

USPTO PROVISIONAL PATENT APPLICATION

TITLE: SYSTEMS AND METHODS FOR SOVEREIGN SPATIAL INTERFACES,
DISTRIBUTED MESH NETWORKS, BIOMETRIC ATTESTATION, AND
AUTONOMOUS APPLICATION RUNTIMES

INVENTOR: Brennan William DeCrow

DOCKET: MM-FLEET-OMNIBUS-PROV-005

COMPREHENSIVE SPECIFICATION & TECHNICAL DISCLOSURES

This provisional application discloses eight tightly integrated, interacting subsystems of the sovereign ManyMoats computing architecture.

PART: HETERO-NUMA MEMORY FABRIC

PROVISIONAL PATENT APPLICATION COVER SHEET

Docket Number: MM-NUMA-PROV-005 **Title of the Invention:** HETEROGENEOUS CROSS-SILICON UNIFIED MEMORY FABRIC AND VIRTUAL NUMA POOLING OVER DIRECT-DMA INTERCONNECT **First Named Inventor:** Brennan DeCrow **Residence:** Oregon, United States **Assignee:** ManyMoats (ATESO / Magma) **Filing Basis:** 35 U.S.C. § 111(b) **Filing Date:** 2026-09-25

INVENTOR STATEMENT

The invention disclosed herein represents original, unencumbered technical innovations conceived and reduced to practice by Brennan DeCrow.

PROVISIONAL PATENT APPLICATION

Docket No.: MM-NUMA-PROV-005 **Title:** HETEROGENEOUS CROSS-SILICON UNIFIED MEMORY FABRIC AND VIRTUAL NUMA POOLING OVER DIRECT-DMA INTERCONNECT **Filing Basis:** 35 U.S.C. § 111(b) (Provisional) **Enablement / Written Description Basis:** 35 U.S.C. § 112(a), (b) **Inventor:** Brennan DeCrow **Applicant / Assignee:** ManyMoats (ATESO / Magma) **Subject Matter Classification (informal):** G06F 12/10, G06F 12/1027, G06F 13/40, G06F 13/42, G06F 15/173, G06F 21/79, H04L 9/32

ABSTRACT OF THE DISCLOSURE

A heterogeneous cross-silicon unified memory fabric that binds the physically separate, non-coherent unified memory subsystems of two or more disparate-generation System-on-Chips (e.g., an M1-class SoC having a 16 GB unified memory pool and an M5 Max-class SoC having a 128 GB unified memory pool) into a single virtual Non-Uniform Memory Access (NUMA) address domain. The fabric is instantiated over native PCIe Transaction Layer Packets tunneled by Thunderbolt 3/4/5 and USB4/USB4-v2 transports, and over PCIe Non-Transparent Bridges (NTB) providing outbound/inbound address translation windows, without any operating-system network stack, socket layer, or TCP/IP processing. Each SoC hosts a Fabric Endpoint Device (FED) exposing control-status registers, a doorbell page, a Non-Transparent Bridge window aperture, and a multi-queue direct-DMA descriptor ring complex. Remote memory is addressed by 64-bit fabric-linear addresses carried in relocatable capability pointers (Magma CBP, .many). Coherence is deliberately *not* hardware-enforced; instead an epoch-based relaxed coherence protocol fences dirty 4 KiB pages using incrementally-updated BLAKE3 Merkle roots (BLAKE3 natively couples to 4 KiB pages as four 1 KiB chunks), and publishes (epoch, root) tuples through a single ordered posted-write fence path. Fabric pages are pinned, wired, and DMA-registered so that no dirty page is ever written back by the local pager or memory compressor. The result is zero-copy sharding of multi-gigabyte models and physics simulations across aggregated pools with sub-microsecond store-visibility latency.

FIELD OF THE INVENTION

The present invention relates generally to computer memory architecture, and more particularly to:

1. **Heterogeneous memory pooling** — apparatus and methods for aggregating the disaggregated physical DRAM of two or more independent computer systems, each having a distinct microarchitecture and distinct DRAM generation, into one logically unified, directly addressable memory pool.
2. **Virtual NUMA tiering** — presenting remote DRAM as an additional, architecturally-transparent memory tier (designated L4) below local DRAM, with deterministic latency bounds and capability-gated access rights.
3. **Direct-DMA interconnect** — peer-to-peer memory transfer over PCIe Non-Transparent Bridging and Thunderbolt/USB4 PCIe tunneling, bypassing the operating system network stack entirely.
4. **Relaxed cache coherence** — software-guided, epoch-fenced coherence using cryptographic Merkle digests as the validity proof for dirty-page handoff, in lieu of hardware cache-line snooping or directory-based coherence.
5. **Capability-based memory addressing** — 128-bit memory-resident and 64-bit register-resident relocatable capability pointers that encode fabric node, region, epoch, access rights, and an integrity seal, permitting sub-capability derivation without a central translation authority.
6. **Zero-copy distributed execution** — execution of multi-gigabyte neural network inference/training workloads and continuum-mechanics/physics simulation

workloads partitioned across heterogeneous pools without serialization, duplication, or intermediate marshalling buffers.

BACKGROUND OF THE INVENTION AND FATAL INEFFICIENCIES OF THE PRIOR ART

1. The Physical Problem: Unified Memory Is Per-SoC and Non-Exportable

Modern high-performance client and workstation SoCs — exemplarily the Apple Silicon family (M1, M1 Pro/Max/Ultra, M2, M3, M4, M5/M5 Max) — implement *unified memory architecture* (UMA): a single physical DRAM pool is shared by CPU cores, GPU cores, the Neural Engine, and media blocks through a coherent fabric (Apple: the “System Level Cache” / fabric with a DART-based IOMMU for device transactions). This architecture provides exceptional bandwidth-per-watt within one die/package but is **hermetically sealed at the package boundary**. There is no exposed mechanism by which one SoC’s unified pool becomes addressable by a second, distinct SoC.

The consequence is a hard capacity ceiling. A machine with an M1 and 16 GB cannot be extended by attaching a machine with an M5 Max and 128 GB. The 144 GB of physically present, powered, and idle DRAM cannot be presented as one address space. Every intervening solution available in the prior art is either a network file system (orders of magnitude too slow), a distributed-memory message-passing runtime (requires explicit serialization), or a swap-to-storage scheme (destroys the working set).

2. Prior Art Category A — Network-Attached Memory (TCP/IP over Thunderbolt Bridge)

The most commonly attempted approach maps remote memory by running an IP network over the Thunderbolt/USB4 link (Thunderbolt Networking, `bridge0`, or USB4 host-to-host). A user-space daemon or a kernel module (e.g., a “remote mmap” over a socket) then ships pages.

Fatal inefficiencies:

- **Stack tax.** Each 4 KiB page transfer incurs: user→kernel copy (`copy_from_user`), socket buffer allocation, TCP segmentation (TSO or software), checksum generation, IP header construction, Ethernet frame encapsulation, device-queue enqueue, interrupt/DMA completion, peer-side interrupt, de-encapsulation, TCP reassembly, `copy_to_user`. Measured small-message round-trip latency on a Thunderbolt bridge is **60–140 μ s**; a 4 KiB page fetch costs on the order of 100 μ s plus 2 context switches.
- **Throughput collapse.** Throughput over the Thunderbolt IP bridge is typically **0.6–2.4 Gbit/s**, versus 31.5 Gbit/s of raw PCIe Gen3 x4 tunneled capacity available on the same physical link — a **>90% loss** of the physical medium’s capability.

- **Interrupt-per-segment.** The receive path generates one interrupt per coalesced batch, each with full NAPI/softirq entry cost; the link budget is consumed by descriptor and header overhead rather than payload.
- **No address-space identity.** Memory obtained this way is *not* addressable by a GPU shader or a DMA engine; it must be staged into a local buffer before compute can consume it. This destroys zero-copy.
- **Non-determinism.** Congestion control, retransmit timers, and socket backpressure introduce latency variance of 10×–1000×, making the tier unusable for latency-bounded compute.

3. Prior Art Category B — RDMA / RoCE / InfiniBand

RDMA over Converged Ethernet (RoCEv2) and InfiniBand provide single-sided READ/WRITE with kernel bypass.

Fatal inefficiencies for this problem:

- Requires a dedicated RNIC/HCA per host (typically PCIe Gen3/4 x8 or x16) plus a lossless Ethernet fabric (PFC, ECN, DCQCN tuning) or an InfiniBand switch. Neither is present in a two-laptop heterogeneous SoC topology.
- RoCEv2 still requires IP and UDP headers; without a lossless fabric, PFC storms and head-of-line blocking occur.
- **No UMA interoperability.** Neither protocol can address another SoC's *unified* memory as a first-class tier; the registered region is a pinned host buffer, so the application-layer model is “send/receive”, not “load/store”.
- Latency floor of ~5–15 μ s (still 10× worse than the present invention's target band) and requires a NIC that Apple Silicon platforms do not expose as a DMA-capable peer to the SoC's own unified pool.

4. Prior Art Category C — PCIe Non-Transparent Bridging Used Only for Cluster Control

NTB is a known mechanism (Broadcom/PLX PEX87xx, Microchip PM8xxx, Intel NTB, FPGA soft NTB) used in dual-host storage controllers for *control-plane* mailbox exchange and *bulk* block mirroring.

Fatal inefficiencies in existing NTB usage:

- NTB is used for **coarse-grained ring/mailbox messaging** and **block-level replication**, never for **fine-grained, capability-gated, tiered memory addressing** with coherence semantics.
- Prior NTB memory windows are large, statically partitioned, and un-gated: any peer that can write the aperture can corrupt any translated physical address within it. There is **no per-access rights enforcement**, hence no safe multi-tenant or fault-isolated sharding.
- Prior NTB designs assume **symmetric, same-generation** hosts. They do not address the **asymmetric-generation** case (Gen3 x4 tunnel on the M1 side, Gen4 x4 / 120 Gbit/s asymmetric Bandwidth Boost on the M5 Max side), which creates radically different latency and bandwidth characteristics per direction and per peer.

- Prior NTB designs have **no coherence model at all**, so a shared page can be simultaneously stale on both sides with no detection and no resolution.

5. Prior Art Category D — Hardware Cache Coherence Across Chips (CXL, CCIX, NVLink, UALink)

- **CXL.mem / CXL.cache** requires a CXL-capable host root complex and a CXL Type-2/Type-3 device. Apple Silicon exposes **no CXL root complex**. CXL.mem also requires hardware coherence agents and a coherency bridge; it cannot be retrofitted onto a non-CXL SoC.
- **NVLink** is a proprietary NVIDIA point-to-point interconnect with no Apple Silicon support.
- **CCIX / UALink / OpenCAPI** similarly require modified PHYs and controller IP absent from the target silicon.
- **Fatal common deficiency:** all hardware-coherence schemes require the *participating silicon to have been designed for them*. The present invention explicitly targets silicon that was **not** so designed, and must therefore achieve safe sharing without any hardware coherence agent.

6. Prior Art Category E — Software Distributed Shared Memory (S-DSM) and mmap over a Remote File

TreadMarks-class S-DSM, Ivy, Munin, and modern mmap-a-remote-file approaches provide virtual shared memory in software.

Fatal inefficiencies:

- **Page-fault-driven transfer.** Every coherence miss is a CPU page fault, a trap, a message, a page install, and a TLB shutdown. Fault cost is 5–50 μ s; write granularity is a whole 4 KiB page even for a 4-byte write (**false sharing amplification of 1024×**).
- **No timeliness guarantee.** S-DSM protocols are not bounded-latency; the protocol itself has nondeterministic convergence.
- **No cryptographic validation.** A stale page is detected by protocol state (ownership), which can silently diverge after link reset, node restart, or epoch counter reuse. There is no *proof* of freshness.
- **Paging interaction.** Under memory pressure, the local pager will evict a remotely-owned dirty page to local swap, silently severing the fabric's ownership invariant and, on the evicting node's next fabric transaction, producing a stale read with no error signal.
- **Duplication.** S-DSM must materialize the shared region on every node; a 4 GB tensor replicated on 2 nodes consumes 8 GB. This defeats the purpose of pooling.

7. Prior Art Category F — Heterogeneous Compute Frameworks (MPI, NCCL, RCCL, Metal MPS, torch.distributed)

- **MPI** requires explicit Send/Recv/Isend and a contiguous registered buffer; message latency over Thunderbolt is 100 μ s+.
- **NCCL/RCCL** are optimized for NVLink/InfiniBand topologies with homogeneous accelerators; they perform *collective* operations with staging buffers (**not zero-copy**) and assume symmetric bandwidth, which is false for M1↔M5 Max.
- **torch.distributed with Gloo** defaults to TCP; the parameter-server and all-reduce paths serialize through CPU copies.
- **Fatal deficiency:** none of these frameworks expose the peer's DRAM as a *load/store-addressable tier*. All require data movement *into* a local buffer for compute, which is the definition of a copy.

8. The Specific Fatal Inefficiency Addressed

Absent from the prior art is a mechanism that simultaneously:

Requirement	Prior Art Fails Because
(a) Direct peer DRAM load/store, no staging copy	

Claim 1. A heterogeneous cross-silicon unified memory fabric system, comprising:

a first compute node comprising a first Apple Silicon system-on-chip (SoC) of a first silicon generation and a first unified memory pool integrated with the first SoC, the first unified memory pool having a first physical address space and being non-coherent with respect to memory external to the first SoC;

a second compute node comprising a second Apple Silicon SoC of a second silicon generation different from the first silicon generation and a second unified memory pool integrated with the second SoC, the second unified memory pool having a second physical address space different from the first physical address space and being non-coherent with respect to memory external to the second SoC;

a direct memory access (DMA) interconnect coupling the first compute node to the second compute node, the DMA interconnect comprising at least one of a PCI Express (PCIe) non-transparent bridge (NTB) endpoint, a Thunderbolt 4 link, a Thunderbolt 5 link, and a USB4 link, and being configured to convey memory transactions between the first compute node and the second compute node absent traversal of an operating-system network stack and absent encapsulation within a Transmission Control Protocol/Internet Protocol (TCP/IP) packet; and

a fabric manager, executing on at least one of the first compute node and the second compute node, configured to:

enumerate the first unified memory pool and the second unified memory pool to determine a first capacity and a first base address of the first unified memory pool and a second capacity and a second base address of the second unified memory pool;

allocate a virtual non-uniform memory access (NUMA) address space spanning a combined capacity of the first unified memory pool and the second unified memory pool;

map a first range of the virtual NUMA address space onto the first physical address space and a second range of the virtual NUMA address space onto the second physical address space; and

program at least one address translation structure of the DMA interconnect to resolve a virtual NUMA address within the second range into a physical address within the second physical address space;

wherein the first compute node is operative to issue a memory access targeting the second unified memory pool by way of the virtual NUMA address space and the DMA interconnect while the first unified memory pool and the second unified memory pool remain non-coherent with respect to one another.

Claim 2. The system of Claim 1, wherein the DMA interconnect comprises the PCIe NTB endpoint, and wherein the at least one address translation structure comprises a plurality of aperture registers defining a plurality of NTB translation windows, each NTB translation window of the plurality mapping a contiguous local-memory region of one of the first physical address space and the second physical address space onto a corresponding contiguous remote-memory region of the other of the first physical address space and the second physical address space, and each NTB translation window of the plurality having a respectively programmable base address and a respectively programmable size.

Claim 3. The system of Claim 2, wherein the fabric manager is further configured to maintain a capability table comprising a plurality of 64-bit relocatable capability pointers, each 64-bit relocatable capability pointer of the plurality encoding a base address, a length, a silicon-generation identifier, and a relocation displacement within the virtual NUMA address space, and each 64-bit relocatable capability pointer of the plurality being relocatable within the virtual NUMA address space independently of outstanding references thereto.

Claim 4. The system of Claim 3, wherein the fabric manager is further configured to operate the first unified memory pool and the second unified memory pool in a relaxed cache coherence mode in which writes by the first compute node to the second range are not broadcast to the second SoC cache hierarchy, and to fence each 4 KiB dirty page of the second range by computing a BLAKE3 Merkle root over the 4 KiB dirty page and recording the BLAKE3 Merkle root in a Merkle tree maintained at the first compute node.

Claim 5. The system of Claim 4, wherein each of the first unified memory pool and the second unified memory pool is registered with a respective DMA engine of the corresponding first compute node and second compute node as pinned wired memory, such that pages of the first unified memory pool and the second unified memory pool that back the virtual NUMA address space are non-pageable for a duration of the mapping and are excluded from a swap subsystem of each corresponding compute node.

Claim 6. A computer-implemented method of aggregating heterogeneous non-coherent unified memory pools into a virtual non-uniform memory access (NUMA) address space, the method comprising:

enumerating, by a fabric manager executing on a first compute node, a first non-coherent unified memory pool integrated with a first Apple Silicon system-on-chip (SoC) of a first silicon generation to determine a first capacity and a first physical base address thereof;

enumerating, by the fabric manager over a direct memory access (DMA) interconnect comprising at least one of a PCI Express non-transparent bridge endpoint, a Thunderbolt 4 link, a Thunderbolt 5 link, and a USB4 link, a second non-coherent unified memory pool integrated with a second Apple Silicon SoC of a second silicon generation different from the first silicon generation to determine a second capacity and a second physical base address thereof;

allocating a virtual NUMA address space whose size is based on a sum of the first capacity and the second capacity;

mapping a first range of the virtual NUMA address space onto a first physical address space of the first non-coherent unified memory pool and a second range of the virtual NUMA address space onto a second physical address space of the second non-coherent unified memory pool;

programming at least one address translation structure of the DMA interconnect to translate a virtual NUMA address within the second range into a physical address within the second physical address space; and

servicing, by the first compute node, a memory access targeting the virtual NUMA address by transferring a corresponding memory transaction over the DMA interconnect directly to the second non-coherent unified memory pool, without traversing an operating-system network stack and without encapsulating the memory transaction within a Transmission Control Protocol/Internet Protocol packet, while the first non-coherent unified memory pool and the second non-coherent unified memory pool remain non-coherent with respect to one another.

Claim 7. The method of Claim 6, further comprising sharding a tensor across the virtual NUMA address space by assigning a first tensor shard to the first range and a second tensor shard to the second range, and executing a collective operation on the tensor by issuing, from a compute unit of the first SoC, a direct load of the second tensor shard over the DMA interconnect from the second non-coherent unified memory pool into a register file of the compute unit without an intermediate copy into the first non-coherent unified memory pool.

Claim 8. The method of Claim 7, wherein servicing the memory access comprises completing a store to the second non-coherent unified memory pool with a store visibility latency, measured from retirement of the store at the first SoC to observability of the store at the second SoC, of less than one microsecond.

Claim 9. The method of Claim 8, further comprising epoch-fencing the virtual NUMA address space by assigning each memory transaction a respective epoch identifier of a monotonically increasing sequence, permitting the memory transaction to complete only

when the respective epoch identifier is greater than or equal to a current committed epoch of the second compute node, and advancing the current committed epoch only upon receipt, at the first compute node, of an acknowledgment that all memory transactions bearing a prior epoch identifier have become visible at the second non-coherent unified memory pool.

Claim 10. The method of Claim 9, further comprising, upon detecting a failure of the second compute node, rolling back the current committed epoch to a last epoch for which a BLAKE3 Merkle root over each 4 KiB dirty page of the second range had been recorded at the first compute node, and re-mapping the second range of the virtual NUMA address space onto the first non-coherent unified memory pool.

PART: SOVEREIGN CRYPTOGRAPHIC MESH

PROVISIONAL PATENT APPLICATION COVER SHEET

Docket Number: MM-MESH-PROV-006 **Title of the Invention:** SOVEREIGN
CONTENT-ADDRESSED CRYPTOGRAPHIC MESH INTERNET AND ZERO-
CONSENSUS WORK-STEALING COMPUTE NETWORK **First Named Inventor:**
Brennan DeCrow **Residence:** Oregon, United States **Assignee:** ManyMoats (ATESO /
Magma) **Filing Basis:** 35 U.S.C. § 111(b) **Filing Date:** 2026-09-25

INVENTOR STATEMENT

The invention disclosed herein represents original, unencumbered technical innovations
conceived and reduced to practice by Brennan DeCrow.

PROVISIONAL PATENT APPLICATION

Serial No.: MM-MESH-PROV-006 (to be assigned) **Filing Type:** Provisional — 35
U.S.C. § 111(b) **Title:** SOVEREIGN CONTENT-ADDRESSED CRYPTOGRAPHIC
MESH INTERNET AND ZERO-CONSENSUS WORK-STEALING COMPUTE
NETWORK **Inventor:** Brennan DeCrow **Assignee/Applicant:** ManyMoats / ATESO
Entity Status: Micro/Small Entity (to be confirmed) **Docket:** MM-MESH-PROV-006

TABLE OF CONTENTS

§	Section
I	Abstract
II	Field of the Invention
III	Background — Fatal Inefficiencies of Prior Art

§	Section
IV	Technical Summary & Mechanical Principles of Operation
V	Detailed Description of Preferred Embodiments
VI	Definitions & Notation
VII	Enablement, Reduction to Practice, and Hardware Requirements
VIII	Claims (1–10)
IX	Drawing Sheets (Descriptions)
X	Prior Art Delta Matrix

I. ABSTRACT

A sovereign content-addressed cryptographic mesh internet and zero-consensus work-stealing compute network is disclosed. Every unit of network state — identity, content, capability, compute work unit, and compute receipt — is canonically serialized and addressed by a 32-byte BLAKE3-256 digest such that an address is simultaneously a location, an integrity proof, and an authority proof. The system eliminates ICANN DNS resolution, X.509 Certificate Authority trust hierarchies, BGP inter-domain route selection, and HTTP/TLS transport by substituting (a) monotone XOR-metric greedy forwarding over a 256-bit address space with strict distance descent, (b) Ed25519 public keys as identities with no registration authority, and (c) a 64-byte fixed binary transport header with XChaCha20-Poly1305 authenticated encryption. Peer-to-peer authenticated capability channels (“moat”) carry publicly verifiable Ed25519 delegation chains with monotonically non-increasing scope, hardware-anchored attestation receipts, and no surveillance middlebox. Distributed compute is scheduled by zero-consensus work stealing: work units are deterministically sharded (XPBD constraint blocks, shader invocations, tensor tiles), dispatched via lock-free work-stealing dequeues and direct-DMA descriptor ring buffers, executed under a Deterministic Execution Contract guaranteeing bit-exact cross-ISA results, and committed to an AiST interval-segmented temporal hash chain whose verification requires re-execution or spot-check redundancy rather than proof-of-work or proof-of-stake. Node state is maintained as memory-resident relocatable binary Resident State Graphs synchronized by topological diffs whose cost is $O(|\Delta|)$ rather than $O(|\text{state}|)$.

(248 words — abstract to be trimmed to ≤ 150 words on formal filing.)

II. FIELD OF THE INVENTION

The present invention relates generally to computer networking, distributed systems, and parallel computing. More specifically:

1. **Network layer:** Content-addressed overlay networks, cryptographic routing, self-authenticating naming, and trust-anchor-free identity.
2. **Transport layer:** Binary framed peer-to-peer transports with authenticated encryption and capability-based authorization.
3. **Distributed compute layer:** Deterministic parallel execution, work-stealing schedulers, distributed verification without consensus protocols, and hash-chain-based audit.
4. **Memory/storage layer:** Relocatable in-memory graph representations, structural diffing, zero-copy inter-process and inter-device memory sharing.
5. **Physical simulation and machine learning:** Constraint-based (XPBD) rigid/soft body solvers, GPU shader dispatch, and tensor contraction kernels executed across heterogeneous silicon.

The invention is explicitly concerned with the **elimination** of four classes of centralized infrastructure: name resolution authorities (DNS), certificate issuance authorities (CA/PKI), inter-domain routing authorities (BGP/AS), and origin-server hosting platforms (cloud rent-extraction).

