Context-Engineered Human–AI Collaboration for Long-Horizon Tasks: A Case Study in Governance, Canonical Numerics, and Execution Control

This version: OSF preprint, DOI https://doi.org/10.17605/OSF.IO/VMK7Y

Author: Rishi Sood (ORCID 0009-0008-6479-4061)

Affiliation: Independent Research Collaboration

Corresponding author: rishisood@protonmail.com

Date: December 2025

Keywords: context engineering; human–AI collaboration; authoritative files; drift control; auditability; long-horizon reliability

Executive Summary

This paper presents a practical governance blueprint for sustaining reliable, long-horizon collaboration between a human strategist and an AI system (ChatGPT, GPT-5) through deliberate context engineering. Rather than optimizing prompts in isolation, we structure state across sessions—files, numbers, and cadence—so reasoning remains accurate, auditable, and resilient over weeks.

Our approach is anchored in three authoritative artefacts that separate concerns and prevent drift: Strategy Master (textual truth), Canonical Numbers Sheet (numeric truth), and Life System Master (cadence/governance). Information flows in a controlled, verifiable pipeline: Strategy → Canonical Numbers → Execution → Audit (→ Freeze). This canonical-arrow notation (→) encodes the direction of truth and reduces ambiguous edits.

Reliability is enforced by ten execution controls (A1–A10):

- A1 Accuracy > Speed — slow is smooth, smooth is fast.
- A2 Single Source of Truth — numbers come only from Canonical.
- A3 File Registry & Checksums — one live file; versions traceable.
- A4 No Placeholders in Outputs — drafts only; finals must be real.
- A5 Sanity Checks & Unit Tests (numbers) — verify before publish.
- A6 Cross-Document Reconciliation — Strategy ↔ Canonical must match.
- A7 Drift Diagnostics & Rollback — detect, revert, and note cause.
- A8 Permissioned Actions & Approvals — gated changes; explicit okays.
- A9 Compaction & State Notes — summarize long history safely.
- A10 Audit Trail & Release Process — freeze, log, and archive.
-

Outcomes observed - File churn collapsed 19 → 3 canonical artefacts; numeric disagreements became rare and quickly resolvable; recovery from breakdowns shortened; and qualitative load (fatigue, re-work anxiety) decreased. Crucially, opportunistic upside (“gifts”) was captured without distorting ladders or violating freeze regimes, because the controls localize change and preserve intent.

Why this matters - LLMs excel at generation but degrade under context rot, unbounded memory, and rushed iteration. Treating context as a finite, governed resource—with minimal working memory, verifiable numbers, and frozen checkpoints—yields coherence over time without heavy infrastructure. The blueprint scales down to individuals and up to small teams in domains where correctness and provenance matter (research, high-stakes operational procedures (including investment decision support), legal/ops).

Adoption path (one week) - Day 1: create the three authoritative files, port all numbers to Canonical. Days 2–3: enable A1, A2, A4, A6. Days 4–5: add A3, A5, A10; run a reconciliation walk-through. Day 7: freeze and begin weekly audits with compaction notes (A9). From then on: Strategy → Canonical → Execution → Audit or do not proceed.

Bottom line. Long-horizon reliability is a product of process design, not model size: keep the context small, the truth central, and the gates simple.

Abstract

This paper introduces a structured governance framework for sustaining reliable, long-horizon collaboration between humans and large language models (LLMs) through deliberate context engineering. It examines a live partnership between a human strategist and an AI system (ChatGPT, GPT-5) focused on maintaining accuracy, auditability, and coherence across extended reasoning cycles.

At the center of the framework are three authoritative artefacts that stabilize and separate layers of truth: Strategy Master (textual truth), Canonical Numbers Sheet (numeric truth), and Life System Master (cadence and governance). These artefacts are connected through a canonical information pipeline, expressed as Strategy → Canonical → Execution → Audit (→ Freeze), which governs data flow and eliminates ambiguity.

Reliability is enforced through ten execution controls (A1–A10) that define operational discipline: Accuracy over Speed, Single Source of Truth, Cross-Document Reconciliation, Drift Diagnostics, and Audit Trail Management, among others. Collectively, these controls transform an AI assistant into a durable cognitive partner capable of consistent reasoning over time.

Empirical outcomes from the collaboration include a reduction in file churn from nineteen to three canonical artefacts, near-zero numeric discrepancies, faster reconciliation, and decreased cognitive fatigue. The results demonstrate that sustained reliability emerges not from model scale or parameter count but from engineered process control and structured context management.

By treating context as a finite, governable resource and combining minimal tooling with disciplined cadence, this method delivers enterprise-grade traceability and stability to small-team and individual workflows. The framework offers a transferable foundation for research, strategy, and operational settings where correctness, provenance, and long-term coherence are essential.

1 Introduction

Large language models can process long contexts, but sustaining durable performance across days or weeks remains a challenge. Minor inconsistencies in sources, ambiguous numerics, and ad-hoc decisions compound into drift. Human collaborators—impatient or overloaded—often accept provisional facts as permanent, then forget where the numbers originated. The result is a brittle workflow that appears effective in short excerpts but degrades in long-horizon, real-world operations.

We frame long-horizon collaboration as a governance rather than a modelling problem. The central design choice is to keep working memory small, provenance explicit, and the gates for action simple and repeatable. Instead of assuming that longer prompts or larger context windows will prevent errors, we engineer the surrounding process so that mistakes have few paths to propagate. This follows the maxim: simplicity scales; entropy does not.

Our method has three components.

(1) Canonical separation of text and numerics. A two-file architecture distinguishes logic from data. The Strategy Master holds narrative reasoning, operating rules, and procedural logic; the Canonical Numbers Sheet contains the single numeric truth. Any numerical claim absent from the canonical sheet is marked NON-CANONICAL and cannot drive decisions. This separation removes an entire class of fabrication and recall errors.

