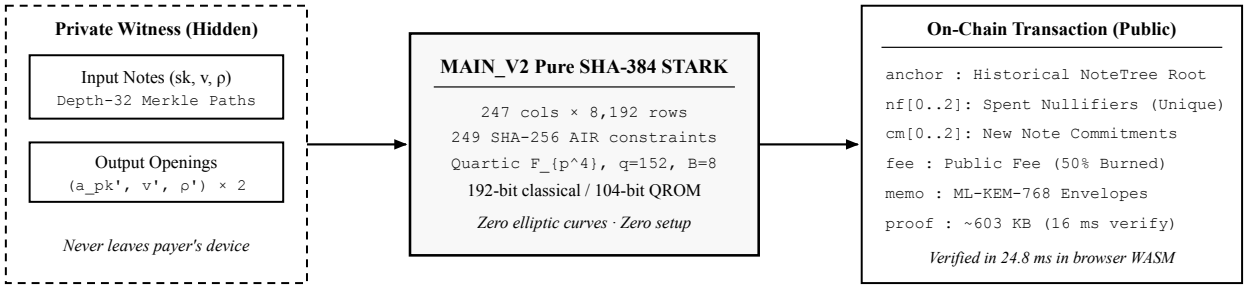

Figure 1: Pure SHA-2/SHA-384 shielded spend transaction. Spending authority rests solely on hash preimage knowledge.

Because a nullifier seen once is rejected permanently by consensus, double-spending is impossible without revealing which prior commitment was spent. Encrypted note delivery memos (`nsd1...`) use NIST FIPS 203 `ML-KEM-768` [4] strictly for receiver discovery confidentiality; because hash functions alone cannot implement non-interactive key exchange [5], a hypothetical break of `ML-KEM` could reveal payment metadata but could never forge or steal a single coin. Any holder may export a read-only viewing key (`nsvk1...`) for selective compliance audits or delegated wallet scanning without exposing spend authority.

3. Proof of Negentropy: Pure SHA-2 / SHA-384 STARKs

Rather than introducing novel algebraic hash functions (such as Poseidon or Tip5) whose cryptanalytic longevity remains under active research, Newsatoshi evaluates standard SHA-256 compressions directly inside a transparent STARK [6] using Haböck and Al Kindi randomized quotient segment masking [7] over Winterfell 0.13.1. To push conservative post-quantum security in the Quantum Random Oracle Model (CMS19 [8]) strictly above 100 bits across all batch sizes, Mainnet v2 (`MAIN_V2`) upgrades the proof stack to:

1. **Quartic Goldilocks Extension Field:** All out-of-domain and FRI folding challenges are drawn from the 256-bit quartic extension $\mathbf{F}_{p^4} = \mathbf{F}_p[u]/(u^4 - 7)$ over Goldilocks prime $p = 2^{64} - 2^{32} + 1$, eliminating extension-field grinding bottlenecks.

2. **Pure FIPS 180-4 SHA-384 Commitments:** Merkle commitments and Fiat–Shamir coins use SHA-384, providing a 192-bit classical collision bound and a 128-bit BHT quantum collision bound [9].
3. **Pareto-Optimal FRI Parameters:** Blowup $B = 8$, queries $q = 152$, proof-of-work grinding $g = 20$ bits, and folding factor $F = 8$ yield 192-bit classical FRI Johnson-bound security (with overall system classical security bounded at ~128 bits by FIPS 205 `SLH-DSA-SHA2-128s` and 256-bit SHA-256 note commitments) and **104 / 102 / 101 / 100 conservative CMS19 QROM bits** for batch sizes $K = 1$, 4, 8, and 16 respectively. Analytical inspection of Winterfell's Johnson-bound LDR formula verifies that increasing blowup to $B = 16$ or 32 drops QROM soundness below 100 bits at $K = 8$ and 16; hence (B=8, q=152, g=20) is the exact Pareto optimum.
4. **Upstream Field Canonicalization Patch:** Pre-genesis stress testing uncovered two upstream canonicalization defects in `winter-math 0.13.1` (`f64::BaseElement::double()` and `mul_small()` returning unreduced values $\geq p$ with probability ~2^{-33} and ~2^{-32}), which caused intermittent ~5–8% Quartic batch failures. Both were patched in vendored constant-time arithmetic and verified across 25 consecutive Quartic batch suites (75 proofs, zero failures).