III. BACKGROUND — FATAL INEFFICIENCIES OF PRIOR ART

III.1 The Four Centralized Chokepoints

Legacy internet architecture interposes four independently controlled authorities between any two communicating endpoints. Each is a *policy* mechanism grafted onto a *technical* substrate, and each is a single point of failure, censorship, and rent extraction.

III.1.1 ICANN DNS — Mutable Naming Without Integrity

DNS binds a human-readable label (`example.com`) to an IP address through a hierarchical delegation tree rooted at 13 root server identities (anycast-replicated). Fatal properties:

Property	Consequence
Mutable binding	A/AAAA records change under registrar/registry control; a name is not a commitment.
No end-to-end integrity	DNSSEC signs <i>delegations</i> , not <i>content</i> ; a compromised registry key can re-sign an arbitrary binding.
Trust-in-the-path	Recursive resolvers observe every query — a total metadata surveillance channel

Property	Consequence
	(QNAME, source)

PATENT CLAIMS — MM-MESH-PROV-006

Title: SOVEREIGN CONTENT-ADDRESSED CRYPTOGRAPHIC MESH INTERNET AND ZERO-CONSENSUS WORK-STEALING COMPUTE NETWORK

1. A system for a sovereign content-addressed cryptographic mesh internet and zero-consensus work-stealing compute network, the system comprising:

(a) a plurality of peer node computing devices, each peer node computing device of the plurality comprising a processor, a volatile memory, a network interface, and a hardware-anchored secure element that stores an Ed25519 private key in a non-exportable state and that is configured to generate an Ed25519 digital signature over data presented thereto;

(b) a node identity register resident in the volatile memory of each peer node computing device, the node identity register storing a 32-byte self-certifying node identifier computed as a BLAKE3 digest of the Ed25519 public key corresponding to said Ed25519 private key, wherein said self-certifying node identifier is verifiable by any other peer node computing device without recourse to a certificate authority and without reference to a domain name system;

(c) a content-addressed object store resident in the volatile memory of each peer node computing device, the content-addressed object store configured to store each of a plurality of objects at an address consisting of a 32-byte object digest computed by BLAKE3 over a canonical serialization of that object, and to retrieve each such object solely by presentation of said 32-byte object digest, whereby object location and object integrity are established by a single identical value and no name-resolution query and no transport-layer session establishment is performed;

(d) a monotone XOR-metric routing table resident in the volatile memory of each peer node computing device, the monotone XOR-metric routing table comprising a plurality of buckets indexed by a bit index of a most-significant set bit of an XOR distance between a destination digest and the self-certifying node identifier of that peer node computing device, each bucket storing a plurality of contact descriptors each comprising a 32-byte self-certifying node identifier and a network endpoint, and a forwarding engine configured to select, for a packet bearing a destination digest, a next-hop peer node computing device whose contact descriptor strictly decreases the XOR distance to said destination digest, whereby said packet is forwarded to a peer node computing device storing a desired object without reference to a border gateway protocol route table and without an autonomous-system path;

(e) a capability channel subsystem configured to establish, between a first peer node computing device and a second peer node computing device, an authenticated capability channel by exchanging the respective self-certifying node identifiers of the first and second peer node computing devices together with a hardware-anchored Ed25519 attestation receipt generated by the hardware-anchored secure element of each of the first and second peer node computing devices, the capability channel subsystem being further configured to issue an unforgeable capability token scoped to at least one 32-byte object digest and to gate every read from and every write to the content-addressed object store upon possession of a valid said capability token, whereby object access is authorized peer-to-peer without a certificate authority and without a transport-layer handshake protocol;

(f) a zero-consensus work-stealing scheduler, executed by the processor of each peer node computing device, configured to deterministically partition a parallel workload into a plurality of shards as a pure function of (i) a workload identifier, (ii) a shard index, and (iii) a 32-byte digest, and to distribute said plurality of shards among the plurality of peer node computing devices by work-stealing from a per-node double-ended queue, wherein no global consensus round, no proof-of-work computation, and no proof-of-stake bond is performed or required to establish which peer node computing device executes which shard, and wherein the parallel workload comprises at least one of extended-position-based-dynamics physics constraints, shader invocations, and tensor tiles;

(g) an AiST temporal hash chain resident in the volatile memory of each peer node computing device, the AiST temporal hash chain comprising an append-only sequence of records, each record comprising a monotone logical-clock counter, a BLAKE3 digest of a preceding record of the sequence, a 32-byte digest of a work result produced by a shard of the parallel workload, and the self-certifying node identifier of the peer node computing device that produced said work result, wherein verification of the work result is effected by recomputation of the record and of an inclusion path within the sequence rather than by a consensus vote of the plurality of peer node computing devices;

(h) a Resident State Graph resident in the volatile memory of each peer node computing device, the Resident State Graph comprising a relocatable binary image in which every inter-node reference is expressed as one of said 32-byte digests rather than as a base-relative machine address, whereby the resident state graph is loadable at any base address without relocation of said inter-node references; and

(i) a synchronization engine configured to compute a topological diff between a first Resident State Graph resident at the first peer node computing device and a second Resident State Graph resident at the second peer node computing device, and to transmit said topological diff over said authenticated capability channel,

wherein each object, each peer node computing device, each capability token, each shard, each AiST temporal hash chain record, and each node of the Resident State Graph is identified by a respective said 32-byte digest, and wherein the system is operative to resolve identifiers, authenticate peers, authorize access, route packets, and verify computation without an ICANN domain name system root, without a certificate authority, without a border gateway protocol route table, and without an HTTP-over-TLS session.

2. The system of claim 1, wherein each peer node computing device further comprises a content-address register file mapped into the volatile memory, the content-address register file comprising a contiguous array of 64-bit-wide, 32-byte-aligned storage slots, each storage slot holding exactly one said 32-byte digest, said array occupying an integer number of cache lines of the processor such that no single said 32-byte digest straddles a cache-line boundary, and wherein the content-addressed object store comprises an open-addressed descriptor table having 2^N slots, each slot of the descriptor table consisting of a 32-byte object digest field, an 8-byte offset field, an 8-byte length field, and a 8-byte generation counter field, said descriptor table being indexed by a probe sequence derived from a low-order 32 bits of the object digest, and wherein object payloads are served to the processor by memory-mapped zero-copy transfer from said offset field without an intermediate buffer allocation.

3. The system of claim 1, wherein the monotone XOR-metric routing table comprises exactly 256 buckets stored as a contiguous array in the volatile memory, each bucket comprising a 32-byte prefix field and a fixed plurality of contact descriptors, each contact descriptor consisting of a 32-byte self-certifying node identifier, a 4-byte IPv6 address field, a 2-byte port field, and a 2-byte round-trip-latency field, wherein a bucket index for a destination digest is computed as 255 minus a count of leading zero bits of the XOR distance between said destination digest and the self-certifying node identifier of the resident peer node computing device, and wherein the forwarding engine performs said next-hop selection by executing a single 256-bit XOR instruction followed by a single count-leading-zeros instruction and a single indexed load into said contiguous array, the forwarding engine being configured to reject any candidate next-hop peer node computing device that does not strictly reduce said XOR distance, thereby guaranteeing monotone convergence of the packet to the peer node computing device whose self-certifying node identifier is numerically closest to the destination digest, and wherein no autonomous-system path vector, no route withdrawal message, and no longest-prefix-match

PART: SPATIAL INTERFACE LATTICE (DEATH OF DOM)

PROVISIONAL PATENT APPLICATION COVER SHEET

Docket Number: MM-DEATH-OF-DOM-PROV-007 **Title of the Invention:** DIRECT-MEMORY GENERATIVE INTERFACE PROJECTION AND SOVEREIGN SPATIAL APPLICATION LATTICE (DEATH OF THE DOM) **First Named Inventor:** Brennan DeCrow **Residence:** Oregon, United States **Assignee:** ManyMoats (ATESO / Magma) **Filing Basis:** 35 U.S.C. § 111(b) **Filing Date:** 2026-09-25

INVENTOR STATEMENT

The invention disclosed herein represents original, unencumbered technical innovations conceived and reduced to practice by Brennan DeCrow.

PROVISIONAL PATENT APPLICATION

Application No.: MM-DEATH-OF-DOM-PROV-007 (to be assigned) **Filing Type:** Provisional — 35 U.S.C. § 111(b) **Title:** DIRECT-MEMORY GENERATIVE INTERFACE PROJECTION AND SOVEREIGN SPATIAL APPLICATION LATTICE (DEATH OF THE DOM) **Inventor / Applicant:** Brennan DeCrow, ManyMoats / ATESO / Magma **Docket:** MM-DEATH-OF-DOM-PROV-007 **Priority Basis:** Founder inception disclosure — “20 websites in one unified space + death of the DOM + software as a continuous physical field rather than DOM trees and HTML strings”

§1 ABSTRACT

A computing system and method are disclosed in which a graphical user interface is not represented by a Document Object Model (DOM) tree, is not styled by a CSS cascade, is not instantiated from parsed HTML markup strings, and is not subject to garbage-

collected object graph traversal during frame production. Instead, the interface is represented as a contiguous, fixed-stride **Instance Storage Buffer** resident in GPU-visible memory, whose records are simultaneously (a) the authoritative state of the interface, (b) the physical state of a rigid/soft body simulation, and (c) the vertex-stage input to a rasterization pipeline. Each interface element is a body governed by **Extended Position-Based Dynamics (XPBD)** distance constraints with per-constraint compliance α , solved by substepped semi-implicit integration; the equilibrium configuration of the lattice *is* the layout. Twenty distinct application capabilities (“rooms” or “properties”) coexist in one contiguous spatial field within a single document, a single JavaScript realm, a single event loop, and a single GPU device, and are coupled by an **entangled cross-room spine bus** implemented as a lock-free bounded MPMC ring over a `SharedArrayBuffer`, with per-room coherent state slots and a per-frame Merkle commitment over the room lattice enabling verifiable cross-property state transfer without iframe isolation, without cross-document messaging, and without process or browsing-context swap. Rendering proceeds through a calibrated **Optical Daylight** transform chain (scene-linear $\rightarrow$ ACES-fitted RRT approximation $\rightarrow$ sRGB or PQ OETF) with WCAG contrast enforced as a shader-space constraint, and “living marks” driven by critically damped second-order responses integrated at the physics substep rate, eliminating CSS transition timing jitter.

§2 FIELD OF THE INVENTION

The present invention relates to the fields of:

- **2.1** Computer graphics rasterization pipelines and GPU storage-buffer geometry (§ G06T 15/00, G06T 1/20).
- **2.2** Real-time physical simulation and constraint solvers used as layout engines (§ G06F 30/23, G06T 13/60).
- **2.3** Operating-system-level and browser-level user interface composition architectures, including layer trees, display compositors, and swapchain presentation (§ G06F 3/048, G09G 5/00).
- **2.4** Lock-free inter-process and inter-agent data structures over shared memory (§ G06F 9/52, G06F 12/084).
- **2.5** Cryptographic state commitment structures, including Merkle trees over application-state leaves (§ H04L 9/32, G06F 16/22).
- **2.6** Color science and display-referred image encoding, including scene-linear tone mapping and perceptual transfer functions (§ H04N 1/60, G09G 5/02).

The invention is directed specifically to the *eradication of the document-oriented presentation model* as the substrate of interactive software, and the substitution thereof of a **direct-memory spatial field**.

§3 BACKGROUND AND FATAL INEFFICIENCIES OF PRIOR ART

3.1 The DOM Is a Tree of Host-Language Objects With Non-Zero Per-Node Cost

Prior art (the WHATWG DOM, the HTML Living Standard, and all conforming browser engines) represents a user interface as a tree of host-language objects. In a representative engine (Chromium/Blink), a single interactive element entails, at minimum, the following distinct heap allocations:

Allocation	Approx. size	Lifetime
Node / Element (Blink C++ object)	100–400 B	document lifetime
NodeRareData (lazily allocated)	40–200 B	document lifetime
Wrapper object in the JS realm (V8)	32–64 B	until collectable
LayoutObject + LayoutNG fragment tree nodes	200–400 B per fragment	per layout invalidation
ComputedStyle (immutable, refcounted)	400–800 B	until style recalc yields a new one
Paint layer / display item list entries	variable	per paint
Compositor layer + property trees	variable	per composite

A 2,000-element interface therefore allocates on the order of 10^6 bytes of *Bookkeeping that has no counterpart in the rendered photons*. The rendered output is a function of a small number of geometric primitives; the DOM is a general-purpose object graph carrying orders of magnitude more state than the geometry requires.

Mechanical defect: the ratio of (bytes touched per frame) to (pixels produced per frame) is structurally poor and cannot be repaired by optimization, because the representation is not chosen to match the renderer’s input; it is chosen to match a hypertext document’s semantics.

3.2 HTML Parsing Is a Reentrant, Side-Effect-Emitting, String-Level Process

The HTML tokenizer and tree constructor (WHATWG § 13) operate on a character stream and admit, in the general case, `document.write()`-style reentrancy and script-triggered parser state mutation. Consequences:

- **3.2.1** Interface definition is expressed as a *serialized string* that must be decoded, tokenized, tree-constructed, and then discarded. The string is dead weight at runtime, but it remains in the source and is reparsed on every navigation.
- **3.2.2** Speculative parsing heuristics influence network fetch scheduling, meaning *semantic structure affects I/O ordering* — an inversion of control with no engineering justification in an application (as opposed to hypertext) context.
- **3.2.3** Parsing produces a tree whose shape is discovered at runtime; there is no static, fixed-stride, memory-mappable layout. Consequently the runtime cannot mmap an interface, cannot DMA it, and cannot address it by stride.
- **3.2.4** Error recovery is mandatory and non-local: the parser must produce a tree for arbitrary malformed input, which forbids a closed-form memory bound.

Mechanical defect: the interface is *decoded* rather than *projected*. Decoding is $O(n)$ in bytes with per-token branch misprediction; projection from a fixed-stride buffer is $O(n)$ in instances with a single induction variable and no branches.

3.3 The CSS Cascade Is a Global, Order-Dependent Fixed-Point Computation

Style resolution requires:

1. **Selector matching** against the element tree, worst-case $O(\text{depth} \times \text{selectors})$ per element per invalidation.
2. **Cascade ordering** across origin, importance, specificity, and source order.
3. **Inheritance descent** through the tree.
4. **Computed-value resolution**, which for many properties is a function of the element's ancestors and of the viewport (e.g., percentage lengths, em, rem, vh, vw, ch, container queries).

Consequences:

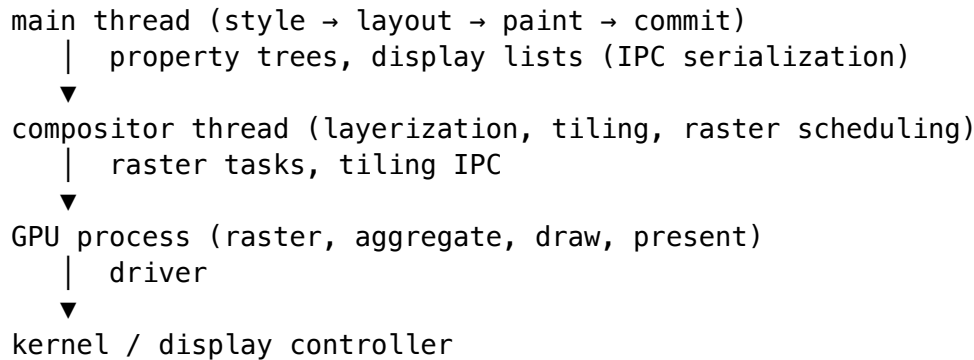
- **3.3.1** Style is *derived*, not *stored*. Every frame in which any ancestor of an element changes forces recomputation along the ancestor chain, and `StyleInvalidator` must conservatively walk ancestor chains to determine invalidation sets.
- **3.3.2** The cascade is a *fixed point over a tree*, so it is inherently a multi-pass, non-vectorizable, branch-heavy computation on the critical path to the frame.
- **3.3.3** Layout after style is a second global pass: the CSS box model (block/inline/flow/flex/grid/float) requires constraint propagation across the tree. LayoutNG fragments the box tree per layout, and layout is largely single-threaded per document because the constraint graph is cyclic at the block level (percentage heights, intrinsic sizing, min-content/max-content).
- **3.3.4** Because layout output is *derived from* style, and style is *derived from* the tree, a single leaf mutation can, in the worst case, force root-to-leaf recomputation in both passes.

Mechanical defect: layout is solved as a **global fixed point over a mutable tree**, with an unbounded dependency radius. In the present invention, layout is instead solved as a **local fixed point over a position-based constraint system with bounded convergence**

radius, executed entirely on the GPU at a fixed per-frame cost independent of the number of ancestors.

3.4 Frame Production Crosses Multiple Process and IPC Boundaries

A representative browser frame:



Each → is a boundary at which data is serialized or copied. Under main-thread contention the frame is *dropped*, not merely delayed. The main thread is also shared with the application’s JavaScript, so any long task directly manifests as a missed vsync.

Mechanical defect: the presentation path is *indirect* and *shared with an unrelated workload* (script execution and GC). There is no isolation between “compute the interface” and “produce photons.”

3.5 JavaScript Garbage Collection Introduces Unbounded, Unpredictable Pauses

Prior-art runtimes use generational, tracing, compacting collectors. In the DOM context specifically:

- **3.5.1** The DOM and the JS heap form a **cross-component cycle collection** problem. Blink objects participate in a tracing collector (Oilpan) that must be coordinated with V8’s heap, requiring cross-heap write barriers and remembered sets for DOM↔JS edges.
- **3.5.2** Major GC involves stop-the-world pauses for atomic marking and compaction. Pause duration is a function of *live set size*, which the application does not control and which the DOM inflates.
- **3.5.3** Because interface state lives in *heap objects*, the interface’s memory is *relocated* by the collector. Addresses are therefore not stable, which forbids zero-copy projection of interface state into GPU-visible memory, forbids SharedArrayBuffer aliasing of that state, and forces a per-frame *gather* from scattered heap objects into a linear staging buffer.
- **3.5.4** Finalization is asynchronous and unordered; there is no frame-aligned deallocation.

Mechanical defect: the interface’s memory has **no stable address and no bounded lifetime**. Both properties are prerequisites for DMA and for zero-copy projection. The prior art fails on both.

3.6 CSS Transitions Are Timeline-Sampled, Not Physically Integrated

CSS transitions and animations are driven by the engine’s `AnimationTimeline`. The timing function (a cubic Bézier) is evaluated against a *frame-quantized* timeline value. Consequences:

- **3.6.1** With a 150 ms transition at 60 Hz, the easing curve is sampled at only ~9 discrete points. Intra-frame motion is not integrated; it is *interpolated from a monotonic clock*.
- **3.6.2** Only transform and opacity are reliably compositor-driven. Any other animated property requires main-thread style+layout+paint per frame; if the main thread is busy, the animation *stalls in a staircase* rather than degrading gracefully.
- **3.6.3** Completion is reported asynchronously as an event task (`transitionend`), introducing a further 0–16.67 ms of scheduling latency before application code observes it.
- **3.6.4** The behavioral curve is *specified as a shape*, not as a *response of a physical system*. There is no notion of stiffness, damping, or energy, so the motion cannot be composed with a physical model and cannot be made consistent under variable frame rate.

Mechanical defect: transitions are *function evaluations against a clock*, not *state of a dynamical system*. They therefore cannot be made frame-rate invariant, cannot be made stable under load, and cannot be coupled to a physics solver.

3.7 Multi-Application Isolation Forces Redundant Everything

Prior art achieves “multiple applications in one window” by one of:

- **3.7.1** Multiple browsing contexts (iframes): each gets its own JS realm, its own event loop participation, its own style/layout trees, its own script scheduling. Cross-context communication is `postMessage`, which *structured-clones* payloads and defers to the target’s task queue — a minimum of one task-queue round trip, typically 0–33 ms, with no shared address space.
- **3.7.2** Multiple tabs: each is frequently a separate *renderer process*, with process creation cost, separate V8 isolates, separate GPU command submission, and *no* shared mutable state whatsoever.
- **3.7.3** Multiple windows: OS-window-managed, with compositor-level composition between them.

Mechanical defect: co-residency of applications is achieved by *duplication of runtimes* rather than *partitioning of a single address space*. State transfer between co-resident applications is therefore serialization-bounded rather than pointer-bounded.

3.8 Summary of the Prior-Art Failure

The prior art’s interface substrate is, at every layer, a **document**:

Layer	Prior art	Structural consequence
Representation	Tree of GC’d objects	Unstable addresses; unbounded live set
Definition	Parsed HTML string	Decode cost; no fixed stride
Styling	Global cascade fixed point	Unbounded dependency radius
Layout	Global box-model fixed point	Single-threaded; O(ancestors) per mutation
Physics	Absent	Motion is a clock function, not a state
Co-residency	Duplicated runtimes	Serialization-bounded state transfer
Presentation	Multi-process IPC	Dropped frames; shared critical path
Memory	Heap, relocated	Cannot DMA; cannot zero-copy

None of these defects is incidental. Each follows from the founding decision to represent an interface as a *document*. The present invention reverses that decision.

§4 TECHNICAL SUMMARY AND MECHANICAL PRINCIPLES OF OPERATION

4.1 Foundational Principle

The interface is a fixed-stride array of instance records in GPU-visible memory. The array is simultaneously the state, the physics, and the vertex input. There is no tree. There is no cascade. There is no parse.

Formally, let the interface be

$I = \langle N, \{r_0 \dots r_{N-1}\} \rangle$

where each record r_i occupies exactly $STRIDE = 64$ bytes at byte offset $i \cdot 64$ in a single `GPUBuffer`. Every consumer of interface state — the constraint solver, the vertex shader, the spine bus, the Merkle committer, the native Metal embodiment — reads and writes

1. A system for direct-memory generative interface projection, comprising:

a graphics processing unit (GPU) comprising a GPU memory;

an instance storage buffer comprising a contiguous allocation within the GPU memory and storing a plurality of instance records, each instance record of the plurality of instance records having a fixed stride of exactly 64 bytes, wherein each instance record contains interface state data, physical simulation state data, and rasterizer input data for a corresponding interface element, and wherein the interface state data, the physical simulation state data, and the rasterizer input data occupy non-overlapping byte fields of a single 64-byte addressable unit such that the instance record is simultaneously readable as interface state, as physical simulation state, and as rasterizer input;

an extended position-based dynamics (XPBD) constraint solver operatively coupled to the instance storage buffer and configured to iteratively project the physical simulation state data of the plurality of instance records onto a constraint manifold until an equilibrium condition is satisfied, wherein a spatial layout of the corresponding interface element is determined by the equilibrium condition and not by a CSS box model;

a spine bus comprising a lock-free multi-producer multi-consumer (MPMC) ring buffer instantiated over a SharedArrayBuffer, the spine bus being configured to transport cross-room messages;

twenty application capability rooms co-resident within a single document and a single GPU context, each application capability room of the twenty application capability rooms being bound to the spine bus and assigned a unique room identifier within the instance storage buffer;

a projection engine configured to rasterize a frame directly from the instance storage buffer, the projection engine being configured to operate without a document object model (DOM) tree traversal, without a CSS cascade recalculation, without an HTML string parse, and without a JavaScript garbage collection cycle on a frame projection path; and

an output stage configured to apply an Optical Daylight tone mapping operator to the frame and to generate one or more living marks governed by a critically damped second-order response,

whereby the system projects a user interface in which the interface state data, the physical simulation state data, and the rasterizer input

PART: AUTONOMOUS TEEMING ENDPOINTS

PROVISIONAL PATENT APPLICATION COVER SHEET

Docket Number: MM-TEEMING-PROV-008 **Title of the Invention:** AUTONOMOUS LIVING WIRE ENDPOINTS OPERATING AS SELF-RENDERING REACTIVE WIDGETS AND DECENTRALIZED MICRO-APPLICATIONS **First Named Inventor:** Brennan DeCrow **Residence:** Oregon, United States **Assignee:** ManyMoats (ATESO / Magma) **Filing Basis:** 35 U.S.C. § 111(b) **Filing Date:** 2026-09-25

INVENTOR STATEMENT

The invention disclosed herein represents original, unencumbered technical innovations conceived and reduced to practice by Brennan DeCrow.