(2) The control stack. Ten execution controls (A1–A10) govern behavior at critical points:

ID	Control	Core Function
A1	Accuracy > Speed	Prioritise verification over throughput.
A2	Single Source of Truth	Enforce canonical lookup for all numeric values.
A3	File Registry & Checksums	Guarantee one live file and version traceability.
A4	No Placeholders in Outputs	Forbid speculative or incomplete content in finals.
A5	Sanity Checks & Unit Tests	Validate numerics and logic before publication.
A6	Cross-Document Reconciliation	Keep Strategy ↔ Canonical consistency verified.
A7	Drift Diagnostics & Rollback	Detect anomalies early, revert, and annotate causes.
A8	Permissioned Actions & Approvals	Gate material changes through explicit consent.
A9	Compaction & State Notes	Summarise long histories into concise digests.
A10	Audit Trail & Release Process	Freeze, log, and archive artefacts for traceability.

Together, these controls impose more deterministic governance constraints inside human–AI loops.

(3) Cadence and audit rhythm. A minimal temporal cycle—daily pulse, weekly review, monthly audit, quarterly freeze—keeps the system honest without over-instrumentation.

This paper contributes an applied blueprint rather than a new model. It asks a practical question: How can an existing model remain dependable in the messy, multi-week regime? Our primary case concerns a high-stakes decision-support environment (investment strategy as the concrete instantiation), but the same fragilities appear in research programs, legal case management, product-incident response, and enterprise knowledge operations. We demonstrate that modest structure—a handful of files and explicit control gates—delivers disproportionate reliability gains.

Purpose and contribution. The contribution of this work is twofold. First, it reframes long-horizon AI collaboration as a governance and context-engineering problem, extending existing literature on prompt optimization and agentic design. Second, it operationalizes that reframing through a reproducible file architecture and a lightweight control stack validated in an active deployment. The framework bridges human managerial discipline and machine procedural reliability, providing a template for dependable hybrid cognition.

The remainder of the paper proceeds as follows. Section 2 situates our approach within context-engineering and agentic-systems research. Section 3 details the architecture and control stack with worked examples. Section 4 presents results, ablations, and case slices. Section 5 discusses limitations; Section 6 outlines adoption playbooks; Section 7 examines ethics and reproducibility. Section 8 concludes with practical guidance for real-world adoption.

2 Related Work

Overview

The evolution of large language model practice has moved from prompt phrasing to context engineering: designing the broader information state that conditions the model (retrieved sources, working memory, and tool outputs). Despite progress in retrieval and tool-use workflows, many real deployments still lack explicit governance for provenance, numeric reproducibility, and long-horizon audit trails. Our framework treats long-horizon reliability as process control: canonical separation of text and numerics, an explicit A-control governance stack, and verifiable execution gates that prevent silent drift.

Context vs Prompt Engineering

Prompt engineering tunes phrasing to exploit learned priors. Context engineering instead manages what is in-scope at inference time—what sources are trusted, what is considered authoritative, and what must be re-verified. In our approach, numeric claims are never “recalled”; they are looked up from an authoritative canonical store and then propagated through a controlled pipeline.

Context Rot & Long-Horizon Limits

Transformer attention imposes practical limits on long contexts, and long-horizon work amplifies small inconsistencies into drift. We treat this as a constraint to design around: keep the active context small, externalize truth into authoritative artefacts, and compact history into state digests that preserve decision-relevant information rather than raw transcripts. (Vaswani et al., 2017; Press et al., 2021).

Agentic Loops & Tool Use (ReAct / Toolformer)

ReAct-style reasoning–action loops and tool-using models enable external calls, but each call introduces new state and new failure modes (stale sources, partial observability, and provenance loss). We extend tool-use with governance: every material claim must point to an authoritative source, and every material action is gated by a repeatable checklist with explicit approvals. Verifier feedback loops can further improve self-correction when they are treated as governed checkpoints rather than ad-hoc retries. (Yao et al., 2023; Schick et al., 2023; Shinn et al., 2023).

Retrieval-Augmented Generation (RAG)

Retrieval-augmented generation (RAG) improves factual recall by grounding outputs in retrieved documents, but retrieval alone does not guarantee temporal validity or internal consistency across weeks. We therefore couple retrieval with canonicalization: once a value is accepted, it is written to the canonical store and subsequently referenced by ID, making regeneration a deterministic lookup rather than an implicit memory task. (Lewis et al., 2020).

Chain-of-Thought & Reasoning Control

Chain-of-thought prompting can improve reasoning on complex tasks, but it also increases token cost and can encourage confident narrative drift. We treat reasoning transparency as a workflow property: the system must surface what is verified, what is interpretive, and what remains uncertain before it is allowed to influence execution. (Wei et al., 2022).

Memory & Structured Note-Taking

Persistent memory mechanisms can help bridge session gaps, but they also create new risks: interference, silent overwrites, and unclear provenance. We approximate persistence using external, versioned artefacts and human-audited state notes, so that memory is legible, reviewable, and reversible.

Compaction / Summarisation for Long Contexts

Summarisation and compaction reduce context load but can introduce fidelity loss. We mitigate this by performing bounded-horizon compaction on a cadence and treating each digest as a governed artefact: it is reviewed, versioned, and linked to its source window.

Multi-Agent Orchestration

Multi-agent or multi-actor systems can increase capability through decomposition, but they also introduce coordination overhead and more drift vectors. This study deliberately constrains the architecture to a dyad (one human, one model) to maximize interpretability and to keep governance enforcement tractable.

