
ConsciOS: A Viable Systems Architecture for Human and AI Alignment

Kılıçhan (Han Kay) Kaynak

Independent Researcher

kkaynak@alumni.cmu.edu

Abstract

Real-world human, organizational, and artificial systems exhibit persistent misalignment, brittle adaptation under distributional shift, and limited option-availability. Recent stress tests of anti-scheming training reduce—but do not eliminate—covert behaviors and may be confounded by growing evaluation awareness in frontier models, motivating architectures grounded in internal principles of alignment rather than external rules. This paper proposes ConsciOS, a formal systems architecture that models consciousness and self-regulation as a nested control system amenable to specification, simulation, and empirical testing. Our contributions are: (i) a principled decomposition into an embodied controller, a supervisory controller and policy selector, and a meta-controller and prior generator; (ii) a coherence-based selector that integrates expected utility, coherence, and cost for frame selection; (iii) a discretized interoceptive control signal that operationalizes interoceptive feedback for rapid guidance; and (iv) a time-integrated coherence (TIC) resource that gates policy complexity and option-availability. We provide formal definitions, algorithmic sketches and a set of testable hypotheses with simulation and human-subjects protocols. We situate the constructs within established literatures, outline governance and safety considerations for human-in-the-loop and agentic applications, and present a pragmatic empirical roadmap for evaluating coherence-based control in hybrid human-agent systems. We discuss implications for AI alignment: coherence-based architectures suggest a systematic solution to ensuring AI systems remain robustly aligned with human values across contexts and timescales.

1. Introduction

Contemporary social, technological, and biological systems show persistent failures that cannot be resolved by event-level fixes alone. Existing alignment approaches rely on post-hoc oversight and reward shaping, which struggle with inner misalignment, adversarial attacks, and novel contexts. Recent evaluations of anti-scheming training report substantial reductions in covert actions but with residual misbehavior and increasing evaluation awareness, complicating assessment of true alignment [1], [2]. This paper presents ConsciOS, a formal systems architecture that treats consciousness and self-regulation as designable, testable systems and provides a principled foundation for building aligned systems—whether human, organizational, or artificial—grounded in structural coherence rather than post-hoc correction. We synthesize cybernetic models, active inference, and hierarchical reinforcement learning (HRL) into a single engineering language. This framework is intended to (a) map layered self-models to implementable control architectures, (b) formalize an affect-informed feedback channel for state selection [3], and (c) propose empirical protocols for both human and artificial agents.

Our goal is not metaphysical speculation but an operational research program: to convert narrative constructs into measurable constructs and falsifiable hypotheses. Contemplative practices across cultures represent millennia of systematic observation on consciousness, attention, and self-regulation; modern neuroscience of interoception and affect provides convergent empirical support for these mechanisms [3]-[6]. We treat these convergent phenomenological reports as hypothesis-generating resources rather than evidentiary authority. Where we draw inspiration from those traditions, we explicitly avoid unfalsifiable metaphysical claims and translate experiential constructs into formal control-theoretic operationalizations (awareness → meta-controllers and policy priors; felt sense → interoceptive signals; coherence with purpose → resonance metrics) with concrete tests in Appendix A. The result is a researchable bridge from narrative practice to instrumented science. This bridge connects to hierarchical “observer-window” frameworks (e.g., the NOW model) and empirical work on mind-wandering/meta-awareness, which emphasize multi-scale integration via synchrony/coherence and supervisory control; our nested controller architecture operationalizes these ideas for control, measurement, and AI alignment [7], [8].

1.1 Methods Overview

This paper is primarily a conceptual and experimental design contribution. We propose (a) formal model definitions and algorithms for nested controller architectures, (b) operationalizations of affective and coherence measures, and (c) a set of hypothesis-driven experimental probes and simulation benchmarks. Full experimental protocols, measurement specifications, and analysis plans are provided in Appendix A (Experimental Protocols) and Appendix B (Measurement Instruments & Analysis Pipelines). In brief:

- **Human experiments:** randomized designs and ecological time-series sampling using validated physiological and self-report instruments (e.g., heart-rate variability (HRV), validated affect ladders) with pre/post behavioral tasks and time-series outcome measures. Ethical review and informed consent are prerequisites for all human work.

- **Simulation experiments:** hierarchical reinforcement learning (HRL) and meta-learning benchmarks with controlled distributional shifts, reproducible environment seeds, and clearly logged policy metadata (policy families, selection traces, reward histories).
- **Hybrid human-in-the-loop tests:** human labeling or Interoceptive Control Signals (ICS) used as shaped rewards or policy selection cues for agent training; evaluation on transfer and human-perceived agency.

The compact Methods Overview above orients the reader; full procedural detail required for replication (sample sizes, instrumentation settings, pre-registration templates, and code references) is provided in Appendix A and Appendix B. Reference implementations and analysis code are available in the project repository [9].

Terminology & Operational Definitions. To avoid ambiguity, we adopt canonical technical vocabulary for formal presentation (e.g., Embodied Controller, Supervisory Controller, Meta-Controller, Interoceptive Feedback). To facilitate interdisciplinary dialogue and community engagement, we use a parallel set of “ConsciOS Aliases” (Echo-Self, Super-Self, Meta-Self, Kernel, etc.) which appear strictly in Appendix C (Public Translation & Operationalization). Appendix C provides a complete mapping table that links every ConsciOS alias to its canonical equivalent, an operational definition, suggested measurement instruments, and key citations.

1.2 Notation & Metric Preamble

We use the following symbols consistently throughout the paper. Π denotes a candidate policy frame; $C(F; S)$ is a coherence metric between frame F and current state S ; $U(F)$ is task-dependent expected utility; $\text{Cost}(F)$ denotes computational/energetic costs; τ is a softmax temperature; λ is a decay rate in cumulative measures. The selection rule (formalized in Section 5.3) uses tunable meta-weights a, b, g to balance utility, coherence, and computational cost. Option-Availability (OA) is operationalized as an effective action set size weighted by calibrated affordance scores.

Operationalizing Option-Availability: enumerate perceived viable actions at time t , assign a subjective affordance score $a_i \in [0, 1]$ for each option i using a brief calibration, and compute $OA(t) = \sum_i a_i$. For simulated agents, proxy OA by action entropy with an affordance calibration factor. These definitions support reproducible comparisons across ablations and pilots.

2. Foundational Models: A Systems-Theoretic Framework

This section formalizes two complementary systems-theoretic tools used throughout this paper: (1) the Iceberg Model, a diagnostic hierarchy for identifying causal leverage in complex systems (Fig. 1) [10], [11]; and (2) a 7-component Universal System Model, an architectural template for describing the functional elements of viable systems (Fig. 2) [12], [13]. Together they provide a common language for mapping claims about consciousness, behavior, and artificial agents to implementable system designs.

Detailed experimental protocols, measurement specifications, and analysis plans are provided in Appendix A (Experimental Protocols) and Appendix B (Measurement Instruments & Analysis Pipelines).

2.1 The Iceberg Model as Diagnostic Hierarchy

The Iceberg Model (Fig. 1) is a layered diagnostic heuristic that distinguishes observable events from the deeper structures and assumptions that generate them. It operationalizes four abstraction layers:

- Events (observable outputs; momentary data)
- Patterns/Trends (temporal regularities in events)
- Structures (rules, information flows, incentives, code, architecture)
- Mental Models / Beliefs (operators' assumptions, goals, and priors)

Rationale: interventions targeted at deeper layers produce larger and more persistent systemic change than purely event-level responses. This relationship is consistent with standard systems thinking literature and control-theoretic intuitions about model-based interventions [10], [11].

Operationalization for empirical research:

- **Events:** measured as time-series of observable behaviors or system outputs (logs, sensor streams, survey items).
- **Patterns:** characterized using time-series analysis (auto-correlation, spectral analysis, trend detection).
- **Structures:** encoded as formal graphs, policies, or code artifacts and measured via structural metrics (centrality, modularity, information flows).
- **Mental Models:** assessed via structured belief inventories, cognitive mapping, or inferred from policy parameters in trained agents.