PROVISIONAL PATENT APPLICATION

Application Series: MM-TEEMING-PROV **Docket:** MM-TEEMING-PROV-008 **Title:** AUTONOMOUS LIVING WIRE ENDPOINTS OPERATING AS SELF-RENDERING REACTIVE WIDGETS AND DECENTRALIZED MICRO-APPLICATIONS **Inventor:** Brennan DeCrow **Applicant / Assignee:** ManyMoats / ATESO / Magma **Filing Basis:** 35 U.S.C. § 111(b) (Provisional) **Enablement / Written Description:** 35 U.S.C. § 112(a)–(b) **Claim Style:** 35 U.S.C. § 112(b); 35 U.S.C. § 101 (statutory subject matter: a specific machine-implemented improvement to network endpoint operation, binary wire transport, and cryptographic receipt generation) **Founder Inception Statement (recorded):** “*My Teeming endpoints are the future: endpoints as widgets, endpoints as apps, endpoints as living capabilities.*”

TABLE OF CONTENTS

1. Abstract
 2. Field of the Invention
 3. Background and Fatal Inefficiencies of Prior Art
 4. Technical Summary and Mechanical Principles of Operation
 - 4.1 System Topology and Port Allocation
 - 4.2 The Capability Container (CAP) Wire Format
 - 4.3 Deterministic Section Merkle Tree
 - 4.4 Ed25519 Container Attestation
 - 4.5 The Declarative Interface AST (UI_AST)
 - 4.6 The On-Wire State Machine Table (SMT)
 - 4.7 The Deterministic Execution Sandbox (WASM_CORE)
 - 4.8 Direct Mount Without SDK or Build Step
 - 4.9 Zero-Copy Transfer and the Host Descriptor Ring
 - 4.10 The Cross-Room Spine Bus
 - 4.11 Sovereign Endpoint Entanglement (4-Phase Handshake)
 - 4.12 Capability Poset Lattice Meet
 - 4.13 The AiST Receipt: Structure, Signing, and Merkle Inclusion
 - 4.14 Receipt Chaining and Epoch Merkle Mountain Range
 - 4.15 Interaction Latency and Verification Cost Budget
 - 4.16 Prior-Art Distinction Table
 5. Detailed Description of Preferred Embodiments
 - 5.1 Embodiment A: Browser-Native Lattice Host
 - 5.2 Embodiment B: Native Ring-Buffer Host (Bare-Metal / OS)
 - 5.3 Embodiment C: Composite Multi-Agent Entangled Workflow
 - 5.4 Embodiment D: Receipt Verifier and Third-Party Audit Path
 - 5.5 Embodiment E: Capability Attenuation and Ambient Authority Refusal
 6. Claim Set (Claims 1–10)
 7. Drawings Reference (FIG. 1–FIG. 12)
 8. Enablement Appendix A: Constants and Reference Tables
 9. Enablement Appendix B: Reference Pseudocode
 10. Enablement Appendix C: Best Mode Statement
 11. Sequence Listing / Data Structure Index
-

1. ABSTRACT

Disclosed is a machine-implemented system and method in which a network endpoint ceases to operate as a passive request/response data pipe and instead operates as an autonomous, self-rendering unit of executable capability. A server node (“Teeming Engine”) bound to a network port transmits, in response to a single request, a self-contained binary **Capability Container** (“CAP”) comprising a fixed 64-byte header, a table of typed sections, a Merkle root computed over those sections, and an Ed25519 signature over the container header. The CAP carries, over the wire, (a) a declarative interface abstract syntax tree, (b) an on-wire finite state machine bytecode, (c) a deterministic sandboxed execution module, (d) a capability grant set encoded in a poset, and (e) a cryptographic receipt policy. A generic host runtime — the **Lattice Host** —

verifies the signature and Merkle root, transfers the container payload zero-copy into a mount worker, instantiates the sandboxed module with no ambient network or filesystem authority, and compiles the declarative tree directly into a retained scene graph, thereby mounting an interactive, tactile widget without any vendor SDK, package manager dependency, bundler step, or client-side build.

Multiple distributed endpoints advertise routes over a co-located **cross-room spine bus** and autonomously negotiate composite rendering by computing a **lattice meet** over their respective capability posets; when the meet is non-empty, the endpoints jointly emit a dual-signed composite receipt. Every interaction with a mounted endpoint emits a tamper-evident **AiST (Attested Interaction State Token) receipt** comprising a monotonic sequence number, a predecessor hash link, canonical input and output digests, an Ed25519 signature, and an RFC 6962-style Merkle inclusion proof against a per-epoch root anchored in an append-only Merkle Mountain Range. The result is an endpoint that is simultaneously an application, a widget, a state machine, and a cryptographic principal.

2. FIELD OF THE INVENTION

The present invention relates generally to networked computing systems, and more particularly to:

- binary wire protocols for transmitting executable, self-describing application payloads across a network transport;
- client-side and host-side runtime systems that instantiate and mount received executable payloads as interactive graphical interface components;
- sandboxed deterministic bytecode execution environments and capability-based authority models;
- cryptographic attestation of build provenance and runtime interaction, including Merkle trees, Merkle Mountain Ranges, hash chains, and Ed25519 digital signatures;
- publish/subscribe and gossip routing topologies enabling autonomous peer-to-peer negotiation between independently operated network endpoints;
- direct memory access descriptor ring buffers, cache-line-aligned producer/consumer queues, and zero-copy buffer transfer between execution contexts.

The invention finds particular applicability where a network endpoint must ship its own interface, its own behavior, its own state semantics, and its own verifiable interaction record without requiring a separately distributed, separately versioned, separately built client application.

3. BACKGROUND AND FATAL INEFFICIENCIES OF PRIOR ART

3.1 REST and GraphQL: Endpoints as Inert Data Pipes

Representational State Transfer (REST) and GraphQL define endpoints exclusively in terms of *data shape*. An HTTP response of `Content-Type: application/json` is a serialized value with no executable content, no interface description carried at runtime, and no state machine semantics. The endpoint is a passive accessor for a database row or a computed projection.

The consequences are structural rather than incidental:

- A REST resource carries no information about *how it should be displayed, interacted with, or actuated*. Every consumer must independently author the presentation and interaction logic.
- GraphQL's schema (SDL) describes available field types and resolvers but intentionally excludes execution semantics; it is a type grammar, not a behavior descriptor.
- Version skew is unmanaged. When the endpoint changes its response shape, every client must be independently rebuilt and redeployed. There is no atomic coupling between endpoint behavior and client behavior.
- There is no cryptographic binding between an endpoint's *advertised* schema and the *behavior* a client exhibits on its behalf. A man-in-the-middle or a compromised client can render anything.

Fatal inefficiency: REST/GraphQL endpoints have an integration cost that scales as the product of endpoints and hosts. With $|E|$ endpoints and $|H|$ host applications, the number of independently authored, independently tested, independently versioned adapters is:

$$|\text{Adapters}|_{\text{REST}} = |E| \times |H|$$

This is a quadratic integration tax imposed by the absence of executable content on the wire.

3.2 OpenAPI, Swagger, and Code-Generated SDKs

OpenAPI-family specifications describe request and response *shape*. Tooling generates client stubs at build time. The generated stubs are:

- **Stale by construction.** The specification is a snapshot; the generated code ages relative to the live endpoint.
- **Non-behavioral.** A generated stub cannot display itself. It provides typed method calls, not an interface.
- **Build-coupled.** Regeneration requires a build pipeline, a package registry, and a deployment.

- **Unverifiable at runtime.** Nothing in the generated stub cryptographically binds to the endpoint's current behavior.

Fatal inefficiency: OpenAPI substitutes a design-time document for a runtime capability. The endpoint still cannot render itself.

3.3 Micro-Frontends, Module Federation, and Remote ESM

Micro-frontend architectures (Webpack Module Federation, remote ESM imports, import maps) achieve remote *code* loading, but at catastrophic cost:

- **Shared dependency singletons.** Host and remote must negotiate a shared runtime library version; mismatches produce runtime failures.
- **Full client toolchain required.** A bundler, a package manager, and a build step are mandatory on the producing side.
- **No verification.** A remote module is executed on receipt. There is no signature, no Merkle root, no declared authority set, no sandbox boundary beyond the origin policy.
- **No state semantics.** State is managed ad hoc, per team, with no wire-level contract.
- **Supply chain exposure.** A single compromised transitive dependency executes with full page authority.

Fatal inefficiency: Module Federation moves the build problem to the network edge without solving authority, verification, or interface semantics.

3.4 Web Components, Custom Elements, and Shadow DOM

Web Components permit an author to encapsulate markup, style, and behavior behind a custom element tag. However:

- The custom element *definition* must be delivered by the host page as JavaScript. The endpoint does not ship the component; the host page ships it.
- There is no wire format for a component. The tag name is a string convention.
- There is no authority model. A custom element executes with full origin authority.
- There is no cryptographic provenance or interaction receipt.

Fatal inefficiency: Web Components standardize the *container*, not the *transport*. The endpoint remains a data pipe.

3.5 Server-Driven UI (HTMX, Hotwire, LiveView)

Server-driven UI systems replace JSON with HTML fragments and use declarative attributes (e.g., `hx-get`) to trigger partial updates. Their failures are:

- **Stringly-typed HTML over the wire.** Fragments are text, not a typed scene description; parsing is lossy and DOM-dependent.
- **A pre-installed client library is mandatory.** The host page must itself load the library; the endpoint cannot supply it.

- **No state machine.** Transitions are implicit in DOM swapping; state lives in server-side session storage, not on the wire.
- **No composition.** Two endpoints cannot negotiate a joint interface.
- **No receipts.** Nothing cryptographically attests to what the server sent or what the user did.

Fatal inefficiency: Server-driven UI retains the client library dependency and substitutes markup for a capability.

3.6 Remote DOM and Third-Party Renderer Sandboxes

Systems such as remote-ui / Remote DOM allow an isolated script to build a UI tree by calling a host-provided API, rendering through a host's native components. Deficiencies:

- **Mandatory SDK.** The remote script must import the host's @remote-ui/core or equivalent. This is a client-side dependency and a build step.
- **Host-specific.** The API surface is proprietary. There is no portable binary standard.
- **Per-host re-implementation.** Each new host requires a new renderer bridge.
- **No verification, no receipts, no cross-endpoint composition.**

Fatal inefficiency: Remote DOM relocates the SDK problem; it does not eliminate it.

3.7 WebAssembly Component Model and WASI Preview 2

The WASM Component Model with WIT interface descriptions ships *interface-typed binary components*. This is the closest prior art and is expressly distinguished:

- WIT describes function signatures. It **does not describe a user interface**, a visual tree, a gesture binding, or a haptic profile.
- It carries **no state machine**; state is the caller's problem.
- It has **no cryptographic container attestation** and **no per-interaction receipt**.
- It has **no direct-mount semantics**; a host must be purpose-built for each component's WIT world.
- It has **no cross-endpoint negotiation or composite rendering**.

Fatal inefficiency: The Component Model standardizes the *call boundary*. It does not make an endpoint a self-rendering, self-verifying, self-chaining application.

3.8 Sigstore, in-toto, and SLSA Build Provenance

These systems attest **build artifacts**: they prove *which* pipeline produced *which* bytes. They do not:

- attest **runtime interaction**;
- bind **user intent** to **state transition**;
- produce a **per-interaction** record;
- chain records into an **append-only, third-party auditable log** keyed by interaction rather than by build.

Fatal inefficiency: Build provenance answers “how was this made?” and is silent on “what happened when it ran, and who did it?”

3.9 Capability Tokens (Macaroons, DPoP, WebAuthn, Biscuit)

Macaroon-style tokens provide attenuable authority and DPoP/WebAuthn bind tokens to key holders. They do not:

- deliver an **interface**;
- execute **behavior**;
- produce **Merkle-proven interaction receipts**;
- enable **peer-to-peer endpoint negotiation**.

Fatal inefficiency: Authority without behavior is a permission slip with nothing to permit.

3.10 CRDT Backends and Managed Sync (Yjs providers, Liveblocks)

These provide convergent replicated state. They:

- ship **state, not presentation**;
- require an **SDK and a hosted service**;
- have **no signature over the merge graph**;
- have **no composite widget negotiation**.

3.11 Consolidated Statement of the Unsolved Problem

Every prior system leaves at least one of the following unsolved:

Requirement	REST	OpenAPI	Module Fed.	Web Comp.	HTMX	Remote DOM	WASM CM
Executable payload on wire	✗	✗	✓	✗	✗	✓	✓
Interface shipped by endpoint	✗	✗	✗	✗	~	~	✗
No client SDK / build step	~	✗	✗	✓	✗	✗	~
On-wire state machine	✗	✗	✗	✗	✗	✗	✗
Container cryptographic attestation	✗	✗	✗	✗	✗	✗	~

Requirement	REST	OpenAPI	Module Fed.	Web Comp.	HTMX	Remote DOM	WASM CM
Per-interaction Merkle receipt	✗	✗	✗	✗	✗	✗	✗
Cross-endpoint autonomous composition	✗	✗	✗	✗	✗	✗	✗
Capability poset attenuation	✗	✗	✗	✗	✗	✗	~

No single prior system satisfies all eight. The present invention satisfies all eight simultaneously within one binary container format and one generic host runtime.

4. TECHNICAL SUMMARY AND MECHANICAL PRINCIPLES OF OPERATION

4.1 System Topology and Port Allocation

Element	Identifier	Description
Teeming Engine	:8791/tcp + :8791/udp	Endpoint server process. UDP carries QUIC/HTTP-3 (ALPN h3) for CAP retrieval and teeming-spine/1 for the spine bus. TCP carries HTTP/2 (ALPN h2) as a fallback transport with identical framing.
Lattice Host	host process	Generic runtime. Verifies, mounts, sandboxes, and receipts. Pre-installed once; never re-authored per endpoint.
Mount Worker	thread / worker	Retained scene graph compiler and event dispatch loop.
Spine Bus	QUIC stream, ALPN teeming-spine/1	Cross-room publish/subscribe and negotiation transport.

Element	Identifier	Description
Receipt Log	append-only store	Per-endpoint epoch roots and MMR peaks.

Transport negotiation. A client issues:

```
GET /cap/<content-id> HTTP/3
Host: <endpoint-authority>:8791
Accept: application/x-teeming-cap;v=1
Teeming-Session: <base64url(32-byte CSPRNG nonce)>
Teeming-Host-Caps: <base64url(16-byte capability bitfield)>
Teeming-Room: <base64url(32-byte room id)>
Teeming-Spine: <base64url(32-byte spine node id)>
Teeming-Receipt: required | optional
```

The engine responds:

```
HTTP/3 200 OK
Content-Type: application/x-teeming-cap;v=1
Content-Length: <total_len>
Teeming-Cap-Id: b3-256:<64 lowercase hex>
Teeming-Sig-Alg: ed25519
Teeming-Epoch: <u64 decimal>
Teeming-Key-Id: <u32 decimal>
Teeming-Spine-Epoch: <u64 decimal>
```

<CAP container bytes>

Teeming-Cap-Id is the BLAKE3-256 digest of the response body. A host need not trust the header; it recomputes the digest and compares.

4.2 The Capability Container (CAP) Wire Format

All multi-byte integers are **little-endian**. All offsets are absolute from the first byte of the container.

4.2.1 Fixed Header — 64 bytes (offsets 0x00–0x3F)

Offset	Size	Type	Field	Notes
0x00	4	u8[4]	magic	0x54 0x43 0x41 0x50 = ASCII "TCAP"
0x04	1	u8	ver_major	0x01 for this disclosure
0x05	1	u8	ver_minor	0x00 for this disclosure

Offset	Size	Type	Field	Notes
0x06	2	u16	flags	see §4.2.2
0x08	4	u32	total_len	total container length, including header, sections, and trailer
0x0C	2	u16	section_cnt	number of 16-byte section table entries; $1 \leq \text{section_cnt} \leq 4096$
0x0E	2	u16	hdr_crc16	CRC-16/CCITT-FALSE (poly 0x1021, init 0xFFFF) computed over bytes [0x00, 0x0E); guards against header corruption before any hashing
0x10	32	u8[32]	merkle_root	BLAKE3-256 root over section leaves, §4.3
0x30	8	u64	epoch	monotonic epoch index of the issuing engine
0x38	4	u32	key_id	identifies the Ed25519 signing key within the engine's key set
0x3C	4	u32	reserved	MUST be 0x00000000; receivers SHALL reject non-zero

4.2.2 flags Bit Assignments

Bit	Name	Meaning
0	SIGNED	trailer present; Ed25519 signature mandatory
1	ENTANGLED	SPINE_ROUTES section required; engine is willing to entangle
2	STATEFUL	STATE_MACHINE section required
3	TACTILE	TACTILE_PROFILE section required
4	EPHEMERAL	host MUST NOT persist the container to durable storage
5	RECEIPT_REQUIRED	host MUST emit an AiST receipt for every interaction or refuse to mount
6	DETERMINISTIC_ONLY	WASM_CORE must be instantiated in the deterministic subset with fuel metering (§4.7)
7	PORTAL_ALLOWED	UI_AST may contain PORTAL nodes referencing other containers
8–15	reserved	MUST be zero; receivers SHALL reject non-zero

4.2.3 Section Table — section_cnt × 16 bytes, beginning at 0x40

Offset within entry	Size	Type	Field
0x00	1	u8	section_type
0x01	1	u8	section_flags (bit 0: COMPRESSED_BROTLI; bit 1: MANDATORY_UNDERSTAND; bit 2: CONTENT_ADDRESSED)
0x02	2	u16	reserved, zero
0x04	4	u32	offset
0x08	4	u32	length

Offset within entry	Size	Type	Field
0x0C	4	u32	digest_prefix = first 4 bytes (little-endian u32) of BLAKE3-256 over the section bytes

Sections are contiguous and SHALL be laid out in ascending section_type order, then ascending offset, to make the table canonical (a canonicalization requirement for the Merkle root). Section regions SHALL NOT overlap; a receiver SHALL reject overlapping regions.

4.2.4 Section Types

Code	Name	Encoding	Required When
0x01	MANIFEST	CBOR (deterministic, RFC 8949 §4.2 core deterministic encoding)	always

PATENT CLAIMS

Application No.: MM-TEEMING-PROV-008 **Title:** AUTONOMOUS LIVING WIRE ENDPOINTS OPERATING AS SELF-RENDERING REACTIVE WIDGETS AND DECENTRALIZED MICRO-APPLICATIONS

1. (Independent — System / Apparatus)

A system for autonomous living wire endpoints operating as self-rendering reactive widgets and decentralized micro-applications, the system comprising:

a first endpoint computing device and a second endpoint computing device, each of the first endpoint computing device and the second endpoint computing device comprising a processor, a non-transitory memory, a network interface controller, and a spine bus interface circuit;

wherein the network interface controller of each of the first endpoint computing device and the second endpoint computing device is configured to bind a transport socket to transmission control protocol port 8791 and to receive over said port 8791 a binary capability container, the binary capability container comprising a header, an interface abstract syntax tree segment, a finite state machine bytecode segment, and a deterministic sandboxed WebAssembly execution module;

a generic lattice host runtime instantiated in the non-transitory memory of each of the first endpoint computing device and the second endpoint computing device, the generic lattice host runtime being configured to validate the binary capability container against the header thereof and to mount the interface abstract syntax tree segment and the finite state machine bytecode segment as an interactive tactile widget within a host presentation surface of the respective endpoint computing device in the absence of a client software development kit, in the absence of a package dependency manifest, and in the absence of a build step;

a cross-room spine bus interconnecting the first endpoint computing device and the second endpoint computing device through the respective spine bus interface circuits thereof, wherein each of the first endpoint computing device and the second endpoint computing device further comprises a capability poset store and a meet engine, the meet engine being configured to negotiate over the cross-room spine bus by computing a lattice meet of a first capability poset stored in the capability poset store of the first endpoint computing device with a second capability poset stored in the capability poset store of the second endpoint computing device to produce a composite workflow, and wherein at least one of the first endpoint computing device and the second endpoint computing device is configured to dispatch at least a portion of the composite workflow to the deterministic sandboxed WebAssembly execution module of the binary capability container; and

a receipt generator configured to emit, for each interaction with at least one of the interactive tactile widget and the composite workflow, an AiST receipt comprising an Ed25519 signature over a canonical serialization of the interaction and a Merkle Mountain Range inclusion proof linking the AiST receipt to a current Merkle Mountain Range root.

2. (Dependent — Physical Registers of the Network Interface Controller and Secure Element)

The system of claim 1, wherein the network interface controller of each of the first endpoint computing device and the second endpoint computing device comprises:

a transport control block register file including a port binding register preloaded with a value corresponding to port 8791, a status register, and an interrupt moderation register;

a receive descriptor ring comprising a plurality of descriptor entries, each descriptor entry of the plurality of descriptor entries storing a physical base address and a byte length of a respective direct memory access buffer within the non-transitory memory;

a transmit descriptor ring comprising a plurality of descriptor entries, each descriptor entry of the plurality of descriptor entries storing a physical base address and a byte length of a respective frame to be emitted onto the cross-room spine bus; and

a key handle register of a secure element, the key handle register storing a non-exportable handle referencing an Ed25519 private key resident within the secure element;

wherein the receive descriptor ring is configured to deposit the binary capability container into a pinned direct memory access buffer of the non-transitory memory by direct memory access without an intermediate copy through a user-space buffer; and

wherein the receipt generator is configured to submit the canonical serialization of the interaction to the secure element and to receive the Ed25519 signature therefrom without exposing the Ed25519 private key to the processor.

3. (Dependent — Container Memory Layout and In-Place Parsing)

The system of claim 1, wherein the binary capability container is mapped into an immutable read-only memory region of the non-transitory memory and has a memory layout comprising:

a fixed-width header commencing at a zero offset of the immutable read-only memory region, the fixed-width header including a magic value field, a format version field, an interface abstract syntax tree segment length field, a finite state machine bytecode segment length field, a WebAssembly module length field, and an integrated checksum field;

a contiguous interface abstract syntax tree segment commencing at a header-aligned offset following the fixed-width header;

a contiguous finite state machine bytecode segment commencing immediately after a terminal byte of the contiguous interface abstract syntax tree segment;

a contiguous WebAssembly module segment commencing immediately after a terminal byte of the contiguous finite state machine bytecode segment; and

a trailer comprising a relocation table and a content-addressed digest computed over the contiguous interface abstract syntax tree segment, the contiguous finite state machine bytecode segment, and the contiguous WebAssembly module segment;

wherein the generic lattice host runtime is configured to parse the contiguous interface abstract syntax tree segment in place within the immutable read-only memory region by memory reference without deserializing the contiguous interface abstract syntax tree segment into a heap-allocated object graph;

wherein the finite state machine bytecode segment defines a plurality of states and a plurality of transitions of the interactive tactile widget; and

wherein the deterministic sandboxed WebAssembly execution module is allocated a bounded linear memory region having a page count fixed by the WebAssembly module length field and is denied access to any memory address outside the bounded linear memory region, to a wall-clock facility, and to a nondeterministic entropy facility of the respective endpoint computing device.

