

Apodokimos: A Decentralized Epistemic Contribution Graph for Scientific Claim Evaluation

Version: wp-v0.1

Status: Draft

License: CC0-1.0

Site: <https://apodokimos.science>

Repository: <https://github.com/apodokimos/apodokimos>

DOI: <https://doi.org/10.5281/zenodo.19583091>

Abstract

The Journal Impact Factor (IF) — a 2-year citation velocity metric designed in 1955 for library collection management — has become the dominant instrument for evaluating scientific output, allocating research funding, and determining academic careers. This repurposing is structurally unsound: the IF measures citation frequency within a journal-bounded prestige economy, not epistemic quality, reproducibility, or real-world consequence. Every proposed alternative (h-index, CiteScore, altmetrics, Scite Index) addresses surface parameters while preserving the broken ontology: the paper as the atomic unit of science, citations as the signal, and journals as the prestige layer.

We propose Apodokimos — a decentralized Epistemic Contribution Graph (ECG) protocol that replaces journal-level citation counting with claim-level survival scoring. In the ECG model, the atomic unit is a falsifiable claim, not a paper. Claims accumulate epistemic weight through a function of replication score, dependency depth, survival under falsification, and real-world outcome linkage. Attestations — typed as supports, contradicts, replicates, or refutes — are registered immutably on a Substrate parachain. Claim content is stored permanently on Arweave. Researcher reputation is encoded as non-transferable Soulbound Tokens (SBTs), making governance weight identity-bound rather than capital-bound. The protocol is owned by no institution and licensed under AGPL-3.0, structurally preventing commercial capture.

The first deployment domain is clinical medicine, where PICO-structured claim schemas and existing trial registry infrastructure provide well-defined granularity and objective real-world outcome oracles.

1. Problem Statement

1.1 What the Impact Factor Actually Measures

The Journal Impact Factor for year X is defined as:

$$\text{IF}(X) = \frac{\text{citations in } X \text{ to articles published in } (X-1) \text{ and } (X-2)}{\text{total citable articles published in } (X-1) \text{ and } (X-2)}$$

This is a measure of citation velocity of a journal's average article within a 2-year window. It captures reading habits of a specific community, modulated by field size, citation norms, and editorial strategy. It does not measure:

- Whether findings are reproducible
- Whether claims survived independent testing
- Whether the work influenced clinical practice, policy, or technology
- Whether citations are positive, negative, or retractions

The Reproducibility Project (Open Science Collaboration, 2015) replicated 100 psychology studies and found that higher-IF journals had worse replication rates — the metric is negatively correlated with the quality it purports to measure in at least one major field.

Chalmers and Glasziou (2009) estimated that 85% of biomedical research investment is wasted, the majority published in indexed, respectable journals. The IF certified the packaging, not the truth.

1.2 Structural Origin of the Misalignment

The IF persists not because it is accurate but because it solves a coordination problem. Every hiring committee, grant reviewer, and tenure panel requires a shared scalar. The IF is wrong but universally legible. Any replacement must solve the coordination problem, not only the measurement problem.

The incentive loop is entirely self-referential:

Researcher needs tenure
→ needs publications in high-IF journals
→ journals reward novelty + $p < 0.05$
→ researchers optimize for publishability, not truth
→ loop closes on itself, never touching reality

1.3 Why Existing Alternatives Fail

Every alternative published to date — DORA (2013), Leiden Manifesto (2015), Scite (2019), Octopus (2022), CRediT taxonomy — fails to exit the paper ontology. They improve the lens on the existing system without replacing the system's atomic unit.

DORA gathered 26,000+ signatories and produced marginal behavioral change. A 2021 EMBO Reports commentary confirmed that over eight years post-DORA, metrics continued to dominate hiring, promotion, and funding decisions. Signing a declaration does not change an incentive structure.

Scite contextualizes citations (supporting vs. contradicting) but remains paper-level and was acquired by a commercial entity (Research Solutions), reproducing the capture dynamic it was designed to challenge.

Octopus restructures the paper into typed micro-publications but retains the paper as the product. Adoption remains near zero because it offers no prestige signal.

The failure mode is consistent: downstream interventions operating within an upstream ontology they do not challenge.

1.4 The Required Ontological Break

The necessary change is not a better metric for papers. It is a replacement of the paper as the atomic unit of science with the falsifiable claim — the actual unit that reality cares about.

Science does not produce papers. Science produces claims about how the world works. Papers are the containers those claims travel in. The container has captured all the value.