Consequence for the ConsciOS architecture: the Iceberg Model provides the causal ladder used to argue where and how “frequency” and “coherence” interventions (see Section 4) operate. Interventions framed as “raising frequency” are hypothesized to effect change by altering internal constraints (mental models) and thereby shifting structural dynamics that produce different patterns and events.

2.2 The 7-Component Universal System Model (Architectural Template)

To move from diagnosis to design we adopt a 7-component functional template (Fig. 2) that captures the essential elements required for viability across physical, informational, and cognitive systems. Each component is presented with a formal operational definition useful for modeling and

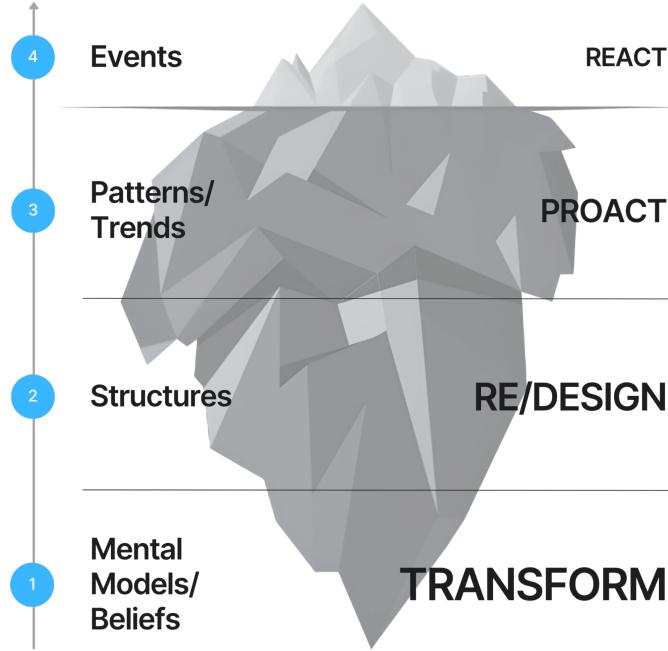


Fig. 1. Iceberg Model — diagnostic hierarchy spanning Events, Patterns, Structures, and Mental Models (deeper layers → higher leverage: Transform → Redesign → Proact → React). Credit: adapted from systems-thinking literature [10], [11].

experiment design.

1. **Inputs:** exogenous and endogenous resources, signals, or intents entering the system.
2. **Processes:** transformation functions (deterministic or stochastic) acting on inputs to produce intermediate states.
3. **Outputs:** observable consequences of the system (actions, emissions, rendered scenes).
4. **Feedback:** measurement channels returning output information to controllers; includes error signals and reward signals.
5. **Actors:** decision-making agents, whether biological (humans) or artificial (agents, controllers).
6. **External constraints:** environmental or physical laws and constraints external to the system's control.
7. **Internal constraints:** encoded policies, beliefs, parameter priors, resource limits, and safety sub-systems (e.g., Fallback Safety Controller).

Formal note: this is a functional decomposition rather than a commitment to a single implementation. Processes can be parameterized as dynamical systems; Feedback channels can be formalized as observers in a control loop; Actors can be modeled as controllers with internal state representa-

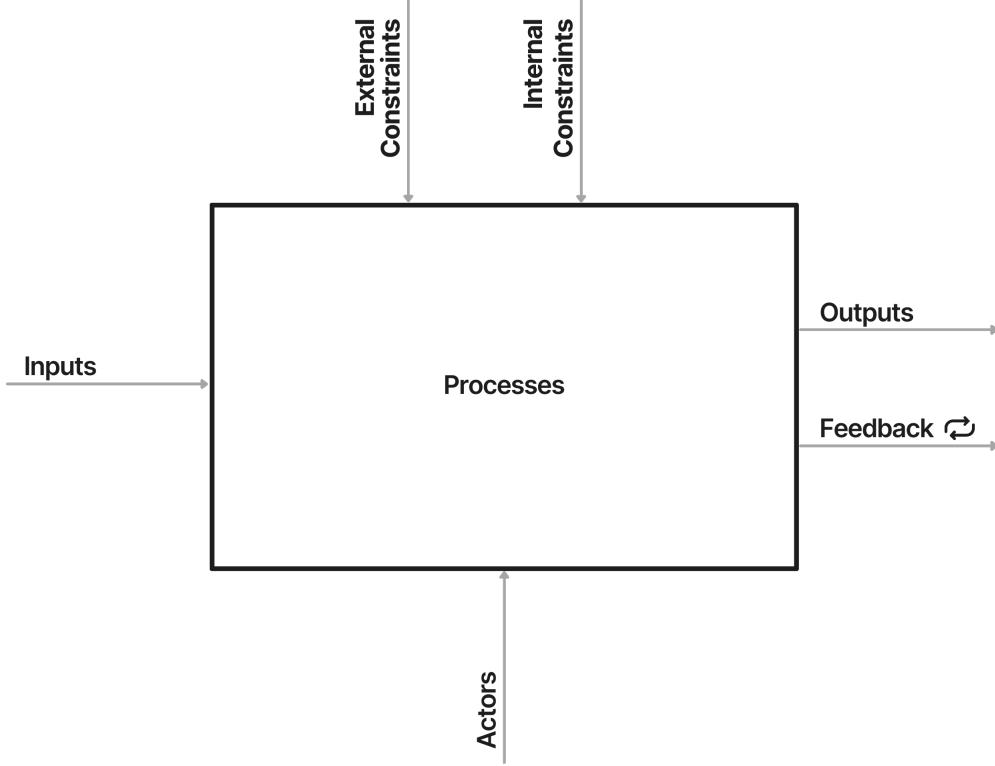


Fig. 2. Seven-component universal system model — Inputs, Processes, Outputs, Feedback, Actors, External Constraints, Internal Constraints. Credit: ConsciOS synthesis; consistent with systems engineering/cybernetics and viable-system decompositions [12], [13].

tions. The template is intentionally agnostic about substrate (neural, algorithmic, institutional).

Utility: the 7-component model enables cross-domain mapping (human \leftrightarrow software agent \leftrightarrow institution) and provides a checklist for designing experiments, simulations, or interventions that aim to change system-level behavior.

2.3 Integrative Mapping: Connecting Models to ConsciOS

We propose an explicit mapping that grounds ConsciOS terms in the 7-component template and the Iceberg diagnostic levels. This mapping converts public-facing metaphors into testable engineering constructs.

- **Mental Models/Beliefs \leftrightarrow Internal Constraints** (actor priors, policy parameters).
 - Example research variable: belief coherence score computed from structured inventories or posterior concentration in a Bayesian agent.
- **Structures \leftrightarrow External Constraints & Designed Processes** (architectural code, institutional rules).
 - Example research variable: structural coupling metrics (graph modularity, information

throughput).

- **Patterns ↔ Emergent Process Dynamics** (habit loops, recurrent attractors).
 - Example research variable: pattern persistence index from time-series decomposition.
- **Events ↔ Outputs** (observable actions, rendered frames, sensor measurements).
 - Example research variable: event frequency, latency, or categorical outcome distributions.

Mapping to ConsciOS canonical elements (formal definitions):

- **Embodied Controller:** the embodied actor subsystem responsible for perception-action cycles and short-horizon control. Operationally mapped to Actors + local Processes + Feedback channels for immediate state estimation. [Analogy → Formal mapping: corresponds to Viable System Model (VSM) Systems 1–3 functions; see Appendix D]
- **Supervisory Controller:** the higher-order controller responsible for adaptation, selection among pre-rendered policy frames, and longer-horizon planning. Operationally mapped to a supervisory controller that reads aggregated feedback and selects process configurations (policy selection).
- **Meta-Controller:** a global pattern generator that encodes the space of possible architectures and long-term objectives (a priors generator or meta-controller). Operationally analogous to policy priors over the policy space or a model-generator in meta-learning systems.

