
// TECHNICAL SPECIFICATION · v3.1.2

Extropy Engine v3.1.2

Technical Specification. R is Rarity. F is Frequency-of-decay. Reputation density (ρ) lives only in CT.

Version: 3.1 Status: Canonical specification, sandbox implementation in progress Date: 2026-05-01
Maintainer: 00ranman (Randall Gossett) Companion work: *Unfuck the World for a Dollar* License:
MIT

Abstract

Extropy Engine v3.1 defines a contribution ledger for measuring and rewarding verified entropy reduction across eight domains of human and civilizational activity. It is a protocol for Digital Autarky: every participant keeps sovereignty over their own intelligence, identity, and local context, while the network provides only the minimum common layer required for coordination, verification, and shared receipts.

No central AI thinks for the network. No service performs decomposition on behalf of users. Intelligence stays at the edge. The shared layer standardizes the handshake, the claim schema, the routing primitives, the validation lifecycle, and the immutable causal DAG that records outcomes.

The `epistemology-engine` package is preserved and redefined. v3.0 mistakenly framed it as the place where decomposition happens. v3.1 recognizes it for what it always was: the mesh's emergent peer-review system, surfaced as a witness and aggregation layer over the network's reputation-weighted validation activity. The engine has no central authority over truth. It runs wherever the mesh runs. It observes consensus emerging from incentive-aligned peer review and exposes that emergent epistemology as a queryable, auditable layer.

This system treats entropy reduction not as metaphor but as measurement. The same mathematical grounding that links thermodynamic entropy and informational entropy is extended into operational instrumentation across cognitive, code, social, economic, thermodynamic, informational, governance, and temporal domains. Value is minted only when measurable disorder decreases in real systems under falsifiable conditions, with action-class rarity weighting, domain weighting, and time-aware settlement.

v3.1 is the first specification that fully integrates the personal AI handshake model, the hybrid OAuth + local KYC + DID + ZKP identity layer, the Personal Signed Local Log, the micro-quest

marketplace, the native substrate decision, and the replacement of the old central decomposition model with edge-native intelligence.

0. What's New in v3.1 (Summary)

- Redefined epistemology-engine. The package name and engine concept are preserved. v3.0 framed it as a central decomposition service; v3.1 recognizes it as the mesh's emergent peer-review witness layer. Decomposition itself moves to personal AI at the edge. The engine now observes, aggregates, and surfaces the epistemology that emerges from incentive-aligned validation across the network.
 - Mandatory hybrid identity layer: OAuth + on-device KYC + ZKP DID with per-context nullifiers.
 - Mandatory Personal Signed Local Log (PSLL) per node.
 - First-class micro-quest marketplace with dynamic reward escalation.
 - Final native-substrate decision; three Holochain patterns borrowed and renamed.
 - Default volunteer micro-validation via 1/10th blind slices.
 - All provisional defaults are explicitly governance-tunable from day one.
-

1. What v3.1 Is

Extropy Engine v3.1 is a protocol and reference architecture for tracking who does what, not who has what. The current ownership economy tracks assets, balances, and transfers. Extropy adds a contribution ledger that tracks verified entropy reduction by person, by context, by domain, and by time.

The system is built around five commitments:

1. Value is measurable as entropy reduction.
2. Entropy manifests across eight domains and their intersectionalities.
3. Intelligence belongs to participants at the edge, not to a central network brain.
4. Verification must be adversarially robust, privacy-preserving, and incentive-aligned.
5. Governance must remain fractal, composable, and bounded against permanent concentration.

v3.1 is also the point where the architecture stops hand-waving. The companion novel makes the civilizational and mathematical case. This document specifies the machine.

2. Implementation Status and Environment Reality

This specification is canonical. The current codebase and any VPS instance running it are not yet a production-hardened final realization of the full design.