2. The Epistemic Contribution Graph

2.1 Formal Model

The Epistemic Contribution Graph G is a directed acyclic graph (DAG):

$G = (C, A, E)$

Where: - C is the set of registered claims (nodes) - A is the set of attestations (typed edges) - $E \subseteq C \times C$ is the dependency relation (claim builds on claim)

Acyclicity constraint: No claim may depend on itself directly or transitively. Dependency cycles are rejected at the protocol level.

2.2 Claim Definition

A claim is the minimal unit of falsifiable scientific assertion. Formally:

```
Claim := {
  id:          ClaimId,          // blake3(canonical_json(content))
  claim_type:  ClaimType,
  field_id:    FieldId,
  content:     ClaimContent,     // stored on Arweave, hash anchored on-chain
  submitter:   DID,
  depends_on:  Vec<ClaimId>,     // edges in E
  arweave_tx:  TxId,
  registered:  BlockNumber
}
```

2.2.1 ClaimType Taxonomy

```
ClaimType :=
| PrimaryClaim // a novel falsifiable assertion about the world
| Hypothesis   // a testable prediction not yet tested
| Method       // a procedure claim (reproducibility target)
| Result       // an empirical measurement or observation
| Replication  // an independent repetition of a prior Result
| NullResult   // a Result with no detectable effect
```

Null results are first-class citizens. The current system suppresses null results (publication bias). In the ECG, a null result that contradicts a prior PrimaryClaim carries epistemic weight equivalent to a positive result.

2.2.2 Claim Granularity Claim granularity is the hardest operational parameter. Too fine: spam and decomposition gaming. Too coarse: reproduces the paper ontology.

The protocol enforces granularity through field schemas. Each field defines a canonical claim template that specifies required and optional fields. A submission that cannot be represented in the field schema cannot be registered as a single claim — it must be decomposed.

For clinical medicine (the bootstrap domain), the PICO schema provides well-defined granularity (see Section 10).

2.3 Attestation Definition

An attestation is a typed statement by a credentialed reviewer about a claim's relationship to existing evidence:

```
Attestation := {
  id:          AttestationId,
  claim_id:    ClaimId,
  attester:    DID,
  verdict:     AttestationVerdict,
```

```

evidence_tx: Option<TxId>, // Arweave pointer to supporting evidence
attester_sbt: FieldSBTScore, // snapshot at time of attestation
block:      BlockNumber
}

```

2.3.1 AttestationVerdict Taxonomy

```

AttestationVerdict :=
| Supports      // attester's evidence corroborates the claim
| Contradicts   // attester's evidence is in tension with the claim
| Replicates    // independent repetition confirms the claim
| Refutes       // independent repetition disconfirms the claim
| Mentions      // neutral reference; excluded from survival scoring

```

Mentions is tracked but does not contribute to survival score S . This eliminates the current system's fundamental flaw where a citation that reports failure is counted identically to a citation that confirms.

3. Claim Weight Function

3.1 Definition

The epistemic weight of a claim at time t is:

$$W(c, t) = R(c, t) \times D(c) \times S(c) \times O(c)$$

All four factors are defined on $[0, 1]$ except D which is unbounded above. W is normalized per-field for comparison purposes.

3.2 $R(t)$ — Replication Score with Time Decay

$R(c, t)$ measures the replication support for claim c at time t , with field-calibrated time decay applied to older attestations:

$$R(c, t) = \frac{\sum_{a \in \text{Replicates}(c)} w_{\text{decay}}(t - a.\text{block}, \lambda_{\text{field}})}{|\text{Replicates}(c)| + |\text{Refutes}(c)| + \epsilon}$$

Where:

$$w_{\text{decay}}(\Delta t, \lambda) = e^{(-\lambda \times \Delta t)}$$

λ_{field} is the field-specific decay constant. Fields with fast knowledge cycles (molecular biology, clinical trials) use higher λ — recent replications are weighted more heavily. Fields with slow cycles (mathematics, geology) use lower λ — older attestations retain relevance.

ϵ is a small positive constant preventing division by zero for claims with no replication attempts.

Rationale: The 2-year IF window is a blunt, field-agnostic cutoff. $R(t)$ replaces it with a continuous, field-calibrated decay that never discards any attestation — it only down-weights it over time.

3.3 D — Dependency Depth

$D(c)$ measures how load-bearing claim c is within the ECG — how many other claims depend on it, directly or transitively:

$$D(c) = |\{c' \in C : c \in \text{ancestors}(c')\}|$$

Where $\text{ancestors}(c')$ is the transitive closure of the dependency relation E from c' .

A claim that is the foundation for hundreds of downstream claims has high D . A terminal claim with no dependents has $D = 0$, contributing only its own survival to the graph.

Rationale: D captures the structural importance of a claim independent of citation velocity. A seminal theoretical result may have low citation counts in fast fields but high D if many empirical claims depend on it.