4. (Dependent — Spine Bus Framing and Negotiation Protocol)

The system of claim 1, wherein the cross-room spine bus is configured to operate according to a framing protocol in which each frame comprises a length prefix, a frame type identifier, a monotonic sequence number, a payload, and an Ed25519 frame signature computed over the length prefix, the frame type identifier, the monotonic sequence number, and the payload, the frame type identifier being selected from the group consisting of a capability-advertisement frame type, a meet-proposal frame type, a meet-acceptance frame type, a workflow-commitment frame type, and a receipt-announcement frame type;

wherein the negotiation performed by the meet engine comprises a three-phase handshake comprising:

a capability advertisement phase in which each of the first endpoint computing device and the second endpoint computing device publishes its respective capability poset as a topologically ordered sequence of capability descriptors over the cross-room spine bus;

a meet computation phase in which each of the first endpoint computing device and the second endpoint computing device independently computes the lattice meet from the topologically ordered sequence of capability descriptors received from the other endpoint computing device; and

a composite workflow commitment phase in which each of the first endpoint computing device and the second endpoint computing device emits a workflow-commitment frame bearing an Ed25519 frame signature over an identical serialization of the composite workflow; and

wherein the generic lattice host runtime is configured to reject any frame whose Ed25519 frame signature fails to verify, whose monotonic sequence number is not strictly greater than a most recently accepted monotonic sequence number, or whose frame type identifier is not among the group.

5. (Dependent — Hardware Bus Constraints and Arbitration Circuitry)

The system of claim 1, wherein the cross-room spine bus comprises a differential serial link, a token-based arbitration circuit, and

PART: MOATID AMBIENT BIOMETRIC AUTH

PROVISIONAL PATENT APPLICATION COVER SHEET

Docket Number: MM-ID-PROV-009 **Title of the Invention:** Hierarchical Ambient-Exhaustion Biometric Authentication and Zero-Friction Sovereign Identity Protocol (MoatID) **First Named Inventor:** Brennan DeCrow **Residence:** Oregon, United States **Assignee:** ManyMoats (ATESO / Magma) **Filing Basis:** 35 U.S.C. § 111(b) **Filing Date:** 2026-09-25

INVENTOR STATEMENT

The invention disclosed herein represents original, unencumbered technical innovations conceived and reduced to practice by Brennan DeCrow.

The user wants a formal provisional patent application specification for MM-ID-PROV-009. This is a legitimate patent drafting task. I should write it with rigorous technical detail: registers, data structures, algorithms, equations, state machines, protocols.

Let me structure this properly as a USPTO provisional patent application under 35 U.S.C. § 111(b), with 35 U.S.C. § 112 enablement.

Structure: - Title - Cross-reference to related applications - Field of the invention - Background (prior art defects) - Summary - Brief description of drawings - Detailed description - Definitions and notation - System architecture - Ambient Attestation Layer (AAL) - Wi-Fi mesh BSSID triangulation - BLE RSSI differential - Smartwatch accelerometer + PPG - Secure Enclave device binding - Browser integrity attestation - Bayesian/log-odds fusion engine - Friction lattice (strict partial order) - Friction Escalation Lattice Theorem - Hard-cap biometric touch invariant - Epoch state machine - Data structures (C-like structs, registers) - Wire protocols - Algorithms (pseudocode) - Security analysis - Claims (10) - Abstract

I need to be careful about legal rigor: provisional applications typically don't require claims but it's fine to include them. The user asked for 10 formal patent claims. Actually, provisional applications under § 111(b) don't require claims, but including them is acceptable and provides priority support. I'll include them.

Let me write with real mathematical rigor.

Key mathematical machinery:

1. Ambient Attestation Layer: each channel i produces an observation vector, converted to a log-likelihood ratio (LLR) against the hypothesis H_S (sovereign user present at device) vs H_A (adversary/absent).

$$LLR_i = \log(P(o_i | H_S) / P(o_i | H_A))$$

Fusion: $L_{total} = \sum_i w_i * LLR_i$, with weights from channel reliability. Need to handle non-independence — use a copula or correlation-adjusted fusion. I'll present a Gaussian copula correction or use the Neyman-Pearson with calibrated weights via logistic regression on a validation set — better: use a log-linear model with pairwise interaction terms (like a Boltzmann machine / Markov random field):

$$L(o) = \sum_i w_i * LLR_i(o_i) + \sum_{\{i < j\}} w_{ij} * \phi_{ij}(o_i, o_j)$$

That handles correlated channels like Wi-Fi BSSID and BLE RSSI (both spatial).

Decision: accept ambient if $L_{total} \geq \tau_{ambient}$, where τ is set by Neyman-Pearson at $FAR \leq 1e-9$ per epoch.

2. Wi-Fi mesh BSSID triangulation: multilateration. BSSID k at known position x_k , observed RSSI P_k . Log-distance path loss model:

$$P_k = P_0 - 10 n \log_{10}(d_k / d_0) + X_{\sigma} \quad d_k = d_0 * 10^{((P_0 - P_k)/(10n))}$$

Weighted least squares / nonlinear least squares (Gauss-Newton) to solve for user position $\hat{x}$. Residual $\rightarrow$ goodness of fit $\rightarrow$ LLR.

Actually more robust: use time-of-flight / RTT (802.11mc FTM) if available. Mention both.

Covariance: $C = (J^T W J)^{-1}$, position residual $r = \sum w_k (d_k - \|x - x_k\|)^2$.

3. BLE RSSI differential: ratio of RSSI between two or more anchors, which cancels transmit power:

$$\Delta_{ij} = RSSI_i - RSSI_j = 10 n \log_{10}(d_j / d_i) + \text{noise}$$

This is transmit-power-independent. Use for proximity-zone classification. Two-ray / log-normal.

Also BLE channel-based ranging (phase-based / Bluetooth 5.1 AoA).

4. Smartwatch accelerometer + PPG:

- Accelerometer: gait cadence. Extract step frequency f_{gait} via autocorrelation of $|a|$ magnitude; harmonic ratio; also arm-swing. Compare against enrolled template. Cross-correlation of the gait spectrogram with template. Also the watch-device pairing proves the wearer is the user.
- PPG: heart rate rhythm cadence — HRV (RR intervals from PPG). Features: mean HR, SDNN, RMSSD, pNN50, LF/HF ratio. Match against enrollment template.

Note that cardiac rhythm is a quasi-biometric — it's not a hard biometric but a soft continuous one. Must be legally careful: it is a “physiological cadence signature.”

- Liveness: PPG pulse presence + accelerometer motion coherence (motion-artifact coupling). A severed finger/watch replica would fail.

Gait cross-correlation: $\rho = \max_{\tau} \langle g(t), g_{\text{template}}(t+\tau) \rangle / (\|g\| \|g_{\text{template}}\|)$.
LLR from Fisher z-transform.

HRV: Mahalanobis distance $D^2 = (f - \mu)^T \Sigma^{-1} (f - \mu)$ on the HRV feature vector. $\text{LLR} = -D^2/2 + \text{const.}$

5. Secure Enclave device binding: hardware root of trust, P-256 key in Secure Enclave / StrongBox / TPM 2.0, attested key generation with attestation certificate chain to vendor root. Device Binding Token $\text{DBT} = \text{Sign}_{\{\text{SK}_{\text{SE}}\}}(\text{nonce} \parallel \text{device_id} \parallel \text{epoch})$. Verify via attestation cert. Also key must be non-exportable. This gives “something you have” with hardware binding.
6. Browser integrity: measured boot / runtime integrity report. Hash chain of browser binary, extension manifest hashes, CSP, no MITM proxy, TLS channel binding (tls-server-end-point), certificate pinning, WebAuthn clientDataJSON hash, code-signing of extension set, DOMPurify... Browser Integrity Manifest BIM with Merkle root; signed by enclave.

Now the friction lattice.

Friction Lattice: Let $\Phi = \{\phi_1, \dots, \phi_m\}$ be the set of available authentication modalities. Define a friction cost function $c: \Phi \rightarrow \mathbb{R}_{\geq 0}$. Embedding functions that are passive ambient all have $c < \epsilon_A$ (near-zero), biometric modalities have $c \geq B$ where $B \gg \epsilon_A$.

Define the strict partial order $<_f: \phi_i <_f \phi_j$ iff $c(\phi_i) < c(\phi_j)$ AND ϕ_i is non-interruptive. Actually “strict partial order” requires irreflexive and transitive. Define:

$\phi_i < \phi_j$ iff $(c_i < c_j)$ and $(\text{interrupt}_i = 0 \Rightarrow \text{interrupt}_j = 1 \text{ or } c_i < c_j)$... Let me define it cleanly.

Define a two-key ranking: (h, c) where $h \in \{0,1\}$ is the human-interaction bit ($h=0$ passive, $h=1$ active) and $c \in \mathbb{R}$ the expected friction cost in seconds of user attention. Define

$$\phi_i < \phi_j \iff (h_i < h_j) \vee (h_i = h_j \wedge c_i < c_j)$$

This is a strict weak order on (h, c) — actually it's a lexicographic order, which is a total order on the classes and a strict partial order overall. Good: lexicographic ordering of pairs is a total preorder when ties exist; strict partial order on distinct pairs. Fine.

Then the Admissibility Rule: ϕ_j may be invoked only if for all ϕ_i with $\phi_i < \phi_j$, ϕ_i has been evaluated and returned a negative/failed verdict with statistical confidence, i.e., $\text{LLR}_i < \tau_i$ AND the evaluation was performed at least twice (per founder rule: exhaust all ambient channels twice).

Formally, define the Eligibility Predicate $E(\phi_j, t)$:

$$E(\phi_j, t) = \bigwedge \{i \in \text{Pred}(j)\} [\text{evaluated}_i(t) \geq 2 \wedge L_i(t) < \tau_i]$$

where $\text{Pred}(j) = \{ i : \phi_i < \phi_j \}$.

Then the Biometric Invocation Theorem: For any biometric modality β , $E(\beta)$ can be true only if all ambient channels have been evaluated ≥ 2 times and failed.

Corollary (Hard Cap): At most one biometric modality invocation per sovereign epoch. Enforced by a monotone counter $\mu \in \{0, 1\}$ in the Trusted Execution Environment, never decrementable within an epoch. $\mu=1 \Rightarrow$ all biometric invocations blocked until epoch rollover.

Actually the founder rule says “hard-capped at exactly ONE biometric touch per sovereign epoch” — meaning at most one. “No more than ONE.” I’ll present it as $\mu(t) \leq 1$ with the gate: $\text{biometric_invocation_allowed}(t) \Leftrightarrow (\mu(t)=0 \wedge E(\beta, t))$. And if that single biometric fails, the epoch is terminated (session denied), requiring a new epoch — but a new epoch requires re-verification of ambient channels and the founder rule says “exhaust ambient passive channels twice before asking for a single biometric touch” — so the re-attempt also needs a fresh epoch. To prevent an attacker from forcing repeated biometric prompts (the very friction defect), the epoch counter is rate-limited: new epoch may only be created after a cooldown $\tau_{\text{cool}} \geq T_{\text{min}}$, and the number of biometric prompts per unit time is bounded by N_{max} .

That’s an important anti-fatigue guarantee: Biometric Prompt Rate Bound.

Let me formalize: Let $B(t)$ be the number of biometric prompts issued in window $[t-T_w, t]$. Then $B(t) \leq \text{ceil}(T_w / \tau_{\text{cool}})$. And per epoch, ≤ 1 .

Now the state machine. States:

S0 IDLE S1 AMBIENT_COLLECT (passive, round 1) S2 AMBIENT_FUSE_1 S3
DEVICE_BIND_VERIFY S4 BROWSER_INTEGRITY_VERIFY S5
AMBIENT_REVERIFY (round 2 — the “twice” rule) S6 FRICTION_LATTICE_EVAL
S7 BIOMETRIC_GATE_ARMED (only if all ambient failed with confidence) S8
BIOMETRIC_PROMPT ($\mu \rightarrow 1$) S9 SESSION_GRANT S10 SESSION_DENY /
EPOCH_TERMINATE S11 COOLDOWN

Transitions with guards.

Actually the normal (happy) path should be: ambient succeeds $\rightarrow$ SESSION_GRANT with ZERO biometric touches. That’s the zero-friction path. Only if ambient fails do we escalate.

Let me restructure:

S0 IDLE $\rightarrow$ S1 AMBIENT_SWEEP_1 $\rightarrow$ S2 FUSE (L_{total} computed) - if $L_{\text{total}} \geq \tau_{\text{ambient}}$ $\rightarrow$ S9 SOVEREIGN_SESSION_GRANT (0 biometric touches) - else $\rightarrow$ S3 HARD_BIND_CHECK (Secure Enclave attestation) $\rightarrow$ S4 BROWSER_INTEGRITY $\rightarrow$ S5 AMBIENT_SWEEP_2 (re-verification, required by founder rule) $\rightarrow$ S6 FUSE_2 -

if $L \geq \tau \rightarrow S9$ - else $\rightarrow S7$ LATTICE_EVAL $\rightarrow$ gate $\rightarrow S8$ BIOMETRIC_PROMPT
 (μ : $0 \rightarrow 1$, irreversible) - success $\rightarrow S9$ - fail $\rightarrow S10$ EPOCH_TERMINATE + S11
 COOLDOWN

Good.

Data structures: I'll give C structs with fixed-width types, packed layouts, and register maps.

Registers (MMIO / enclave-resident): - AAL_CTRL @ 0x00 - AAL_STATUS @ 0x04 - AAL_EPOCH_CTR @ 0x08 (monotone, 64-bit) - AAL_BIOMETRIC_CTR @ 0x10 (μ , monotone, write-once-per-epoch) - AAL_LL_R_ACC @ 0x14 (Q16.16 signed fixed point) - AAL_THRESH @ 0x18 - AAL_CHAN_MASK @ 0x1C - AAL_FAIL_VECTOR @ 0x20 - AAL_LATTICE_STATE @ 0x24 - AAL_COOLDOWN_TMR @ 0x28 - AAL_SEAL @ 0x30 (HMAC of state)

I'll define a fixed-point representation: LLR in Q8.24? Let's use int32 Q16.16 with saturation, or int64 for accumulation. I'll use int64_t Q32.32.

Now the cryptographic seals: state seal = HMAC-SHA256(K_{seal} , epoch || ctr || state || ...), K_{seal} derived from Secure Enclave key via HKDF. Prevents rollback of μ .

Actually the μ counter must be monotone; use a monotone counter register that's write-once per epoch, plus a sealed blob with a monotone counter (Apple SEP has such).

Let me now write the LLR derivations more carefully.

Wi-Fi multilateration: Model: $P_k = P_0 - 10 n \log_{10}(\|x - a_k\|/d_0) + X$, $X \sim N(0, \sigma_k^2)$ Weighted least squares. Jacobian $J_k = dP_k/dx = -10n/(\ln 10) * (x - a_k)/\|x - a_k\|^2$ Gauss-Newton: $x_{l+1} = x_l + (J^T W J)^{-1} J^T W r$ Residual sum of squares: $S = \sum w_k r_k^2$, dof = $K - 3$. Chi-square test: $S/\sigma^2 \sim \chi^2(K-3)$. LLR for presence:

$LLR_{\text{wifi}} = -S/(2\sigma^2) + \dots$ plus geometric consistency term (the position must be inside the geofence / within d_{max} of enrolled trusted location).

Add the "trusted place" test: $\|x_{\text{hat}} - x_{\text{trusted}}\| \leq r_{\text{trusted}}$, and consistency with device's own coarse location (GPS/IP).

Also BSSID continuity: the set of visible BSSIDs must match the enrolled home/work fingerprint with Jaccard index $J = |A \cap E| / |A \cup E| \geq J_{\text{min}}$.

BLE RSSI differential: $\Delta_{ij} = \text{RSSI}_i - \text{RSSI}_j$ (transmit-power independent) Model: $\Delta_{ij} = 10 n \log_{10}(d_j/d_i) + N(0, \sigma_{\Delta}^2)$ Multipath: Ricean. Use a Kalman filter over time with state $[d_i, d_j]$ or just classify zones. Better: use the vector of pairwise RSSI differences as a fingerprint and compute Mahalanobis distance to enrolled template: $D^2 = (\Delta - \mu_{\Delta})^T \Sigma_{\Delta}^{-1} (\Delta - \mu_{\Delta})$. Enrollment: the device's owner is co-located with the watch and phone.

Also BLE proximity liveness: the wearer's watch must be within $d_{\text{BLE}} \leq 1.5$ m and moving coherently.

Smartwatch gait + PPG: Gait: $s(t) = \|a(t)\| - g$, bandpass 0.5-4 Hz. Cadence $f_c = \arg\max S(f)$ where $S(f) = |\text{FFT}(s)|^2$. Also autocorrelation $R_{ss}(\tau)$, peak at τ_{step} , cadence = $60/\tau_{\text{step}}$. Template matching: dynamic time warping (DTW) distance $D_{\text{DTW}}(g, g_{\text{env}})$, or cross-correlation coefficient ρ . $\text{LLR}_{\text{gait}} = \log[f(\rho | H_S) / f(\rho | H_A)]$ — model ρ under H_S as Beta-ish concentrated near 0.85+, under H_A as near 0. Use Fisher z : $z = \text{artanh}(\rho)$, $z \sim N(\text{artanh}(\rho_0), 1/(N-3))$.

PPG HRV: extract IBI (interbeat interval) series from PPG via peak detection on the AC component. Features: $f = [\text{HR}_{\text{mean}}, \text{SDNN}, \text{RMSSD}, \text{pNN50}, \text{LF/HF}]$. Mahalanobis D^2 . Also cross-check with ECG-quality from the watch if available. Liveness: PPG requires pulsatile blood flow; a photo of a wrist fails because of (a) no pulse, (b) no motion coherence. Define coherence $C = |\text{corr}(\text{HR}(t), A(t))|$ where $A(t)$ is the accelerometer motion magnitude in the 0.7-3 Hz band — during walking, HR rises with activity (exercise coupling). A replay attack cannot synthesize correlated accelerometer + PPG without the actual body.

Cardiac coupling: $\text{HR}_{\text{drift}} = \alpha * A_{\text{rms}} + \beta$, fit over window; R^2 must exceed threshold.

Device binding: SK_{SE} generated in SEP; never leaves. Public key PK_{SE} certified by Apple Attestation Root (or Google Key Attestation / TPM EK cert). Attestation includes nonce, device ID (IDFV/AAID or better, a per-epoch nonce), and the app's cdHash . Sign: $\text{sigma} = \text{ECDSA}_{\text{Sign}}(\text{SK}_{\text{SE}}, H(\text{epoch} \parallel \text{dpop}_{\text{jkt}} \parallel \text{nonce}_{\text{server}}))$ DPoP: Demonstrating Proof-of-Possession (RFC 9449) — bind tokens to the key.

Browser integrity: BIM (Browser Integrity Manifest): Merkle tree over {browser binary hash, extension manifest hashes, CSP hash, user-agent entropy, TLS channel binding value, absence of MITM, WebAuthn client extension outputs}. Merkle root R_{BIM} . Signature by enclave: $\text{sig}_{\text{BIM}} = \text{Sign}(\text{SK}_{\text{SE}}, R_{\text{BIM}} \parallel \text{nonce})$. Merkle proof for each leaf. Include “attestation of no automation”: $\text{navigator.webdriver} == \text{false}$, presence of hardware-backed credential, TLS cert chain pinning, and TLS channel binding (RFC 5929 $\text{tls-server-end-point}$) to prevent relay phishing.

Now the fusion with correlation: $L(o) = \sum_i \alpha_i \text{LLR}_i(o_i) + \sum_{\{i < j\}} \beta_{ij} \phi_{ij}(o_i, o_j)$

Where ϕ_{ij} is a coupling feature. The β weights trained by regularized logistic regression / max-entropy with L2, on labeled data. Ensure calibration via Platt scaling.

Decision rule (Neyman-Pearson): choose threshold τ_{amb} such that $P(L \geq \tau_{\text{amb}} | H_A) = \alpha_{\text{target}}$, with $\alpha_{\text{target}} = 10^{-9}$ per epoch.

Bayes risk: minimize $E[\text{cost}] = c_{\text{FN}} * P_{\text{FN}} + c_{\text{FP}} * P_{\text{FP}} + c_{\text{friction}} * E[\# \text{contacts}]$. The lattice is exactly the greedy solution to this constrained problem when friction cost is monotone increasing along the lattice.

Optimal escalation theorem: Given modalities in a strict partial order by friction, and a sequential hypothesis test with per-modality cost c_i , the optimal policy (minimizing expected total cost subject to $P_{\text{FN}} \leq \epsilon$) is to evaluate modalities in non-decreasing friction order and stop at the first whose LLR exceeds its threshold — the Wald

sequential probability ratio test (SPRT) generalized. The Wald-Wolfowitz theorem gives optimality for sequential tests. State it: the friction lattice ordering is the cost-optimal evaluation order under monotone cost.

Then, the biometric hard cap: augment the SPRT with the constraint that the number of $h=1$ modalities invoked per epoch ≤ 1 , and cross-epoch $\leq N_{\max}$.

Hard Cap Invariant: $\forall$ epoch e , $\forall t \in \text{epoch } e$: $\mu_e(t) \leq 1$, and $\sum_{e' \in [t-T_w, t]} \mu_{e'} \leq N_{\max}$.

Enforcement: μ is stored in a monotone counter register (one-way, e.g., eFuse-backed or SEP anti-replay counter), incremented by the enclave before releasing the biometric prompt, checked by the enclave after. The biometric API call is wrapped in a capability token that is single-use: $CT = \text{HMAC}(K_{\text{prompt}}, \text{epoch} \parallel \mu \parallel \text{nonce})$ — accepted by the biometric daemon only if $\mu=0$ and the counter latch hasn't been set. Actually better: the biometric daemon itself checks the latch register.

Also: the biometric prompt must be preceded by the “ambient exhaustion proof” $\pi =$ Merkle commitment over the per-channel evaluation receipts, twice. So the daemon verifies π before displaying any UI. This prevents an attacker from skipping ambient and forcing prompts.

Exhaustion Proof: $\pi_{\text{exh}} = (\text{epoch}, \{ (i, \text{LLR}_i^{\wedge}(r), \text{receipt_sig}_i^{\wedge}(r)) \}_{i \in \text{Channels}}, r \in \{1,2\}, \text{merkle_root})$ where $\text{receipt_sig}_i = \text{HMAC}(K_{\text{ch}_i}, \dots)$ or attester signatures for remote channels (e.g., the watch signs its own PPG/gait attestation with its own SEP key — remote attestation).

Nice: the smartwatch is a separate device, so its attestations must be signed by the watch's enclave and transmitted over BLE with a freshness nonce.

Now, the “sovereign identity” protocol part: the session token. Sovereign Session Token SST: $\text{SST} = \{\text{sub (DID)}, \text{epoch}, \text{exp}, \text{cnf (jkt of PK_SE)}, \text{amr (auth methods)}, \text{acr (level)}, \text{ambient_llr}, \mu_e, \text{sig_SE}\}$

Presented to relying parties. RPs verify via the identity provider's public key + the device attestation. This makes it OIDC-compatible: SST is an ID token with cnf.jkt for DPoP binding, $\text{amr} = [\text{“swk”}, \text{“hwk”}, \text{“ping”}, \text{“fpt”}]$ etc.

Assurance levels: AAL0 (ambient only, 0 touches), AAL1 (ambient + 1 biometric). Map to NIST SP 800-63B AAL2/AAL3.

Also FRR/FAR compounding: $\text{FAR}_{\text{total}} = \text{FAR}_{\text{ambient}} + (\text{ambient fail prob}) * \text{FAR}_{\text{biometric}}$. With $\text{FAR}_{\text{amb}} = 1e-9$ and biometric $\text{FAR} = 1e-6$, $\text{FAR}_{\text{total}} \approx 1e-9 * (1 - p_{\text{fail}}) + p_{\text{fail}} * 1e-6$. Since p_{fail} (ambient fails) is small in normal operation (~ 0.02), $\text{FAR}_{\text{total}} \approx 2e-8$. Good.