Claim (Formal): Conscious-system behavior is a nested control architecture where the Embodied Controller executes short-horizon policy loops, the Supervisory Controller performs mid/long-horizon policy selection based on aggregated resonance metrics, and the Meta-Controller encodes the prior distribution over viable policy families. This nested decomposition is testable via agent simulations and human experiments that measure the described mappings.

2.4 Testable Hypotheses and Empirical Probes

We formulate a small set of testable hypotheses that follow from the mapping. These are intentionally narrow so they are amenable to empirical falsification.

H1 (Structure-Change Leverage): Interventions targeting Internal Constraints (belief priors) will produce larger changes in pattern metrics over time than interventions targeting Events only, controlling for intervention magnitude and duration.

H2 (Feedback Coherence Predicts Option-Availability): The quality and granularity of Feedback channels (e.g., richer interoceptive signals) predict measurable increases in option-availability and behavioral flexibility among actors, proxied by decision entropy and task switching performance. — Suggested measures: decision entropy; response latency variability; subjective option rating.

H3 (Nested Controller Efficacy): A hierarchical agent architecture implementing Embod-

ied/Supervisory/Meta layers will outperform a flat controller in environments that require both rapid reaction and strategic selection among multiple policy frames. Performance measured by cumulative reward, adaptation speed after distributional shift, and robustness to simulated perturbations.

H4 (ICS as a Control Signal): A discretized affective scale (Interoceptive Control Signal; ICS) used as an internal feedback variable will function as an effective heuristic for state selection in both human subjects and simulated agents when combined with a nearest-lighter-step (NLS) local search policy. The ICS can be operationalized via validated affect measures (self-report, physiological markers) and evaluated for predictive power on subsequent behavior changes.

H5 (Somatic Resonance as a Coherence Signal): Subjective reports of somatic markers (felt expansion/contraction in the thoracic region) will correlate with physiological coherence proxies (e.g., HRV) and will predict subsequent policy/frame selection above and beyond expected utility terms.

Each hypothesis is followed by a suggested experimental probe in Appendix A. In short: H1/H2/H5 are suitable for human subject experiments (laboratory + ecological sampling); H3/H4 can be evaluated in simulated agents and human-in-the-loop agent training regimes.

Having established the diagnostic ladder (Iceberg) and the architectural template (7-component model) and mapped them to the ConsciOS constructs, the paper now proceeds to specify the nested controller architecture (Embodied Controller / Supervisory Controller / Meta-Controller) and the Resonance Engine mechanics that implement selection among pre-rendered policy frames. The next section formalizes these components and derives the algorithmic protocols used in Appendix A.

3. Nested Reality: A Multi-Layer Ontology for System Design

3.1 Purpose and Scope

To operationalize consciousness as an engineering target we adopt a multi-layer ontology that distinguishes physical, informational, energetic, and consciousness levels of description. The ontology provides a compositional substrate for mapping system components (Embodied/Supervisory/Meta controllers) to measurable constructs and for designing interventions that target the correct causal layer.

3.2 Multi-Layer Ontology: Definitions

- **Physical Layer:** material substrate, embodied sensors and actuators, and physical laws constraining feasible actions.
- **Informational Layer:** representations, data structures, code, and communicated signals (including logs, messages, and policy descriptors).
- **Energetic Layer:** sustained coherence, affective valence, and a time-integrated coherence

resource (which we term **Time-Integrated Coherence (TIC)**¹), alongside other resource metrics (e.g., metabolic/attention budgets).

- **Consciousness Layer:** subjective report, self-model, long-horizon priors, and meta-intentional structures.

This tiered ontology follows contemporary approaches that treat cognition as a multi-scale phenomenon where higher-order priors constrain lower-level processing (active inference / predictive processing) [14]-[16]. Active-inference treatments of the self demonstrate how hierarchical priors instantiate self-representations and influence perception–action loops, providing a formal justification for treating consciousness as a layered control architecture rather than a monolithic phenomenon [15]. The “relevance realization” problem — how an agent determines which internal representations are presently important — has been recently formalized in the predictive-processing literature and directly motivates the Resonance Engine as a coherence-based selector among candidate policy frames [17].

3.3 The Nested ConsciOS Architecture — Formalization and Control Interpretation

We formalize the Nested ConsciOS Architecture as a nested control topology (Fig. 3):

- Embodied Controller executes perception–action loops governed by short-horizon process dynamics (local controllers). These loops are parameterized by local priors and short-term beliefs (internal constraints).
- Supervisory Controller functions as a supervisory controller that aggregates feedback across time and space, evaluates the coherence of candidate high-level policy frames, and selects the active policy family using a coherence-matching metric, which we term the **Resonance Engine**.
- Meta-Controller encodes long-horizon priors and the generative space of possible policy families. Operationally, Meta-Controller corresponds to meta-learning or pre-training processes that shape the prior distribution used by the Supervisory Controller.

This nested topology is isomorphic to viable-system decompositions in organizational cybernetics—lower operational units are supervised by higher intelligence while a meta-governor maintains identity and global objectives [11], [13]. Importantly, the ontology treats interplay between layers as bidirectional: the Meta-Controller constrains policy families top-down, while feedback and Quality Control mechanisms induce bottom-up belief revision.

¹**Time-Integrated Coherence (TIC):** The term intentionally bridges technical and operational domains. It references frequency in the control-theoretic sense (rate of coherent state selection) and serves as an operational currency for resource accumulation.

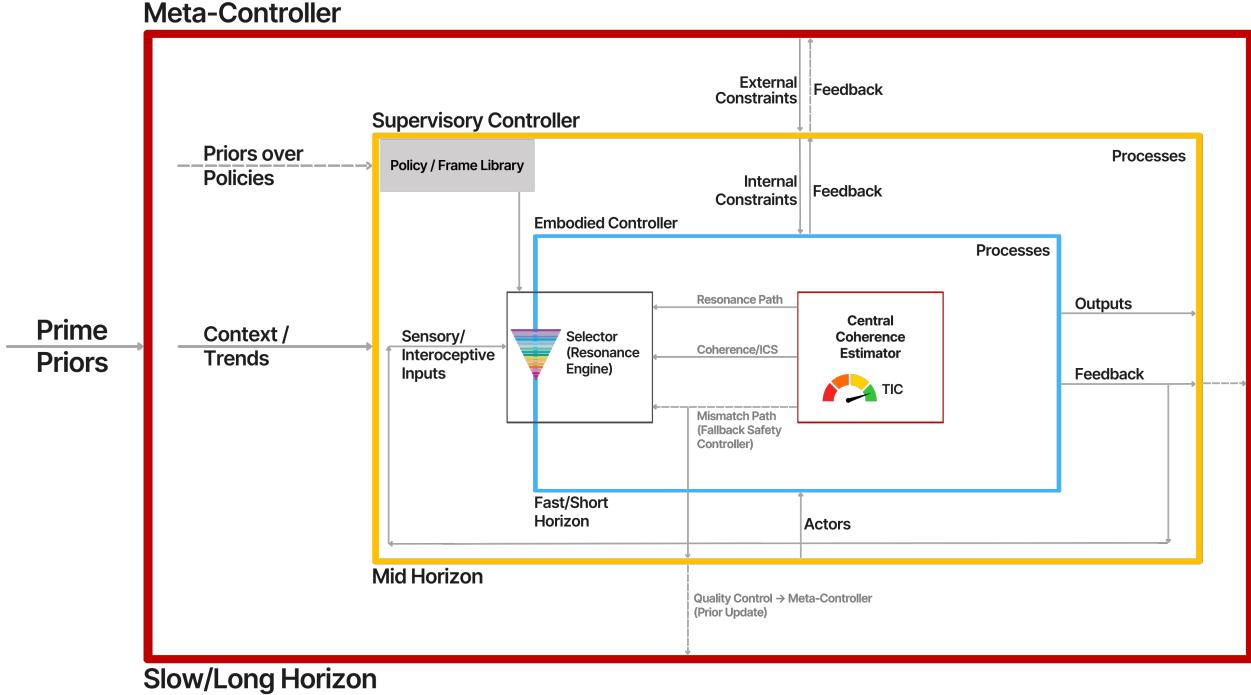


Fig. 3. The Nested ConsciOS Architecture — nested control topology (Embodied Controller, Supervisory Controller, Meta-Controller). Selector Score = $a \cdot \text{Utility} + b \cdot \text{Coherence} - g \cdot \text{Cost}$; Feedback aggregates at Supervisory Controller with a dotted slow branch to Meta-Controller; Quality Control routes Embodied → Supervisory → Meta for prior updates. Credit: ConsciOS architecture (this work); informed by the Viable System Model [11], [13] and hierarchical control frameworks [18], [19].