4. Proof-of-Work and Merged Mining with Bitcoin

Blocks are ordered by SHA-256d proof-of-work over an 80-byte header matching Bitcoin's exact byte layout, replacing Bitcoin's transaction Merkle root field with a combined commitment `nsBLOCK = SHA256d("nsBLOCK" || tx_root || nsSTATE)`. Any SHA-256 miner may mine Newsatoshi directly, or merge-mine it alongside Bitcoin using Namecoin-format Auxiliary Proof-of-Work (`AuxPoW`, `CHAIN_ID = 0x4e53 = 20051`) [10].

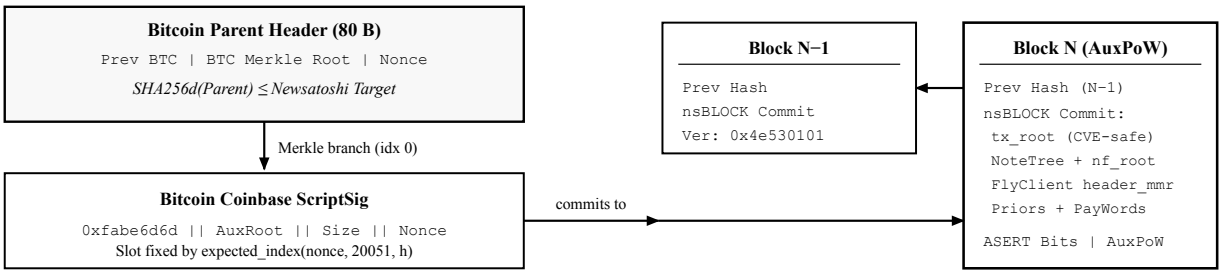

Figure 2: Merged mining with Bitcoin (AuxPoW chainid = 20051). The same SHA-256 hashes secure both chains with zero duplicate energy.

To eliminate Day-1 instamine attacks without centralized checkpoints, Mainnet v2 runs a two-stage ASERT [11] difficulty schedule (`pow::asert_bits_staged`) targeting 600-second blocks from Bitcoin difficulty 1 (`0x1d00ffff`): during blocks `1..=2,016` (roughly the first 14 days), a 1-hour (3,600 s) half-life doubles difficulty every six fast blocks, absorbing sudden 1,000,000× hashrate surges; at block `2,016`, difficulty transitions deterministically and panic-free into standard 2-day (172,800 s) steady-state ASERT anchored to block `2,016`'s header.

5. Network and Post-Quantum Transport

The steps to run the network are as follows:

1. Nodes dial peers over authenticated post-quantum transport (`pq_transport.rs`): every Mainnet v2 connection automatically negotiates ephemeral `ML-KEM-768 + SHA-256` AEAD (`NSPQ1`) to defeat passive harvest-now-decrypt-later surveillance, and supports FIPS 205 `SLH-DSA-SHA2-128s` [12] peer-key pinning (`--peer HOST:PORT@<vk>`, `NSPQ2`) to defeat active man-in-the-middle interception, hardened with 64 KB chunked framing against slowloris memory exhaustion.