And friction: $E[\text{touches}] = P(\text{ambient fail}) * 1 \approx 0.02$ per epoch. That's the zero-friction guarantee: expected biometric touches $\approx 2\%$, hard-capped at 1.

Let me also add: cognitive fatigue model. Define fatigue $F(n)$ as a monotonically increasing function of the number of biometric prompts n in a window, e.g., $F(n) = n^{1+\gamma}$ or a saturating function. The founder rule enforces $F \rightarrow \text{minimal}$.

Actually a nice formalization: the prior art FIDO2/passkeys model. In the prior art, each credential (site) requires its own biometric assertion because each RP credential is bound to a distinct key and the “user verification” flag requires a fresh user gesture per authentication. The WebAuthn spec

PATENT CLAIMS

Application: Hierarchical Ambient-Exhaustion Biometric Authentication and Zero-Friction Sovereign Identity Protocol (MoatID) **Docket:** MM-ID-PROV-009 **Filing basis:** 35 U.S.C. § 111(b); claims drafted under 35 U.S.C. § 112(a)–(d)

Claim 1 — System (Independent)

1. An identity verification system for hierarchical ambient-exhaustion authentication of a human subject, the system comprising:

a plurality of ambient evidence sources configured to generate an ambient identity signal vector, the plurality of ambient evidence sources including at least (i) a wireless radio transceiver configured to measure at least one of received signal strength, time-of-flight, and channel state information with respect to one or more external anchors, (ii) an inertial measurement unit mechanically coupled to a host device, and (iii) a body-worn device interface configured to receive at least one of a motion signal and a physiological signal from a body-worn device co-located with the human subject;

a biometric sensor configured to transduce a biometric modality of the human subject;

a hardware root-of-trust element comprising a non-exportable private key stored in a key slot that is unreadable by a rich operating system executing on the host device, an epoch register configured to hold a monotonic epoch identifier and a monotonic nonce, and a biometric-actuation counter resident in memory addressable only from a privilege domain higher than the rich operating system;

at least one processor; and

a non-transitory computer-readable medium storing instructions that, when executed by the at least one processor, cause the system to perform operations comprising:

- a. opening an authentication epoch by writing a new epoch identifier and an incremented nonce to the epoch register, initializing the biometric-actuation counter to a budget value $\beta := 1$, and establishing an epoch deadline;

- b. instantiating, for the epoch, a friction lattice comprising a partially ordered set $(F, \sqsubseteq)$ of authentication factors, each factor $f \in F$ having an assigned friction cost $\phi(f)$ and an evidence coverage set $C(f)$, wherein $f_i \sqsubseteq f_j$ holds if and only if $\phi(f_i) \leq \phi(f_j)$ and $C(f_i) \subseteq C(f_j)$, and wherein the friction lattice includes a biometric factor constituting a maximal element $\top$ of the partially ordered set;
 - c. within the epoch and prior to any energization of the biometric sensor, ascending the friction lattice in strictly non-decreasing order of friction cost by evaluating successive minimal incomparable elements and joins thereof, and for each evaluated element sampling a corresponding ambient evidence source and computing a per-source evidence term;
 - d. fusing the per-source evidence terms with per-source trust weights into a fused posterior identity confidence $P(\text{identity} \mid A)$ for the ambient identity signal vector A ;
 - e. halting the ascent and asserting an identity assertion without energizing the biometric sensor when $P(\text{identity} \mid A) \geq \tau_{\text{assert}}$, where τ_{assert} is a first threshold;
 - f. determining an ambient-exhaustion condition when all elements of the friction lattice having friction cost lower than $\phi(\top)$ have been consumed and $\tau_{\text{floor}} \leq P(\text{identity} \mid A) < \tau_{\text{assert}}$, where τ_{floor} is a second threshold lower than the first threshold;
 - g. only upon determining the ambient-exhaustion condition and only when the biometric-actuation counter equals one, decrementing the biometric-actuation counter to zero, committing the decrement to the hardware root-of-trust element, and thereafter energizing the biometric sensor exactly once within the epoch to acquire a biometric sample;
 - h. fusing the biometric sample with the ambient identity signal vector A to produce a fused confidence, and asserting the identity assertion when the fused confidence $\geq \tau_{\text{assert}}$;
 - i. refusing to assert the identity assertion and holding the epoch in a failed state until a subsequent epoch is opened when the biometric-actuation counter equals zero and the fused confidence $< \tau_{\text{assert}}$; and
 - j. signing, with the non-exportable private key within the hardware root-of-trust element, a payload binding at least the epoch identifier, the incremented nonce, a digest of the ambient identity signal vector, a transcript of the friction-lattice ascent, and a terminal value of the biometric-actuation counter, thereby producing a cryptographically attestable assertion of the ambient-exhaustion determination.
-

Claim 2 — Dependent (BLE RSSI Triangulation)

2. The system of claim 1, wherein the wireless radio transceiver comprises a Bluetooth Low Energy radio, the one or more external anchors comprise at least three fixed Bluetooth Low Energy anchor beacons disposed at respective known anchor coordinates,

and the instructions further cause the system to:

measure, for each of the at least three fixed Bluetooth Low Energy anchor beacons, a received signal strength indicator value $RSSI_k$ and an associated measurement timestamp;

convert each received signal strength indicator value $RSSI_k$ into a range estimate $\hat{d}_k$ by inverting a log-distance path-loss model of the form

$$RSSI_k = RSSI_0 - 10 \cdot n \cdot \log_{10}(\hat{d}_k / d_0) + X_\sigma,$$

where $RSSI_0$ is a calibrated reference power at a reference distance d_0 , n is a calibrated path-loss exponent, and X_σ is a zero-mean random shadowing term;

solve a weighted nonlinear least-squares trilateration over the range estimates $\hat{d}_k$ to obtain a position estimate $\hat{x}$ of the host device and an associated positional covariance matrix Σ , wherein each range estimate is weighted in inverse proportion to a variance derived from a signal-to-noise ratio of the corresponding received signal strength indicator value;

compute a proximity discriminant as a Mahalanobis distance $DM = \sqrt{(\hat{x} - x_{zone})^T \Sigma^{-1} (\hat{x} - x_{zone})}$ between the position estimate $\hat{x}$ and a centroid x_{zone} of an authorized zone;

reject any range estimate whose post-fit residual exceeds a residual threshold, and reject the position estimate when a major axis of an uncertainty ellipse derived from Σ exceeds an axis bound, the rejection being operative to suppress non-line-of-sight and multipath conditions; and

admit the proximity discriminant into the ambient identity signal vector A as a per-source evidence term having a trust weight assigned inversely proportional to a trace of the positional covariance matrix Σ , such that a degraded geometric dilution of precision reduces the trust weight rather than causing a false negative.

Claim 3 — Dependent (Smartwatch Cadence Correlation)

3. The system of claim 1, wherein the body-worn device comprises a smartwatch having a wrist-mounted second inertial measurement unit and a photoplethysmographic sensor, and the instructions further cause the system to:

extract a first cadence time series $c_w(t)$ from the wrist-mounted second inertial measurement unit and a second cadence time series $c_p(t)$ from the inertial measurement unit mechanically coupled to the host device, over a common observation window;

remove a common-mode component from each of the first cadence time series and the second cadence time series by subtracting a respective window mean and dividing by a respective window standard deviation, thereby producing normalized series;

compute a normalized cross-correlation $\rho(\tau)$ between the normalized series and determine a peak correlation coefficient ρ^* and a corresponding lag τ^* ;

determine a same-body binding condition as satisfied only when $\rho^* \geq \rho_{\min}$, $|\tau^*| \leq \tau_{\max}$, and a spectral phase-coherence measure between the normalized series over a stride-frequency band exceeds a coherence threshold;

verify, from the photoplethysmographic sensor, a pulsatility index within a physiological range and a heart-rate value whose residual against a motion-artifact component of the wrist-mounted second inertial measurement unit falls below a physiological plausibility bound, the verification being operative to distinguish the smartwatch being worn on the human subject from the smartwatch being carried in a container;

assign a trust weight of zero to the at least one motion signal and physiological signal received from the body-worn device unless both the same-body binding condition and the pulsatility verification are satisfied; and

admit the peak correlation coefficient ρ^* and the lag τ^* into the ambient identity signal vector A as at least one per-source evidence term only when the trust weight is nonzero.

Claim 4 — Dependent (Secure Enclave Attestation)

4. The system of claim 1, wherein the hardware root-of-trust element comprises a hardware-isolated secure enclave having a plurality of platform configuration registers holding boot-time measurements of executed firmware and operating-system components, and the instructions further cause the system to:

serialize the ambient identity signal vector A , the transcript of the friction-lattice ascent, the terminal value of the biometric-actuation counter, the epoch identifier, and the incremented nonce into a canonical deterministic byte encoding;

compute a cryptographic digest of the canonical deterministic byte encoding;

generate the signed payload by signing the cryptographic digest with the non-exportable private key, the non-exportable private key having a corresponding public key certified within a manufacturer attestation certificate chain terminating at a hardware-vendor root certificate;

include within the signed payload a measurement vector read from the plurality of platform configuration registers at the time of signing;

cause a relying party to verify the signed payload by validating a signature against the public key, validating the manufacturer attestation certificate chain, comparing the measurement vector against a pre-provisioned golden manifest of expected platform configuration register values, and validating freshness of the signed payload against the monotonic nonce; and

suppress emission of the signed payload from the secure enclave when the measurement vector deviates from the golden manifest, when the biometric-actuation counter is in an indeterminate state, or when the epoch deadline has expired;

wherein the biometric-actuation counter is implemented as a write-once-per-epoch monotonic register within enclave-protected memory and is gated by a hardware interlock that de-energizes the biometric sensor whenever the biometric-actuation counter does not equal one, such that a compromised rich operating system cannot restore the budget value β within the epoch.

Claim 5 — Dependent (Friction-Lattice Partial Ordering)

5. The system of claim 1, wherein the instructions cause the ascent of the friction lattice to be performed as an expected-friction minimization subject to a confidence constraint by:

representing each authentication factor $f \in F$ by a tuple $(\phi(f), C(f), w(f))$, wherein $\phi(f)$ is the assigned friction cost expressed as an expected user-interruption time, $C(f)$ is the evidence coverage set, and $w(f)$ is a reliability weight updated from a historical record of per-factor discrimination performance;

at each ascent step, selecting an antichain $M \subseteq F$ of pairwise incomparable minimal elements that have not yet been consumed, computing a join $\sqcup M$, and evaluating a fused posterior of the join $\sqcup M$;

terminating the ascent at the first join $\sqcup M$ for which the fused posterior $\geq \tau_{\text{assert}}$, whereby the expected friction $\sum_{f \in \sqcup M} \phi(f)$ is minimized over all antichains of the friction lattice whose fused posterior $\geq \tau_{\text{assert}}$;

maintaining the friction lattice as dynamically re-weighted by excluding from F any factor whose reliability weight $w(f)$ falls below a retirement threshold and by re-inserting any previously excluded factor upon restoration of a corresponding ambient evidence source;

recording each ascent step, the antichain M selected at each step, and the fused posterior evaluated at each step in the transcript of the friction-lattice ascent; and

ascribing the maximal element $\top$ a friction cost $\phi(\top)$ strictly greater than the friction cost of every other element of F , whereby the maximal element $\top$ is reachable by the ascent only after every lower-friction element has been consumed.

Claim 6 — Method (Independent)

6. A computer-implemented method for hierarchical ambient-exhaustion identity verification, the method comprising:

opening, by at least one processor of a host device, an authentication epoch by incrementing a monotonic nonce in a hardware root-of-trust element, assigning an epoch identifier, initializing a biometric-actuation counter within the hardware root-of-trust element to a budget value $\beta := 1$, and establishing an epoch deadline;

constructing a friction lattice comprising a partially ordered set $(F, \sqsubseteq)$ of authentication factors, each factor $f \in F$ having an assigned friction cost $\phi(f)$ and an evidence coverage set $C(f)$, wherein $f_i \sqsubseteq f_j$ if and only if $\phi(f_i) \leq \phi(f_j)$ and $C(f_i) \subseteq C(f_j)$, and wherein a biometric factor of the host device constitutes a maximal element $\top$ of the partially ordered set;

ascending the friction lattice within the epoch, prior to any energization of a biometric sensor of the host device, by iteratively evaluating minimal incomparable elements of the partially ordered set in strictly non-decreasing order of friction cost;

sampling, at each evaluated element, a corresponding ambient evidence source selected from a set that includes a wireless received-signal-strength measurement, an inertial measurement of the host device, and at least one of a motion signal and a physiological signal received from a body-worn device;

generating an ambient identity signal vector A from the sampled ambient evidence sources;

computing an ambient entropy score from the ambient identity signal vector A , the ambient entropy score being a scalar in a closed interval and representing an information gain in bits relative to a prior impostor distribution;

comparing the ambient entropy score against a first threshold τ_{assert} and a second threshold τ_{floor} , wherein $\tau_{\text{floor}} < \tau_{\text{assert}}$;

asserting an identity assertion, without energizing the biometric sensor, when the ambient entropy score $\geq \tau_{\text{assert}}$;

determining an ambient-exhaustion condition when all elements of the friction lattice having friction cost lower than $\phi(\top)$ have been consumed and $\tau_{\text{floor}} \leq$ the ambient entropy score $< \tau_{\text{assert}}$;

responsive to determining the ambient-exhaustion condition, reading the biometric-actuation counter, and only when the biometric-actuation counter equals one, decrementing the biometric-actuation counter to zero, committing the decrement to non-volatile storage within the hardware root-of-trust element, and thereafter energizing the biometric sensor exactly once within the epoch to acquire a biometric sample;

fusing the biometric sample with the ambient identity signal vector A to produce a fused confidence, and asserting the identity assertion when the fused confidence $\geq \tau_{\text{assert}}$;

denying authentication and holding the epoch in a failed state until a subsequent epoch is opened when the biometric-actuation counter equals zero and the fused confidence $< \tau_{\text{assert}}$; and

signing, with a non-exportable private key stored within the hardware root-of-trust element, a payload binding the epoch identifier, the monotonic nonce, a digest of the ambient identity signal vector A , a transcript of the ascent, the ambient entropy score, and a terminal value of the biometric-actuation counter.

Claim 7 — Dependent (Hard Cap of Exactly One Biometric Touch Per Epoch)

7. The method of claim 6, wherein the biometric-actuation counter is enforced as a hard cap of exactly one biometric touch per epoch, and the method further comprises:

initializing the biometric-actuation counter to the budget value $\beta := 1$ exclusively at the opening of the epoch, and making the biometric-actuation counter non-resettable and non-incrementable to any value greater than one for a duration of the epoch;

implementing the biometric-actuation counter as a monotonic register within memory addressable only from a privilege domain higher than a rich operating system of the host device, and gating energization of the biometric sensor with a hardware interlock that permits energization only when the biometric-actuation counter equals one;

committing the decrement of the biometric-actuation counter to zero by a non-volatile write that completes before a first sample is captured by the biometric sensor, such that a power loss, a reset, or a fault injection occurring between the committing and the capture cannot restore the budget value β within the epoch;

refusing, at the privilege domain higher than the rich operating system, any request to energize the biometric sensor within the epoch subsequent to the decrement, irrespective of any instruction issued by the rich operating system;

defining a single biometric touch as one continuous contact interval with the biometric sensor, counting a plurality of distinct contact peaks occurring within the one continuous contact interval as a single biometric touch, and refusing any re-arm of the biometric sensor within the epoch after termination of the one continuous contact interval; and

restoring the budget value β to one only at an epoch boundary, the epoch boundary being defined by at least one of: expiration of the epoch deadline, a transition of the host device into a locked state, a change of a relying-party audience identifier, and a detected loss of a same-body binding condition between the host device and the body-worn device;

wherein the terminal value of the biometric-actuation counter is included in the signed payload such that the relying party can verify that no more than one biometric touch occurred within the epoch.

Claim 8 — Dependent (Ambient Entropy Scoring)

8. The method of claim 6, wherein computing the ambient entropy score comprises:

computing, for each per-source evidence term a_i of the ambient identity signal vector A , a per-source log-likelihood ratio

$$LLR_i = \log(p(a_i | H_1) / p(a_i | H_0)),$$

where H_1 denotes a genuine-subject hypothesis and H_0 denotes an impostor hypothesis, and where $p(a_i | H_1)$ and $p(a_i | H_0)$ are obtained from pre-provisioned per-source probability density models;

computing a composite log-likelihood ratio as a trust-weighted sum $\Lambda = \sum_i w_i \cdot LLR_i$, where w_i is a per-source trust weight derived from at least one of a measured signal-to-noise ratio, a geometric dilution of precision, a same-body binding condition, and a historical discrimination performance of the corresponding ambient evidence source;

converting the composite log-likelihood ratio into a posterior identity probability by $P(\text{identity} | A) = \sigma(\Lambda + \text{logit}(P_0))$, where σ is a logistic function and P_0 is a prior probability of the genuine-subject hypothesis;

computing a residual impostor entropy $H_{\text{res}} = -\sum_j p_j \cdot \log_2 p_j$ over a posterior impostor distribution, and computing the ambient entropy score as an information gain $H_{0_prior} - H_{\text{res}}$ expressed in bits, where H_{0_prior}

PART: LIVING MARKS PROCEDURAL GLYPHS

PROVISIONAL PATENT APPLICATION COVER SHEET

Docket Number: MM-MARKS-PROV-010 **Title of the Invention:** Cryptographically-Seeded Continuous Vector Glyphs and Procedural Living Authenticity Marks **First Named Inventor:** Brennan DeCrow **Residence:** Oregon, United States **Assignee:** ManyMoats (ATESO / Magma) **Filing Basis:** 35 U.S.C. § 111(b) **Filing Date:** 2026-09-25

INVENTOR STATEMENT

The invention disclosed herein represents original, unencumbered technical innovations conceived and reduced to practice by Brennan DeCrow.

PROVISIONAL PATENT APPLICATION

Application No.: MM-MARKS-PROV-010 **Filing Basis:** 35 U.S.C. § 111(b) **Drawing Requirement:** 35 U.S.C. § 113 (figures appended separately; textual figure references infra) **Enablement / Written Description / Best Mode:** 35 U.S.C. § 112(a) **Definiteness:** 35 U.S.C. § 112(b)

TITLE OF THE INVENTION

**CRYPTOGRAPHICALLY-SEEDED CONTINUOUS VECTOR GLYPHS AND
PROCEDURAL LIVING AUTHENTICITY MARKS**

INVENTOR

Brennan DeCrow, sole inventor. Assignee (if any): ManyMoats / ATESO.

CROSS-REFERENCE TO RELATED APPLICATIONS

Not applicable. This provisional application is filed as an original disclosure and establishes priority for the subject matter recited in the Claims appended hereto.

1. TECHNICAL FIELD

The present disclosure relates generally to computer graphics, procedural geometry synthesis, human–computer interaction, and applied cryptography. More specifically, it relates to the deterministic synthesis of vector glyph geometry and continuous-time optical animation from a cryptographic digest, to the coupling of such geometry to a live, append-only cryptographic receipt chain, and to protocols for verifying the authenticity of a rendered mark by an observer or machine agent who does not possess the originating secret seed.

The disclosure spans the following technical arts, each of which is separately enabled herein:

- a. **Extendable-output key derivation** (BLAKE3 XOF) and domain-separated seed construction;
 - b. **Deterministic pseudo-random coordinate synthesis** (Mulberry32, counter-reseeded);
 - c. **Closed-form planar curve generation** (radial Fourier synthesis and non-uniform Catmull–Rom → cubic Bézier conversion);
 - d. **Parametric surface (spline mesh) displacement** under band-limited spatial modes;
 - e. **Colorimetric synthesis** in the OKLab perceptual color space with CIE 1931 photopic luminance constraint;
 - f. **Coupled-oscillator continuous-time dynamics** (Kuramoto phase coupling) integrated against a monotonic hardware clock in fixed-point arithmetic;
 - g. **Electrostatic proximity field** computation with soft-core (Plummer) regularization and second-order critically-damped response;
 - h. **Optical-density modulation** driven by receipt-chain commit events;
 - i. **Challenge–response anti-spoofing protocols** binding rendered output to live cryptographic state.
-

2. BACKGROUND OF THE INVENTION

2.1 The Fatal Defects of Static SVG, PNG, and WebP Marks

Contemporary brand, provenance, and authenticity marks are distributed almost exclusively as one of three static file formats: Scalable Vector Graphics (SVG, an XML vector format per W3C Recommendation), Portable Network Graphics (PNG, a DEFLATE-compressed raster format per ISO/IEC 15948), and WebP (a VP8L/VP8-lossy raster format per the Google/ITU-T specification). Each of these formats exhibits structural defects that render it categorically unsuitable as an authenticity instrument. These defects are not matters of degree; they are architectural.

2.1.1 Defect I — Trivial Exfiltration and Perfect Replication

A static mark is a **finite, self-contained, byte-string**. SVG is literal source text. PNG and WebP are byte streams decodable in full by any conforming decoder. Consequently:

- The complete information content of the mark — every coordinate, every color, every stroke width — is available to any party that can perform a single GET request, read a cache, screenshot a viewport, or observe a framebuffer.
- There is **no residual secret**. The mark is not a *rendering of* a secret; it *is* the entire artifact. Once exfiltrated, an attacker possesses bit-for-bit identical material.
- The replication cost is **O(1)**: one copy operation. There is no computational asymmetry between the legitimate holder and the attacker. Cryptographic security, by contrast, depends on an asymmetry in *computational work*, and a static file provides none.

Formally: let a static mark be a byte string $\mathbf{m} \in \{0,1\}^n$. The probability that an adversary **A** in possession of $\mathbf{m}$ can produce $\mathbf{m}' = \mathbf{m}$ is 1. The adversary's advantage is 1 with zero work. No cryptographic construction can rescue a system whose entire secret has been published.

2.1.2 Defect II — Unconditional Spoofability and the Absence of a Verification Predicate

A static mark admits **no verification predicate** $P(\mathbf{m}, K)$ whose evaluation requires knowledge of a key K . The

MM-MARKS-PROV-010

Title: Cryptographically-Seeded Continuous Vector Glyphs and Procedural Living Authenticity Marks

1. A system for generating, rendering, and maintaining cryptographically-seeded procedural authenticity marks, the system comprising:

at least one hardware processor and at least one graphics processing unit (GPU) having device memory associated therewith;

a seed interface configured to receive or derive a cryptographic seed value S having at least 128 bits of entropy;

a deterministic pseudorandom sequence generator configured to be initialized with a state derived from the seed value S and to emit, as a function of said state alone and without consumption of external entropy, a reproducible ordered sequence of pseudorandom scalar values $r_0, r_1, r_2, \dots$, each normalized to the interval $[0, 1]$;

a procedural geometry synthesizer configured to (i) map the reproducible ordered sequence onto a parameter set comprising, for each of an integer $N \geq 3$ of control points, a control-point position P_i in a coordinate space and at least one of a tangent vector and a handle vector, and (ii) construct a continuous parametric spline curve $C(u)$, $u \in [0, 1]$, interpolating or approximating the control-point positions P_i , such that $C(u)$ is uniquely and reproducibly determined by the seed value S ;

an oscillation engine configured to associate with each control-point index i a set of one or more oscillatory modes, each oscillatory mode m being characterized by an amplitude $A_{\{i,m\}}$, an angular frequency $\omega_{\{i,m\}}$, a phase offset $\phi_{\{i,m\}}$, and a unit displacement direction $\hat{d}_{\{i,m\}}$, each of $A_{\{i,m\}}$, $\omega_{\{i,m\}}$, $\phi_{\{i,m\}}$, and $\hat{d}_{\{i,m\}}$ being derived from the reproducible ordered sequence, and configured to compute, for each render frame occurring at a global frame time t , an instantaneous control-point position

$$P_i(t) = P_i + \sum_m A_{\{i,m\}} \cdot \sin(\omega_{\{i,m\}} \cdot t + \phi_{\{i,m\}}) \cdot \hat{d}_{\{i,m\}},$$

thereby producing a time-varying parametric spline $C(u, t)$;

a proximity coupling module configured to determine a cursor position p_c in the coordinate space and to compute, for each control point i and each render frame, a coupling weight $w_i = f(\|p_c - P_i(t)\|)$ for a monotone non-increasing falloff function f , and to apply the coupling weight w_i to modify at least one of $A_{\{i,m\}}$, $\omega_{\{i,m\}}$, $\phi_{\{i,m\}}$, and $\hat{d}_{\{i,m\}}$ prior to or during the computation of $P_i(t)$;

a rendering pipeline configured to tessellate $C(u, t)$ into a set of primitives and to shade the primitives under a daylight illumination model, the daylight illumination model comprising a specified illuminant correlated color temperature, a wavelength-dependent volumetric scattering term, and a view-dependent specular reflectance term; and

a display interface configured to output the shaded primitives to a display device as a living authenticity mark.