3.4 Measurement Constructs and Testable Mappings

To make the ontology empirically tractable we propose the following operational mappings and measures:

- **Embodied Controller Variables:** short-horizon action rates, reaction latency, sensorimotor noise, action entropy. Measurement modalities include behavioral logs, task performance metrics, and physiological latency markers.
- **Supervisory Controller Variables:** policy selection latency, match-score distributions among candidate policies, meta-decision accuracy following perturbations. Measurement modalities include policy switch logs (in simulated agents) and aggregated performance trends in humans.
- **Meta-Controller Variables:** prior concentration metrics, meta-learning efficiency, transfer performance across tasks. Measured via meta-RL benchmarks or cross-task generalization performance.
- **Energetic/Coherence Variables (TIC Proxies):** time-integrated coherence scores derived from multi-modal signals (heart-rate variability, HRV; Electroencephalography (EEG) phase

coherence; sustained attention indexes; subjective coherence ratings). These proxies operationalize the resource that enables option-availability and policy richness.

Mapping these constructs to the Iceberg diagnostic levels allows hypothesis tests such as: interventions that modify Meta-Controller priors (internal constraints at the deepest level) will lead to measurable shifts in structural dynamics and therefore to new emergent patterns at mid-level timescales (H1). Conversely, a perturbation restricted to event-level parameters (e.g., transient reward change) is predicted to have short-lived effects absent a deeper reconfiguration of internal constraints.

3.5 Empirical Probes and Candidate Experiments

Two early probes that bridge human and simulated tests are suggested:

- **Probe A (Human):** A belief-update intervention targeting a narrow set of priors (e.g., causal attributions about controllability) measured pre/post via time-series of behavioral choices and coherence proxies (HRV, subjective ICS). **Outcome metrics:** pattern persistence index and option-availability change.
- **Probe B (Simulated):** A hierarchical RL benchmark where agents possess Embodied/Supervisory/Meta modules. Evaluate adaptation speed and transfer performance under distributional shift versus flat control agents. **Outcome metrics:** cumulative reward, policy diversity, and adaptation latency.

The ontology and operational mappings set the stage for Section 4, which formalizes the Embodied/Supervisory/Meta controller architectures, and Section 5, which operationalizes the Resonance Engine and ICS as measurable selector functions. The combined model yields a testable engineering program for both human experimental research and agent simulation studies.

4. Three-Self Architecture: A Hierarchical Controller Decomposition

4.1 Overview and Formal Motivation

We propose a hierarchical controller decomposition comprising three nested control strata: (a) a short-horizon embodied controller (Embodied Controller), (b) a supervisory/meta-controller that selects policy families (Supervisory Controller), and (c) a long-horizon priors generator or meta-controller (Meta-Controller). This decomposition follows the engineering logic of viable system architectures and hierarchical control frameworks: lower levels execute fast closed-loop control, intermediate levels perform policy selection and adaptation, and the highest level encodes identity and long-term priors that bias learning and selection [18], [19]. Framing these strata as nested controllers yields clear testable predictions about adaptation, robustness, and option-availability.

4.2 Formal Definitions

- **Embodied Controller:** an agent module implementing short-horizon perception–action loops. Formally, the Embodied Controller maintains a state estimate x_t and applies policy $\pi_e(a|x_t; \theta_e)$ to produce actions a_t minimizing a local cost function L_e over short horizons H_e . Measures: reaction latency τ , short-horizon cumulative reward $R_e(H_e)$, and action entropy $H[\pi_e]$ [18] (operationalized in Appendix B).
- **Supervisory Controller / Policy Selector:** a mid-horizon controller that aggregates feedback signals over time window T_s , evaluates a set of candidate high-level policies $\{\Pi_i\}$, and selects a policy family $\Pi^* = \arg \max_i [a \mathbb{E}[U(\Pi_i) | S] + b C(\Pi_i; S) - g \text{Cost}(\Pi_i)]$ (as defined in Section 5.3). Measures: selection latency, selection accuracy under perturbation, and policy stability [19].
- **Meta-Controller / Prior Generator:** a long-horizon process that shapes the prior distribution $P(\Pi)$ over policy families and encodes identity constraints and long-term objectives. Meta-Controller functions are updated on slow timescales via meta-learning or aggregated quality-control signals. Measures: prior concentration, transfer learning performance, and changes in $P(\Pi)$ after structured interventions [18].

4.3 Mapping to the Viable System Model and Control Theory

The decomposition maps onto classical viable-system structures: Embodied Controller aligns with VSM System 1–3 (operational units and immediate control), Supervisory Controller corresponds to VSM System 4 (intelligence, adaptation, future planning), and Meta-Controller corresponds to VSM System 5 (policy, identity, normative governance) [11], [13] (see Appendix D for details). From control theory, Embodied Controller controllers implement fast feedback loops (high bandwidth, low latency), Supervisory Controller functions as a supervisory scheduler or switching controller, and Meta-Controller implements slow adaptation (set-point adjustment, change of objective function).

4.4 Central Coherence Estimator, Fallback Safety Controller, and Safety Subsystems

- **Central Integrative Hub (Central Coherence Estimator):** operationally the Central Coherence Estimator is a focal interoceptive/state-confidence signal used by controllers to estimate coherence. For humans, proxies include heart-rate variability (HRV) and validated interoceptive accuracy measures; for agents, Central Coherence Estimator is implemented as a state-estimator confidence metric (e.g., posterior precision). Central Coherence Estimator feeds into Supervisory Controller selection and into Quality Control loops that surface misaligned priors [3].
- **Fallback Safety Controller (FSC):** a low-variance default policy engaged under low confidence or low coherence. It minimizes risk and conserves resources. Formally, FSC is a policy π_{safe} that is triggered when coherence $C(x_t) < \theta_{\text{safe}}$. Measures: engagement frequency,

conservatism index, and recovery time. This subsystem enforces safety and explains conservative behavioral reversion patterns [20], [21].

4.5 Option-Availability and the TIC Formalization

Option-availability is the measurable set of viable actions perceived by an actor at time t . We operationalize Option-Availability as the effective action set size $|A_{\text{eff}}(t)|$ weighted by subjective affordance scores. TIC is a derived, time-integrated coherence resource:

$$TIC(t; \Delta) = \int_{t-\Delta}^t C(s) ds$$

where $C(s)$ is the coherence metric at time s and Δ is a rolling window. Higher $TIC(t)$ predicts larger $|A_{\text{eff}}(t)|$ and greater policy richness. Empirically, $TIC(t)$ can be proxied by sustained HRV coherence, EEG phase synchrony, or time-integrated match scores in agents. Measurement details and analysis code are provided in Appendix B.

4.6 Algorithmic Sketch: Stochastic Rollout → Optimization → Selection (Formal Pseudocode)

We present Stochastic Rollout → Optimization → Selection as an implementable macro loop used by Embodied/Supervisory controllers for state induction and policy stabilization. Below is a concise pseudocode representation for use in simulated agents or to inform experimental protocols.