The currently uploaded Git repository and VPS deployment should be treated as a sandbox implementation of the v3.1 architecture. It is a live engineering testbed, not a finished production deployment, not a hardened adversarial-internet node image, and not a recommendation to run the stack as-is on arbitrary public infrastructure.

That distinction matters.

The spec defines the intended protocol, service boundaries, trust model, economic loop, and validation logic. The present repository reflects a kernel implementation and an evolving service mesh used to exercise the loop, test assumptions, expose failure modes, and close remaining gaps. The code is where the theory gets punched in the mouth.

This is not an apology. It is an honesty clause.

The stack is meant to be run, broken, observed, patched, and iterated until the remaining engineering gaps are gone. The fact that the current deployment is a sandbox does not weaken the architecture. It is how the architecture earns the right to exist.

See `docs/VPS_NODE.md` for the current sandbox-node deployment posture and the local↔VPS handshake harness.

3. Core Vision: Digital Autarky

Digital Autarky means every participant remains sovereign over their own intelligence stack, identity material, decision context, and local event history. The network does not become a supermind. The network becomes a coordination and accounting layer.

Under v3.1:

- Every user runs a personal AI or local multi-model consensus on their own hardware or node.
- That personal AI interprets real-world signals and private context locally.
- The network never needs raw personal context to coordinate shared work.
- Shared infrastructure is reduced to the minimum common substrate needed for claim exchange, routing, validation, rewards, and receipts.

The purpose is verifiable coordination without surrendering personal sovereignty.

This means the protocol explicitly rejects:

- No central AI deciding what reality means.

- No central epistemology engine decomposing the world for everyone.
- No platform-owned private reasoning layer that becomes a hidden control point.
- No requirement that users expose raw identity material or full local logs to the network.

The system only works if edge intelligence remains edge intelligence.

See [architecture/AUTARKY.md](#) for the full vision document.

4. This Is Not a Metaphor

The engine inherits its philosophical and mathematical backbone from the same claim established in the companion novel: entropy reduction is the unit of all value, and this is not a metaphor.

Entropy is not being used here as motivational branding, poetic framing, or aesthetic vocabulary. Entropy is a measurable property of system state. Thermodynamic entropy and informational entropy share formal mathematical grounding (Shannon 1948; Landauer 1961; Bennett 2003). That grounding lets the engine treat teaching, debugging, conflict mediation, scheduling, governance repair, and waste reduction not as incommensurable vibes but as domain-specific instances of the same underlying phenomenon: the reduction of disorder in a system.

The domains differ. The instruments differ. The falsifiability thresholds differ. The human meaning differs. The underlying structure does not.

That is why the system can compare contributions across domains without collapsing into price worship. It is not because all things are identical. It is because all measured value is translated through a common physical and informational frame.

Epistemic note. The Extropy Engine draws its conceptual vocabulary from thermodynamics and information theory. The term "entropy reduction" is used as a generalized measurement framework — operationalized through domain-specific metrics — rather than as a claim of literal thermodynamic equivalence in every domain. Each domain defines its own measurement protocol for what constitutes measurable disorder reduction. See [docs/operationalization/](#) for per-domain instruments and falsification conditions.

5. The Eight Domains

The engine recognizes eight domains in which entropy appears in human systems. Every valid contribution claim must reduce entropy in one or more of these domains.

5.1 Cognitive Entropy

Disorder in knowledge, understanding, mental models, skill formation, and conceptual coherence.

- *Examples:* Teaching a concept clearly. Correcting a misconception. Building a curriculum. Producing documentation that improves comprehension.
- *Typical instruments:* Assessment delta. Retention rate. Competency demonstration. Knowledge graph coherence.

5.2 Code Entropy

Disorder in software systems, architecture, maintainability, correctness, and operational clarity.

- *Examples:* Fixing a bug. Refactoring a brittle module. Increasing test coverage. Reducing complexity.
- *Typical instruments:* Cyclomatic complexity. Failing vs. passing tests. Error frequency. Coverage metrics. Static analysis scores.