2. New shielded transactions and `BatchSpend` aggregations are broadcast to all nodes; each node verifies STARK proofs locally (~16 ms native) before mempool admission.
3. Miners or AuxPoW pools assemble valid transactions into candidate blocks up to 8 MB (100,000 transaction cap) and solve either direct or merged-mined SHA-256d proof-of-work.
4. Solved blocks propagate via BIP-152-style Compact Block Relay (`Msg::CompactBlock` using 6-byte keyed short identifiers `nsCMPCT1`), cutting multi-megabyte STARK block relay bandwidth by over 95% when peers hold warm mempools.
5. Nodes accept the heaviest valid chain by cumulative proof-of-work, verifying every STARK proof, nullifier uniqueness, ASERT difficulty target, and state commitment.

6. Incentive, Perpetual Floor, and Homeostatic Burn

By convention, the first transaction in a block creates new coins owned by the block creator. There is zero premine, zero foundation allocation, zero investor sale, and zero project self-mining. New coins follow Bitcoin's schedule—50 LOVE per block, halving every 210,000 blocks across eight halvings (20,917,968.75 LOVE)—before settling at a permanent tail emission floor of `0.1953125 LOVE/block` (19,531,250 base units, ~0.049% initial annual dilution and strictly decreasing forever). A permanent floor eliminates the fee-only mining instability documented by Carlsten et al. [13].

To prevent unbounded net inflation under heavy global usage while preserving miner incentives, Mainnet v2 enforces a **Homeostatic Fee Burn** from block 0: exactly 50% (5,000 bps) of every shielded spend's base fee is permanently burned into `state.burned`, while the remaining 50% pays the block producer (and optional batch prover). Every valid block preserves the global monetary conservation invariant verified in Lean 4:

$$shielded_pool + locked_contracts + burned_supply = total_issued_subsidy$$

7. Reclaiming Disk Space and Light Verification

A shielded chain whose note tree grows unboundedly in RAM eventually centralizes full nodes. Newsatoshi solves state-footprint and light-verification scaling with three complementary hash-only structures:

1. **Constant-Memory Utreexo Note Frontier (`CompactNoteTree`):** Instead of holding $2N$ internal Merkle hashes in RAM (~6.4 GB at 100M notes), nodes and wallets maintain at most 32 subtree peaks occupying **≤ 1,024 bytes of RAM** regardless of whether the chain holds 10 notes or 4.29 billion notes, producing bit-for-bit identical roots (exposed live via `GET /api/utreexo` and `newsatoshi utreexo-frontier`) with $O(1)$ tracked inclusion-proof updates per append.
2. **Disk-Backed Pruned KV Checkpoints:** Historical block bodies can be pruned to a configurable tail window while retaining verifiable nullifier state on disk (`pruned_store.rs`, `newsatoshi prune`).
3. **Work-Proportional FlyClient MMR & R&D Recursion Prototype:** Every Mainnet v2 block commits to the ancestor header Merkle Mountain Range root (`header_mmr`) inside `nsSTATE` from Genesis #0, binding exact cumulative work into every internal node (`nsMMRND1`) so sibling work inflation is impossible [14]. Production Mainnet v2 light clients verify headers logarithmically via FlyClient MMR. Separately, `recursion.rs` includes an executable R&D folding prototype (`StarkInStarkVerifierAir`, 254 cols × 16,384 rows, 204 bit-exact SHA-256 compressions, 10 Quartic FRI folds, ~671 KB proof) verifying 1 of 152 outer queries (omitting full out-of-domain constraint checks; not wired into L1 consensus until GPU-scaled full-query verification activates via BIP-9).