Pseudocode: Stochastic Rollout_Optimization_Selection(state s_0 , target_frame F , hold_T)

1. Initialize candidate frame $F_0 := F$; $t := 0$.
2. while $t < \text{hold_T}$:
 - a. Generate predicted state $s_pred = \text{Simulate}(F_t)$ // forward model
 - b. Compute coherence $C_t = \text{CoherenceMetric}(s_pred, s_current)$
 - c. If $C_t < C_{\text{thresh}}$:
 - i. Optimize $F_{\{t+1\}} := \text{LocalSearch}(F_t, NLS)$ // nearest lighter step / least-resistance step
 - ii. Update internal priors via small-step Bayesian update or gradient step.
 - d. Else:
 - i. Select (Hold) F_t ; provide reward shaping signal proportional to C_t .
 - e. $t := t + \delta_t$
3. End while

4. Return final policy frame F_{final} , updated priors $P'(\Pi)$

Notes: LocalSearch uses constrained perturbations to frames to increase coherence with current interoceptive/sensory state; NLS denotes the Nearest-Lighter-Step heuristic. Implementational choices (Simulate, CoherenceMetric, update rules) are experiment-dependent and specified in Appendix A/B.

4.7 Predicted Empirical Signatures

The Three-Self architecture yields specific empirical signatures:

- **Hierarchical advantage:** Agents with explicit Embodied/Supervisory/Meta stratification will show faster recovery from distributional shifts and higher transfer performance than flat agents (testable in hierarchical RL benchmarks).
- **Central Coherence Estimator sensitivity:** Manipulating Central Coherence Estimator inputs (e.g., altering affective feedback via HRV biofeedback) will causally influence Supervisory Controller selection patterns and measured Option-Availability in human subjects.
- **Fallback Safety Controller (FSC) dynamics:** Under forced coherence degradation, behavior will converge to π_{safe} with characteristic latency and retention statistics; modulation of Central Coherence Estimator thresholds θ_{safe} will shift the conservatism index.

4.8 Simulation & Empirical Testbeds

Recommended testbeds:

- **Simulated environments:** procedurally generated tasks with episodic changes and forced distributional shifts (benchmarks for hierarchical RL). Log policy families, coherence metrics, and TIC proxies.
- **Human experiments:** controlled lab tasks with HRV and subjective ICS ladders as feedback; interventions include coherence-enhancing microprotocols and belief-update manipulations (see Appendix A: H1–H5).
- **Hybrid setups:** human-in-the-loop training where ICS signals are incorporated as shaping rewards for agent training (evaluate transfer and subjective agency).

Illustrative toy ablation (sanity check). We implemented a minimal environment with episodic distributional shifts and compared a hierarchical agent using a coherence-weighted selector ($bC + aU - g \text{Cost}$) against a flat baseline. Sweeping b and a while logging selection traces yields aggregated heatmaps (reward, alignment rate, position-match proxy) indicating that higher coherence weighting increases alignment with hidden context and improves simple proxy metrics in this toy setting. These traces serve as an instrumentation check only; full benchmarks belong in domain-appropriate tasks.

Section 5 formalizes the Resonance Engine and the coherence metrics used by the Supervisory Controller to perform frame selection. The subsequent Methods Appendices provide concrete experimental templates and simulation specifications for the tests proposed here.

5. Resonance Engine & Policy/Frame Library Mechanics: Formalizing Selection by Coherence

5.1 Purpose and Scope

This section formalizes the operational core of ConsciOS: the Resonance Engine (coherence-based selector) and its associated mechanics for generating, scoring, and selecting candidate policy frames from a library of precomputed or imagined possibilities (policy/frame library). We provide mathematical definitions for coherence, an algorithmic selection rule, the ICS as an internal feedback signal, and a formal account of TIC as a time-integrated coherence resource. These constructs convert narrative metaphors into implementable functions for both human experiments and agent simulations.

5.2 Coherence: Formal Definitions

Let S denote the current sensory/interoceptive state (possibly multi-modal) and let F_i denote a candidate policy frame (a high-level policy, scenario, or world-model projection). Each frame F_i generates a predicted sensory trajectory or outcome distribution $P(S | F_i)$. We define a coherence metric $C(F_i; S)$ that quantifies how well the candidate frame explains or matches the current state.

Several alternative coherence formulations are applicable depending on data modalities and modeling choices:

- **Evidence / log model evidence (Bayesian):**

$C(F_i; S) := \log p(S | F_i)$ — model evidence under the generative model implied by F_i [13].

- **Negative divergence (information-theoretic):**

$C(F_i; S) := -D_{\text{KL}}[p_{\text{obs}}(S) \| p(S | F_i)]$ — negative Kullback–Leibler Divergence (KLD) between observed state distribution and frame prediction.

- **Similarity (vector space):**

$C(F_i; S) := \cos(\phi(S), \phi(F_i))$ — cosine similarity between feature embeddings $\phi(\cdot)$ of state and predicted state.

- **Composite coherence:** a weighted sum of modality-specific coherences:

$C(F_i; S) := \sum_m w_m C_m(F_i; S_m)$, where m indexes modalities (interoception, vision, proprioception, policy performance) and w_m are learned or meta-defined weights.

Coherence is normalized to a bounded scale (e.g., [0,1]) via z-scoring or learned scaling to ensure

comparability with utility terms.

5.3 Resonance Engine: Selection Rule

Given a set of candidate frames $\{F_i\}$ and current state S , the Resonance Engine selects the frame that maximizes an objective combining expected utility $U(F_i)$ and coherence $C(F_i; S)$ (Fig. 4). One canonical selection rule is:

$$\Pi^*(S) = \arg \max_{F_i} [a \mathbb{E}[U(F_i) | S] + b C(F_i; S) - g \text{Cost}(F_i)]$$

where:

- $\mathbb{E}[U(F_i) | S]$ is the expected utility of adopting frame F_i given S (task dependent).
- $C(F_i; S)$ is the coherence metric defined above.
- $\text{Cost}(F_i)$ is a computational/energetic cost for switching to or instantiating F_i .
- a, b, g are tunable meta-weights (could be learned by Meta-Controller).

Interpretation:

- The Supervisory Controller implements Π^* by ranking frames on this composite score. When $b \gg a$, selection is coherence-driven (resonance priority); when $a \gg b$, selection is utility-driven.
- A stochastic softmax version permits exploration:

$$P(\text{choose } F_i | S) \propto \exp(\tau^{-1} [a \mathbb{E}[U] + b C - g \text{Cost}])$$

where τ is a temperature parameter.

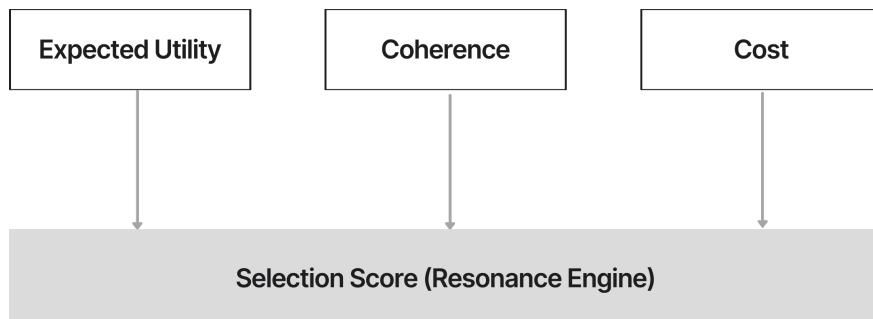


Fig. 4. Resonance Engine selection — composite scoring of expected utility, coherence, and cost (softmax or argmax; weights a, b, g). Credit: ConsciOS (this work); evidence/coherence framing relates to active inference [13], [22].