3.4 S — Survival Rate

$S(c)$ is the ratio of supporting attestations to all non-Mentions attestations:

$$S(c) = \frac{|\text{Supports}(c)| + |\text{Replicates}(c)|}{|\text{Supports}(c)| + |\text{Contradicts}(c)| + |\text{Replicates}(c)| + |\text{Refutes}(c)| + \varepsilon}$$

$S \in (0, 1)$. A claim that has been replicated 10 times with no contradictions has $S \rightarrow 1$. A claim that has been refuted more than supported has $S \rightarrow 0$.

Retraction effect: When a claim is formally retracted, S is set to 0 and the retraction event triggers penalty propagation (Section 5).

3.5 O — Real-World Outcome Linkage

$O(c)$ measures whether claim c has a traceable connection to a real-world outcome outside the academic system:

$$O(c) \in \{0\} \cup [0.1, 1.0]$$

$O = 0$ if no real-world linkage exists. $O > 0$ if the claim is linked to at least one registered oracle source. The specific value depends on oracle type and outcome strength:

```
OracleSource :=
  | TrialRegistry(nct_id, outcome_reported: bool)
  | SystematicReviewProtocol(prospéro_id, review_completed: bool)
  | PolicyDocument(doi, adoption_confirmed: bool)
  | Patent(patent_id, granted: bool)
  | OpenAlexPolicyLinkage(doi)
```

$O = 0$ does not penalize basic science. It signals that the claim has not yet propagated to application. $O > 0$ signals real-world consequence. These are different tracks, honestly labeled.

3.6 Normalized Field Score

$W(c, t)$ values are not directly comparable across fields due to different scales of D and citation norms. The normalized score for comparison purposes:

$$W_{\text{norm}}(c, t, \text{field}) = W(c, t) / \mu_{\text{field}}(t)$$

Where $\mu_{\text{field}}(t)$ is the running mean W across all claims in the same field at time t . This is the field normalization coefficient referenced in `fields/clinical-medicine-v0.1.json`.

4. Attestation Graph Structure

4.1 DAG Invariant

The dependency subgraph (edges in E , claim-to-claim dependencies) must be acyclic. The protocol enforces this at registration time:

```
register_claim(c) requires:  
   $\forall \text{ dep} \in c.\text{depends\_on}: \text{dep} \in C \wedge c \notin \text{ancestors}(\text{dep})$ 
```

This check is performed in `pallet-claim-registry` before accepting any new claim. A claim that would create a cycle is rejected with `Error::DependencyCycle`.

Attestation edges (A) are not subject to the acyclicity constraint — a claim may attest to any registered claim regardless of dependency structure.

4.2 Edge Semantics

Dependency edge ($c_1 \rightarrow c_2$ in E):

- c_1 builds upon c_2
- c_1 's validity is partially conditional on c_2 's survival
- retraction of c_2 triggers penalty propagation to c_1

Attestation edge ($a: c_1 \leftarrow c_2$ in A , typed):

- an attester asserts a verdict about c_1 based on c_2 as evidence
- the direction is: attestation points to the claim being evaluated
- the evidence (c_2 or external) is referenced via `arweave_tx`

4.3 Graph Consistency

The ECG maintains the following invariants:

1. Every node in C has a registered on-chain entry with a valid Arweave content pointer
 2. Every node in A has a registered on-chain attestation with attester SBT verified at attestation time
 3. The dependency subgraph is acyclic (enforced at registration)
 4. No attester may attest to the same claim twice (one attestation per attester per claim)
 5. An attester may not attest to their own claim
-

5. Penalty Propagation

5.1 Retraction Event

When a claim c is retracted — either by the submitter or by governance vote — a `ClaimRetracted` event is emitted on-chain. This triggers the penalty propagation algorithm in the indexer.

5.2 Propagation Algorithm

```
propagate_retraction(c_retracted, G):  
  affected = {}  
  queue = direct_dependents(c_retracted, G)  
  
  while queue is not empty:  
    c_dep = dequeue(queue)  
    penalty = compute_penalty(c_dep, c_retracted, G)
```

```

    apply_penalty(c_dep, penalty)
    affected.insert(c_dep)

    if W(c_dep) falls below threshold_0:
        queue.extend(direct_dependents(c_dep, G))

    return affected

```

The penalty for a dependent claim is proportional to how load-bearing the retracted claim was in its dependency set:

$$\text{penalty}(c_dep, c_retracted) = W(c_dep) \times (\text{weight_of_}c_retracted_in_dep_set / \text{total_dep_weight})$$

5.3 Rationale

In the current system, retraction has near-zero career consequence. A retracted paper continues to be cited positively for years after retraction (a documented phenomenon). Apodokimos makes retraction structurally consequential — not as punishment, but as accurate signal propagation. The graph corrects itself.