5.3 Social Entropy

Disorder in trust networks, cooperation, conflict dynamics, and community coherence.

- *Examples:* Mediation. Trust restoration. Organizing a fractured group. Reducing communication breakdown.
- *Typical instruments:* Conflict incident reduction. Participation quality. Trust survey changes. Network cohesion proxies.

5.4 Economic Entropy

Disorder in allocation, throughput, matching, waste, bottlenecks, and coordination of scarce resources.

- *Examples:* Better matching supply to need. Removing a useless middle step. Improving workflow efficiency. Reducing idle capacity.
- *Typical instruments:* Utilization rates. Throughput improvement. Waste reduction. Fulfillment times.

5.5 Thermodynamic Entropy

Physical disorder expressed through waste heat, physical inefficiency, environmental degradation, or energy loss.

- *Examples:* Improving insulation. Recycling systems. Waste reduction in physical production. Environmental restoration.

- *Typical instruments:* Energy use delta. Heat loss reduction. Material recovery rates. Emissions changes.

5.6 Informational Entropy

Disorder in records, data quality, accessibility, signal-to-noise ratio, and archival coherence.

- *Examples:* Cleaning a dataset. Organizing records. Fact-checking and source reconciliation. Improving discoverability.
- *Typical instruments:* Error rate reduction. Completeness improvement. Retrieval latency. Consistency scores.

5.7 Governance Entropy

Disorder in decision systems, accountability structures, legitimacy, responsiveness, and rule coherence.

- *Examples:* Fixing a broken decision process. Making accountability enforceable. Reducing policy contradiction. Improving transparency.
- *Typical instruments:* Decision latency. Reversal frequency. Participation quality. Auditability metrics.

5.8 Temporal Entropy

Disorder in time allocation, sequencing, synchronization, bottlenecking, and operational cadence.

- *Examples:* Better scheduling. Queue reduction. Workflow synchronization. Eliminating waiting and dead time.
- *Typical instruments:* Cycle time. Wait time. Scheduling conflict reduction. Throughput per unit time.

5.9 Intersectionality Across Domains

The eight domains are not silos. Real contributions often land across multiple domains at once. A teacher may reduce cognitive, social, and temporal entropy in one act. A clean software deployment may reduce code, informational, economic, and temporal entropy simultaneously. Governance reform may cascade into social and economic order.

The engine does not force a claim into one exclusive box. It measures a domain vector and weights it contextually.

6. Canonical XP Formula

The canonical XP formula is:

$$XP = R \times F \times \Delta S \times (w \cdot E) \times \log(1/T_s)$$

Where:

Variable	Range	Description
R	[0.1, 10.0]	Rarity coefficient — action-class scarcity of the loop being closed. Property of the loop, not the actor.
F	(0.0, 1.0]	Frequency-of-decay penalty — multiplicative dampener on repeated submissions of the same claim shape.
ΔS	[0.0, ∞)	Entropy reduction magnitude — the heart of the system.
w	[0.0, 5.0] vector	Domain-weight vector — governance-adjustable per DFAO.
E	[0.0, 1.0] vector	Eight-domain entropy vector — measured per claim.
T_s	(0.0, 1.0]	Settlement-time fraction; $\log(1/T_s)$ rewards faster settlement and prevents farming via instant closure.

The formula lives in one place: `packages/xp-formula/src/index.ts`. Every service that mints XP imports from there. No reimplementations.

v3.1.2 correction. v3.1.1 conflated R with Reliability/Reputation and F with Falsifiability. v3.1.2 corrects both. R is Rarity. F is Frequency-of-decay. Reputation never re-enters XP. Reputation density (ρ) lives only in CT. This is a hard architectural invariant. See `docs/THREE_LAYER_SEPARATION.md` and `docs/CONTRIBUTION_GRAPH.md`.