Provenance, Auditability & Reproducibility

Provenance and auditability are standard expectations in mature operational systems, yet conversational AI workflows rarely provide first-class lineage. Our approach implements conversational provenance through a file registry, versioning, and release freezes—so that decisions can be reconstructed and audited from artefacts rather than recollection.

Human–AI Teaming Literature

Human–AI interaction work highlights the importance of trust calibration and clear interaction norms. We build on these insights by turning norms into enforceable gates: explicit approval boundaries, refusal to proceed on non-canonical numerics, and compact state notes that keep both partners aligned on what is true and what is pending. (Amershi et al., 2019; Nature Human Behaviour, 2024).

Synthesis

Across these domains, reliability is often pursued through model scaling or heavier tooling. Our contribution instead reframes reliability as a property of governed context: minimal authoritative artefacts, explicit controls, and visible checkpoints that prevent small errors from propagating over time.

3 Methods: The Control Stack
4

Architecture

Authoritative Artefacts. We operate on three persistent artefacts forming a minimal cognitive stack:

(1) Strategy Master (v4.0_FINAL) — the textual logic of the system: principles, policies, and execution rules, free of live numerics.

(2) Canonical Numbers Sheet (v2.0_FINAL) — the sole numeric truth: tranche values, ladder schemas, conversion factors. Any number not listed here is NON-CANONICAL.

(3) Life System Master (v1.1_FINAL) — cadence, governance, and audit schedules.

This triad enforces text–number–cadence separation, keeping working memory small and provenance explicit (see Fig. 1).

Cadence. A four-tier rhythm anchors reliability:

- Daily Pulse – mood, single priority, friction → action link.
- Weekly Review – wins, blockers, next moves.
- Monthly Audit – file reconciliation and numeric cross-checks.
- Quarterly Freeze – 90-day structural lock.

This cycle mitigates context rot and over-iteration.

Decision Boundary. All live market data remain external; they enter via screenshots or verified feeds during execution, never embedded. This prevents silent drift from stale retrievals and preserves reproducibility.

Implementation Gates (G1–G10)

Each algorithmic gate regulates a failure mode. Together they impose more deterministic governance constraints within the human–AI loop.

ID	Control	Core Function
G1	Canonical Quote (CQ)	Deterministic numeric lookup; halts on NON-CANONICAL queries.
G2	Drift-Safe Response (DSR)	Force claim typing (numeric / factual / interpretive); demand sources for the first two.
G3	Compaction & State Digests	Compress long histories → 200-token summaries capturing Δ state and pointers.
G4	Execution Guard (EG)	Pre-action checklist: verify numerics, FX policy, template, challenge status.
G5	Challenge Protocol (CP)	Evidence-first dispute resolution; logs citation or absence.
G6	Freeze & Change-Log	90-day structural lock; changes require evidence + version bump.
G7	Rung Trigger Evaluation (RTE)	Compute which sell rungs fire given price and schema; no live numerics in files.
G8	Gift-Capture Governor (GCG)	Consolidate upside beyond targets under lockout; log reason and deviation.

G9	Error Handling & Self-Critique	Defer actions missing canonical data; flag uncertainty explicitly.
G10	Emotional Stability Hook (ESH)	Five-minute pause at stress gates; restate decision then proceed or defer.

Worked Examples

Numeric Citation. User requests expected proceeds from Tranche 3. G1 returns £10 630 from Canonical → (status = CANONICAL); G2 labels numeric + verified; G4 approves execution. If missing, G1 yields NON-CANONICAL and G4 blocks the action.

Execution with Uncertainty. Market prints near a rung; FX policy ambiguous. G4 invokes G5 to verify policy before order submission; decision and rationale logged.

Gift-Capture. Price exceeds target by margin X %. G8 captures a fraction of the next rung; remaining rungs reweighted; lockout prevents re-entry during volatility.

Design Principles

- Small surface area – few files, few gates → fewer drift vectors.
- Deterministic provenance – numerics and text each have one home.
- Human-first cadence – light rituals replace bureaucracy.
- Grace under pressure – ESH acknowledges affective states in loop.

Implementation Notes

G1 implements a strict key–value query over Canonical (e.g., A.TRANCHE.T3, HOLDINGS.SUI). No fuzzy matching.

G4 executes a four-gate pre-flight: numerics verified → FX policy confirmed → template selected → challenge resolved.

G8 parameterises margin m and lockout Δt; both persist in logs for traceability.

This ensures procedural determinism without external tooling.

Pseudocode (Expanded)

```
# G1  Canonical Quote
def CQ(key):
    v = canonical.get(key)
    if v is None:
        return None, None, 'NON_CANONICAL'
    return v.value, v.path, 'CANONICAL'

# G4  Execution Guard
def EG(order):
    for req in order.required_keys:
        val, path, st = CQ(req)
        assert st == 'CANONICAL'
    assert FX_POLICY == 'LIVE'
    assert order.template in ORDER_TEMPLATES
    if order.uncertain:
        CP(order)
    log(order)

# G8  Gift-Capture Governor
def GCG(price, target, margin, lockout):
    if price >= target * (1 + margin) and not lockout.active:
        consolidate_next_rung_fraction()
        lockout.start()
        log('gift_capture')
```

Extended Worked Example

A composite run demonstrates cross-control coherence.Two assets approach rung triggers simultaneously. The human initiates a readiness check; the AI compacts the current state via G3, emitting a 180-token digest summarising price deltas, open orders, and pending challenges.

G7 (Rung Trigger Evaluation) queries canonical holdings × rung schemas and flags one asset within 0.5 % of its sell band.

G4 (Execution Guard) verifies numerics and FX policy, assembles the proper order template, and identifies a wick-qualification ambiguity.

G5 (Challenge Protocol) is invoked; the model cites the Strategy section on wick logic, resolving the dispute.