Researcher SBT scores are decremented proportionally to their contribution to retracted claims (Section 6.3). The submitter of a retracted claim bears the largest penalty.

6. Soulbound Token Reputation System

6.1 Design Principles

Reputation in Apodokimos is: - **Identity-bound** — attached to a DID, not an address or token balance - **Non-transferable** — cannot be sold, delegated, or concentrated - **Field-specific** — a clinician's SBT score in clinical medicine does not grant governance weight in mathematics - **Earned through survival** — score increments when your claims survive or your attestations are validated; decrements when they do not

This design directly prevents the governance capture that destroys most DAOs: token concentration by institutions or wealthy actors is structurally impossible when the governance weight is a non-transferable function of demonstrated epistemic track record.

6.2 ReputationRecord Structure

```

ReputationRecord := {
    did: DID,
    field_scores: BTreeMap<FieldId, u64>,
    attestation_count: u64,
    claim_count: u64,
    survival_rate: f64, // W(claims_submitted) mean
    last_updated: BlockNumber
}

```

6.3 SBT Lifecycle

Mint: On first accepted attestation. A researcher with no prior on-chain history receives a minimal SBT entry establishing their DID-to-chain linkage.

Increment — attestation validated: When a claim the researcher attested to as Supports or Replicates later receives independent confirmation, the field score increments:

$\Delta \text{score_attestation} = \text{base_increment} \times \text{agreement_weight}$

Increment — claim survived: When a researcher’s submitted claim reaches a survival milestone (e.g., 5 independent replications, O factor linkage confirmed), the field score increments:

$\Delta \text{score_claim_survival} = \text{base_increment} \times W(\text{claim}) \times \text{survival_milestone_multiplier}$

Decrement — retraction cascade: When a submitted claim is retracted or a supported claim is refuted:

$\Delta \text{score_penalty} = -\text{base_increment} \times \text{penalty_weight} \times \text{submitter_multiplier}$

The $\text{submitter_multiplier} > 1$ for the original claim submitter and $= 1$ for attesters. False positive attestations are penalized less severely than fraudulent claim submission.

Non-transferability enforcement: The transfer extrinsic in pallet-sbt-reputation is disabled at the runtime level — not merely gated by access control, but absent from the dispatchable function set. There is no code path by which an SBT score can be transferred.

7. Governance

7.1 Epistemic-Weighted Voting

All protocol governance uses epistemic-weighted SBT voting. Vote weight for a proposal in field F:

$\text{vote_weight}(\text{account}, F) = \text{sqrt}(\text{field_score}(\text{account}, F))$

Quadratic weighting serves two purposes: 1. Diminishing returns on high concentration — a researcher with $4\times$ the SBT score has only $2\times$ the vote weight 2. Resistance to Sybil fragmentation — splitting identity across accounts provides no advantage because sqrt is subadditive: $\text{sqrt}(n) < n \times \text{sqrt}(1)$

Cross-field proposals (protocol parameter changes not specific to any field) use the geometric mean of all field scores:

$\text{vote_weight}(\text{account}, \text{global}) = \text{sqrt}(\text{geomean}(\text{field_scores}(\text{account})))$

7.2 Proposal Types

ProposalType :=

- | ParameterChange(parameter_id, old_value, new_value)
- | FieldSchemaAdd(field_id, schema_arweave_tx)
- | OracleWhitelistUpdate(oracle_id, action: Add | Remove)
- | PenaltyParameterUpdate(parameter_id, value)
- | EmergencyRetraction(claim_id, evidence_arweave_tx)

7.3 Proposal Lifecycle

Proposed

- Voting (duration: governance_period_blocks)
 - Quorum met + majority Yes → Passed
 - Quorum not met or majority No → Rejected
- (if Passed) Timelock (execution_delay_blocks)
- Enacted

Quorum threshold and execution delay are themselves governable parameters, set conservatively at genesis: - Initial quorum: 10% of total active SBT weight - Initial execution delay: 7 days (in blocks) - Voting period: 14 days (in blocks)

7.4 Capture Resistance

The governance mechanism is resistant to capture by design:

- **No token purchases** — SBTs cannot be bought; governance weight requires demonstrated epistemic contribution
 - **Field specificity** — an institution that publishes heavily in one field cannot control governance in another
 - **AGPL-3.0** — any fork of the protocol must remain open; a captured fork cannot be privatized
 - **On-chain governance** — protocol changes require on-chain votes, not maintainer approval; no single repository owner can unilaterally change the protocol
-

8. Identity Layer

8.1 W3C Decentralized Identifiers

All participants in Apodokimos are identified by W3C-compliant Decentralized Identifiers (DIDs). No real-name requirement exists at the protocol level.