6.1 Rarity Coefficient R

- R is a property of the loop, not the actor. Same loop closed by a different person yields the same R. Math is invariant under actor swap.
- Measures action-class scarcity within a domain: how hard-to-replicate the contribution shape is, not how trusted the submitter is.
- Past actions never inflate new XP. There is no "R bank" tied to a person, a wallet, or a validator history. R is set by the rarity table for the claim class, not by who is closing it.

- Bounded to [0.1, 10.0] in formula application. Common claim classes sit near 0.1; structurally rare, hard-to-fake reductions sit near 10.0.
- Reputation, reliability, accuracy history, and validator track record do not enter R. They are CT-side concerns and surface only as ρ (reputation density) inside the CT formula.

6.2 Frequency-of-Decay Penalty F

- F is a multiplicative dampener that reduces XP yield when the same submitter repeats the same claim shape too often. Anti-grind, anti-spam, anti-Goodhart. Not a measure of how testable a claim is.
- Fingerprint: F is keyed on the tuple (submitter_id, claim_type, primary_domain, DFAO_id). Different domains, different DFAOs, and different claim types do not penalize each other.
- Production curves shipped in code:
 - extropialingo: $F = 1 / (1 + \log_{1p}(\text{sessionFrequency} - 1))$ — log-tail decay; long-running learners are not punished as harshly as the harmonic curve would punish them.
 - levelup-academy: $F = 1 / (\text{attempts} + 1)$ — harmonic decay; sharper drop, used where each attempt should yield meaningfully less than the last.
- Both curves bounded in $(0, 1]$ and approach zero asymptotically. $n=1$ yields $F=1$, meaning the first instance of a claim in a fingerprint is full strength.
- Reset rules: F resets per fingerprint when (a) the season rolls over under temporal policy, (b) the DFAO redefines the claim type, or (c) a calibration event triggers.
- Falsifiability is a separate concept. Falsifiability is a claim-quality criterion enforced by validation rules and surfaced through the mesh/falsifiability route on the redefined epistemology layer (see §17.2). It is not a variable in the XP formula. v3.1.1 conflated the two. v3.1.2 corrects the record.

6.3 Entropy Reduction Magnitude ΔS

- General method: measure state before, measure state after, compute reduction under the domain's instrument model.
- If no meaningful ΔS can be measured, the system should not mint meaningful XP.

6.4 Weighted Domain Vector ($w \cdot E$)

- E is the eight-domain vector of measured entropy reduction.
- w is set by context. A software-oriented DFAO may weight code and informational entropy heavily. A neighborhood repair DFAO may weight social, governance, and temporal entropy differently.
- This gives the system contextual nuance without abandoning a universal metric.

6.5 Time Factor $\log(1/T_s)$

- Faster resolution of real value under valid verification counts more than value that takes far longer to settle.
- The time term remains logarithmic to preserve boundedness and avoid runaway incentives.

6.6 Irreducible Form

$$XP = \Delta S / c_L^2$$

Where c_L is a domain-specific propagation constant — the empirical maximum rate at which validated information propagates within a given domain's measurement infrastructure. This is a structural analogy to $E=mc^2$, not a physical law. c_L values are calibrated per domain, not derived from fundamental physics. Default values are governance-tunable initial estimates, refined as operational data accumulates.

7. The Personal AI Handshake Model

v3.1 makes the personal AI handshake model canonical.

This corrects the older architectural reading in which the **epistemology-engine** was treated as a central decomposition service. That reading was wrong. Decomposition was never supposed to happen in a single network service — that would centralize intelligence, drift toward hidden ontology control, and violate Digital Autarky. Decomposition is a personal-AI responsibility. The engine itself is something different (see §13).

7.1 Personal AI Responsibilities

Each user's personal AI is responsible for:

- Interpreting real-world requests, complaints, observations, and opportunities locally.
- Translating those into micro-claims.
- Decomposing them into short actionable quests, with 2–5 minutes as the default task grain.
- Maintaining a local signed event history (PSLL).
- Producing claim packages suitable for network submission.
- Managing local identity and consent surfaces.