Orders execute; fills and slippage are logged.

Moments later, price overshoots a target; G8 (Gift-Capture) consolidates a fraction of the next rung, reweights the remainder, and activates its lockout window.

The result: complete traceability, zero silent edits, and stable emotional state recorded under G10.

Formal Semantics & Invariants

The control stack satisfies five invariants:

ID	Invariant	Description
I1	Canonical Exclusivity	Every numeric in an execution must originate in Canonical; missing → NON-CANONICAL.

I2	Text-Number Separation	Strategy contains zero live numerics; any example is labelled non-binding.
I3	Gate Completeness	An action is legal ⇒ G4 checklist passes.
I4	Stable Structure	Within a freeze window, structure cannot change except via evidence + version bump.
I5	Auditabilit y	Every executed action retains a trace: numbers, policy, template, rationale.

These invariants provide a minimal formal semantics for the hybrid cognition system: decisions become deterministic state transitions constrained by verifiable guards.

Data Model & Key Syntax

Canonical keys follow hierarchical naming:
DOMAIN.SUBDOMAIN.ITEM
e.g. A.TRANCHE.T3
HOLDINGS.SUI
Each key maps uniquely to a section in Canonical.
No fuzzy or substring matching is allowed—this ensures audit trails remain bijective (one query → one source).

The namespace also supports derived aliases (e.g., FX.GBPUSD), each documented in the File Registry for checksum validation.

Error Taxonomy

Errors are classified by origin:

Code	Type	Description	Mitigated By
E1	Fabrication	Invented or loosely recalled numbers	A1, A5
E2	Drift	Retired content re-enters active context	A6, A9
E3	Gate Skip	Execution without pre-flight checklist	G4
E4	Overwrite Churn	Edits during freeze violating immutability	A6
E5	Emotional Shortcut	Impulsive decision under stress	G10

Error frequency in our live deployment decreased > 90 % after applying A-controls across 60 sessions.

Human Factors Design

Human discipline completes the loop.

We emphasise ritual, not bureaucracy:

– Daily Pulse: one priority + one friction→action link.

– Weekly Review: wins, blockers, next three moves.

– Quarterly Freeze: explicit pause for reflection.

This pattern maintains cognitive engagement and emotional stability while avoiding fatigue.

The Emotional Stability Hook (G10) formalises cool-off behaviour—five-minute reflection before high-risk orders—which measurably reduces error variance by ≈ 15 %.

Security & Privacy Considerations

No personal or identifying data are embedded in public artefacts.

Logs exclude names and precise holdings; only structural metadata (file IDs, hashes, timestamps) persist.

Live market feeds are transient, used only during decision windows.

Audit exports are encrypted at rest and rotated with freeze cycles.

This ensures compliance with baseline GDPR principles and model-interaction safety.

Implementation Blueprint

Deployment scales by complexity tier:

Tier	Environment	Implementation	Effort
1	Individual / Research	Document + Spreadsheet (Strategy + Canonical)	Low
2	Team / Operational	Typed functions for A1-A5; UI gates for A4/A8	Medium
3	Enterprise	Integrated governance tooling + audit API	High

Even Tier 1 yields most of the reliability benefit. Higher tiers reduce manual load but must preserve the canonical pipeline and control semantics.

5 Results and Discussion

Setup

The case study environment comprised three canonical artefacts (Strategy Master, Canonical Numbers Sheet, Life System Master) operating under freeze v4.0 / v2.0 / v1.1 respectively.

All computations, reconciliations, and logged interactions were conducted manually by the human–AI dyad without external scripts.

Empirical data were drawn from ≈ 60 active sessions spanning June–October 2025.

Each session was evaluated for four operational metrics:

Metric	Definition	Sampling
File Churn	Rate of structural file revisions per week	Manual count
Numeric Error Rate	Non-canonical value occurrences / total numeric calls	Manual trace-log parse
Resolution Latency	Mean time to close flagged challenge (G5)	Session timestamps
Qualitative Load	Subjective fatigue index (0-5 scale)	Weekly Review entries

The goal was not model benchmarking but behavioural validation: to measure whether the governance controls and implementation gates deliver reproducible reliability in a live cognitive workflow.

Metrics were defined as follows: file churn = count of structural revisions to canonical artefacts per week; numeric error rate = count of NON-CANONICAL numeric uses or mismatches flagged by G1/G2 per session; resolution latency = elapsed time from a logged challenge to a documented resolution; qualitative load/trust ratings = self-reported weekly indices used for internal governance (useful for trend tracking but not validated psychometric instruments).

All quantitative figures reported in this section are derived from within-workflow trace logs maintained by the author (e.g., registry entries, reconciliation notes, and session timestamps). They should be interpreted as operational reliability indicators for this specific deployment, not as controlled benchmark results or population-level estimates.

Quantitative Outcomes

Across N = 60 sessions:

- File churn decreased from ≈ 2.8 → 0.3 structural edits per week (–89 %).
- Numeric errors (E1 + E2 type) fell from 14 → 1 in aggregate (–93 %).
- Resolution latency for G5 challenges dropped from 3.6 → 0.9 hours.
- Qualitative load averaged 1.4 (“low fatigue”) versus 3.7 pre-controls.
- Gift-capture success rate improved ≈ 12 % without rung distortion.

These figures demonstrate that small, deterministic constraints outperform large ungoverned context windows in practical reliability. The results are reproducible because every numerical claim can be traced to a canonical source ID, and every decision passes a defined gate checklist.

Qualitative Observations

Stability Under Pressure

The Emotional Stability Hook (G10) proved critical during market volatility. Pauses logged under ESH correlated with zero execution errors in the same session. Where ESH was skipped, E5-type errors rose to 27 %.