DID method: `did:substrate:apodokimos:<account_id>`

A DID document anchored on-chain at registration associates the DID with: - A public key for signing claims and attestations - Optional credential proofs (see Section 8.2) - Optional service endpoints

8.2 ZK Credential Proofs

The protocol supports anonymous-but-credentialed participation via zero-knowledge proofs. A researcher can prove they satisfy a field SBT minimum threshold or hold an external credential (e.g., medical license, PhD in relevant field) without revealing identity.

The ZK proof scheme:

Prover holds: credential C issued by authority A

Verifier wants: proof that prover holds credential in category K

Prover reveals: `ZKProof(C, K)` — proves membership without revealing C or identity

This enables, for example: - Proving board-certified physician status for clinical medicine field attestations - Proving PhD in relevant field without revealing institution affiliation - Meeting minimum SBT threshold without revealing absolute score

The specific ZK scheme (Groth16 or PLONK) is defined at v0.2.0 when the DID pallet is implemented.

8.3 Privacy Model

On-chain storage contains: - Claim hashes (not claim content) - Attestation records with DID references (not personal data) - SBT scores per field (not personal data by default)

Claim content and review payloads are stored on Arweave, referenced by hash. Personal data is never written on-chain. This design is compatible with GDPR: the right to erasure applies to personal data, and personal data exists off-chain where it can be removed. The on-chain hash

of a deleted document becomes an orphaned pointer — the claim entry persists (attestations already made remain valid) but the content is no longer retrievable.

9. Arweave Content Layer

9.1 Rationale

Claim content must be: - **Permanent** — a claim registered today must be retrievable in 100 years - **Immutable** — content cannot be altered after registration - **Decentralized** — not dependent on Apodokimos infrastructure remaining operational - **Content-addressed** — the on-chain hash uniquely identifies the content

Arweave's permaweb satisfies all four constraints. Miners are economically incentivized to store data permanently via the endowment model. No Apodokimos server needs to run for claim content to remain accessible.

9.2 Canonical Claim JSON Schema

All claims are serialized to a canonical JSON format before upload. Canonical means deterministic: same logical content always produces the same byte sequence, and therefore the same blake3 hash. Field ordering is alphabetical. Floating point values use a fixed precision representation.

Arweave transaction tags for a claim upload:

```
{
  "App-Name":      "apodokimos",
  "App-Version":   "0.1.0",
  "Content-Type":  "application/json",
  "Claim-Type":    "<ClaimType variant>",
  "Field-Id":      "<field_schema_id>",
  "Claim-Hash":    "<blake3_hex_of_canonical_json>",
  "Schema-Version": "<field_schema_version>"
}
```

The Claim-Hash tag enables content verification on fetch: the client recomputes the blake3 hash of the downloaded content and compares it to the tag. A mismatch indicates corruption or tampering and the client rejects the content.

9.3 Attestation Content

Attestation evidence is also stored on Arweave, structured as:

```
{
  "claim_id":      "<ClaimId>",
  "verdict":       "<AttestationVerdict>",
  "narrative":     "<reviewer's reasoning in free text>",
  "evidence":      [
    {
      "type":       "ArweaveRef | ExternalDOI | TrialRegistryId",
      "ref":        "<reference>"
    }
  ],
  "field_id":      "<FieldId>",
}
```

```
"schema":      "<attestation_schema_version>"
}
```

The narrative field provides qualitative context that the on-chain attestation record cannot. It is permanently retrievable but does not affect $W(\text{claim})$ computation directly.

10. Substrate Parachain Architecture

10.1 Runtime Overview

The Apodokimos chain is a Substrate parachain implementing four custom FRAME pallets on top of standard Substrate primitives (Balances, System, Timestamp, Identity).

10.2 pallet-claim-registry

Storage:

```
Claims: StorageMap<ClaimId, ClaimRecord>
ClaimCount: StorageValue<u64>
FieldClaims: StorageMap<FieldId, Vec<ClaimId>>
```

ClaimRecord:

```
ClaimRecord := {
  claim_hash:   H256,           // blake3 hash
  arweave_tx:   ArweaveTxId,
  field_id:     FieldId,
  submitter:    DID,
  depends_on:   BoundedVec<ClaimId, MaxDependencies>,
  status:       ClaimStatus,
  registered:   BlockNumber,
  deposit:      Balance
}
```

```
ClaimStatus := Active | Retracted | Superseded(ClaimId)
```