7.2 Network Responsibilities

The shared protocol is responsible for:

- Standardizing claim and quest schema.
- Routing via SignalFlow.

- Matching validators (volunteer micro-validation by default).
- Recording receipts to the DAG.
- Running mint, burn, and settlement logic.
- Enforcing shared protocol rules.

7.3 Why This Matters

This model prevents centralized AI drift.

The network does not get to decide what you meant. It does not get to own your local context. It does not get to accumulate hidden world models over everyone's lives. It only receives the minimum interoperable outputs necessary for shared action.

8. Identity and Accountability Layer

v3.1 requires a hybrid identity architecture combining usability, privacy, and Sybil resistance.

8.1 Design Goals

The identity system must satisfy five constraints:

1. Easy onboarding for normal humans.
2. Strong resistance to one-person-many-identity abuse.
3. No raw PII exposure to the network DAG.
4. Selective reveal under governance conditions.
5. Compatibility with local edge intelligence.

8.2 Canonical Identity Flow

1. User signs in locally using familiar credentials via OAuth or OpenID.
2. User performs a one-time on-device KYC binding (ID scan, biometric bind, or trusted issuer handoff).
3. Personal AI generates a DID and corresponding Verifiable Credential.
4. Credential is wrapped in zero-knowledge proofs (BBS+ default; zk-SNARKs supported).
5. Network receives only proof material and per-context nullifier outputs required to establish uniqueness and permissions.

8.3 What the Network Sees vs. Doesn't See

Network sees	Network does not see
Proof of uniqueness	Raw identity documents
Proof of valid onboarding	Full biometric material
Contextual nullifier material	Private local onboarding state
Governance-relevant accountability hooks	Real-world identity tied to DID by default

8.4 Accountability

This is not optional anonymity. It is selective privacy under enforceable accountability.

If governance thresholds are met under valid process, specific reveal or neutralization actions may be triggered against a DID. The provisional default is 7-of-12 ecosystem-validator threshold-keyed escrow, governance-tunable.

Full spec: [docs/IDENTITY.md](#).

9. Personal Signed Local Log (PSLL)

The Personal Signed Local Log is the v3.1 name for the local append-only provenance log maintained by each participant's edge intelligence. The pattern is borrowed (with credit) from Holochain's source-chain concept and reimplemented natively.

9.1 Purpose

PSLL exists to:

- Preserve per-node provenance.
- Record decomposition and decision flow locally.
- Support auditability without leaking the full private log to the network.
- Anchor verifiable receipts into the DAG.

9.2 Properties

The PSLL must be:

- Append-only.
- Hash-chained.
- Cryptographically signed.
- Locally controlled.

- Selectively disclosable.

9.3 Anchoring Model

The network does not ingest raw PSLN payloads.

Instead, periodic Merkle-root (or equivalent) commitment receipts are anchored into the DAG. Under dispute, subsets of the PSLN can be revealed with inclusion proofs or ZKP-based selective disclosure.

9.4 Why PSLN Matters

Without a local signed log, edge intelligence becomes hand-wavy. With PSLN, the system gains provenance without surrendering sovereignty.

Full spec: `docs/PSLN.md`. Implementation skeleton: `packages/psln-sync/`.

10. Operational Model: Micro-Quests and the Task Marketplace

The user interface and operational model in v3.1 are built around a living marketplace of micro-quests.

10.1 Inputs

Real-world requests and complaints become structured micro-claims through personal AI mediation.

Examples: Lawn needs mowing. Trash on the road at mile marker 14. A dog killed the chickens. This service crashes under load. This neighborhood group is deadlocked.

10.2 Decomposition

These inputs are decomposed into do-it-now tasks, with 2–5 minutes as the default granularity. The point is operational tractability. A contribution economy fails if contribution units are too vague, too large, or too slow to verify.