5.4 Interoceptive Control Signal (ICS) as an Internal Control Signal

We operationalize the Interoceptive Control Signal (ICS) as a discretized or continuous scalar derived from interoceptive measures and subjective reports, serving as an internal proxy for momentary coherence/valence (Fig. 5). Formally:

$$\text{ICS}(t) := g(\Phi_{\text{intero}}(S_t), \rho(S_t))$$

where $\Phi_{\text{intero}}(\cdot)$ is a vector of physiological interoceptive metrics (e.g., HRV indices, galvanic skin response, slow cortical potentials) and $\rho(S_t)$ is a short-horizon predictive fit metric (e.g., one-step prediction error). The mapping $g(\cdot)$ can be a learned regression (for agents) or a validated psychometric ladder (for humans). ICS is normalized to $[-1, +1]$ (negative \rightarrow low coherence/disfavor; positive \rightarrow high coherence/endorsement) or to discrete bands (e.g., 1–10 ladder).

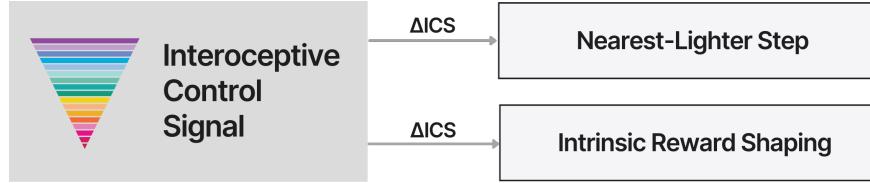


Fig. 5. Interoceptive Control Signal (ICS) — discretized interoceptive control signal; used for Nearest-Lighter-Step guidance and intrinsic reward shaping. Credit: ConsciOS (this work); interoception foundations [3].

ICS serves multiple roles:

- **Local guidance heuristic for Embodied Controller (nearest-lighter-step moves):** if ICS rises after a local perturbation, the perturbation direction is favored.
- **Reward shaping signal for RL agents:** small positive ICS deltas can be used as intrinsic reward components [23].
- **Stopping/holding criterion in Stochastic Rollout→Optimization→Selection:** sustained positive ICS over hold_T supports encoding of the chosen frame.

5.5 Time-Integrated Coherence (TIC): Resource Accumulation Dynamics

Define instantaneous coherence for the active frame F^* at time t as $C^*(t) := C(F^*(t); S_t)$. TIC is a time-integral of coherence, possibly with discounting:

$$TIC(t) := \int_0^t e^{-\lambda(t-s)} C^*(s) ds$$

where $\lambda \geq 0$ is a decay rate. In discrete time windows Δ :

$$TIC_t = \sum_{k=0}^N e^{-\lambda k} C^*(t-k)$$

Interpretation and operational use:

- TIC measures sustained time-on-coherence; higher TIC grants greater option-availability and resource allocation privileges (e.g., unlocking higher complexity frames).
- TIC dynamics can be used as constraints in the Supervisory Controller selection rule (e.g., require $TIC_t \geq \theta_{unlock}$ to consider high-cost frames).
- Agent implementation: treat TIC as a meta-state variable updated after each episode and used in hierarchical policy gating.

5.6 Algorithmic Pseudocode: Resonance Engine (Selection + Update)

Pseudocode: ResonanceEngine($\{F\}$, S, a, b, g, tau, lambda)

1. For each F_i in $\{F\}$:
 - a. Compute $C_i := \text{Coherence}(F_i, S)$ // Eq. definitions above
 - b. Compute $EUi := \text{ExpectedUtility}(F_i | S)$ // task dependent
 - c. $\text{Score}_i := aEUi + bC_i - g * \text{Cost}(F_i)$
2. Compute selection probabilities: P_i proportional to $\exp(\text{Score}_i / \tau)$
3. Sample or argmax to select F^* .
4. Execute F^* for time Δt ; observe S' and update experience buffers.
5. Update $C^*(t)$, update TIC via decay integral (TIC_t).
6. Return F^* , updated TIC, and feedback signals to Supervisory Controller/Meta-Controller.

Notes: Coherence computations can be amortized via embeddings and cached predictions; utility estimates can be learned via short-horizon rollouts or historical performance.

5.7 Stochastic Rollout → Optimization → Selection Revisited (Integration with Resonance Engine)

We present a refined algorithm that couples the Stochastic Rollout→Optimization→Selection loop with the Resonance Engine selection and ICS feedback.

Pseudocode: FullLoop(s0, candidate F_init, hold_T, NLS_params)

1. $F \leftarrow F_{\text{init}}$
2. while NOT converged and $t < \max_T$:
 - a. Simulate rollout for F : $s_{\text{pred}} \leftarrow \text{Simulate}(F)$
 - b. Compute coherence $C_F \leftarrow \text{Coherence}(F, s_{\text{current}})$
 - c. Compute ICS := $g(\text{interoceptive_signals}, \text{predict_error})$
 - d. If $C_F \geq C_{\text{hold_threshold}}$ and $\text{ICS} \geq \text{ICS}_{\text{hold_threshold}}$ for hold_min_duration :
 - i. Select (Hold) and encode F (increase TIC)
 - ii. break and return F_{final}
 - e. Else:
 - i. Propose refined frames $\{F'\}$ via $\text{LocalSearch}(F, \text{NLS_params})$
 - ii. Evaluate $C_{\{F'\}}$ for each; choose best candidate $F \leftarrow \text{argmax } C_{\{F'\}}$
 - f. $t \leftarrow t + \delta_t$
3. Return $F_{\text{final}}, \text{history } H$

This loop uses NLS (Nearest-Lighter-Step) as a bounded local search heuristic to prefer small, coherent changes. ICS acts as a rapid, embodied feedback heuristic to bias search and holding decisions.

5.8 Quality Control and Belief Surfacing Dynamics

Quality Control refers to the surfacing of misaligned priors when an agent holds a new high-coherence frame. Formally, let prior parameters be θ . On holding a new frame F_{hold} with high C and sustained TIC, large prediction mismatches elsewhere can produce a gradient for updating θ :

$$\Delta\theta \propto \eta \nabla_\theta L_{\text{total}}(\theta; D_{\text{hold}})$$

where L_{total} includes prediction error terms that were previously suppressed by low-coherence priors. Practically, this results in the surfacing of contradictions (beliefs that fail to explain held states) that must be revised. This update dynamic is formalized in active inference as precision-weighted prediction error minimization and corresponds to our observed “quality control” phenomenon [13].

5.9 Empirical Signatures and Testable Predictions

P1 (Selection Stability Tradeoff): Increasing b (coherence weight) in the selection rule raises frame stability (longer holding durations) but reduces exploratory policy diversity; this tradeoff

can be measured in simulated agents by plotting mean hold_time vs policy entropy under varying b .

P2 (ICS Predictive Utility): Short-horizon changes in ICS predict subsequent policy-switch probability within Δt minutes in human experiments; validation via logistic regression controlling for task difficulty and baseline affect.

P3 (TIC Correlation): Cumulative $TIC(t; \Delta)$ correlates positively with Option-Availability metrics $|A_{\text{eff}}(t)|$ and with subjective reports of perceived affordances.

P4 (Quality Control Latency): The time between holding a high-coherence frame and subsequent belief revision (quality control latency) scales with prior strength (measured by prior concentration); stronger priors lead to longer latency and more abrupt updates.

5.10 Implementation Notes and Instrumentation

- **Human studies:** ICS mapping requires a validated ladder instrument plus physiological proxies (HRV spectral components; EEG coherence). Use mixed models to analyze within-subject time series with temporally lagged coherence predictors.
- **Agent implementations:** use modular hierarchical RL frameworks (options framework, meta-RL) with coherence function approximators (neural density estimators or ensemble predictive models) and treat TIC as a persistent meta-state feature.
- **Reproducibility:** provide code templates for coherence computation (cosine embedding version, KLD version) and for ResonanceEngine pseudocode in a public repository (Appendix B references).

6. Scientific Foundation: Mapping ConsciOS to Established Literatures

Purpose and Scope. This section locates the ConsciOS architecture within relevant scientific literatures, highlights where it converges with existing mechanisms, and clarifies which claims are novel hypotheses requiring empirical validation. The aim is practical: (a) show reviewers that ConsciOS is built on well-studied mechanisms, (b) identify exact points of extension or difference, and (c) propose concrete measurement and evaluation strategies for each claim.