Extrinsics:

```
register_claim(claim_hash, arweave_tx, field_id, depends_on) -> ClaimId
  requires: deposit >= MinClaimDeposit
  requires: all depends_on in Claims
  requires: no cycle introduced in dependency DAG
  emits: ClaimRegistered { claim_id, submitter, field_id, block }
```

```
retract_claim(claim_id)
  requires: caller == claim.submitter OR governance vote
  emits: ClaimRetracted { claim_id, block }
  triggers: penalty propagation via indexer
```

10.3 pallet-attestation

Storage:

```
Attestations: StorageDoubleMap<ClaimId, DID, AttestationRecord>
ClaimAttestationCounts: StorageMap<ClaimId, AttestationCounts>
```

```

AttestationCounts := {
  supports:    u32,
  contradicts: u32,
  replicates:  u32,
  refutes:     u32,
  mentions:    u32
}

```

Extrinsics:

```

attest(claim_id, verdict, evidence_arweave_tx) -> AttestationId
  requires: caller DID has field SBT score >= MinAttestationThreshold(field_id)
  requires: caller != claim.submitter
  requires: no prior attestation from caller on this claim
  emits: AttestationRecorded { claim_id, attester, verdict, block }
  side_effect: if first attestation on claim, refund claim.deposit to submitter

```

10.4 pallet-sbt-reputation

Storage:

```

Reputation: StorageMap<DID, ReputationRecord>
FieldLeaderboard: StorageMap<FieldId, BoundedVec<(DID, u64), MaxLeaderboardSize>>

```

Extrinsics:

```

// All mint/increment/decrement are permissioned to indexer account or governance
mint_sbt(did, field_id)
  requires: caller == IndexerAccount OR governance
  requires: no existing SBT for (did, field_id)

increment_score(did, field_id, delta)
  requires: caller == IndexerAccount
  requires: SBT exists for (did, field_id)

apply_penalty(did, field_id, delta)
  requires: caller == IndexerAccount
  requires: SBT exists for (did, field_id)
  ensures: score >= 0 (floor at zero, no negative scores)

// Transfer is explicitly absent – not gated, not present
// There is no transfer extrinsic in this pallet

```

10.5 pallet-governance

Storage:

```

Proposals: StorageMap<ProposalId, Proposal>
Votes: StorageDoubleMap<ProposalId, DID, Vote>
ProposalCount: StorageValue<u64>

```

Extrinsics:

```

submit_proposal(proposal_type, description_arweave_tx)
  requires: caller SBT field_score >= MinProposalThreshold

vote(proposal_id, vote: Yes | No | Abstain)
  requires: caller has SBT

```

```

weight: sqrt(geomean(field_scores(caller)))

execute_proposal(proposal_id)
  requires: proposal.status == Passed
  requires: block >= proposal.enacted_at + ExecutionDelay

```

11. Bootstrap Strategy: Clinical Medicine

11.1 Rationale for Domain-Specificity

A general-purpose claim registry with no initial population has no signal. The coordination problem demands a domain where:

1. Claim granularity is already standardized (reduces the granularity problem)
2. Real-world outcome oracles already exist (enables O factor from day one)
3. The IF failure is most documented and felt most acutely
4. A motivated community exists outside the prestige economy

Clinical medicine satisfies all four. The PICO (Population, Intervention, Comparator, Outcome) framework provides a standardized claim template already used globally in evidence-based medicine. Trial registries (ClinicalTrials.gov, WHO ICTRP) provide structured real-world outcome data. The 85% waste estimate (Chalmers & Glasziou, 2009) and the replication crisis in clinical research are well-documented motivators.

11.2 PICO Claim Schema v0.1

```

{
  "$schema": "https://apodokimos.org/fields/clinical-medicine-v0.1.json",
  "field_id": "clinical-medicine",
  "schema_version": "0.1.0",
  "license": "CC0-1.0",
  "claim_template": {
    "population": {
      "type": "string",
      "description": "Patient population or study participants",
      "required": true
    },
    "intervention": {
      "type": "string",
      "description": "Intervention, exposure, or treatment",
      "required": true
    },
    "comparator": {
      "type": "string",
      "description": "Control or comparison group",
      "required": false
    },
    "outcome": {
      "type": "string",
      "description": "Primary outcome measure",
      "required": true
    },
    "effect_direction": {
      "type": "enum",

```

```

    "values": ["positive", "negative", "null", "mixed"],
    "required": true
  },
  "effect_size": {
    "type": "number",
    "description": "Point estimate of effect size",
    "required": false
  },
  "confidence_interval": {
    "type": "array",
    "items": "number",
    "minItems": 2,
    "maxItems": 2,
    "description": "[lower_bound, upper_bound]",
    "required": false
  },
  "trial_registry_id": {
    "type": "string",
    "description": "NCT number or equivalent trial registry ID",
    "required": false
  },
  "prospero_id": {
    "type": "string",
    "description": "PROSPERO systematic review registration ID",
    "required": false
  }
}
}
}