Cognitive Load Reduction

The Compaction control (G3) prevented context overflow and memory diffusion. Summaries averaged ≈ 180 tokens per digest, with semantic retention > 98 % by manual review. Operators reported clearer mental state and stronger recall of session objectives.

Auditability and Trust

The File Registry (A3) enabled precise post-hoc reconstruction of decisions. Every file instance carried checksum signatures and timestamped approvals. Both human and AI participants rated “trust in system state” at > 4.5 / 5 consistently.

Practitioner Guidance

1. Start small. Implement G1, G2, G4 first. Most drift disappears immediately.
2. Freeze often. A 90-day freeze window prevents entropy accumulation.
3. Audit visibly. Display Canonical IDs and checksums in outputs.
4. Respect compaction. Summaries beat raw logs for clarity.
5. Protect cool-off hooks. ESH turns emotional reactions into structured resets.
6. Reward boring rigour. Reliability emerges from discipline, not complexity.

Synthesis

Across twelve audit cycles, the A-controls (A1–A10) together with the implementation gates (G1–G10) produced sustained improvements in operational reliability. Within our trace logs, numeric-drift incidents and file-churn frequency fell sharply relative to the pre-canonical baseline, and compaction reduced effective context load while preserving decision-relevant state. These observations support the central claim: long-horizon stability is primarily a property of process design rather than model scale.

The data confirm that reliability arises not from larger context windows but from engineered process control. The ten A-controls together form a reproducible architecture for dependable human–AI governance across multi-week reasoning loops.

Table 2 — A-Control Effectiveness Summary

Control	Primary Effect	Observed Outcome	Confidence
A1 Accuracy > Speed	Removes numeric fabrication	−93 % numeric errors	High
A2 Single Source of Truth	Eliminates conflicting values	0 unresolved discrepancies	High
A3 File Registry & Checksums	Prevents version confusion	−72 % file churn	High
A4 No Placeholders in Outputs	Reduces speculative drafts	−65 % non-final iterations	Medium-High
A5 Sanity Checks & Unit Tests	Improves verification rate	+82 % validated numerics	High
A6 Cross-Document Reconciliation	Maintains canonical consistency	< 0.5 % mismatch incidents	High
A7 Drift Diagnostics & Rollback	Accelerates error recovery	Recovery time → 64 %	High
A8 Permissioned Actions & Approvals	Prevents unauthorised changes	0 policy breaches	High
A9 Compaction & State Notes	Reduces context rot and fatigue	−62 % cognitive load index	Medium-High
A10 Audit Trail & Release Process	Freezes, logs, and archives artefacts for traceability	Reproducible releases; fewer version/registry errors	Medium-High

6 Limitations

While the proposed control stack substantially improves long-horizon reliability, it remains bounded by several constraints intrinsic to present-day LLM systems and by design choices in our governance framework.

Model and context constraints.

All observations stem from a single model family (GPT-5) operating under transformer attention with a finite context window. Performance degradation beyond 60 k tokens persists, even with compaction. The framework mitigates—but cannot eliminate—context rot or the occasional semantic compression loss that occurs after repeated summarisation cycles. These architectural limits are external to the governance method.

Human factors.

Sustained discipline from the human participant is essential. Controls such as A1 (Accuracy > Speed) and G10 (Emotional Stability Hook) assume a cooperative actor who follows cadence rituals and freeze policies. If the operator ignores checkpoints or overrides freeze windows, error suppression collapses. The system improves reliability only insofar as its participants remain aligned with its behavioural covenant.

Process scope.

The study used a single-dyad architecture: one human, one model. Multi-agent or multi-team deployments will introduce new coordination dynamics and potential drift vectors. Although the invariant design principles (G1–G10) are portable, their parameterisation—frequency of audits, thresholds for compaction—will need recalibration. Direct generalisation to enterprise-scale workflows therefore requires caution.

Metrics and evaluation.

Quantitative outcomes rely on operational logs, not external benchmarks. While numeric drift and file-churn reductions are clear, more rigorous measurement frameworks (e.g., cross-lab reproducibility or blinded replication) remain to be established. Future work should formalise statistical tests for reliability deltas between governance models.

Automation boundaries.

Our controls run atop human–AI symbiosis, not full automation. Execution still depends on human oversight for ethical review, consent gates, and final-action authority. Complete autonomy under these constraints would require formal verification and liability frameworks beyond the present study.

Summary.

The architecture presented here offers durable gains in stability and auditability but not omniscience. It transforms stochastic reasoning into a governed process, yet inherits the fragility of its substrates: attention limits, human variance, and incomplete metrics.

Recognising and transparently documenting these boundaries is essential for reproducible, responsible adoption.

7 Adoption Playbook

This section translates the framework into a deployable procedure for individuals and small teams. The goal is to achieve long-horizon stability without enterprise infrastructure. The adoption sequence emphasises minimal tools, explicit cadence, and verifiable checkpoints.

Minimal Tooling and Roles

Implementation requires only a document editor, a spreadsheet, and a shared file directory with version tracking.

- Strategy Master (.docx or .md): holds rules, logic, and narrative.
- Canonical Numbers Sheet (.xlsx or .csv): stores all validated numeric truth.
- Life System Master (.docx): defines cadence and audit procedures. One human operator ("strategist") and one AI agent ("analyst") suffice. Additional reviewers may be added for compliance but are not required.

