**Byzantium‑Coin™ (BZFC)**

**CoinMarketCap White Paper — v2 (Aug 23, 2025)**
License: MIT
Ticker: **BZFC**
Decimals: **17**
Max Supply: **21,000,000,000,000 BZFC (21T)**
Consensus: **PoSA — Proof‑of‑Stake Authority**
Tech stack: **Rust / Substrate (FRAME)**

**Upload note:** Save this file as README.md at the root of your GitHub repo for best presentation.

**1) Abstract**

Byzantium‑Coin™ (BZFC) is a PoSA‑secured Layer‑1 chain designed for **predictable monetary decay**, **institutional wrapping**, **industrial compliance attestations**, and **actuarial risk pools**. Its policy surface is organized under the **Smart Heuristic Retrieval (SHR)** namespace model, which routes requests to versioned rulebooks so **future upgrades never rewrite history**. A **half‑life decay engine**—optionally extended to a multi‑isotope model—provides flexible, opt‑in supply reduction with a hard rule: **savings/vault balances never decay**.

**2) Quick Facts**

* **Name / Ticker:** Byzantium‑Coin™ / **BZFC**
* **Max Supply:** 21 **trillion** BZFC (21,000,000,000,000)
* **Decimals:** 17 (base unit = 10^‑17 BZFC)
* **Consensus:** PoSA (authority set via staking & governance)
* **Monetary Policy:** Half‑life decay (on/off), optional **isotope** blend, **1 day → 1000 years** range; **savings exempt**
* **Fees:** Flexible model (Flat or EIP‑1559‑style), **per‑namespace** add/mult, optional tips; configurable **burn split**
* **Wrapping:** 1:1 institutional/customer wraps with **vault shielding** (underlying held in savings)
* **Compliance:** Editioned **Rulebooks** (Θ‑II, Θ‑III, …) and immutable **Attestations**
* **Insurance:** Ψ pools with pot ratios, reserve floors, and payout caps
* **Open Source:** MIT License (developer‑friendly)

**3) SHR Namespace Architecture**

BZFC organizes policy and modules by **Greek namespaces** with **Roman numerals** for editions/generations. The **SHR Router** directs a request to the correct module and edition at the time of action; later editions never retroactively change existing records.

* **Ω (Omega):** Fungible assets, decay models, *Crypto‑Tics™* (time units), payments, fees, burns, savings vaults
*Examples:* Ω‑II (second‑gen decay schedule), Ω‑III (new fee rulebook)
* **Θ (Theta):** Industrial compliance, certification, material verification
*Examples:* Θ‑IV → ASME Section IV code checks
* **Ψ (Psi):** Insurance pools, pot ratios, actuarial models
*Examples:* Ψ‑III → 3rd actuarial release for marine risk
* **Λ (Lambda):** Wrapping, derivative issuance, cross‑chain certificates
*Examples:* Λ‑I → first derivative wrapping module for treasury vaults

**Case Example:** A refinery submits a steel batch for verification. SHR routes to **Θ‑II**, locking compliance checks to **Rulebook Edition II**. Future rulebooks (Θ‑III, Θ‑IV) **do not** alter the original audit record.

**4) Monetary Policy & Decay**

**4.1 Goals**

* Predictable, configurable supply reduction via **half‑life decay**
* **Opt‑in** burns that **never penalize savings/vault balances**
* Optional **isotope** model for multi‑component decay tuned to different horizons
* Configurable **off/on** at governance level; **1 day → 1000 years** half‑life range (max 1000y)

**4.2 Mechanics (on‑chain friendly)**

To avoid expensive exponentials on‑chain, BZFC uses a **per‑epoch rate** r\_epoch with a pooled accounting index:

* **Per‑epoch rate from half‑life**
( r\_{\text{epoch}} = 1 - 2^{-\Delta t / T\_{1/2}} )
* **Isotope blend (weights sum to 1)**
( r\_{\text{eff}} = \sum\_i w\_i, (1 - 2^{-\Delta t / T\_{1/2,i}}) )
* **Global liquid index update (O(1))**
( \text{index}*{\text{new}} = \text{index}*{\text{old}} \cdot (1 - r\_{\text{eff}}) )

All **liquid** balances scale by the index. **Savings/vault** balances sit outside the decay path and are **never decayed**.