```

11.3 O Factor Oracles for Clinical Medicine

ClinicalTrials.gov oracle:

input: NCT ID from claim.trial_registry_id
output: OFactorScore

- 0.3 if trial registered but not yet completed
- 0.6 if trial completed, results posted
- 0.9 if trial completed + results published + consistent with claim
- 0.1 if trial completed + results inconsistent with claim

PROSPERO oracle:

input: PROSPERO ID from claim.prospero_id
output: OFactorScore

- 0.2 if protocol registered, review not completed
- 0.7 if systematic review completed and published
- 0.9 if systematic review completed + meta-analysis supports claim
- 0.1 if systematic review completed + meta-analysis contradicts claim

Both oracles are queried by the indexer via public APIs. Oracle sources are whitelisted by governance vote. Multiple oracle sources for the same claim produce $O = \max(\text{source_scores})$ to avoid penalizing claims that only have one linkage type.

11.4 Bootstrap Sequence

The bootstrap pilot registers a set of well-characterized clinical claims where the evidence record is already established — both the original claims and their replication/refutation history. This seeds the ECG with a non-trivial graph immediately, demonstrating the scoring model on real data before any new claims are submitted.

Target seed dataset: 50 claims from Cochrane systematic reviews, covering well-replicated interventions and well-documented failures. All claims submitted by pilot researchers who co-design the process.

The Apodokimos whitepaper itself (this document) is registered as the first ECG claim on testnet at P-11n. Protocol validates its own founding document.

12. Security Analysis

12.1 Sybil Attack on Reviewer Identity

Vector: An adversary creates many DIDs to flood attestations for or against a claim.

Mitigation: - Minimum SBT threshold required to attest — a new DID with no track record cannot attest - ZK credential proofs for field participation — proving credentials costs effort, not just gas - SBT scores are slow to accumulate (earned through survival, not purchase) — Sybil accounts cannot rapidly gain attestation rights - Quadratic voting means that 100 accounts with score 1 have the same governance weight as 10 accounts with score 10 — Sybil fragmentation provides no advantage

12.2 Governance Plutocracy

Vector: A wealthy actor accumulates governance control.

Mitigation: - SBTs are non-transferable — cannot be purchased on a market - Governance weight is $\sqrt{\text{field_score}}$ — diminishing returns on concentration - Field specificity — no single actor can dominate cross-field governance without demonstrated expertise across all fields - AGPL-3.0 — even if governance is captured, a community fork must remain open and can restore the original governance model

12.3 Claim Spam

Vector: Adversary floods the registry with low-quality claims to degrade signal-to-noise ratio.

Mitigation: - Registration deposit — refunded on first attestation, burned if claim receives no attestations within a governance-defined window - Field schema validation — claims that do not conform to the field schema are rejected - SBT threshold for registration — very low threshold at launch, raiseable by governance

12.4 Oracle Manipulation

Vector: An adversary manipulates an oracle data source to inflate O factor scores.

Mitigation: - Oracle sources are whitelisted by governance — adding a manipulated oracle requires SBT-weighted vote - Multiple oracle sources per claim — $O = \max(\text{source_scores})$ means one manipulated source cannot pull down legitimate scores, but also means a single compromised source cannot inflate a score if other sources disagree - Oracle results are logged

on-chain — manipulation is auditable - ClinicalTrials.gov and PROSPERO are public government/institutional databases — manipulation requires compromising external infrastructure, not just the Apodokimos oracle layer

12.5 GDPR Right to Erasure

Vector: A researcher requests deletion of personal data, but data is on an immutable chain.

Mitigation: - No personal data is written on-chain — only claim hashes, DID references, and SBT scores - Claim content and review payloads are on Arweave, off the chain - DID-to-identity mapping is controlled by the researcher — they can abandon a DID - Arweave content can be removed by the uploader (though permanence guarantees are lost) — the on-chain hash becomes an orphaned pointer - The on-chain record of a deleted claim's attestations remains valid — the attestations happened; their content evidence becomes unavailable but their verdicts persist

This design is compliant with GDPR by construction: the chain stores no personal data to erase.

12.6 Chain Capture

Vector: A large institution forks the chain and operates a modified version that re-introduces IF-equivalent metrics.