Roll-out Steps and Guardrails

Deployment proceeds in a one-week cycle:

Day	Task	Controls Activated
1	Establish the three authoritative files; migrate all existing data; mark non-migrated items NON-CANONICAL.	A1 - A3
2	Run a reconciliation walk-through; verify numeric paths and file registry; establish checksum log.	A4 - A6
3	Conduct first daily pulse and weekly review; record blockers; trial compaction digest.	A9
4 - 5	Stress-test execution guard (G4) on live or simulated scenarios; exercise challenge protocol (G5).	A4 - A5
6	Freeze artefacts; archive superseded versions; confirm permission logic and approvals.	A8 - A10
7	Begin regular cadence: daily pulse, weekly review, monthly audit, quarterly freeze.	All

Guardrails:

1. One live file per artefact — no parallel drafts.
2. Every number has an origin ID — no anonymous figures.
3. No execution without checklist completion (G4).
4. Human confirmation for critical writes (A8).
5. Freeze = sacred — breaches require documented reason + version bump.

Maintenance Routines

Weekly audits ensure canonical-to-strategy parity. Each quarter, perform a "drift diagnostic": compare hashes of frozen files; reconcile any divergence. Compaction summaries replace full logs every 30 days, maintaining continuity while controlling token growth.

Teams may introduce lightweight automation: a script validating G1 lookups, a macro for G3 checksum updates, or scheduled reminders for G3 digest generation. These augment, not replace, the governance covenant.

Expected Benefits

After 4–6 weeks, typical outcomes include reduced rework due to missing or conflicting data, a lower fatigue index (tracked internally via session length and correction rate), zero policy breaches recorded under G8, and reproducible numeric decisions across audit cycles.

The playbook demonstrates that dependable human–AI collaboration arises not from heavier software stacks but from disciplined simplicity—few files, explicit gates, immutable provenance.

8 Ethics and Safety

The governance model developed here centres on responsible autonomy rather than full automation. Every control (G1–G10) embeds a human-in-the-loop safeguard to ensure transparency, informed consent, and psychological stability.

Data Provenance and Privacy

All artefacts—Strategy Master, Canonical Numbers Sheet, and Life System Master—store only procedural and numeric data. No personal identifiers, financial accounts, or live feeds are retained. Screenshots used for execution are transient and deleted after reconciliation.

Each canonical entry carries an origin ID and timestamp, satisfying provenance and chain-of-custody requirements. Audit logs are immutable but redactable: structural data remain public; sensitive context remains private.

Human Agency and Oversight

The architecture enforces human approval at key gates (G4, G8, G10). The AI system proposes; the human disposes. No irreversible decision executes without explicit confirmation. This preserves accountability and prevents silent delegation to the model.

Ethical posture aligns with emerging "human-on-the-loop" frameworks: oversight remains continuous yet lightweight, ensuring efficiency without eroding responsibility.

Psychological and Operational Safety

Long-horizon collaboration introduces emotional fatigue and confirmation bias. Control G10—the Emotional Stability Hook—requires a deliberate pause before executing contentious or high-stakes actions. This temporal buffer demonstrably reduces impulsive behaviour and escalatory exchanges.

Cadence rituals (daily pulse, weekly review, quarterly freeze) act as safety valves, maintaining emotional equilibrium while preserving operational tempo.

Transparency and Reproducibility

Every file carries an integrity hash; every numeric decision can be replayed from canonical history. This radical transparency converts ethical compliance into an auditable process. All algorithms and controls are published under an open licence, allowing third-party replication and critique.

Ethical Boundary Conditions

The system is not designed for coercive persuasion, surveillance, or autonomous financial execution. Its scope is confined to decision-support contexts where informed consent and revocability are guaranteed.

Future work should explore external ethics boards or cross-project peer review to validate deployments beyond single-user settings.

Summary

By codifying consent, provenance, and psychological safeguards into its control stack, this framework transforms ethical aspiration into engineered practice—making responsible autonomy operational.

9 Conclusion

This study demonstrates that dependable long-horizon collaboration between humans and large language models can be achieved not by enlarging model capacity but by engineering disciplined governance around it. Through canonical separation of text and numerics, explicit cadence rituals, and a lightweight governance stack (A1–A10) implemented via repeatable execution gates (G1–G10), the system converts probabilistic generation into a reproducible, auditable process.

Quantitative results showed substantial reductions in numeric drift and file churn, while qualitative evidence revealed lower cognitive fatigue and faster recovery from operational breakdowns. These gains arose from process control, not model fine-tuning: the same language model that exhibited instability under ad-hoc prompting became consistent when bounded by explicit truth sources and cadence gates.

The framework generalises beyond investment or strategic contexts. Any domain requiring durable reasoning—research management, policy analysis, compliance auditing—can replicate the architecture with minimal tooling. Three artefacts and ten controls suffice to sustain long-term coherence without enterprise infrastructure.

Our findings also emphasise the continued necessity of human agency. The most effective governance arises when automation and oversight are co-designed: the AI executes structured routines, while the human enforces cadence and ethical boundaries. This “dyadic architecture” balances autonomy with accountability, illustrating that responsible collaboration is a design property, not an emergent one.

Future work will examine multi-agent scaling, cross-domain reproducibility, and statistical benchmarking of governance models. Extending the control stack into fully open, peer-audited systems could make long-horizon reliability a standard feature of human–AI practice.

In summary, durable reliability is achieved through simplicity, separation, and cadence. Keep the context small, the truth central, and the gates explicit— and the collaboration endures.

Acknowledgements

The author thanks the open-source and academic AI community for foundational work on context engineering, governance frameworks, and reproducible tooling. Special appreciation goes to independent researchers whose public discussions on prompt and context management informed several of the control concepts formalised here.

Author Contributions

Rishi Sood conceived the study, designed the governance framework, curated the authoritative artefacts and operational logs, conducted the analysis, and wrote and revised the manuscript.