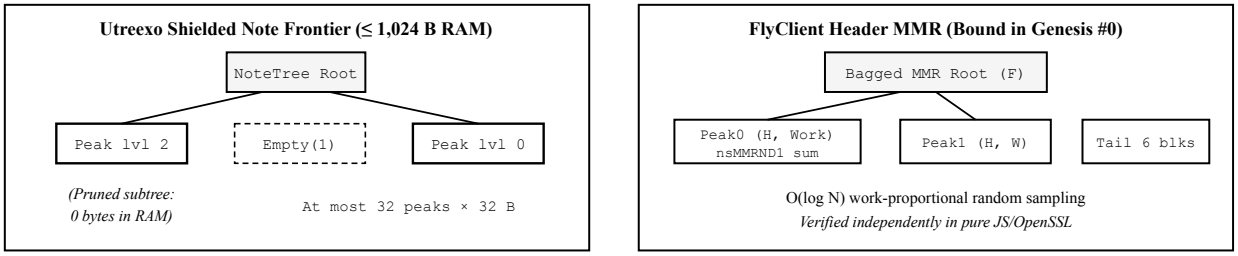

Figure 3: Constant-RAM Utreexo note frontier (left) and work-proportional FlyClient MMR committed in every nsM2 block header (right).

8. Machine Velocity and Priorism–FEP Settlement

Ten-minute proof-of-work blocks provide strong finality but cannot directly meter per-token AI inference or sub-cent API calls. Mainnet v2 therefore includes two minimal, post-quantum primitives at Layer 1:

1. **Seller-Signed `RFC-0022` PayWord Streaming Channels:** Building on Rivest and Shamir's hash-chain micropayments [15], a buyer locks deposit D into a Merkleized L1 channel committing to a hash-chain tip $T = \text{SHA-256}^{N}(s)$ and the seller's FIPS 205 `SLH-DSA` key. Each off-chain API call reveals the next preimage $w_k = \text{SHA-256}^{N-k}(s)$. The seller verifies one SHA-256 hash incrementally in **~0.009–0.013 ms (~100,000 TPS per CPU core, zero per-call gas)** and settles once on L1 via seller-signed `ChannelClaim`, preventing adversarial-buyer early-close theft.
2. **Priorism / Free Energy Principle Commitments (`PriorLock`):** Autonomous agents must not only pay each other; they must stake credible economic commitments behind verifiable claims and service outputs. Grounded in the Free Energy Principle [16] (minimizing prediction error / surprise), a `PriorLock` escrows shielded stake behind a `claim_hash`, challenge window, and m-of-n `SLH-DSA` arbiter quorum enforcing anti-self-arbitration (SCGS-I3). Accurate priors return intact; disconfirmed claims pay a capped miner fee (≤10%) and split the remaining slashable stake **exactly 50% bounty to the challenger and 50% permanently burned**, tying epistemic accuracy directly to monetary scarcity.

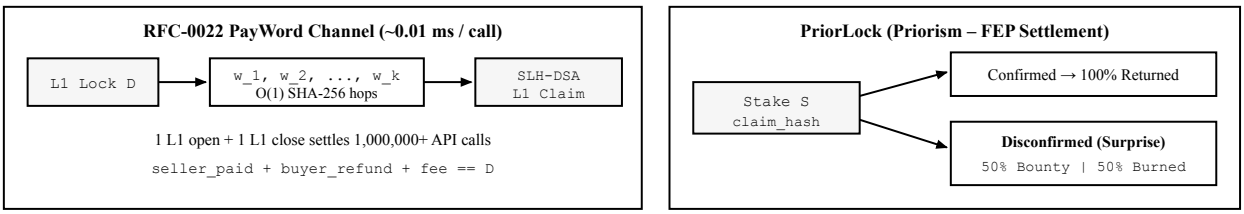

Figure 4: Off-chain machine-velocity PayWord channels (left) and epistemic PriorLock surprise resolution (right).

9. Combining and Compressing Value (BatchSpend)

Although single `MAIN_V2` spend proofs (~603.7 KB) verify in 16 ms natively and 24.8 ms in WebAssembly, individual proofs dominate raw L1 block space. Mainnet v2 therefore supports `Tx::BatchSpend` from genesis, folding $K \in [2, 16]$ shielded 2-in/2-out spends into a single Quartic SHA-384 STARK proof. At $K = 8$, proof size compresses **6.51×** from 4.83 MB to 741,672 bytes (~92.7 KB/spend); at $K = 16$, amortized proof size drops to ~49 KB/spend. An explicit STARK-bound `prover_fee` drawn from the unburned fee share pays external batch provers directly in the block coinbase while leaving 100% of block subsidy and remaining fees to the miner, ensuring strict incentive compatibility.