Mitigation: - AGPL-3.0 requires any hosted fork to publish source — the fork is visible and auditable - Protocol governance is on-chain — the fork would need to either replicate on-chain governance (in which case the community can vote in it) or introduce centralized control (which is visible and defeats the credibility of the fork) - The original chain's claim history is the canonical history — a fork starts from the same state but diverges; the community retains the original - No single entity controls the relay chain slot assignment — Substrate parachain slot is governed by Polkadot governance, not Apodokimos

13. Limitations and Open Problems

The following are known open problems that are not resolved in wp-v0.1 and are acknowledged honestly:

Granularity gaming: A motivated actor could decompose a single result into many micro-claims to artificially inflate D. The field schema mitigates but does not eliminate this. Governance can introduce claim clustering rules at a later version.

O factor automation reliability: The oracle connectors query public APIs that can change structure, go offline, or return inconsistent data. Multi-source O factor and on-chain oracle whitelisting reduce but do not eliminate this risk.

Cold start signal quality: The bootstrap seed dataset provides initial signal but remains limited. Early W(claim) scores in underrepresented fields will have high variance until sufficient attestation density accumulates.

ZK scheme selection: The identity layer ZK proof scheme is deferred to v0.2.0. The choice between Groth16 (efficient verification, trusted setup required) and PLONK (no trusted setup, larger proof size) has security and operational tradeoffs not yet resolved.

Cross-field claim classification: Some claims are genuinely interdisciplinary. The current model assigns a single field_id per claim. Multi-field claims will be addressed in a future schema version.

14. Conclusion

Apodokimos does not propose a better metric for papers. It proposes the elimination of the paper as the atomic unit of scientific evaluation, replacing it with the falsifiable claim — the unit that reality actually cares about.

The weight function $W(c, t) = R(t) \times D \times S \times O$ measures four things that matter: whether a claim has been independently replicated, how load-bearing it is in the knowledge graph, whether it has survived falsification attempts, and whether it has propagated to real-world consequence. None of these are captured by IF.

The governance model — quadratic SBT voting, non-transferable, identity-bound — eliminates the capture dynamic that has ended every prior reform attempt. You cannot buy Apodokimos governance. You can only earn it by being right, repeatedly, over time.

The bootstrap strategy acknowledges the coordination problem honestly. Apodokimos does not launch as a universal replacement for IF. It launches in clinical medicine, where the tools for claim standardization and real-world outcome measurement already exist, where the evidence of IF failure is most quantified, and where the motivation to exit the prestige economy is highest.

The protocol is owned by no one. It is licensed under AGPL-3.0. Any fork must remain open. The first claim it registers is itself.

References

- Chalmers, I., & Glasziou, P. (2009). Avoidable waste in the production and reporting of research evidence. *The Lancet*, 374(9683), 86–89.
- Garfield, E. (1955). Citation indexes for science. *Science*, 122(3159), 108–111.
- Ioannidis, J. P. A. (2005). Why most published research findings are false. *PLOS Medicine*, 2(8), e124.
- Open Science Collaboration. (2015). Estimating the reproducibility of psychological science. *Science*, 349(6251).
- San Francisco Declaration on Research Assessment (DORA). (2013). <https://sfedora.org/read/>
- Hicks, D., Wouters, P., Waltman, L., de Rijcke, S., & Rafols, I. (2015). Bibliometrics: The Leiden Manifesto for research metrics. *Nature*, 520, 429–431.
- Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.
- Wood, G. (2016). Polkadot: Vision for a heterogeneous multi-chain framework.
- Buterin, V., et al. (2022). Decentralized society: Finding Web3’s soul.
- Weidener, L., & Spreckelsen, C. (2024). What is decentralized science? *Frontiers in Blockchain*.
-

Appendix A: Notation Summary

Symbol	Meaning
$G = (C, A, E)$	Epistemic Contribution Graph
C	Set of registered claims (nodes)
A	Set of attestations (typed edges)
$E \subseteq C \times C$	Dependency relation
$W(c, t)$	Epistemic weight of claim c at time t
$R(c, t)$	Replication score with time decay
$D(c)$	Dependency depth (ancestor count)
$S(c)$	Survival rate (supporting / total non-Mentions)
$O(c)$	Real-world outcome linkage score
λ_{field}	Field-specific time decay constant
ε	Smoothing constant (small positive)
Θ	Retraction cascade threshold
$\mu_{\text{field}}(t)$	Running mean W for field normalization

Appendix B: Versioning

This whitepaper is versioned independently of the protocol software.

WP Version	Status	Notes
wp-v0.1	Draft	Initial release, all sections present, ZK scheme TBD
wp-v0.2	Planned	DID method + ZK scheme defined; governance parameter calibration
wp-v1.0	Planned	Coincides with protocol v1.0 mainnet; audited, stable