10.3 Marketplace Behavior

Quests are published into a marketplace where SignalFlow routes them based on:

- Reputation
- Skill profile
- Location

- Current demand
- Availability
- DFAO policy

10.4 Dynamic Reward Escalation (Provisional)

Neglected work automatically gets higher potential XP/rewards until someone accepts.

Provisional curve: linear 1.0x → 3.0x over 7 days, then logarithmic to a cap of 10.0x.

Governance-tunable.

10.5 Validation by Volunteer Micro-Slices (Default)

Volunteer validators can validate 1/10th blind slices of a claim without seeing the full context.

Aggregation produces F. This dilutes single-validator influence and supports privacy.

Full spec: `docs/QUEST_MARKET.md`. Implementation skeletons: `packages/quest-market/`, `packages/validation-neighborhoods/`.

11. Substrate and Architecture (Final Decision)

v3.1 commits to building a native substrate end-to-end. Extropy is not deployed as a hApp on Holochain or any other existing framework.

11.1 Reasoning

Full Digital Autarky requires owning the lowest shared layer (handshake + DAG). Dependency on another project's plumbing would compromise the vision and create a supply-chain control point we do not control.

11.2 Borrowed Patterns (Reimplemented Natively, Credit Given)

Holochain pattern	Extropy name	Purpose
Source chain	Personal Signed Local Log (PSLL)	Per-node append-only provenance
Neighborhood DHT	Validation Neighborhoods	Sharded validation load balancing + task discovery
Zomes / DNA modules	Rule Modules	Composable, fractal DFAO inheritance and evolution

Credit: the patterns are good. The implementations are ours.

Full spec: `architecture/SUBSTRATE.md`.

12. XP / Loop / Game Theory Layer (Confirmed Intact)

- Provisional mint + retroactive validation/burn + domain-specific falsification conditions remain central.
- Anti-prisoner's-dilemma mechanics strengthen at ~10+ person rings (reputation decay, dynamic quorums, conviction voting).
- Goodhart pressure is treated as diagnostic fuel for instrument refinement, not a fatal flaw.

Loop lifecycle: OPEN → VALIDATING → CONSENSUS → CLOSED → SETTLED (or FAILED / ISOLATED).

13. Service Architecture (v3.1)

13.1 Core Protocol Services

Package	Status	Purpose
contracts/	Active	Shared types, schemas, single source of truth
xp-formula/	Active	Canonical formula implementation (pure function)
loop-ledger/	Active	Loop lifecycle state machine
signalflow/	Active	Validator routing and dispatch
xp-mint/	Active	Two-phase minting (provisional + settle/burn)
reputation/	Active	Per-domain reputation, decay, anti-Sybil scoring
dag-substrate/	Active	Causal DAG ledger, vertex/edge primitives
dfao-registry/	Active	Fractal organization registry
governance/	Active	Proposals, conviction voting, threshold execution
temporal/	Active	Seasons, decay scheduling, loop timeouts

Package	Status	Purpose
token-economy/	Active	6-token economy (XP, CT, EP, IT, GT, RT)
credentials/	Active	Verifiable credential issuance/verification helpers

13.2 New in v3.1

Package	Status	Purpose
identity/	Skeleton	OAuth + on-device KYC + DID + ZKP wrapper
psll-sync/	Skeleton	PSLL anchoring service (Merkle commitments to DAG)
quest-market/	Skeleton	Micro-quest marketplace + dynamic reward escalation
validation-neighborhoods/	Skeleton	Sharded micro-validation routing

13.3 Redefined in v3.1

Package	Status	Redefinition
epistemology-engine/	Active (redefined)	No longer performs central decomposition. Now a mesh-observability and emergent-consensus witness layer. See §13.4.

13.4 The Epistemology Engine — Redefined

The engine name and concept are preserved because the name carries the actual purpose: an engine of epistemology, not a decomposition pipeline. v3.0 misread its role. v3.1 corrects the reading.