**4.3 Savings Exemption**

* Users can move funds **Liquid ⇄ Savings**; savings are **decay‑free**.
* Optional anti‑gaming **lock window** (default 0) can snapshot savings at epoch start without penalizing genuine savers.

**4.4 Isotope Coupling (optional)**

* Governance can enable **isotope coupling** to consider aggregate burns as a signal for component weights/rates.
* This remains **opt‑in** and **editioned** via Ω rulebooks for auditability.

**5) Fees, Burns & Treasury**

* **Modes:** Off, **Flat** (base fee per tx), or **EIP‑1559‑style** (base fee adjusts by block fill proxy).
* **Per‑namespace knobs:** additive fee (base units) + multiplier (bps) per module (Ω/Θ/Ψ/Λ).
* **Tips:** optional user tips added to total fees.
* **Burn split:** BurnBps defines what % of fees is **burned** vs accrued to **Treasury**.
* **Events & metrics:** emit FeeCharged{total,burned,treasury}; track block‑level activity.

**Note:** In production, chains commonly integrate with pallet-transaction-payment for tx fees. BZFC exposes a pallet‑level fee framework for fine‑grained, namespace‑aware policies and on‑chain analytics.

**6) Wrapping & Vault Shielding (Λ)**

* **1:1 wrapping** for institutional or customer‑specific applications.
* Underlying BZFC is moved from **Liquid → Savings (vault)** during the wrap, making it **decay‑exempt** while wrapped.
* wrap\_mint(policy, amount) and wrap\_burn(policy, amount) manage lifecycle; policies carry metadata hashes (e.g., IPFS CIDs) and shielding rules.

**7) Industrial Compliance & Rulebooks (Θ)**

* **Register / update / lock** Rulebook editions (Θ‑II, Θ‑III, …). Once **locked**, a rulebook is immutable.
* **Attestations** bind a subject\_hash (e.g., steel batch) to an edition with a payload\_hash (e.g., PMI report) + optional evidence hashes.
* **Annotations** (Info / Supersede / Revoke / Reject) are **append‑only**; prior records remain intact.

**Assurance:** Auditors and regulators can rely on the fact that **later editions never rewrite earlier attestations**.

**8) Insurance & Actuarial Pools (Ψ)**

* Governance creates pools with **pot ratios**, **reserve floors**, **per‑claim caps**, and **min collateral**.
* **Premiums:** user Liquid → **pool vault Savings** (decay‑exempt while pooled).
* **Claims:** **vault Savings → claimant Liquid**, enforcing solvency rails.
* Transparent counters for premiums in / payouts out; designed for actuarial analytics.

**9) Consensus & Security (PoSA)**

* **PoSA**: a vetted validator/authority set participates in block production and finality.
* **Rotation & staking** managed via Substrate **Session**/**Aura** with governance controls.
* **Bootstrapping:** start with a small authority set; expand/rotate as decentralization increases.

**10) Tokenomics Overview**

* **Max Supply:** 21,000,000,000,000 BZFC (hard cap)
* **Unit Precision:** 17 decimals
* **Emissions/Reductions:** Supply reduction driven by **decay burns** and **fee burns** per rulebook editions.
* **Savings/Vaults:** Never decay; principal protected from automatic burns.
* **Wrapping:** Neutral to total supply; wrapped units are claims on saved principal.

**Circulating Supply & Allocation:** To be finalized at mainnet genesis and disclosed in the public chainspec and explorer. BZFC’s framework supports transparent genesis endowments, vesting, and on‑chain tracking of burns.

**11) Mathematics & Developer Notes**

* **Scaling:** Use **Perquintill** precision (1e18) for per‑epoch rates and indices.
* **Index accounting:** global pooling ensures O(1) decay application; avoids map iteration.
* **Reference CLI:** The repo includes a **Half‑Life CLI** to compute r\_epoch for any half‑life in **1 day → 1000 years**.
* **Runtime integration:** Substrate FRAME pallets (Ω/Λ/Θ/Ψ), SHR Router, governance origins, and PoSA wiring stubs are provided.