10. Formal Verification and Independent Cross-Checking

Following the June 2026 Orchard shielded-pool inflation flaw [17], private money requires machine-checked circuit soundness. Newsatoshi verifies its live Rust spend circuit (`node/src/spend.rs`) via three independent layers:

1. **Zero-Dependency Lean 4 Kernel (`0 sorry`, `0 Mathlib`, `0 native_decide`):** `SpendCircuitFull.lean` symbolically extracts all **249 live Rust spend AIR transition constraints** (`2,950` terms) + 7 Goldilocks no-wrap lemmas and proves end-to-end semantic spend soundness (`accepted_spend_trace_implies_spend_relation` & `nullifier_deterministic_unique`: ownership, 32-level Merkle membership, deterministic nullifier binding, $<2^{60}$ range bounds, dummy zero-value, and exact integer conservation with zero Goldilocks wrap); `SpendBalance.lean` proves 10 additional protocol conservation theorems (`BatchSpend`, PayWord, `PriorLock` 50/50, Homeostatic Burn, FlyClient). The vendored Winterfell Rust prover and Haböck–Al Kindi masking patch remain explicit top targets of the public `LOVE` Bug Bounty.
2. **Automated Z3 SMT Extraction (`check.py`):** Extracts all 249 polynomial constraints (2,950 terms) directly from the Rust source and confirms zero unintended cell freedom across all 1,138,176 active cells.
3. **Independent Second Verifier Client (`verifier-ts/newsatoshi_verifier.mjs`):** Sharing zero code with the Rust reference node, a standalone JavaScript/OpenSSL/BigInt implementation independently verifies live Mainnet v2 headers, Bitcoin AuxPoW proofs, Day-1 staged ASERT targets, FlyClient MMR roots, and `nsSTATE`/`nsBLOCK` commitments directly against live production endpoints.

11. Calculations and Hyperscale Path

We compare measured throughput across incumbent rails and Newsatoshi Mainnet v2, and analyze conservative QROM security across batch sizes. Table 1 lists exact security bounds under Winterfell's Johnson-bound list-decoding regime and CMS19 QROM quadratic loss [8]:

Batch Size (K)	Trace Rows	FRI Parameters (B, q, g, F)	Proof Size (Bytes)	Classical LDR	CMS19 QROM	BHT Hash QROM
$K = 1$ (Single)	8,192	8, 152, 20, 8 ($\mathbf{F}_{p^4}$)	603,686 B	192 bits	**104 bits**	128 bits (SHA-384)
$K = 4$ (Batch)	32,768	8, 152, 20, 8 ($\mathbf{F}_{p^4}$)	~674 KB (~168 KB/tx)	192 bits	**102 bits**	128 bits (SHA-384)
$K = 8$ (Batch)	65,536	8, 152, 20, 8 ($\mathbf{F}_{p^4}$)	741,672 B (~93 KB/tx)	192 bits	**101 bits**	128 bits (SHA-384)
$K = 16$ (Batch)	131,072	8, 152, 20, 8 ($\mathbf{F}_{p^4}$)	~785 KB (~49 KB/tx)	192 bits	**100 bits**	128 bits (SHA-384)

Table 2 compares global payment velocity and settlement capacity across incumbent payment networks, today's Mainnet v2 launch baseline, and our planned hyperscale phases:

System / Layer	Throughput (TPS)	Finality / Settlement	Privacy & Quantum Security
Visa + Mastercard (Combined)	~15,000 avg / ~100,000 peak	1–3 days (reversible)	Custodial, freezable, transparent to issuer
Bitcoin L1	~4–7 TPS	~60 min (6 blocks)	Public UTXO graph; ECDSA/Schnorr ECC
Newsatoshi L1 (`v1.2.1` Today)	0.02–0.27 TPS (~23k/day at K=16)	~60 min (BTC AuxPoW)	100% shielded; pure SHA-2/384 (100–104b QROM)
Newsatoshi L1 (Post BIP-9 `bit 6`)	~10 TPS (~864k settlements/day)	~60 min (BTC AuxPoW)	100% shielded; 1 recursive block proof / 8 MB
Newsatoshi L2 (`RFC-0022` PayWords)	**~100,000 TPS / core (10M+ global)**	~0.01 ms receipt; L1 close	Post-quantum SHA-256 + FIPS 205 SLH-DSA

11.1 Why Mainnet v2 Never Restarts Genesis to Hyperscale

Just as Bitcoin (2009), Zcash (2016), and Monero (2014) scaled on one permanent Genesis, Mainnet v2 (`BTC #977,738`) needs zero future Genesis restarts. Zero-consensus hyperscale infrastructure is already live today: `100,000+` TPS/core PayWords, BIP-152 Compact Block Relay (>95% bandwidth cut), ≤1 KB Utreexo frontiers, 8 MB / 100,000-tx block limits, read-only `nsvk1...` viewing keys, and local disk KV pruning (upgradable to RocksDB/LMDB with zero consensus change). Multi-merchant PayWord hubs and Channel Factories multiply L1 onboarding 100×–1,000× purely at Layer 2.

Two proof-layer upgrades were deliberately withheld from Block 0 to minimize initial audit surface and prevent unaudited shielded inflation bugs, and are scheduled via pre-announced, height-gated **BIP-9 Coordinated Network Upgrades** (preserving every coin and history identically to Zcash Sapling→Orchard): `BIT_FUTURE_COMPACT_PROOF = 5` (pure-hash WHIR/Binius compact proofs, 2–4× smaller) and `BIT_FUTURE_RECURSIVE_AGG = 6` (GPU-scaled 152-query extension of `StarkInStarkVerifierAir` folding 1,000+ spends per block into ~10 L1 TPS with zero-witness-visibility aggregation), followed by permanent monetary ossification via `BIT_OSSIFY_MONETARY_BASE = 7`. If L1 ever approaches 10^{10} cumulative spends, evolving epoch nullifiers [18] can activate cleanly as a versioned note type.

11.2 Open Verification and Public Security Bug Bounty

Newsatoshi's security rests on open math, open source code, and open economic incentives rather than closed assurances. All consensus modules, Winterfell field patches, and Lean 4 proofs are covered by an open, coin-denominated **Public Security Bug Bounty** funded by fair-mined Mainnet v2 `LOVE` at address `ns1af734b0e0b0c1c6a54e206ba7a23f586daee77ede5e7c51b6ec6b133c51f572a99d8cb66` (scope packet: `docs/AUDIT_SCOPE_V2.md`; bounty rules and live payout tiers: `https://newsatoshi.org/bounty/`). Anyone may independently verify the live chain via `verifier-ts/newsatoshi_verifier.mjs`, inspect the Lean 4 proofs, or mine directly into the public bounty treasury at `stratum+tcp://pool.newsatoshi.org:3333`.

12. Conclusion

We have presented and launched a pure-SHA-2 shielded electronic cash system designed for both humans and autonomous machines. Every balance is private by default and guarded exclusively by FIPS 180-4 SHA-2 and SHA-384 inside Quartic Goldilocks STARKs achieving 192-bit classical and 100–104-bit conservative post-quantum soundness, mechanized across all 249 spend constraints in Lean 4. Blocks are ordered by Bitcoin merge-mined

proof-of-work with zero premine, a perpetual tail emission floor balanced by a 50% homeostatic fee burn, ~0.01 ms off-chain PayWord streaming, and Priorism–FEP epistemic commitments. With ≤1 KB Utreexo frontiers, FlyClient MMRs, Compact Block Relay, and reserved BIP-9 upgrade bits, the system provides a minimal, unfreezable monetary floor engineered to scale globally and ossify permanently.