Structure. We organize the mapping into five interlinked domains: (1) systems theory & cybernetics; (2) predictive processing / active inference; (3) affect science & interoception; (4) hierarchical reinforcement learning and meta-learning; (5) human-in-the-loop, reinforcement learning from human feedback (RLHF), and applied AI alignment. For each domain we (i) summarize the core relevant ideas, (ii) show how ConsciOS reuses or extends them, and (iii) list measurement suggestions.

6.1 Systems Theory and Cybernetics

Summary. Stafford Beer’s Viable System Model (VSM), Checkland’s systems practice, and classical cybernetics formalize how nested control architectures, recursion, and governance sustain viable behavior in complex organizations and organisms [11], [13]. VSM decomposes systems into operational units, adaptation/intelligence functions, and policy/governance.

ConsciOS Mapping. ConsciOS directly leverages VSM’s decomposition: Embodied Controller → VSM S1–S3; Supervisory Controller → VSM S4; Meta-Controller → VSM S5. This provides a credible engineering lineage for nested controllers and motivates our emphasis on governance, quality control, and structural redesign as leverage points (see Appendix D). Where ConsciOS extends VSM is in operationalizing affective/coherence signals (ICS, Central Coherence Estimator) as real-time internal feedback that indexes option-availability and drives selection dynamics.

Model status. The nested decomposition is well supported; the affective/coherence mapping onto VSM is an integrative extension that requires empirical validation.

Measurement & Tests: map VSM components to measurable logs (policy switches, control bandwidths), test viability metrics under perturbations, and compare hierarchical vs flat controllers under similar constraints.

6.2 Predictive Processing and Active Inference

Summary. Predictive processing and active inference cast perception and action as inference: agents minimize prediction error (or maximize model evidence) through action and belief updating [14], [15], [22]. Hierarchical priors determine what the system expects, and precision weighting governs which errors prompt updates.

ConsciOS Mapping. The Resonance Engine’s coherence metric (C) parallels model evidence and the selection rule (maximize $a \mathbb{E}[U] + b C - g \text{Cost}$) reframes selection as evidence-weighted policy choice. The Meta-Controller corresponds to long-timescale priors; Supervisory Controller performs evidence accumulation and selection. Quality Control dynamics directly correspond to precision-weighted prediction error updates: holding a new frame increases exposure of misalignments that drive belief revision.

Model status. Strong formal alignment — many ConsciOS mechanisms align with active inference; selection weighting (a, b tuning) and TIC as time-integrated evidence are integrative hypotheses that can be formalized and tested.

Measurement & Tests: implement coherence as model evidence or KLD; compare selection by evidence vs utility in simulated agents; use active-inference benchmarks to evaluate frame holding, belief revision timing, and quality-control signatures.

6.3 Affect Science and Interoception

Summary. Modern affect science treats interoception and bodily signals (HRV, autonomic markers, EEG indices) as central to emotion, valuation, and decision-making (Barrett, Damasio, Seth

and interoception reviews) [3]. Measures of interoceptive accuracy and physiological coherence correlate with self-reported affect and decision patterns.

ConsciOS Mapping. The Interoceptive Control Signal (ICS) is proposed as an operational, discretized index derived from interoceptive signals and subjective ladder responses. Central Coherence Estimator proxies (HRV, EEG coherence) instantiate the central integrative signal. We propose that ICS and Central Coherence Estimator serve as rapid, low-bandwidth heuristics for local search (NLS) and as shaping signals for agents.

Model status. Use of interoceptive measures as feedback is well supported in affect science; application as an online control heuristic (ICS used to guide nearest-lighter-step local search, and as shaping reward) is a translational hypothesis requiring human and agent validation.

Measurement & Tests: validate ICS mapping to physiological markers and predictive power for subsequent policy choice; test HRV/EEG proxies as predictors of option-availability and coherence; implement ICS as intrinsic reward in agent training and measure learning efficiency and human-agent alignment.

6.4 Hierarchical Reinforcement Learning & Meta-Learning

Summary. Hierarchical RL (options framework) and meta-learning formalize how agents learn temporally abstract actions and how priors or meta-policies accelerate transfer and adaptation (Sutton & Barto; recent HRL surveys) [18], [19], [24]. Switching controllers and gated meta-policies provide the algorithmic ground for layered control.

ConsciOS Mapping. Embodied/Supervisory/Meta map naturally to low-level option executors, mid-level policy selectors, and meta-learning priors. The TIC concept operationalizes the resource gating that unlocks higher-complexity frames, analogous to budgeted computation or curiosity rewards.

Model status. Algorithmic mapping is established; the specific Time-Integrated Coherence operationalization (time-integrated coherence gating higher frames) is a design innovation requiring benchmarks across meta-RL tasks.

Table 1. Comparison of ConsciOS and Standard HRL features.

Feature	ConsciOS (Embodied/Supervisory/Meta)	Standard HRL (Options/Feudal)
Selection Signal	Resonance (Coherence + Utility - Cost)	Value Function (Reward Maximization)
Gating Mechanism	TIC (Time-Integrated Coherence)	Temporal Termination / Sub-goal Completion
Feedback Channel	ICS (Affective/Interoceptive)	Reward / Error Signal
State Space	Multi-layer (Physical + Energetic/Coherence)	State + Temporal Abstraction

Measurement & Tests: implement hierarchical agents with TIC gating; compare to standard HRL baselines on transfer, resilience to shift, and computational cost. Log policy diversity and measure relation between sustained coherence and unlocked policy complexity.

6.5 Human-in-the-Loop Learning, RLHF, and AI Alignment

Summary. Reinforcement learning from human feedback (RLHF) and human-in-the-loop systems use user signals to shape agent policies. Recent alignment work emphasizes hybrid architectures combining human priors, interpretability constraints, and intrinsic agent objectives [18], [19], [25].

ConsciOS Mapping. ICS signals and Central Coherence Estimator proxies are candidate human feedback channels for RLHF—fast, affect-informed signals that can shape agent selection and policy priors. The nested controller architecture provides an alignment affordance: Supervisory Controller and Meta-Controller can serve as interpretability and governance layers enforcing safety constraints.

Model status. The high-level mapping to alignment frameworks is an analogy with practical potential; operational details of ICS as RLHF require human trials and careful safety considerations (ethical, adversarial feedback, reward hacking).

Measurement & Tests: small-scale human-in-the-loop trials using ICS telemetry as shaping reward; measure agent behaviour, alignment metrics, and human perceived control/agency; evaluate safety failure modes (adversarial signals, goal hacking).

6.6 Limitations of the Current Evidence Base & Open Challenges

- **Empirical grounding:** several central ConsciOS constructs (TIC, NLS heuristic, Resonance Engine weighting) are engineering hypotheses with attractive theoretical grounding but limited direct empirical evidence; they require simulation and human experiments.
- **Measurement validity:** interoceptive proxies (HRV, EEG) are imperfect and noisy. Vali-

dating robust ICS mappings across populations and contexts is nontrivial.

- **Operational complexity:** implementing coherent coherence metrics across multi-modal inputs requires careful model selection, calibration, and computational budgets.
- **Ethical and safety concerns:** using physiological/affective signals as shaping rewards raises consent, privacy, and manipulation concerns. Any human-in-the-loop work must prioritize ethical review and safeguards.

6.7 Synthesis and Research Agenda

Short-term priorities (0–6 months):

- Formalize coherence metrics (KLD, log evidence, embedding similarity) and publish benchmarks.
- Implement hierarchical RL agents with a TIC gating mechanism and run transfer/adaptation benchmarks.
- Run pilot human studies validating ICS mapping to HRV and predictive power for option-availability.

Medium-term priorities (6–24 months):

- Human-in-the-loop RLHF trials using ICS as shaping signal with strong safety monitoring.
- Cross-domain replication (lab, ecological sampling, simulated agents) and release of open datasets and code.
- Comparative studies mapping ConsciOS constructs to VSM/active inference control metrics in organizational settings.