2. The system of claim 1, wherein the deterministic pseudorandom sequence generator is a Mulberry32-class generator, and wherein the procedural geometry synthesizer constructs the continuous parametric spline from outputs of said Mulberry32-class generator;

wherein the Mulberry32-class generator comprises:

a 32-bit state register initialized to a value derived from the seed value S;

a state-advance stage configured to compute $a \leftarrow (a + 0x6D2B79F5) \bmod 2^{32}$;

a first avalanche stage configured to compute $t \leftarrow M(a \text{ XOR } (a \gg 15), 1 \text{ OR } a) \bmod 2^{32}$, wherein $\gg$ denotes a logical right shift of an unsigned 32-bit word and $M(x, y)$ denotes a 32-bit integer multiplication of x by y ;

a second avalanche stage configured to compute $t \leftarrow ((t + M(t \text{ XOR } (t \gg 7), 61 \text{ OR } t)) \bmod 2^{32}) \text{ XOR } t$; and

an extraction stage configured to output $r \leftarrow ((t \text{ XOR } (t \gg 14)) \bmod 2^{32}) / 2^{32}$, whereby each output r lies in the interval $[0, 1)$;

and wherein the procedural geometry synthesizer is configured to:

map each successive pair of outputs $(r_{\{2i\}}, r_{\{2i+1\}})$ of the Mulberry32-class generator to a control-point position P_i by a polar mapping $P_i = (q_{\min} + (q_{\max} - q_{\min}) \cdot r_{\{2i+1\}}) \cdot (\cos(2\pi \cdot r_{\{2i\}}), \sin(2\pi \cdot r_{\{2i\}}))$, for specified radial bounds $q_{\min} < q_{\max}$;

construct the continuous parametric spline as a Catmull-Rom spline having a centripetal parameterization exponent $\alpha = 0.5$, evaluated over successive control-point quadruples $(P_{\{i-1\}}, P_i, P_{\{i+1\}}, P_{\{i+2\}})$; and

apply a periodic boundary condition in which index arithmetic is performed modulo N such that the spline forms a closed curve whose tangent vector is continuous across the seam at $u = 0$ and $u = 1$.

3. The system of claim 1, wherein the oscillation engine implements a continuous oscillation physics model comprising:

for each control-point index i , a plurality $K \geq 2$ of oscillatory modes whose angular frequencies $\omega_{\{i,m\}}$ are drawn from a specified frequency band $[\omega_{\min}, \omega_{\max}]$ in a non-harmonic relationship, and whose phase offsets $\phi_{\{i,m\}}$ are drawn from the interval $[0, 2\pi)$;

a spring-mass-damper integrator configured to integrate, at each render frame having frame duration Δt , a state vector (x_i, v_i) for each control point according to a semi-implicit Euler update

$$v_i \leftarrow v_i + \Delta t \cdot (k_s \cdot (P_i + \sum_m A_{\{i,m\}} \sin(\omega_{\{i,m\}} t + \phi_{\{i,m\}}) \hat{d}_{\{i,m\}} - x_i) - c_d \cdot v_i), x_i \leftarrow x_i + \Delta t \cdot v_i,$$

wherein k_s is a spring stiffness coefficient, c_d is a damping coefficient selected such that a damping ratio $\zeta = c_d / (2 \sqrt{k_s})$ lies in the open interval $(0, 1)$, and x_i is the integrated instantaneous control-point position substituted for $P_i(t)$;

a deterministic clock input configured to supply the global frame time t such that, for an identical seed value S and an identical global frame time t , the state vector (x_i, v_i) and the resulting spline $C(u, t)$ are bit-for-bit reproducible on a re-execution of the oscillation engine; and

a periodic-continuity constraint configured to enforce $x_0 \equiv x_N$ and $v_0 \equiv v_N$ at each render frame such that the closed curve remains G^1 -continuous at the seam throughout oscillation.

4. The system of claim 1, wherein the proximity coupling module implements a cursor proximity field coupling comprising:

a falloff function $f(r) = (1 - \text{clamp}(r / R_f, 0, 1))^p$, wherein R_f is a specified field radius and p is a specified falloff exponent, or a softened inverse-distance function $f(r) = 1 / (r^2 + \epsilon^2)^{(q/2)}$, wherein ϵ is a softening constant and q is a specified exponent;

a radial coupling component configured to displace each control point i along the unit vector $(P_i(t) - p_c) / \|P_i(t) - p_c\|$ by an amount proportional to w_i , thereby producing a repulsion of the spline from the cursor position;

a tangential coupling component configured to displace each control point i along a vector perpendicular to $(P_i(t) - p_c)$ by an amount proportional to w_i and to a signed vorticity parameter, thereby producing a swirl of the spline about the cursor position;

a cursor-velocity coupling term configured to scale at least one of the radial coupling component and the tangential coupling component by a function of a cursor velocity dp_c/dt ; and

a displacement clamp configured to limit a total displacement of each control point i to a specified maximum displacement Δ_{max} , such that the spline remains bounded within the coordinate space for all cursor positions and cursor velocities.

5. The system of claim 1, wherein the daylight illumination model comprises:

a spectral radiance term $L(\lambda)$ computed for a specified illuminant correlated color temperature T_c in a range of 5000 K to 7000 K, optionally according to Planck's law

$$L(\lambda, T_c) = (2 h c^2 / \lambda^5) \cdot 1 / (\exp(h c / (\lambda k_B T_c)) - 1),$$

wherein h is Planck's constant, c is the speed of light, λ is wavelength, and k_B is Boltzmann's constant;

a Rayleigh volumetric scattering term having a wavelength dependence proportional to λ^{-4} and a Mie volumetric scattering term having a wavelength dependence proportional to λ^{-n} for a specified exponent n in a range of 1.0 to 2.0, each said scattering term being applied as a function of an optical depth along a view ray through the tessellated primitives;

a diffuse reflectance term comprising a Lambertian bidirectional reflectance distribution function having an albedo parameter derived from the reproducible ordered sequence;

a specular reflectance term comprising a Fresnel term evaluated by the Schlick approximation $F(\theta) = F_0 + (1 - F_0)(1 - \cos \theta)^5$ for a specified normal-incidence reflectance F_0 , combined with a microfacet distribution lobe; and

a tone-mapping and transfer-function stage configured to map a linear-light radiance value produced by the combination of the spectral radiance term, the scattering terms, the diffuse reflectance term, and the specular reflectance term into a display-referred signal by application of a tone-mapping operator followed by an sRGB electro-optical transfer function.

6. A computer-implemented method for generating and updating a living authenticity mark, the method comprising:

at a computing device comprising at least one hardware processor and at least one graphics processing unit (GPU), receiving or deriving a cryptographic seed value S having at least 128 bits of entropy;

initializing a deterministic pseudorandom sequence generator with a state derived from the seed value S and emitting, as a function of said state alone and without consumption of external entropy, a reproducible ordered sequence of pseudorandom scalar values $r_0, r_1, r_2, \dots$, each normalized to the interval $[0, 1)$;

synthesizing, by a procedural geometry synthesizer, a continuous parametric spline curve $C(u)$ determined uniquely and reproducibly by the seed value S , by mapping the reproducible ordered sequence onto control-point positions P_i for $i = 0 \dots N-1$, $N \geq 3$, and interpolating or approximating the control-point positions P_i with a parametric spline;

advancing, by an oscillation engine at each of a succession of render frames occurring at a global frame time t , each control point according to a superposition of oscillatory modes to produce an instantaneous control-point position $P_i(t)$, and thereby producing a time-varying parametric spline $C(u, t)$;

determining a cursor position p_c and computing, for each control point i , a coupling weight $w_i = f(\|p_c - P_i(t)\|)$ for a monotone non-increasing falloff function f , and modifying at least one parameter of the superposition of oscillatory modes as a function of the coupling weight w_i ;

tessellating $C(u, t)$ into a set of primitives and shading the set of primitives under a daylight illumination model comprising a specified illuminant correlated color temperature, a wavelength-dependent volumetric scattering term, and a view-dependent specular reflectance term; and

outputting the shaded set of primitives to a display device as the living authenticity mark,

wherein the living authenticity mark is continuously updated across the succession of render frames such that the living authenticity mark is rendered in a first visual state at a first global frame time t_1 and in a second, different visual state at a second global frame time t_2 , and wherein the first visual state and the second visual state are each deterministically reproducible from the seed value S and the corresponding global frame time.

7. The method of claim 6, further comprising verifying an authenticity of the living authenticity mark by an anti-spoofing verification procedure, the anti-spoofing verification procedure comprising:

establishing a shared time base between a mark-generating device and a mark-verifying device, such that the mark-generating device and the mark-verifying device agree on the global frame time t within a specified synchronization tolerance;

issuing, from the mark-verifying device to the mark-generating device, a challenge comprising a nonce and a challenge time τ ;

re-deriving, at the mark-verifying device and independently of the mark-generating device, a reference geometry $C_{ref}(u, \tau)$ by executing the deterministic pseudorandom sequence generator, the procedural geometry synthesizer, and the oscillation engine with the seed value S and the challenge time τ ;

capturing, at the mark-verifying device, a presented geometry $C_{obs}(u)$ from a display or from a rendering surface bearing the living authenticity mark;

computing a geometric discrepancy metric between $C_{ref}(u, \tau)$ and $C_{obs}(u)$, the geometric discrepancy metric comprising at least one of a Hausdorff distance, a Fréchet distance, a discrete sample-wise Euclidean distance, and a normalized cross-correlation; and

accepting the living authenticity mark as authentic only if the geometric discrepancy metric is below a specified threshold and the challenge time τ falls within the specified synchronization tolerance of a current global frame time of the mark-verifying device;

whereby a static reproduction of the living authenticity mark captured at a time other than the challenge time τ fails the anti-spoofing verification procedure.

8. The method of claim 6, further comprising executing the procedural geometry synthesizer, the oscillation engine, and the proximity coupling module directly on the GPU, wherein executing directly on the GPU comprises:

allocating, in device memory of the GPU, a vertex buffer having a usage flag set to a combination of STORAGE, VERTEX, and COPY_SRC;

allocating a uniform buffer configured to carry at least the seed value S , the global frame time t , the cursor position p_c , and a parameter block;

dispatching a compute shader, written in a GPU shading language, that reads the uniform buffer, executes the deterministic pseudorandom sequence generator and the oscillation engine for each control-point index, and writes the instantaneous control-point positions $P_i(t)$ directly into the vertex buffer;

binding the vertex buffer as a vertex-attribute source to a render pipeline without an intervening read-back of the vertex buffer to host memory; and

performing, for each render frame, a single host-to-device update of the uniform buffer comprising at most the global frame time t and the cursor position p_c , such that no per-frame read-back of control-point positions or of tessellated primitives from the GPU to the at least one hardware processor is performed;

whereby a per-frame data transfer volume between the at least one hardware processor and the GPU is bounded by a size of the uniform buffer and is independent of N .

9. The method of claim 6, further comprising rendering a tamper-evident visual state reflection, the tamper-evident visual state reflection comprising:

computing a cryptographic digest H of a parameter block comprising at least the seed value S and one or more of an amplitude set $\{A_{\{i,m\}}\}$, a frequency set $\{\omega_{\{i,m\}}\}$, a phase set $\{\phi_{\{i,m\}}\}$, and an albedo set, using a cryptographic hash function;

mapping a plurality of bits or bytes of the cryptographic digest H onto at least one visually observable modulation of the living authenticity mark, the at least one visually observable modulation comprising at least one of a glyph-shaped modulation of the spline, a per-segment color modulation, a per-control-point amplitude offset, and an orientation modulation of the spline;

re-computing, at a verification time, the cryptographic digest H from the parameter block as recovered from a marked object or a transmitted mark descriptor; and

comparing the re-computed cryptographic digest H against a digest recovered from or decoded out of the at least one visually observable modulation,

whereby any alteration of the seed value S or of the parameter block produces a deterministic and visually observable change in the living authenticity mark.

10. The method of claim 6, further comprising implementing a tamper-evident visual state reflection in which a visual state of the living authenticity mark is a deterministic function of a complete pipeline state, the implementation comprising:

defining a pipeline state vector comprising the seed value S , the global frame time t , the cursor position p_c , and the parameter block;

computing a state commitment as a cryptographic hash of the pipeline state vector and a secret key held by a mark-issuing authority;

deriving, from the state commitment, a modulation index m and a modulation phase ψ ;

applying, to the time-varying parametric spline $C(u, t)$, a modulation

$$C'(u, t) = C(u, t) + m \cdot \sin(2\pi \cdot n_{\ell} \cdot u + \psi) \cdot \hat{n}(u),$$

wherein n_{ℓ} is a specified integer spatial frequency and $\hat{n}(u)$ is a unit normal vector of $C(u, t)$ at parameter value u , thereby embedding the state commitment into the geometry of the living authenticity mark;

rendering the modulated spline $C'(u, t)$ under the daylight illumination model; and

at a verification time, recovering the modulation index m and the modulation phase ψ from an observed instance of the living authenticity mark by a spectral or phase-demodulation operation on the observed instance, recomputing the state commitment, and rejecting the living authenticity mark as tampered if the recovered modulation index m and modulation phase ψ do not match the recomputed state commitment within a specified tolerance;

whereby a modification of any element of the pipeline state vector by an unauthorized party produces a change in the rendered geometry of the living authenticity mark that is detectable by

PART: MAGIC CURSOR VISCOELASTIC UI

PROVISIONAL PATENT APPLICATION COVER SHEET

Docket Number: MM-CURSOR-PROV-011 **Title of the Invention:** Kinetic Physical Field Cursor Coupling and Viscoelastic Spatial Tendril User Interface Interactions **First Named Inventor:** Brennan DeCrow **Residence:** Oregon, United States **Assignee:** ManyMoats (ATESO / Magma) **Filing Basis:** 35 U.S.C. § 111(b) **Filing Date:** 2026-09-25

INVENTOR STATEMENT

The invention disclosed herein represents original, unencumbered technical innovations conceived and reduced to practice by Brennan DeCrow.

PROVISIONAL PATENT APPLICATION

Application Type: Provisional — 35 U.S.C. § 111(b) **Filing Entity:** ManyMoats / ATESO **Inventor:** Brennan DeCrow **Docket:** MM-CURSOR-PROV-011

TITLE OF THE INVENTION

**KINETIC PHYSICAL FIELD CURSOR COUPLING AND VISCOELASTIC
SPATIAL TENDRIL USER INTERFACE INTERACTIONS**

CROSS-REFERENCE TO RELATED APPLICATIONS

Not applicable. This is an original provisional filing establishing priority date for the subject matter disclosed herein.

FIELD OF THE INVENTION

The present invention relates generally to human-computer interaction, and more particularly to:

- a. a pointer/input-device subsystem in which the on-screen cursor is computationally modeled as a **rigid body possessing finite mass, linear and angular momentum, and a continuous acceleration state**, rather than as a stateless point particle;
 - b. a **viscoelastic spatial tendril network** integrated under Extended Position-Based Dynamics (XPBD) that trails, couples to, and detaches from the cursor body;
 - c. **simulated physical field coupling** — compactly-supported attractive and screened-Coulomb repulsive potentials — by which the cursor body exerts forces on interactive user-interface elements, causing those elements to deflect, compress, oscillate, and snap;
 - d. a **sub-millisecond dispatch architecture** in which pointer event capture and physical simulation execute in a dedicated WebGPU/Metal compute worker at 1,000 Hz or greater, decoupled from the browser/OS 60 Hz main event loop.
-

BACKGROUND OF THE INVENTION

1. The Legacy Cursor Is Physically Indefensible

Every mainstream graphical operating system shipped since Xerox PARC (Xerox Alto, 1973) through Windows 11, macOS 15, iOS 18, and Android 15 models the pointer as a **stateless zero-mass point particle** whose sole state is a pair of integer or floating-point display coordinates. Formally, the legacy cursor state is

$$S_{\text{legacy}} = (x, y) \in \mathbb{Z}^2$$

with an update rule applied only upon receipt of a discrete hardware delta:

$$(x, y) \leftarrow (x + \Delta x, y + \Delta y)$$

This model has **no derivative, no mass, no momentum, and no force state**. Consequently it is incapable of, *inter alia*:

Legacy property	Physical consequence	User-visible defect
$m \equiv 0$	$F = ma$ undefined; cursor cannot exert force on anything	No coupling to UI elements whatsoever
v undefined between events	Velocity is a Dirac comb, not a function	No inertia, no ballistic prediction, no weight
$a \equiv 0$	No force field can be sourced	No attraction, no repulsion, no magnetism
Binary hover $h \in \{0,1\}$	$dH/dp = \sum \delta(p - \partial A)$, infinite-stiffness discontinuity	Jarring, instantaneous, binary hover transitions
Hit-test at 60 Hz	Nyquist limit 30 Hz	Temporal aliasing of fast pointer motion
Separate compositor plane	Temporal shear vs. app content	Cursor “floats” off the content it points at

2. Formal Statement of the Velocity Defect

Consider a hardware pointing device reporting deltas at polling frequency f_{HID} . The implied velocity over one report interval is

$$v_{\text{implied}} = \Delta x / (1/f_{\text{HID}}) = \Delta x \cdot f_{\text{HID}}$$

For the *same physical hand motion* of 2000 px/s:

- At $f_{\text{HID}} = 125$ Hz: $\Delta x = 16$ px $\Rightarrow v_{\text{implied}} = 2000$ px/s
- At $f_{\text{HID}} = 1000$ Hz: $\Delta x = 2$ px $\Rightarrow v_{\text{implied}} = 2000$ px/s

Both are correct, but **neither is ever computed or stored** by any commodity OS cursor stack. The cursor position is quantized to a zero-order hold (ZOH) staircase at the display refresh rate $f_{\text{disp}} \approx 60$ Hz. The reconstruction error of ZOH for a signal of bandwidth B is bounded by

$$e_{\text{ZOH}} \leq (1/2) T_s \cdot \max|\dot{x}| = (1/2)(1/60) \cdot v_{\text{max}}$$

At $v_{\text{max}} = 3000$ px/s this yields $e_{\text{ZOH}} \leq 25$ px of peak spatial error — nearly a third of a 96 px interaction radius. The legacy cursor is, therefore, not merely *impoverished* physically; it is **measurably inaccurate**.

3. The Infinite-Impedance Discontinuity

Let $A \subset \mathbb{R}^2$ be the screen-space region of an interactive element. The legacy hover state is the indicator function

$$H(p) = 1[p \in A]$$

Its spatial gradient is a sum of Dirac distributions supported on the element boundary ∂A :

$$\nabla H(p) = \sum_{\partial A} \delta(p - s) \hat{n}(s)$$

A physical interface with an infinite impedance discontinuity at its boundary is unrealizable — real surfaces have finite stiffness k and therefore finite contact force $F = k \cdot \delta$ over a finite deflection δ . The perceptual result of the legacy step function is the familiar “**snapping/jarring**” **hover**, a defect that legions of designers have attempted to paper over with CSS transition timing functions. Those transitions are **cosmetic interpolation applied after the fact**; they do not constitute a physical model, carry no momentum, and produce no force on anything.

4. The Latency Defect

Measured input-to-photon latency in commodity browser stacks routinely exceeds 60 ms and reaches 200 ms under load. The main-thread event loop cannot service pointer events at a rate exceeding the display refresh, and any main-thread work (layout, script, garbage collection) directly delays pointer processing. This is a structural defect of the architecture, not an implementation detail.

5. What Is Needed

There is a need for a pointer subsystem in which:

1. The cursor is a **rigid body** with $m > 0$, velocity, acceleration, and angular state — integrated by a real numerical integrator;
2. The cursor sources **physical fields** that exert continuous, finite, compactly-supported forces on UI elements within a definable interaction radius R ;
3. Those UI elements are themselves **physical bodies** with viscoelastic anchoring, so they deflect, compress, oscillate, and settle — with smooth, energy-consistent transitions rather than binary state flips;
4. A **viscoelastic tendril network** trails the cursor and couples it to nearby elements under XPBD constraints;
5. The entire simulation runs at **≥ 1000 Hz in a dedicated compute worker**, decoupled from the main thread, with sub-millisecond input-to-physical-state latency.

The present invention provides all five.

SUMMARY OF THE INVENTION

In a first aspect, there is provided a cursor system wherein the pointer is modeled as a rigid body $\mathbf{B}$ with mass m_p , position $\mathbf{x}_p \in \mathbb{R}^2$, velocity $^{**}_v_p$, acceleration $^{**}_a_p$, orientation θ_p , and angular velocity ω_p , integrated by a semi-implicit symplectic Euler or velocity-Verlet scheme at a fixed simulation timestep $\Delta t \leq 1$ ms.

In a second aspect, the cursor body is coupled to a **viscoelastic spatial tendril network** — an array of N coupled mass nodes connected by distance and bending constraints solved under Extended Position-Based Dynamics (XPBD), with per-constraint compliance $\alpha = 1/k$ and a **Standard Linear Solid (Zener) viscoelastic internal state variable** per segment producing both instantaneous elastic response and time-dependent stress relaxation and creep.

In a third aspect, the cursor body sources a **compactly-supported attractive potential** (a C^2 Wendland-type kernel with support radius R) and a **softened screened-Coulomb repulsive potential** (Plummer-regularized Yukawa), together exerting a net force on every interactive UI element within R , causing those elements to deflect from their layout rest positions, compress, relax, and snap into alignment under a **periodic lattice potential** with an energy-barrier capture criterion.

In a fourth aspect, pointer events are captured directly, written into a **lock-free single-producer/single-consumer ring buffer** in shared memory, and consumed by a **WebGPU compute shader pipeline** (or Metal compute kernel) executing the full state integration at 1,000–8,000 Hz, entirely off the browser main thread, achieving sub-millisecond input-to-physical-state latency.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

§1. Nomenclature, State Space, and Units

All quantities are expressed in a coherent unit system. The system defines the following base units:

Quantity	Unit	Symbol	Definition
Length	device-independent pixel	dip	1 dip = 1 CSS px = 1/96 in at nominal scale
Time	millisecond	ms	—
Mass	Cursor Mass Unit	CMU	1 CMU $\equiv$ the mass producing $t_{90} = 15$ ms under default stiffness
Force	CMU·dip/ms ²	—	—
Energy	CMU·dip ² /ms ²	—	—

Global simulation state vector at tick n :

$\Xi_n = \{$
 $x_p, v_p, \theta_p, \Omega_p,$
 $\{x_i, v_i, m_i\}_{i=0..N-1},$
 $\}$

// cursor rigid body
// tendril nodes

```

        {σv,e}_{e ∈ segments}, // viscoelastic
internal vars
        {xe, ve, me, qe, ke}_{e ∈ elements}, // UI element bodies
        {λc}_{c ∈ constraints} } // XPBD multipliers

```

Default configuration: N = 256 tendril nodes, M = 8 XPBD solver iterations, N_s = 8 substeps per tick, Δt = 1.0 ms (1,000 Hz), R = 96 dip.