What it is now:

A peer-review mesh observability layer. The real epistemology engine is the network itself — every personal AI submitting claims, every volunteer validator running 1/10th blind slices, every reputation update, every retroactive burn or settle. Truth-finding is what *emerges* from those primitives running on incentives at scale. The `epistemology-engine` package is the formal witness,

aggregator, and queryable surface for that emergent process.

What it does:

- Aggregates validation outcomes across the mesh and surfaces consensus drift, dissent clusters, and contested-claim patterns.
- Computes mesh-wide falsifiability statistics (F-distributions per domain, per DFAO).
- Tracks reputation graph evolution and exposes Sybil-suspicious clusters.
- Surfaces emergent ontologies — recurring claim patterns, naming convergence, instrument standardization across DFAOs.
- Provides queryable hooks for governance proposals ("show me all claims in domain X with contested validation in the last cycle").
- Does not decide what is true. Does not perform claim decomposition. Does not arbitrate disputes. Does not own a private world model.

Why this preserves the original vision:

Real peer review was never a service. Real peer review is what happens when honest validators with reputation skin in the game evaluate falsifiable claims under incentive alignment. v3.0 implicitly tried to instantiate that as a microservice, which would have collapsed the property that made it work. v3.1 recognizes that the mesh, running on incentives alone, *is* the engine. The package wraps that emergent process with observability and aggregation, not control.

Architectural posture:

- Read-mostly. Writes only metadata about the network's own state, not new claims.
- Indexes the DAG and reputation graph; does not extend them.
- Stateless under restart; can be rebuilt from DAG replay.
- Multiple instances can run independently — there is no canonical engine instance, by design.

14. Provisional Defaults (All Governance-Tunable)

Knob	Provisional default	Governance authority
ZKP scheme	BBS+	Ecosystem DFAO supermajority
Reward escalation curve	linear→3× over 7d, log to cap 10×	Per-DFAO override allowed
Reveal threshold	7-of-12 ecosystem validators + cause-shown proposal	Ecosystem DFAO

Knob	Provisional default	Governance authority
XP decay rate	$\rho = 0.01$ per 30-cycle period (~1%/month)	Ecosystem DFAO
Transfer friction	$\delta = 0.02$ (2% loss per transfer)	Ecosystem DFAO
Domain weights w	1.0 default per domain	Per-DFAO override
Essentiality factor E	0.8 default	Per-DFAO override
Default task grain	2–5 minutes	Per-DFAO override
Validation slice	1/10th blind	Per-DFAO override

Every knob has a default so the system has shape now. Every knob is votable. Nothing is locked.

15. Falsification Conditions

Each domain's measurement protocol is falsifiable. If domain XP scores show no correlation with independently measured outcome metrics over a defined window with a defined N , that domain's measurement protocol is invalid and must be revised.

See [docs/operationalization/](#) for per-domain falsification thresholds and review cadence.

16. Open Engineering Gaps

63 identified implementation gaps across 13 categories. See [GAPS.md](#) for the full enumeration. The most significant unresolved questions:

- Gödel Boundary Watchdog: paradox-safe self-referential claim handling remains incomplete.
- `c_l` calibration bootstrap: per-domain propagation constants need operational data we do not yet have.
- Network density threshold: voluntary adoption may not reach minimum density without institutional backing.
- Regulatory exposure: CT-to-fiat bridges may trigger securities classification (Howey test in US).
- Cultural friction with 7-day-cycle institutions if the base-10 temporal layer activates.

These are features of an honest engineering specification, not buried caveats.

17. References

See `docs/CHANGELOG.md` and the canonical paper *XP Timekeeping System: Temporal DAG Infrastructure, Entropy Economics, and the Post-Calendar Coordination Problem* (Gossett, 2026) for full bibliographic references.

Co-written and curated by Randall Gossett. The system is designed to be falsifiable, not infallible. Every domain defines what would prove it wrong. That is the difference between engineering and ideology.

// Read more at lladnaros.com · Source on [GitHub](#)