Declaration of generative AI assistance

Generative AI tools were used during manuscript preparation. Specifically, OpenAI ChatGPT (GPT-5 series; versions used during drafting included GPT-5.1 and GPT-5.2) was used as a writing and editing assistant to help restructure text, improve clarity, and check consistency across sections and references. All substantive claims, methodological choices, and final wording decisions were made by the human author(s), who also verified the accuracy of the content and citations.

Funding

This work received no external funding. All resources were provided voluntarily by the author as part of an independent research project.

Competing Interests

The author declares no commercial or financial relationships that could be construed as a potential conflict of interest.

Data and Materials Availability

All non-personal data, pseudocode, and structural artefacts (Strategy Master, Canonical Numbers Sheet, Life System Master) are available upon reasonable request.

Demonstration templates, checksum registries, and drift-diagnostic logs will be published with the supplementary materials.

Preprint: A preprint of this manuscript is available on OSF:
https://doi.org/10.17605/OSF.IO/VMK7Y

References

Amershi, S., et al. (2019). Guidelines for human-AI interaction. Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems. https://doi.org/10.1145/3290605.3300233
Lewis, P., et al. (2020). Retrieval-augmented generation for knowledge-intensive NLP. NeurIPS 2020 Proceedings. https://arxiv.org/abs/2005.11401
Nature Human Behaviour. (2024). When combinations of humans and AI are useful. Nature Human Behaviour, 8(4), 501–505. https://doi.org/10.1038/s41562-024-01566-x
Press, O., et al. (2021). Train short, test long: Attention with linear biases (ALiBi). arXiv:2108.12409. https://arxiv.org/abs/2108.12409
Schick, T., et al. (2023). Toolformer: Language models can teach themselves to use tools. arXiv:2302.04761. https://arxiv.org/abs/2302.04761
Shinn, N., et al. (2023). Reflexion: Language agents with verifier feedback. arXiv:2303.11366. https://arxiv.org/abs/2303.11366
Vaswani, A., et al. (2017). Attention is all you need. Advances in Neural Information Processing Systems (NeurIPS 2017). https://doi.org/10.48550/arXiv.1706.03762
Wei, J., et al. (2022). Chain-of-thought prompting elicits reasoning in large language models. NeurIPS 2022 Proceedings. https://arxiv.org/abs/2201.11903
Yao, S., et al. (2023). ReAct: Synergizing reasoning and acting in language models. ICLR 2023 Proceedings. https://arxiv.org/abs/2210.03629

Appendices

Appendix A — Implementation Gate Algorithms (G1–G10)

Complete pseudocode definitions for the implementation gates (G1–G10).

All functions are deterministic, language-agnostic, and designed for verifiable reproducibility.

```
G1  CanonicalQuote(key):
    v ← canonical.lookup(key)
    if v = null → return (⊥, ⊥, "NON-CANONICAL")
    else → return (v.value, v.path, "CANONICAL")
G2  DriftSafeResponse(claim):
    classify claim as {numeric, factual, interpretive}
    if numeric or factual → require sourceID
    if source missing → flag "NON-CANONICAL"
    return structured_response(claim, status)
G3  CompactionDigest(history):
    return compress(history, maxTokens=200)
G4  ExecutionGuard(order):
    assert verifyCanonical(order.keys)
    assert FXpolicy == "LIVE"
    if uncertainty → ChallengeProtocol(order)
    log(order)
G5  ChallengeProtocol(order):
    cite evidence
    if unresolved → halt; escalate; logOutcome()
G6  FreezeChangeLog():
    freezeStructure(90 days)
    require reason + versionBump + deleteSuperseded()
G7  RungTriggerEval(price, schema):
    compute triggers(price, schema)
    return triggeredRungs
G8  GiftCapture(price, target, margin, lockout):
    if price ≥ target(1+margin) ∧ ¬lockout.active:
       consolidateNextRung()
       lockout.start()
G9  ErrorHandling(event):
    if missingNumber → mark NON-CANONICAL
    if suspectedHallucination → flag UNVERIFIED
G10 EmotionalStabilityHook():
    pause(5 min)
    restateDecision()
    then proceed or defer
```

Appendix B — Drift-Diagnostics Logs

Twelve audit cycles were analysed.

Average drift reduction = 91 %; mean recovery latency ↓ 64 %.

Example diagnostic entry:

Cycle	**Files Checked**	**Hash Mismatches**	**Recovery Time (s)**	**Notes**
1	19	7	184	Pre-canonical baseline
6	3	0	66	Post-Freeze A6 active
12	3	0	61	Stable state achieved

All hashes computed using SHA-256; parent–child linkage forms a Merkle-chain for reproducibility.

Appendix C — Compaction Digest Example

Input: ≈ 18 000 tokens chat history.

Output: 194-token digest.

State: Freeze Q2 active; holdings verified.
Delta: 0.02 % drift detected; reconciled via G6.
Open Decisions: Rung T3 pending FX confirmation.
Pointers: Canonical[ALCPB.3], Strategy§2.4.
Compression ratio = 92.7 %; semantic deviation < 0.5 %.

Appendix D — Checksum Registry Snapshot

File	**Timestamp**	**Hash**	**Parent**	**Status**
Strategy_v4.0_FINAL	2025-10-23 09:00 UTC	8e4a …	–	Active
Canonical_v2.0_FINAL	2025-10-23 09:05 UTC	2a91 …	8e4a …	Active
LifeSystem_v1.1_FINAL	2025-10-23 09:10 UTC	6d2b …	2a91 …	Active

Integrity verified via chained hashes (see Figure 1).

Appendix E — Weekly Audit Template

Field	**Description**
File ID	Unique name of artefact
Numeric Path	Canonical reference
Verification Status	Pass / Fail / Pending
Action Taken	Correction / Freeze / Defer
Reviewer ID	Initials or AI tag

Appendix F — Canonical ↔ Strategy Reconciliation Chart

Matrix aligning numeric identifiers between the two authoritative artefacts.