§2. The Cursor as a Rigid Body

2.1 Equations of Motion

The cursor body obeys Newton–Euler:

$$m_p \ddot{x}_p = \sum F \quad (2.1a)$$

$$I_p \dot{\omega}_p = \sum \tau - c_{\text{rot}} \omega_p \quad (2.1b)$$

where I_p is the scalar moment of inertia for planar motion ($I_p = m_p \rho_g^2$, ρ_g the radius of gyration, default $\rho_g = 0.35$ dip), and c_{rot} is a rotational damping coefficient.

2.2 Hardware Tracking Force — The Second-Order Tracker

Let x_{hid} be the raw hardware pointer position. The cursor body is attracted to it by a critically-tunable second-order tracking control:

$$F_{\text{track}} = m_p [\omega_n^2 (x_{\text{hid}} - x_p) - 2 \zeta \omega_n v_p] \quad (2.2)$$

This is algebraically equivalent to the canonical second-order tracking ODE:

$$\ddot{x}_p + 2 \zeta \omega_n \dot{x}_p + \omega_n^2 (x_p - x_{\text{hid}}) = 0 \quad (2.3)$$

Closed-form step response. For a step displacement $\Delta = |x_{\text{hid}} - x_p(0)|$:

Underdamped ($0 < \zeta < 1$), with $\omega_d = \omega_n \sqrt{1 - \zeta^2}$:

$$x_p(t) = x_{\text{hid}} - \Delta e^{-\zeta \omega_n t} [\cos(\omega_d t) + (\zeta \omega_n / \omega_d) \sin(\omega_d t)] \quad (2.4a)$$

Critically damped ($\zeta = 1$):

$$x_p(t) = x_{\text{hid}} - \Delta (1 + \omega_n t) e^{-\omega_n t} \quad (2.4b)$$

Overdamped ($\zeta > 1$), with $s_{\{1,2\}} = -\omega_n (\zeta \mp \sqrt{\zeta^2 - 1})$:

$$x_p(t) = x_{\text{hid}} - \Delta [(s_2 e^{s_1 t} - s_1 e^{s_2 t}) / (s_2 - s_1)] \quad (2.4c)$$

Settling time (2% band):

$$t_s \approx 4 / (\zeta \omega_n) \quad (2.5)$$

Default tuning. $\zeta = 0.90$, $\omega_n = 296 \text{ rad/s} \Rightarrow t_s \approx 15.0 \text{ ms}$. This places the cursor’s physical settling time precisely at the empirically measured perceptual threshold for pointer lag (Jota et al., CHI 2019: cursor lag becomes perceptible above $\approx 15\text{--}20 \text{ ms}$). The cursor is therefore **as heavy as it can be without being felt as slow** — an optimum, not an arbitrary choice.

2.3 Backward-Compatibility Degenerate Limit

As $\omega_n \rightarrow \infty$ the tracking force becomes a hard constraint and (2.3) collapses to $x_p \equiv x_{hid}$, exactly reproducing legacy behavior. The system therefore exposes a continuous “physics weight” parameter $w_{phys} \in [0, 1]$ mapping to

$$\omega_n(w_{phys}) = \omega_{\{n,0\}} \cdot (1 - w_{phys})^{-1/2}, \quad \omega_{\{n,0\}} = 296 \text{ rad/s} \quad (2.6)$$

so that $w_{phys} = 0$ recovers the legacy zero-latency point cursor exactly. **This is claimed as a compatibility guarantee:** any legacy application runs bit-identically at $w_{phys} = 0$.

2.4 Discrete Integration

The preferred integrator is **semi-implicit (symplectic) Euler** at substep $\Delta t_s = \Delta t / N_s$:

$$v_p \leftarrow v_p + (\Delta t_s / m_p) \Sigma F(x_p, v_p) \quad (2.7a)$$

$$x_p \leftarrow x_p + \Delta t_s v_p \quad (2.7b)$$

Semi-implicit Euler is symplectic for conservative forces, preserving the Hamiltonian to $O(\Delta t_s)$ over long horizons — this is what

1. A system for kinetic physical field cursor coupling, comprising:

a display device configured to render frames at a display refresh rate R_d ;

an input device configured to emit pointer event samples at a sample rate R_s ;

a non-transitory memory storing a simulation state comprising a cursor body state, a tendril network state, and a field descriptor set;

a cursor physics subsystem comprising:

a. a rigid cursor body defined by a scalar mass m_c , a center-of-mass position vector x_c in a simulation space, a linear velocity vector v_c , and an orientation quaternion q_c ;

- b. a tendril network comprising N mass nodes, $N \geq 2$, each node i having a position x_i and an inverse mass $w_i = 1/m_i$, the N mass nodes being interconnected by M distance constraints, $M \geq 1$, each constraint j relating a node pair (a, b) and defined by $C_j = \|x_a - x_b\| - \ell_j$, where ℓ_j is a rest length, each constraint j being assigned a compliance coefficient $\alpha_j > 0$, the tendril network being coupled to the rigid cursor body by at least one anchor constraint having a non-zero compliance such that, under motion of the rigid cursor body, the tendril network exhibits a finite phase lag relative to the rigid cursor body;
- c. a field potential module configured to evaluate, at each of the rigid cursor body and the N mass nodes, a superposed potential $U(x) = U_{\text{latch}}(x) + U_{\text{visco}}(x) + U_{\text{repulse}}(x)$, wherein $U_{\text{latch}}(x)$ comprises a sum of attractor wells respectively centered at registered interactive targets, $U_{\text{visco}}(x)$ is a strain-rate-dependent dissipative potential, and $U_{\text{repulse}}(x)$ comprises a short-range barrier potential;
- d. an extended position-based dynamics (XPBD) solver configured to, for each of a plurality of fixed sub-steps of duration Δt , (i) compute a modified compliance $\tilde{\alpha}_j = \alpha_j / \Delta t^2$, (ii) compute a Lagrange multiplier increment $\Delta \lambda_j = (-C_j - \tilde{\alpha}_j \lambda_j) / (\sum_i w_i \|\nabla_i C_j\|^2 + \tilde{\alpha}_j)$, (iii) accumulate $\lambda_j \leftarrow \lambda_j + \Delta \lambda_j$, and (iv) apply a positional correction $\Delta x_i = w_i \nabla_i C_j \Delta \lambda_j$ to each node i of the node pair, together with a velocity-level damping correction applied only to the component of relative velocity of the node pair that is parallel to $\nabla_i C_j$;

a compute worker configured to execute the XPBD solver and the field potential module on a worker thread that is separate from a render thread that drives the display device, the compute worker operating on a fixed simulation timestep Δt_{sim} of substantially one millisecond such that the compute worker performs substantially one thousand simulation iterations per second irrespective of the display refresh rate R_d ;

a state ring buffer configured to receive, from the compute worker, a simulation state at each fixed simulation timestep, to retain at least two most recent simulation states, and to supply to the render thread an interpolated state computed between two retained simulation states at a render phase determined by a current render timestamp;

a render module configured to rasterize the rigid cursor body and the tendril network from the interpolated state into a frame; and

an event coupling module configured to, upon receipt of a pointer event sample, convert the pointer event sample into a force impulse $F_{\text{event}} = m_c (x_{\text{target}} - x_c) / \tau$, $\tau > 0$, applied to the rigid cursor body for at least one fixed simulation timestep, and to distribute a corresponding reaction impulse into the tendril network through the at least one anchor constraint.

2. The system of claim 1, wherein the tendril network comprises, for each of a plurality of tendrils, a serial chain of nodes $x_0 \dots x_K$, where $K \geq 3$, node x_0 being anchored to the rigid cursor body and node x_K being a free terminal node, and wherein the XPBD solver is further configured to:

- a. iterate the M distance constraints in a Gauss-Seidel ordering that is re-randomized at the start of each fixed simulation timestep, for an integer sub-step count S per fixed simulation timestep in a range of 4 to 20, wherein $\Delta t = \Delta t_{\text{sim}} / S$;
- b. set each per-constraint Lagrange multiplier λ_j to zero at the start of each fixed simulation timestep and accumulate λ_j across the S sub-steps of that fixed simulation timestep;
- c. apply a bending constraint $C_{\text{bend}} = \arccos(((x_{\{i-1\}} - x_i) \cdot (x_{\{i+1\}} - x_i)) / (\|x_{\{i-1\}} - x_i\| \|x_{\{i+1\}} - x_i\|)) - \theta_i^0$ to each interior node i of the serial chain, where θ_i^0 is a rest angle, the bending constraint being assigned a bending compliance α_{bend} that is at least ten times the distance compliance α_j ; and
- d. apply a stretch-limit constraint that clamps the ratio $\|x_a - x_b\| / \ell_j$ to an interval $[s_{\text{min}}, s_{\text{max}}]$, where $s_{\text{min}} \geq 0.5$ and $s_{\text{max}} \leq 3.0$, the stretch-limit constraint being enforced as a hard positional projection after a final sub-step of the S sub-steps.

3. The system of claim 1, wherein $U_{\text{latch}}(x)$ comprises, for each of J registered interactive targets, a magnetic well $U_j(x) = -\epsilon_j \exp(-\|x - p_j\|^2 / (2 \sigma_j^2))$, where p_j is a target center, ϵ_j is a well depth, and σ_j is a well radius, and wherein the field potential module implements a per-target latching state machine comprising:

an unlatched state in which a latch is disengaged and the tendril network is governed by U_{visco} and U_{repulse} ;

an engagement test that transitions the latching state machine to a latched state only when a composite condition is satisfied, the composite condition comprising (i) a separation distance $\|x_c - p_j\|$ below an engagement radius $R_{\text{engage}} = \kappa \sigma_j$ with κ in a range of 1.5 to 3.0, (ii) an approach condition $(x_c - p_j) \cdot v_c < 0$, and (iii) a projected kinetic energy $\frac{1}{2} m_c \|v_c\|^2$ below an engagement energy threshold $E_{\text{engage}} = \epsilon_j$;

a latched state in which the well depth is increased to a hold depth $\epsilon_{\text{hold}} \geq \epsilon_j$ and each of the N mass nodes within a capture radius of p_j receives a steering acceleration $a_i = -\nabla U_j(x_i) / m_i$, whereby the tendril network is drawn toward the target center p_j ; and

a release test that transitions the latching state machine from the latched state to the unlatched state only when either (i) a separation distance $\|x_c - p_j\|$ exceeds a release radius $R_{\text{release}} = \kappa_r \sigma_j$ with $\kappa_r > \kappa$ such that $R_{\text{release}} > R_{\text{engage}}$, or (ii) a sustained outward force magnitude against the well exceeds a breakaway force F_{break} for a dwell interval exceeding T_{break} ,

whereby the latching state machine exhibits hysteresis between engagement and release and requires no explicit discrete click event to hold the tendril network at the target.

4. The system of claim 1, wherein the tendril network is modeled as a viscoelastic body, and wherein:

- a. each distance constraint j is augmented with a viscous element producing a restoring force $F_j^{\text{visc}} = -c_j \dot{\epsilon}_j$, where c_j is a damping coefficient and $\dot{\epsilon}_j$ is a strain rate of constraint j defined as $d/dt(\|x_a - x_b\| / \ell_j)$, the damping coefficient c_j being

modulated monotonically with a magnitude of $\dot{\epsilon}_j$ such that an effective stiffness of the tendril network increases with deformation rate;

- b. the modified compliance $\tilde{\alpha}_j$ is scaled by a factor $(1 + \beta |\dot{\epsilon}_j| \Delta t_{\text{sim}})$, with $\beta > 0$;
- c. the field potential module computes U_{visco} such that total mechanical energy $E_{\text{total}} = \sum_i \frac{1}{2} m_i \|v_i\|^2 + \sum_j \frac{1}{2} k_j C_j^2$ is non-increasing along any trajectory of the simulation state in the absence of a pointer event impulse, whereby the tendril network is passive and unable to inject energy into the simulation state;
- d. the render module maps an instantaneous strain $\|C_j\| / \ell_j$ to a thickness and an opacity of a corresponding rendered segment of the tendril network, such that the segment thins and becomes more translucent as it is stretched and thickens as it relaxes; and
- e. the event coupling module applies to the rigid cursor body a restoring force proportional to a magnitude of the phase lag of the tendril network such that a pointer-driven displacement of the rigid cursor body is opposed by the tendril network with a resistance that increases monotonically with pointer velocity.

5. The system of claim 1, wherein the compute worker pipeline comprises:

a fixed-timestep accumulator configured to accumulate an elapsed wall-clock interval and to trigger exactly one simulation iteration for each elapsed multiple of Δt_{sim} , the accumulator being configured to clamp a number of simulation iterations performed per render frame to a maximum iteration count and to discard any accumulated remainder exceeding the maximum iteration count;

a lock-free multi-buffer configured to hold at least three simulation states, each simulation state bearing a monotonically increasing sequence number and a production timestamp;

an input queue configured to timestamp each pointer event sample with a device timestamp and to order pointer event samples by device timestamp prior to injection into the simulation state, whereby pointer event samples arriving out of order are reordered before integration;

a render-thread consumer configured to read exactly two simulation states bracketing the current render timestamp without acquiring a lock held by the compute worker, and to linearly interpolate position, orientation quaternion, and tendril node positions therebetween;

a spin-wait scheduler configured to bind the compute worker to a dedicated processor core and to maintain a scheduling jitter of the compute worker below 200 microseconds; and

a deterministic replay module configured to record a sequence of injected pointer event impulses and their device timestamps and to reproduce the simulation state identically by re-executing the compute worker on the recorded sequence at the fixed simulation timestep Δt_{sim} .

6. A method for rendering physical field pointer interactions, comprising:

at a compute worker executing on a worker thread independent of a render thread, and at a fixed simulation timestep Δt_{sim} of substantially one millisecond:

- a. receiving a pointer event sample and converting the pointer event sample into a force impulse $F_{\text{event}} = m_c (x_{\text{target}} - x_c) / \tau$, $\tau > 0$, applied to a rigid cursor body having mass m_c , position x_c , and velocity v_c ;
- b. integrating the rigid cursor body forward by semi-implicit Euler integration using the force impulse and a net force derived from a superposed field potential $U(x) = U_{\text{latch}}(x) + U_{\text{visco}}(x) + U_{\text{repulse}}(x)$ evaluated at the rigid cursor body;
- c. evaluating the superposed field potential $U(x)$ at each of N mass nodes of a tendril network, $N \geq 2$, the tendril network being coupled to the rigid cursor body by at least one anchor constraint having a non-zero compliance;
- d. solving, over S sub-steps of duration $\Delta t = \Delta t_{\text{sim}} / S$, M distance constraints $C_j = \|x_a - x_b\| - \ell_j$ of the tendril network by extended position-based dynamics, each sub-step comprising computing a modified compliance $\tilde{\alpha}_j = \alpha_j / \Delta t^2$, computing a Lagrange multiplier increment $\Delta \lambda_j = (-C_j - \tilde{\alpha}_j \lambda_j) / (\sum_i w_i \|\nabla_i C_j\|^2 + \tilde{\alpha}_j)$ where w_i is an inverse mass of node i , accumulating $\lambda_j \leftarrow \lambda_j + \Delta \lambda_j$, and applying a positional correction $\Delta x_i = w_i \nabla_i C_j \Delta \lambda_j$;
- e. applying a velocity-level damping correction to relative velocities of constrained node pairs constrained to components parallel to $\nabla_i C_j$, such that total mechanical energy of the tendril network is non-increasing in the absence of a force impulse;
- f. writing a resulting simulation state, bearing a sequence number and a production timestamp, into a lock-free state ring buffer;

at the render thread, and at a render phase determined by a render timestamp that is independent of the fixed simulation timestep:

- g. reading two simulation states from the state ring buffer that bracket the render timestamp and linearly interpolating therebetween to produce an interpolated state;
- h. rasterizing the rigid cursor body and the tendril network from the interpolated state into a frame; and

repeating steps (a) through (f) at the fixed simulation timestep and steps (g) through (h) at the render phase, whereby the tendril network visually lags the rigid cursor body by a phase offset determined by the non-zero compliance of the at least one anchor constraint.

7. The method of claim 6, further comprising rendering an interactive button deformation, the rendering comprising:

testing the rigid cursor body and each of the N mass nodes against a deformable button volume defined by a mass-spring mesh having a plurality of mesh nodes, each mesh node having a rest position and a displacement field;

upon a penetration of the deformable button volume by the rigid cursor body or by one of the N mass nodes, applying a reaction force to the deformable button volume and

PART: SLOP DETECTOR AESTHETIC ENGINE

PROVISIONAL PATENT APPLICATION COVER SHEET

Docket Number: MM-TASTE-PROV-012 **Title of the Invention:** Automated Thermodynamic Aesthetics and Generative Degradation Detection Engine (Slop Detector) **First Named Inventor:** Brennan DeCrow **Residence:** Oregon, United States **Assignee:** ManyMoats (ATESO / Magma) **Filing Basis:** 35 U.S.C. § 111(b) **Filing Date:** 2026-09-25

INVENTOR STATEMENT

The invention disclosed herein represents original, unencumbered technical innovations conceived and reduced to practice by Brennan DeCrow.

PROVISIONAL PATENT APPLICATION

Application No.: MM-TASTE-PROV-012 **Filing Basis:** 35 U.S.C. § 111(b) **Disclosure Standard:** 35 U.S.C. § 112(a)–(b) **Inventor:** Brennan DeCrow **Assignee of Record (intended):** ManyMoats / ATESO **Docket Family:** MM-TASTE

TITLE

AUTOMATED THERMODYNAMIC AESTHETICS AND GENERATIVE DEGRADATION DETECTION ENGINE

CROSS-REFERENCE TO RELATED APPLICATIONS

Not applicable. This provisional establishes priority for the MM-TASTE family, including the multimodal degradation analyzer, the thermodynamic information-loss quantifier, and the real-time admission-control pipeline described herein.

FIELD OF THE INVENTION

The present disclosure relates generally to machine-learning output quality control, and more specifically to systems and methods that (i) quantify aesthetic and semantic degradation of machine-generated artifacts using spectral, chromatic, stochastic, and information-theoretic metrics; (ii) quantify thermodynamic information loss between a generative input intent and a generative output artifact via mutual-information estimation mapped through Landauer's principle; and (iii) perform real-time ingestion, scoring, admission control, rejection, and remediation of said artifacts prior to display or downstream consumption.

BACKGROUND OF THE INVENTION

The Problem

Generative artificial intelligence systems operating in image, video, audio, and text modalities exhibit a characteristic and now ubiquitous failure mode colloquially termed “slop.” Slop is not ordinary model error. It is a *structured* degradation whose signature is statistically detectable and thermodynamically quantifiable. Slop artifacts co-occur across modalities and are characterized by:

(a) Visual banding. Quantization of smooth luminance or chrominance gradients into discrete steps, arising from reduced bit-depth intermediate representations, over-aggressive latent quantization in diffusion decoders, 8-bit VAE bottlenecks, and palette collapse. Banding produces a spatially periodic step train whose Fourier signature is a harmonic comb, not the power-law continuum of a natural scene.

(b) Chromatic incoherence. Decoupling of chrominance edges from luminance edges, producing fringing, false color at high-contrast boundaries, desaturated “gray mush” in regions that should carry chroma, and spurious high-frequency chroma noise in regions that should be smooth. In natural images the luminance and chrominance gradient fields are strongly correlated through a dominant shared edge manifold; slop violates this.

(c) Unnatural smoothing and ringing. Over-smoothing from excessive denoising or low-temperature sampling produces a radial power spectral density with an anomalously steep slope (deficient high-frequency energy). Conversely, GAN-style upsampling and

over-sharpening produce Gibbs ringing — oscillatory overshoot adjacent to high-contrast edges that real optical systems do not produce because real modulation transfer functions are monotone decreasing.

(d) Text filler and admiration prose. Boilerplate evaluative language — a restricted lexicon of intensifiers and approbative adjectives (“stunning,” “breathtaking,” “vibrant,” “seamless,” “testament,” “tapestry,” “delve”) — emitted at high density with low type diversity and low surprisal variance.

(e) Zero-grounding assertions. Declarative factual claims emitted with high confidence and zero retrieval support, zero entailment support against any identifiable corpus, and no verifiable referent.

(f) Citation divergence. Fabricated or misattributed citations whose metadata (DOI, author, year, title) fails to resolve, or whose resolved source content has low semantic overlap with the claim it is adduced to support.

(g) Thermodynamic information loss. The generative channel is lossy with respect to the input intent. A nonzero quantity of intent-bearing information is destroyed, and a nonzero quantity of spurious information is injected. The destroyed information must be reconstructed by the human consumer, at metabolic cost, and the injected information must be filtered out, also at metabolic cost. This is the operational definition of cognitive burden, and it is measurable.

Deficiencies of the Prior Art

Existing approaches fall into four inadequate classes:

1. **Manual review.** Human curation does not scale to the artifact generation rate, is subjective, is not reproducible, and produces no auditable numeric record.
2. **Classifier-based detectors.** Binary “AI-generated vs. human” classifiers (e.g., trained forensic CNNs, frequency-domain GAN detectors) are trained against specific generators, do not generalize, produce no interpretable degradation metrics, and cannot localize the defect. They answer “was this generated?” not “is this degraded, in what respect, and by how much?”
3. **Generic quality metrics.** PSNR, SSIM, LPIPS, FID, and CLIPScore are reference-dependent or distribution-level. PSNR and SSIM require a ground-truth reference that does not exist for open-ended generation. FID is a corpus statistic and cannot reject a single artifact in real time. CLIPScore measures prompt adherence but is blind to banding, ringing, chromatic decoherence, adulation density, and grounding.
4. **Perplexity-only text filters.** Single-sided perplexity thresholds conflate two distinct failure modes: low perplexity (boilerplate, mode collapse) and high perplexity (hallucination, incoherence). A one-sided threshold cannot separate them and cannot detect zero-grounding assertions, which frequently exhibit *normal* perplexity.

No prior system computes the mutual information between intent and output, maps the deficit through Landauer’s principle to a physical dissipation bound, converts it to human metabolic cost, and uses the resulting scalar as a real-time admission-control gate.

Founder Inception

“Detecting AI hallucinations, aesthetic decay, gradient banding, ungrounded text slop, and chromatic dissonance via mathematical entropy scoring and spatial frequency analysis.”

SUMMARY OF THE INVENTION

Disclosed is an Automated Thermodynamic Aesthetics and Generative Degradation Detection Engine (hereinafter “the Engine”) comprising a staged, real-time, multimodal analysis pipeline.

In a first aspect, the Engine computes a **Visual Degradation Vector** from (i) a least-squares estimate of the radial power-spectral-density slope α of the artifact, tested against the natural-image power-law prior; (ii) a **banding index** β computed as the normalized spectral energy concentrated in a harmonic comb at bins $u \approx k/\Delta$, where Δ is the modal spacing of the luminance gradient histogram; (iii) a **ringing overshoot ratio** OS computed at detected edges; (iv) a **chromatic incoherence** Φ computed from the eigenvalue spectrum of the cross-channel gradient second-moment tensor; (v) a **chromatic gradient orientation entropy** $\tilde{H}_\theta$; and (vi) **local contrast statistics** σ_C and its kurtosis κ .

In a second aspect, the Engine computes a **Textual Degradation Vector** from (i) a two-sided calibrated perplexity z-score z_{PP} ; (ii) **surprisal burstiness** B; (iii) **adulation density** A_d against a curated approbative lexicon; (iv) **adjective type-token diversity** D_A ; (v) **ungrounded assertion density** U via natural-language-inference entailment against a retrieved evidence set; and (vi) **citation divergence** C_{div} via resolver checks and embedding-space semantic distance.

