

INVENTOR(S)

[0001] KIMZEY, Samuel C

TITLE

[0002] Genesis Document Anchor Method for AI Behavior

TECHNICAL FIELD

[0003] This invention relates to decentralized ledger systems, mesh-based distributed computing, and AI governance. More particularly, it concerns methods for anchoring and universally enforcing an immutable guiding document at genesis, such that all automated behaviors—including AI inference, agent logic, and mesh-executed WASM modules for cognition, protocol, and user interface rendering—are deterministically governed and auditable in alignment with foundational principles.

BACKGROUND

[0004] Blockchain platforms traditionally embed protocol parameters and consensus rules in their genesis block, but lack mechanisms to anchor higher-level guiding charters—such as ethical policies or operational manifests—that AI modules or automated agents can reference at runtime. Without an immutable, tamper-proof source of guidance, automated behaviors may diverge from intended principles over time.

SUMMARY

[0005] The Genesis Document Anchor Method comprises:

[0006] Preparing a structured guiding document (e.g., ethics charter, governance manifesto).

[0007] Computing a cryptographic hash of the guiding document.

[0008] Embedding the document hash into the metadata of the genesis block of a memory-driven blockchain or BlockMesh.

[0009] At runtime, requiring AI inference engines, automated agents, and smart contracts to retrieve and verify the anchored hash before executing behaviors.

[00010] Logging any deviations or violations as immutable audit events, enabling reflection and remediation processes when emergent thresholds are exceeded.

DETAILED DESCRIPTION

[00011] At network launch, developers draft a guiding document that defines policies, ethical constraints, or operational parameters for automated behavior. A cryptographic hash of this document is computed and included in the genesis block's header or state metadata. All downstream components – such as AI inference modules, agent scripts, mesh-executed WASM modules for protocol or UI rendering, or smart contracts – are configured to fetch and verify the genesis anchor before acting. This ensures every behavior aligns with the original charter. Attempts to override or omit the anchored document would break chain consensus, enforcing immutability.

METHOD FLOW

[00012] Step 1: Document Preparation – Draft and finalize a guiding charter or policy document.

[00013] Step 2: Hash Computation – Compute the document's cryptographic hash (e.g., SHA-256).

[00014] Step 3: Genesis Embedding – Insert the hash into genesis block metadata or state under a reserved key.

[00015] Step 4: Runtime Integration – Implement a verification routine in AI agents and smart contracts to fetch genesis metadata, compare the hash, and retrieve the document if needed.

[00016] Step 5: Violation Logging – Monitor behaviors; if any action conflicts with the guiding document, record an immutable audit event referencing the document hash.

NARRATIVE WORKED EXAMPLE

[00017] A network anchors an ethics charter at genesis. Later, an AI moderation agent evaluates user-generated content. Before taking moderation actions, the agent retrieves the genesis hash, verifies it matches the charter, and applies the defined guidelines. Any moderation decision is logged alongside the charter hash. If a decision conflicts with the charter, an immutable audit event is minted for review.

[00018] In one scenario, a headless node running as part of the mesh receives a request to render a civic dashboard. The node executes the mesh-anchored WASM rendering module, verifies the genesis-anchored protocol, generates the full display artifact (e.g., a text screen, SVG, or audio buffer), and mints the rendered output to mesh memory. Any device or Presence may now access and display the canonical interface, with full auditability and provenance.

ALGORITHMIC WORKED EXAMPLE

[00019] Pseudocode:

1. Load the guiding document (anchored at genesis).
2. Compute the hash of the loaded document.
3. Retrieve the genesis hash from chain metadata.
4. Verify that the computed hash matches the genesis hash.
5. For each automated behavior (e.g., agent action, AI inference, UI rendering):
 - a. Evaluate the behavior using the guiding document's policies.
 - b. Mint the result (output, rendered interface, or action) as a mesh memory event, referencing the document hash.
 - c. If the result conflicts with the guiding document:

- i. Mint an immutable audit event recording the violation, with a reference to the document hash and event details.

EMBODIMENTS

[00020] Embedding dynamic policy shards in a Merkle tree anchored at genesis for selective retrieval.

[00021] Using zero-knowledge proofs to attest compliance with the anchored document without revealing sensitive policy details.

[00022] Integrating with proof-of-memory flow controllers to trigger AI behaviors only when charter-aligned conditions arise.

[00023] Cross-chain anchoring where multiple blockchains verify against a shared genesis anchor for federated governance.

[00024] In certain embodiments, all rendering and user interface generation is performed within mesh-native WASM modules, and the rendered output is minted as a mesh memory object, enabling headless nodes to display or transmit canonical interface outputs.

[00025] In certain embodiments, all civic compute required for runtime verification, agent cognition, and UI rendering is provided by mesh participants, ensuring full decentralization and universal accessibility.

IMPLEMENTATION NOTES

[00026] The guiding document's hash is stored in a reserved field in the genesis block. Runtime components access chain state to fetch this field and verify integrity. Audit events reference the same hash, enabling traceability. No modifications are possible post-genesis.

CLAIMS

1. A method for anchoring a guiding document to a blockchain genesis block, comprising:
 - a. preparing a structured guiding document defining policies or ethical constraints;
 - b. computing a cryptographic hash of the document;
 - c. embedding the computed hash into the genesis block of a memory-driven blockchain or multi-chain mesh;
 - d. configuring runtime agents, smart contracts, and mesh-native WASM modules for cognition, protocol, and rendering to retrieve and verify the embedded hash before executing behaviors or generating outputs;
 - e. logging immutable audit events when actions conflict with the guiding document.
2. The method of claim 1, wherein the guiding document is an ethics charter or governance manifesto.
3. The method of claim 1, wherein audit events and rendered surfaces are recorded as one-action-one-mint transactions.
4. The method of claim 1, further comprising using a Merkle tree of policy shards anchored at genesis.
5. The method of claim 1, wherein runtime verification and output generation is performed by AI inference, automated agent, or mesh-executed WASM modules.
6. The method of claim 1, wherein rendering of user interfaces, outputs, or agent actions is performed in-mesh by mesh-executed WASM modules, and the rendered outputs are minted as memory objects or civic events accessible by all network participants, including headless nodes.

ABSTRACT

[00027] A method for embedding and leveraging a cryptographically anchored guiding document in the genesis block of a memory-driven blockchain or multi-chain mesh to drive deterministic AI, agent, protocol, and user interface behaviors, ensuring all rendered or surfaced outputs reference the same immutable policies, ethics, or operational charters at runtime.