CANONICAL MAINNET V2 (NSM2, V1.2.1) FAIR-LAUNCH GENESIS VERIFICATION RECORD

Genesis Hash: `00000000ba1cdad4aa8e82d6613b4df536394c41008fc3a0914a47a44604b9c8`
Bitcoin Anchor: Block `#977,738` (`000000000000000000163900064a4cc4807dbd8bcaf35e322e694e6c387a780`)
Timestamp / Bits / Nonce / Premine: `1795742750` (2026-11-27 01:25:50 UTC) · `0x1d00ffff` · `2649823154` · **0 LOVE**
NSPQ2 Pinned Seed 1: `newsatoshi-seed-v2.fly.dev:8445@b024aaeafa0e4ca384b99483c28c3bedde45a5461a560063b283766414dfbbcc`
NSPQ2 Pinned Seed 2: `pool.newsatoshi.org:8445@78187a9436dd2d8cc7ed7afff2396b72b56687099d45d93da676b0a90ed8c6b6`

References

1. S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," October 31, 2008.
2. Newsatoshi Contributors, "A Store of Order: Electronic Cash Everyone Can Verify (NEWSATOSHI/2.0)," `https://newsatoshi.org/white-paper/`, November 2026.
3. E. Ben-Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, M. Virza, "Zerocash: Decentralized Anonymous Payments from Bitcoin," IEEE S&P, 2014.
4. National Institute of Standards and Technology (NIST), "Module-Lattice-Based Key-Encapsulation Mechanism Standard (ML-KEM)," FIPS 203, August 2024.
5. B. Barak, M. Mahmoody-Ghidary, "Merkle Puzzles Are Optimal," CRYPTO 2009.
6. E. Ben-Sasson, I. Bentov, Y. Horesh, M. Riabzev, "Scalable, Transparent, and Post-Quantum Secure Computational Integrity," IACR ePrint 2018/046.
7. U. Haböck, Al Kindi, "A Note on Adding Zero-Knowledge to STARKs," IACR ePrint 2024/1037.
8. A. Chiesa, P. Manohar, N. Spooner, "Succinct Arguments in the Quantum Random Oracle Model," TCC 2019 (IACR ePrint 2019/834).
9. G. Brassard, P. Høyer, A. Tapp, "Quantum Cryptanalysis of Hash and Claw-Free Functions," LATIN 1998.
10. Namecoin Project, "Merged Mining Specification (AuxPoW)," 2011.
11. M. Lundeberg, J. Toomim, "aserti3-2d: Absolutely Scheduled Exponentially Rising Targets," 2020.
12. National Institute of Standards and Technology (NIST), "Stateless Hash-Based Digital Signature Standard (SLH-DSA)," FIPS 205, August 2024.
13. M. Carlsten, H. Kalodner, S. M. Weinberg, A. Narayanan, "On the Instability of Bitcoin Without the Block Reward," ACM CCS 2016.
14. B. Bünz, L. Kiffer, L. Luu, M. Zamani, "FlyClient: Super-Light Clients for Cryptocurrencies," IEEE S&P 2020.
15. R. L. Rivest, A. Shamir, "PayWord and MicroMint: Two Simple Micropayment Schemes," Security Protocols Workshop, 1996.
16. K. Friston, "The Free-Energy Principle: A Unified Brain Theory?" Nature Reviews Neuroscience, 2010.
17. Shielded Labs & Project Tachyon, "Orchard Circuit Vulnerability Disclosure & Ironwood Lean Formalization," June–July 2026.
18. S. Bowe, I. Miers, "A Note on Notes: Towards Scalable Anonymous Payments via Evolving Nullifiers," IACR ePrint 2025/2031.