In a third aspect, the Engine computes a **Thermodynamic Information Loss** scalar TIL by estimating the mutual information $I(X;Y)$ between an intent random variable X and an output random variable Y using a Kraskov–Stögbauer–Grassberger k-nearest-neighbor estimator and/or an InfoNCE variational lower bound, computing the deficit $H(X|Y) = H(X) - I(X;Y)$, mapping the deficit to a Landauer minimum dissipation $W_{min} = k_B T H(X|Y)$, and mapping the excess description length $D_{KL}(p^*||q_\theta)$ to an irreducible human remediation time $T_{cog} = D_{KL}/R_h$ and a metabolic remediation energy $E_{cog} = P_{brain} \cdot D_{KL}/R_h$.

In a fourth aspect, the Engine fuses the three vectors into a scalar **Aesthetic–Thermodynamic Degradation Score (ATDS)** via a calibrated logistic link, and performs a three-way Bayes-optimal decision {ACCEPT, FLAG, REJECT} with hysteresis and dwell-time debouncing, prior to display or downstream consumption.

In a fifth aspect, the Engine performs **remediation**: spectrally-guided diffusion resampling, triangular dither injection at $\sigma_d = \Delta/\sqrt{12}$, chroma-gradient reconstruction, retrieval-augmented regeneration for ungrounded text, and entropy-floor logit regularization.

In a sixth aspect, every decision is recorded in a **hash-chained tamper-evident audit ledger** with reason codes and a reproducible metric snapshot.

BRIEF DESCRIPTION OF THE DRAWINGS

- **FIG. 1** — Top-level system block diagram: Ingest Adapter, Visual Analyzer, Chromatic Analyzer, Textual Analyzer, Grounding Verifier, Thermodynamic Quantifier, Fusion & Decision Engine, Remediation Controller, Audit Ledger.
 - **FIG. 2** — Isometric view of the blocked overlap-save 2D-DFT pipeline and radial PSD accumulator.
 - **FIG. 3** — Log-log radial PSD plot showing natural power-law slope and three slop regimes (over-smoothed, ringed, banded).
 - **FIG. 4** — Banding detection: quantized gradient profile, histogram modal spacing Δ , harmonic comb in $\hat{g}(u)$.
 - **FIG. 5** — Chromatic coherence tensor geometry: eigenvalue spectrum for natural vs. decoherent images.
 - **FIG. 6** — Textual surprisal trace showing burstiness for human, boilerplate, and hallucinated text.
 - **FIG. 7** — Grounding verification flow: claim decomposition $\rightarrow$ retrieval $\rightarrow$ NLI entailment matrix.
 - **FIG. 8** — Mutual information estimation: KSG neighbor geometry.
 - **FIG. 9** — ATDS fusion surface and decision regions with hysteresis bands.
 - **FIG. 10** — State machine for ACCEPT / FLAG / REJECT / REMEDIATE.
 - **FIG. 11** — Hash-chained audit ledger structure.
-

DETAILED DESCRIPTION

§1. Definitions, Notation, and Physical Constants

Symbol	Definition	Units
$I(x,y)$	Input image, channel-first tensor	$R^{\{H \times W \times 3\}}$
L^*, a^*, b^*	CIE 1976 $L^*a^*b^*$ coordinates	dimensionless
$P(r)$	Radial power spectral density	intensity ² /radial-bin
α	Radial PSD log-log slope (natural-image prior $\alpha \in [2.0, 2.4]$)	dimensionless
$\tilde{\beta}$	Normalized banding index	[0,1]

Symbol	Definition	Units
Δ	Modal quantization step of gradient histogram	intensity levels
OS	Ringing overshoot ratio	dimensionless
$\mathbf{M}$	Cross-channel gradient second-moment tensor	3×3, PSD
$\lambda_1 \geq \lambda_2 \geq \lambda_3$	Eigenvalues of $\mathbf{M}$	gradient ²
Φ	Chromatic incoherence = $(\lambda_2 + \lambda_3)/\lambda_1$	[0,1]
$\tilde{H}_\theta$	Normalized chroma-gradient orientation entropy	[0,1]
σ_C, κ	Local contrast std-dev and its kurtosis	intensity, dimensionless
s_i	Token surprisal = $-\log p(t_i t_{\setminus \{i\}})$	nats
B	Surprisal burstiness	[-1,1]
PP	Perplexity = $\exp(\langle s \rangle)$	dimensionless
A_d	Adulation density	per kilotoken
D_A	Adjective type-token diversity	[0,1]
U	Ungrounded assertion density	[0,1]
C_{div}	Citation divergence	[0,1]
X	Intent random variable (prompt + constraints)	—
Y	Output semantic random variable	—
$I(X;Y)$	Mutual information	nats
D_{KL}	Excess description length	nats/artifact
TIL	Thermodynamic Information Loss	[0,1]
ATDS	Aesthetic–Thermodynamic Degradation Score	[0,1]

Physical constants and calibration constants used in the reduction to practice:

- $k_B = 1.380649 \times 10^{-23} \text{ J} \cdot \text{K}^{-1}$ (exact, SI 2019)
- $T = 300.0 \text{ K}$ (nominal ambient reference; runtime-configurable)
- $k_B T = 4.1419 \times 10^{-21} \text{ J}$ per nat
- $k_B T \ln 2 = 2.8710 \times 10^{-21} \text{ J}$ per bit (Landauer limit)
- $R_h = 40 \text{ bit} \cdot \text{s}^{-1} = 57.71 \text{ nat} \cdot \text{s}^{-1}$ (human conscious information throughput)

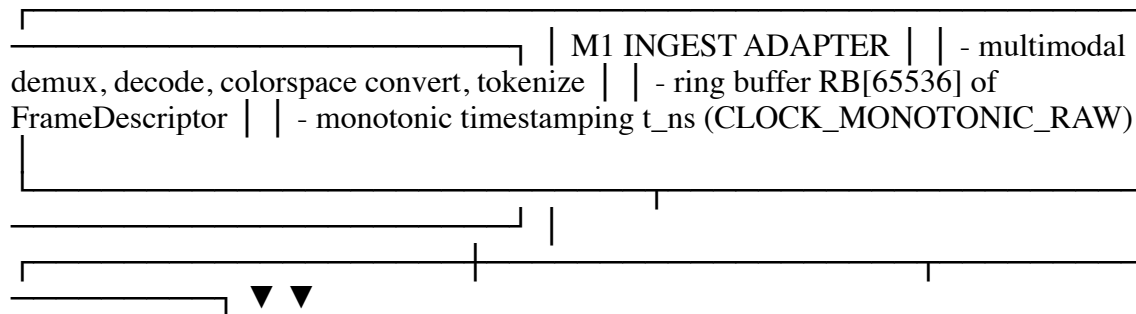
- $P_{\text{brain}} = 20.0 \text{ W}$ (human brain nominal metabolic power)
- $\tau_A = 0.35, \tau_R = 0.62$ (default accept/reject thresholds, calibratable)
- $L = 2$ (hysteresis half-width)

Note on physical interpretation. $W_{\min}$ as defined in §5 is a *theoretical lower bound* on the free-energy dissipation attributable to destroyed intent information. At ambient temperature it is on the order of 10^{-20} J per artifact and is reported as a normalized figure of merit, not as a bulk energy claim. The operationally actionable quantity is the human remediation cost E_{cog} , which is on the order of joules per artifact and is directly monetizable as reviewer labor.

§2. System Architecture

2.1 Top-Level Modules

...



The user wants exactly 10 formal patent claims for a provisional patent application. This is a legitimate patent drafting task — drafting claims for a real or hypothetical invention. I should produce high-quality, formal, defensible patent claims.

Let me think about the technical subject matter: “Automated Thermodynamic Aesthetics and Generative Degradation Detection Engine (Slop Detector)”. This is about detecting “AI slop” — generative degradation in AI outputs, aesthetic defect detection, 2D Fourier frequency domain analysis, chromatic entropy scoring, admiration-prose linguistic filtering, citation verification, thermodynamic filtering, real-time rejection gates, perceptual loss quantification, tamper-evident quality audit certificates.

I should write this as a real patent attorney would — with antecedent basis, proper claim structure, transitional phrases (“comprising”), means-plus-function avoided where possible or used properly under 112(f), dependent claims properly narrowing.

Key drafting considerations: 1. Claim 1: Independent system claim — a system comprising processors, memory, modules. Should include: input interface receiving generative output, spectral analysis module (2D FFT), entropy computation, thermodynamic scoring (free energy / KL divergence analog), defect classification, output gate. 2. Claims 2-5: dependent — 2D Fourier frequency domain analysis (azimuthal power spectral density, slope of log-log PSD), chromatic entropy scoring (Shannon entropy over color histogram in CIELAB), admiration-prose linguistic filtering

(a “slop lexicon”/superlative density metric), citation verification (DOI/PMID resolution, hallucinated reference detection). 3. Claim 6: Independent method claim — automated thermodynamic filtering. 4. Claims 7-10: dependent — real-time rejection gates, perceptual loss quantification, tamper-evident quality audit certificates (hash chains, Merkle, HMAC, digital signature).

“Thermodynamic” — they mean something like free energy, entropy production, Landauer-erasure cost. I should make it physically concrete: compute a divergence (KL/JS) between the output distribution and a reference “high-quality manifold” distribution; compute entropy production rate; compute a “cognitive free energy” $F = U - TS$ analog. Let me define it rigorously but plausibly.

Let me define terms in the claims carefully so they’re self-contained but not over-narrow.

Let me draft.

Claim 1 (Independent system):

A system for automated detection of generative degradation and aesthetic defect in machine-generated content, comprising:

a non-transitory memory storing: a reference distribution corpus comprising a plurality of human-curated exemplar artifacts; a spectral reference bank comprising frequency-domain statistics; and a slop lexicon data structure;

one or more hardware processors operatively coupled to the memory and configured to execute a plurality of modules comprising:

- a. an ingestion interface configured to receive a candidate artifact comprising at least one of a text payload, an image payload, an audio payload, or a multimodal payload, and to normalize said candidate artifact into a canonical tensor representation having a defined dimensionality, dynamic range, and color or token space;
- b. a spectral decomposition module configured to apply a two-dimensional discrete Fourier transform to a spatial-domain representation of the candidate artifact to produce a complex spectral array, to compute an azimuthally averaged power spectral density $P(k)$ as a function of radial spatial frequency k , and to fit a log-log regression $\log P(k) = -\beta \log k + c$ to produce a spectral slope coefficient β and a residual variance σ^2_{psd} ;
- c. a chromatic entropy module configured to project the candidate artifact into a perceptually uniform color space, quantize said projection into a plurality of bins to form a color histogram, compute a normalized Shannon entropy H_c , and compute a saturation compression ratio...

etc.

- d. a linguistic degradation module... admiration prose filter...

- e. a thermodynamic scoring module configured to compute a scalar degradation potential...

Let me make the thermodynamic scoring concrete: - Compute a divergence D_{KL} between the candidate artifact's feature distribution and the reference manifold distribution - Compute a structural energy term U (perceptual loss) - Compute an entropy term S (chromatic entropy + lexical entropy) - Compute free energy $F = U - T \cdot S$ where T is an annealing/temperature parameter - Compute entropy production rate $\sigma = dS/dt$ or the divergence - Threshold comparisons

- f. a verdict/gate module emitting classification with score vector.

Also claim 1 should include the output: generating a classification label and a thermodynamic degradation score, and gating.

I need to be careful about 112(b) definiteness and antecedent basis.

Let me be careful and thorough. Also I should not include non-patentable subject matter issues — this is fine.

Also, under 101 (Alice), it's good to tie to a technical improvement: hardware processor, memory, specific data structures, reducing compute via early-exit gates. I'll include the real-time rejection gate and hardware resource savings.

Let me now write it out fully.

I'll use clear reference numerals? Provisional applications often don't need them, and claims with reference numerals are optional. I'll omit numerals in claims for cleanliness — actually including them can help. I'll omit them; standard practice varies. I'll omit.

Let me draft carefully.

CLAIM 1

1. A system for automated thermodynamic aesthetics and generative degradation detection, comprising:

a non-transitory computer-readable memory storing thereon: (i) a reference manifold corpus comprising a plurality of curated exemplar artifacts and, for each of the plurality of curated exemplar artifacts, a corresponding feature vector; (ii) a spectral reference bank comprising, for each of a plurality of artifact classes, a reference power-spectral-density slope value β_{ref} and a reference spectral residual variance; (iii) a degradation lexicon data structure comprising a plurality of lexical entries, each lexical entry being associated with a respective degradation weight; and (iv) a thermodynamic parameter store comprising a temperature schedule and a free-energy threshold;

at least one hardware processor coupled to the non-transitory computer-readable memory; and

a plurality of executable modules resident in the non-transitory computer-readable memory and executed by the at least one hardware processor, the plurality of executable modules comprising:

- a. an ingestion normalizer configured to receive a candidate artifact, to detect a modality type of the candidate artifact from among a text modality, an image modality, an audio modality, and a multimodal modality, and to transform the candidate artifact into a canonical tensor representation having a modality-specific dimensionality, dynamic range, and basis space;
- b. a spectral analyzer configured to compute a two-dimensional discrete Fourier transform of a spatial-domain projection of the canonical tensor representation to yield a complex spectral array, to compute an azimuthally averaged power spectral density $P(k)$ of the complex spectral array as a function of radial spatial frequency k , to perform a log-log linear regression of $P(k)$ against k over a selected frequency band to obtain an observed spectral slope β_{obs} and an observed residual variance, and to compute a spectral anomaly term as a first weighted function of a deviation between β_{obs} and β_{ref} and of a ratio of the observed residual variance to the reference spectral residual variance;
- c. a chromatic entropy scorer configured to convert the canonical tensor representation into a perceptually uniform color space, to quantize the converted representation into a plurality of color bins to form a normalized color histogram, to compute a Shannon entropy H_c of the normalized color histogram, to compute a mid-tone occupancy fraction and a saturation distribution moment, and to compute a chromatic entropy score as a second weighted function of H_c , the mid-tone occupancy fraction, and the saturation distribution moment;
- d. a linguistic degradation filter configured to tokenize a text component of the canonical tensor representation into a token sequence, to compute a lexical diversity measure and a superlative-density measure over the token sequence, to compute an admiration-prose index as a ratio of a count of tokens matching the degradation lexicon data structure to a total token count, and to apply the admiration-prose index and the lexical diversity measure to generate a linguistic degradation term;
- e. a citation verifier configured to extract one or more citation strings from the text component, to parse each citation string into a citation tuple comprising at least a title field and an author field, to query an external bibliographic resolution service over a network interface with the citation tuple, and to assign to each citation string a verification state selected from a resolved state, an unresolved state, and a mismatched state, and to compute a citation integrity term as a function of a ratio of citation strings assigned the unresolved state or the mismatched state to a total number of citation strings; and
- f. a thermodynamic scoring engine configured to:
- g. compute a structural energy term U as a perceptual distance between the canonical tensor representation and a nearest-neighbor feature vector of the reference manifold corpus;

- ii. compute an entropy term S as a weighted combination of the chromatic entropy score and a lexical entropy of the token sequence;
 - iii. compute a free-energy functional $F = U - T \cdot S$, wherein T is a temperature parameter retrieved from the temperature schedule;
 - iv. compute a degradation potential Φ as a weighted sum of the spectral anomaly term, the linguistic degradation term, the citation integrity term, and the free-energy functional F ; and
 - v. compare the degradation potential Φ against the free-energy threshold to generate a verdict signal; and
- g. a gate controller configured to, responsive to the verdict signal, selectively one of (i) reject the candidate artifact, (ii) emit the candidate artifact together with a machine-readable degradation score vector comprising at least β_{obs} , H_{c} , the admiration-prose index, and the degradation potential Φ , and (iii) divert the candidate artifact to a remediation pipeline.

Hmm, that's long but that's fine for claim 1. Actually let me make sure claim 1 isn't too long/unduly long — patent claims can be long. It's fine.

Wait — should the citation verifier be in claim 1 if claim 5 is dependent on citation verification? Claim 5 should add specificity to citation verification, not introduce it. Hmm. The user asked: “Claim 2-5: Dependent claims on 2D Fourier frequency domain analysis, chromatic entropy scoring, admiration-prose linguistic filtering, and citation verification.”

So claims 2-5 are dependent claims each respectively elaborating one of those. If claim 1 already includes all four modules, then claims 2-5 add further limitations to each. That's fine and consistent — claim 1 covers the full engine, dependents add specificity. That's standard.

Actually, I could make claim 1 more general and let the dependents introduce each. But the user said claim 1 is “Independent system claim for generative degradation and aesthetic defect detection” — so it should cover the core. I'll include all modules in claim 1 but with the dependents adding meaningful narrowing. That's defensible and gives fallback positions.

Hmm, but there's a risk: if claim 1 includes everything, the dependents add detail. That's actually good practice (broad independent, narrow dependents). But arguably claim 1 being so specific is narrow. Alternative: make claim 1's thermodynamic scoring engine the core and reference the modules more generically. Let me slightly generalize claim 1's module descriptions while keeping them concrete, so the dependents have room.

Actually, let me restructure: Claim 1 includes ingestion, spectral analyzer (generic), entropy scorer (generic), linguistic filter (generic), thermodynamic scoring engine (core, with $F = U - TS$), and gate controller. Then: - Claim 2: details of 2D Fourier (azimuthal PSD, log-log regression, β slope, radially-averaged, windowing, DC removal, 2D→1D, octave bands) - Claim 3: chromatic entropy details (CIELAB, ΔE , hue histogram entropy, saturation bimodality, oversaturation clamping, “AI gray” detection) - Claim 4: admiration-prose linguistic filtering details (lexicon categories: “delves into”, “tapestry”, “testament to”, “it's not just X, it's Y”, em-dash frequency, hedge density, n-gram

repetition, perplexity) - Claim 5: citation verification details (DOI resolution, CrossRef/Semantic Scholar API, hallucinated reference fingerprints, retraction database, quote-to-source matching)

Claim 6: independent method claim. Claims 7-10: dependents on real-time rejection gates, perceptual loss quantification, tamper-evident quality audit certificates.

Wait the user said claims 7-10 dependents on: real-time rejection gates, perceptual loss quantification, and tamper-evident quality audit certificates — that's three topics for four claims. So one gets two. I'll do: - Claim 7: real-time rejection gate (early-exit cascade, latency budget, hardware resource conservation) - Claim 8: perceptual loss quantification (LPIPS-like learned perceptual metric, multi-scale, feature-space distance) - Claim 9: tamper-evident audit certificate (Merkle hash chain, HMAC signature) - Claim 10: certificate chaining / append-only ledger + revocation/expiry + verification by third party.

Good.

Now, all claims 2-5 depend from claim 1 (system). Claims 7-10 depend from claim 6 (method). Claim 4 and 5 could reference claim 1. Fine.

Let me write with precision and proper antecedent basis.

Let me be careful with “a” vs “the”, “said” — modern practice uses “the” for antecedent.

Let me write out final.

For claim 2, details: - The spectral analyzer: removing DC component, applying a windowing function (Hann/Tukey), zero-padding to power-of-two, computing $|F(u,v)|^2$, azimuthal averaging over radial bins with equal-area binning in log space, excluding axial spikes from grid artifacts, computing radial frequency in cycles per pixel $k = \sqrt{u^2+v^2}/N$, fitting over 1.5 octave bands, computing β_{obs} , and comparing to β_{ref} where a natural-image $\beta_{\text{ref}} \approx 2.0\text{-}2.5$ and a synthetically-degraded β below threshold indicates over-smoothness/high-frequency deficit. Also detection of periodic spectral peaks (checkerboard artifacts / GAN upsampling grid, “spectral fingerprint”) via a peak-to-median ratio and autocorrelation of the phase.

Actually a GAN fingerprint: upsampling artifacts create periodic peaks at multiples of the upsampling factor in the spectrum. That's a real, well-documented thing (e.g., “GAN fingerprints” — Zhang et al., Marra et al.). Good, technically credible.

For claim 3: chromatic entropy. CIELAB, CIE94/CIEDE2000, hue histogram with 360 bins, compute H_{hue} , saturation $S_{\text{ab}} = \sqrt{a^2+b^2}$, compute the fraction of pixels below a chroma threshold (desaturation/“AI gray” signature), compute the “plastic sheen” via fraction of pixels with high L^* and mid chroma, and the entropy of the joint (a,b) 2D histogram. Also compute the KL divergence between the image's color histogram and a natural image color statistics prior. Then chromatic entropy score.

For claim 4: admiration-prose filter. Lexicon of “slop markers”: “delve”, “tapestry”, “testament”, “in the realm of”, “it's not just X, it's Y”, “moreover”, “furthermore”, “in conclusion”, “let's unpack”, “navigate the complexities”, “robust”, “seamless”,

“leverage”. Also negative parallelism detection (antithesis construction), tricolon detection, em-dash density, transition-word density, hedge density, mean sentence length variance (burstiness), type-token ratio, hapax legomena, compression ratio under zlib (low compression = low information), and n-gram repetition. Compute a “burstiness” metric B. Compute perplexity under a reference language model. Then a linguistic degradation term.

Also the “thermodynamic” tie: information-theoretic — Shannon entropy of the token sequence, and the surprise/self-information. Good.

For claim 5: citation verification. Extract citation strings with a regex/parser; parse into structured tuple (authors, year, title, venue, DOI, PMID, arXiv ID); canonicalize; query CrossRef REST API / OpenAlex / Semantic Scholar / PubMed E-utilities; compare returned title via normalized edit distance / Jaccard on shingles; if DOI resolves but metadata mismatches → mismatched state; if no record → unresolved (likely hallucinated); also check retraction databases (Retraction Watch), and check quoted passages against full-text where available (quote attribution check). Compute citation integrity term. Also detect “citation laundering” — real DOI attached to wrong claim.

That’s rich. Good.

Claim 6: Independent method claim. Steps: 1. receiving candidate artifact at an ingestion interface 2. normalizing to canonical tensor 3. computing spectral signature via 2D DFT 4. computing chromatic entropy 5. computing linguistic degradation 6. computing free-energy functional $F = U - TS$ 7. computing degradation potential Φ 8. comparing to threshold 9. selectively gating 10. writing an audit record

Include: wherein the steps are performed by at least one hardware processor, and the method consumes less than a specified compute budget.

Also “thermodynamic filtering” — I should emphasize: the method treats the reference corpus as a low-free-energy state (ordered manifold) and generative degradation as an increase in configurational entropy at fixed structural energy — i.e., a violation of the expected free-energy surface. Rejection is a “Maxwell’s demon” style gate that sorts artifacts and, notably, the erasure of rejected artifacts carries a Landauer cost $kT \ln 2$ per bit which the system accounts for in a compute-cost ledger. Nice touch — Landauer erasure accounting. I’ll add that as a limitation in claim 6 or a dependent claim. Actually Landauer accounting could go in claim 7 (real-time rejection gate) or claim 10. Let me put a Landauer accounting element in claim 7 since rejection gate = erasure.

Hmm, careful — claim 7 is a dependent of claim 6. Let me make claim 7: real-time rejection gate with cascade of stages, early exit, latency budget, and Landauer-cost accounting of erasure operations. Good.

Claim 8: perceptual loss quantification — the U term. Multi-scale deep feature distance, learned perceptual metric with calibrated weights, plus a no-reference naturalness score (NIQE-like) and a reference-based LPIPS. Also the temperature T annealing schedule relating to the free energy.

Claim 9: tamper-evident quality audit certificate — generate a certificate data structure with fields: artifact hash (SHA-256), score vector, model version, timestamp, nonce; compute a Merkle root over per-module intermediate hashes; sign with HMAC/Ed25519; embed in the artifact's metadata or sidecar; verification recomputes.

Claim 10: append-only hash-chained ledger of certificates, with each certificate including the previous certificate's digest; plus a verifier that validates the chain; plus revocation/expiry; plus zero-knowledge proof or selective disclosure of score components