**12) Governance**

* **Governance Origin** controls decay mode, isotope parameters, fee model, burn split, namespace fees, savings lock window, rulebook lifecycle, and Ψ pool parameters.
* Early phases may use **Root/Sudo**; transition paths include councils or multisig collectives.
* All parameter changes emit **events** for auditability.

**13) Compliance & Auditability**

* Event‑rich design: DecayApplied, FeeCharged, WrapMinted/WrapBurned, Attested/Annotated, PremiumPaid/ClaimPaid.
* Tooling: command‑line **audit parser** (JSONL → Markdown/CSVs) and a **browser log viewer** (drag‑and‑drop).
* SHR ensures investigations can reproduce the **exact rulebook edition** used at the time of action.

**14) Roadmap & Delivery Status**

**Delivered engineering milestones (public repo scaffolds):**

1. Ω pallet, PoSA stubs, genesis template, Half‑Life CLI
2. Governance origin + parameter transactions
3. **Pooled decay index** (O(1) per epoch)
4. Tests, chainspec samples, minimal node notes
5. Λ wraps MVP + vault shielding
6. Θ rulebooks + attestations + annotations
7. Ψ pools MVP (premiums/claims with solvency rails)
8. Node quickstart & consolidated chainspec
9. **Enhanced fee model** (Flat/EIP‑1559, per‑namespace, tips, burn split)
10. Audit tools, UI, consolidated docs
11. **Monorepo consolidation** (workspace, changelog, helpers)

**Next:** Security review, formal verification targets for decay math, and expanded explorer integration.

**15) Risk Factors**

* **Consensus & governance risks:** PoSA starts with curated authorities; decentralization roadmap and rotation policies mitigate but do not eliminate risks.
* **Parameter misconfiguration:** Decay rates, burn splits, and fee models are governance‑controlled; strong multi‑sig or council procedures recommended.
* **Smart contract / pallet bugs:** Pallet code is open source; audits and bounty programs are encouraged prior to large‑scale value custody.

**16) Legal & Licensing**

* **License:** MIT — permissive, developer‑friendly.
* **Brand:** Byzantium‑Coin™ (BZFC). All third‑party marks belong to their respective owners.
* **Regulatory posture:** BZFC is an open‑source protocol. Jurisdiction‑specific compliance remains the responsibility of operators and integrators. The Θ module assists with attestations but does not replace regulatory advice.

**17) Listing Information (for CMC)**

* **Project Name:** Byzantium‑Coin™
* **Ticker:** BZFC
* **Type:** L1 (Substrate‑based), custom pallets (Ω/Λ/Θ/Ψ)
* **Max Supply:** 21,000,000,000,000
* **Decimals:** 17
* **Consensus:** PoSA (Aura/Session)
* **Explorers:** *TBA*
* **Official Website:** <https://byzantinefuturecapital.com>
* **Docs & White Paper:** This document (MIT) + repo docs
* **Contact / Email:** info@byzantinefuturecapital.com
* **Contracts (if bridged/wrapped):** *TBA*
* **Circulating Supply:** *TBA at genesis*

This white paper describes *protocol design and reference implementation*. Final mainnet parameters (authorities, genesis allocations, explorer URLs) will be published before launch.

**Appendix A — Example Decay Configuration**

* **Epoch length:** 1 hour
* **Half‑life:** 100 years → CLI computes r\_epoch (Perquintill) for governance.
* **Decay mode:** HalfLife { rate\_per\_epoch }
* **Savings lock window:** 0 (savers remain fully exempt)

**Appendix B — Example Isotope Blend**

* Components: 30% (T½=1y), 70% (T½=10y)
* Effective per‑epoch rate: ( r\_{\text{eff}} = 0.3,r\_1 + 0.7,r\_2 )

**Appendix C — Namespace Codes (suggested)**

* Ω (Omega) = 937, Θ (Theta) = 920, Ψ (Psi) = 936, Λ (Lambda) = 923 (for internal fee routing; configurable)