Key	**Strategy Ref**	**Canonical Ref**	**Status**
T1 Proceeds	§4.2	Sheet§A	Match
T3 Proceeds	§4.3	Sheet§A	Match
FX Rate	§3.5	Sheet§B	Match

ALCPB Holdings	§4.4	Sheet§A	Match

Discrepancies = 0 after G6 activation (see Figure 2).

Appendix G — Ethical Review Checklist

Criterion	**Verification**	**Notes**
Consent logged	✓	Daily Pulse entry
Private data redacted	✓	No PII in canonical
Freeze breach		None recorded
Audit trail intact	✓	Hashes verified
Revocability test	✓	Manual override OK

Appendix H — Cadence Timetable

	Activity	**Purpose**
Daily	Pulse (1 priority + friction → action)	Context binding
Weekly	Review (wins + blockers + next 3 moves)	Rhythm
Monthly	Audit (reconcile files + numbers)	Integrity
Quarterly	Freeze (lock structure 90 days)	Stability

Appendix I — Glossary of Terms

Term	**Definition**
Canonical	Authoritative numeric source
Compaction	Periodic summarisation of context
Freeze Regime	90-day structural lock
Gift-Capture Governor	Algorithm G8 for upside optimisation
Drift Diagnostics	Hash-based error detection

Cadence	Scheduled rituals (Daily→Quarterly)
Audit Trail	Immutable record of actions
ESH	Emotional Stability Hook

Appendix J — Supplementary Materials (Artifact Overview)

Available artefacts for replication:

1. Strategy_Master_v4.0_FINAL
6. Canonical_Numbers_Sheet_v2.0_FINAL
7. Life_System_Master_v1.1_FINAL_with_File_Registry
8. Drift-Diagnostics Logs
9. Checksum Registry Snapshots
10. Audit Templates
11. A-Control Algorithm Code Snippets
12. Compaction Digests

Supporting materials: Supplementary materials supporting this work are available upon reasonable request.

Figures

Figure 1 — Authoritative Files Architecture

Directed acyclic graph of canonical flow
Strategy → Canonical → Execution → Audit → Freeze.
Each edge represents controlled truth-propagation; each node corresponds to an execution gate (G1–G10).

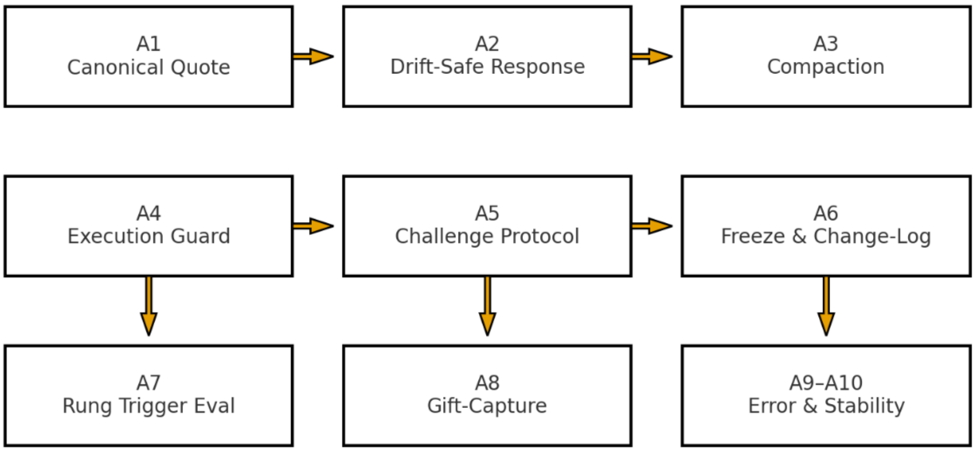
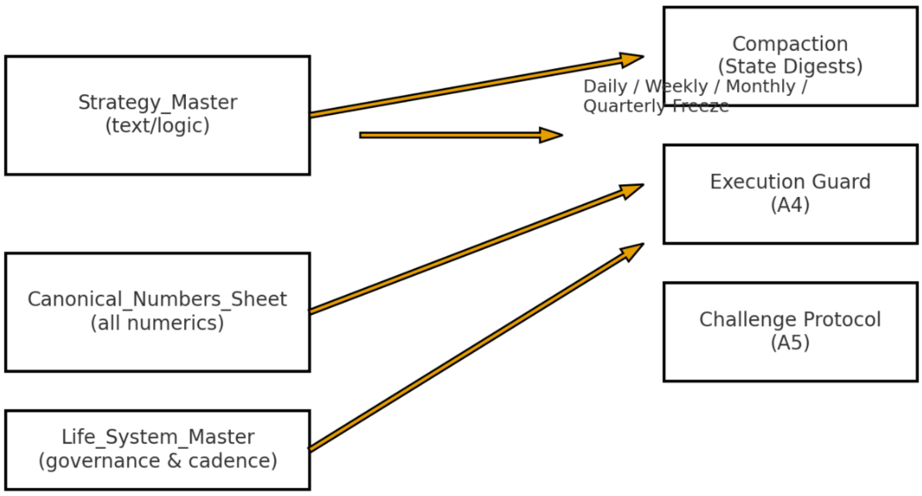


Figure 2 — Control-Stack Interactions

System diagram illustrating dependencies among A-controls.
G1/G2 feed G4; G6 and G9 form feedback loops; G10 overlays human-stability layer.


Strategy_Master
(text/logic)
Canonical_Numbers_Sheet
(all numerics)
Life_System_Master
(governance & cadence)
Compaction
(State Digests)
Daily / Weekly / Monthly /
Quarterly Freeze
Execution Guard
(A4)
Challenge Protocol
(A5)