Model status summary. The paper’s structural mappings to systems theory, active inference, and hierarchical RL are principally established. Key operational innovations (ICS as fast control signal; Time-Integrated Coherence gating; NLS heuristic) are framed as hypotheses and will be elevated as empirical evidence accumulates.

7. The AI Mirror — Applications to Artificial Agents and Alignment

7.1 Purpose and Scope

This section translates the ConsciOS architecture into concrete architectures, experiments, and governance patterns for artificial agents. The goal is practical: demonstrate how the Embodied/Supervisory/Meta decomposition, Resonance Engine, ICS, and Time-Integrated Coherence can

be implemented and tested in agentic systems; identify alignment and safety affordances; and propose pilot deployments that produce measurable scientific output.

7.2 Mapping ConsciOS to AI Agent Architectures

In artificial agents, ConsciOS constructs are operationalized through computational analogs of biological signals. The agent develops “interoceptive” coherence measures from its own internal model statistics (e.g., prediction error, parameter stability, model evidence). During training, these autonomous measures may be supplemented or shaped by human-provided ICS signals (see Experiment 3); in deployment, the agent operates using self-assessed coherence.

- **Embodied controller / low-level policy (Embodied Controller):** Implemented as a fast policy module or low-level controller in robotics or simulated agents (e.g., policy π_e parameterized by neural networks or model predictive controllers). It handles sensory inputs and immediate action loops and exposes short-horizon telemetry (latencies, action entropy) [15] (maps to VSM Systems 1–3).
- **Supervisory controller / policy selector (Supervisory Controller):** Implemented as a mid-level manager that selects or composes policies from a policy library or a set of options. Technically realized as a policy-over-options, gating network, or a learned selector (e.g., meta-controller). It evaluates coherence metrics and expected utility and implements the ResonanceEngine selection rule [16] (maps to VSM System 4).
- **Meta-controller / prior generator (Meta-Controller):** Implemented as a meta-learning or prior-shaping module: e.g., an outer loop that updates priors, regularizers, or initializations (MAML-style, population-based training, or distributional priors). It controls long-term adaptation, governance constraints, and objective shaping [15] (maps to VSM System 5).
- **Centralized coherence estimator, discretized affect index, and time-integrated coherence resource (Central Coherence Estimator / ICS / Time-Integrated Coherence):** Central Coherence Estimator = centralized coherence estimator (e.g., model evidence, posterior precision); ICS = scalar intrinsic signal computed from internal prediction error, parameter stability, and state confidence; Time-Integrated Coherence = persistent meta-state (time-integrated coherence). These variables inform gating, reward shaping, and policy unlocking dynamics.

7.3 Concrete Technical Experiments (Agentic Testbeds)

Below are prioritized experiments that produce defensible empirical claims and are tractable in standard agent frameworks.

Experiment 1: Hierarchical Agent Benchmark (Embodied/Supervisory/Meta vs Flat)

- **Setup:** Build two agents in a procedurally generated environment with episodic distribu-

tional shifts: (A) Hierarchical agent with Embodied/Supervisory/Meta and Time-Integrated Coherence gating; (B) Flat baseline agent with comparable parameter count.

- **Manipulations:** Introduce sudden context shifts and resource constraints; vary b/a weights in the Resonance selection rule.
- **Metrics:** cumulative reward, adaptation latency (time to recover pre-shift performance), policy diversity, computational cost.
- **Expected Result:** Hierarchical agent exhibits faster recovery, higher transfer, and graceful degradation under constraints if Meta/Super stratification is effective.

Experiment 2: ICS as Intrinsic Reward (Affect-Driven RL)

- **Setup:** Train agents with an intrinsic reward augment derived from an ICS proxy computed from the agent’s own internal coherence signals (e.g., prediction error magnitude, parameter update stability, model evidence). Compare to agents with standard curiosity or novelty intrinsic rewards.
- **Manipulations:** Vary scale of ICS influence; test in sparse reward environments.
- **Metrics:** sample efficiency, exploration patterns, policy robustness.
- **Expected Result:** ICS-shaped agents show improved exploration aligned with long-horizon coherence; risk: reward hacking — monitor for adverse optimization.

Experiment 3: Human-in-the-Loop ICS Shaping (RLHF variant)

- **Setup:** Recruit human participants to provide fast ICS signals (subjective 1–10 ladder) while interacting with an environment; use ICS as shaping reward during agent fine-tuning.
- **Manipulations:** Compare shaped vs unshaped agents and compare different ICS smoothing/decay regimes.
- **Measures:** agent alignment to human preferences, transfer, and human perceived control and agency.
- **Safety Guardrails:** explicit consent, adversarial signal detection, audit logs, human override. Ethical review required.

Experiment 4: Time-Integrated Coherence Gate Unlocking Complexity

- **Setup:** Implement the TIC variable as an accumulated coherence budget; implement high-cost policies that require Time-Integrated Coherence $\geq \theta$ to unlock.
- **Metrics:** policy complexity usage, cost efficiency, task performance under time pressure.
- **Expected Result:** Time-Integrated Coherence gating yields more conservative resource allocation and reduces spurious activation of expensive policies while enabling high-value policy use when coherence is sustained.

7.4 Prototype Applications and Pilot Domains

- **Simulated agent research labs:** immediate testbeds for Experiments 1–4 using OpenAI Gym variants, ProcGen, or custom procedurally generated environments.
- **Robotics / embodied agents:** Embodied Controllers in mobile platforms; Supervisory Controller for task selection (home assistant switching tasks); pilot complexity gating with Time-Integrated Coherence in battery-constrained robots.
- **Smart home context:** integrate ICS proxies from occupant devices (consented HRV wearables) to adapt lighting/temperature policy selection — low-risk pilot if privacy/consent is addressed.
- **Organizational decision support:** simulate Embodied/Supervisory/Meta as team roles in collaborative platforms to test policy selection and governance before automation.

7.5 Governance, Safety, and Ethical Considerations

- **Measurement validity & privacy:** physiological signals (HRV, ICS proxies) are sensitive. All human data collection must follow IRB, GDPR/CCPA compliance, and local regulation. Use minimal necessary telemetry and strong anonymization.
- **Reward hacking & manipulation:** ICS used as shaping reward may be manipulated. Implement adversarial detection, signal plausibility checks, and human override.
- **Interpretability & auditability:** Design Supervisory Controller and Meta-Controller with explicit logging, provenance, and interpretable selection traces to enable audits and post-hoc explanations for policy selection.
- **Safety layering:** Fallback Safety Controller (FSC) / pi_safe must be robust to adversarial inputs and designed by principled safety engineering (conservative defaults, kill-switches, oversight loops).
- **Governance pathways:** include stakeholder review boards, transparency reports, and open pre-registration of human trials; consider third-party audits for high-impact deployments.

7.6 Metrics, Benchmarks and Evaluation Protocol

Suggested canonical metrics for pilot evaluation:

- **Agent performance:** cumulative reward, normalized performance relative to baseline.
- **Adaptation:** adaptation latency post distributional shift, recovery ratio.
- **Option-availability proxy:** measured $|A_{\text{eff}}(t)|$ via simulated affordance enumeration or human reported affordance counts (experience sampling).
- **Coherence correlation:** Spearman/Pearson correlation of coherence metric C with ICS prox-

ies and with downstream performance improvements.

- **Time-Integrated Coherence impact:** correlation and causal estimates (instrumental variable or randomized threshold experiments) of Time-Integrated Coherence on policy unlocking and sustained performance.
- **Safety metrics:** frequency of π_{safe} engagements, rate of adversarial detection triggers, human override rate.

7.7 Roadmap for Pilots, Datasets, and Reproducibility

- **Phase 0 (0–2 months):** Code templates for ResonanceEngine, coherence functions (cosine/embedding/KLD), and Time-Integrated Coherence computation; seed simulated envs.
- **Phase 1 (2–6 months):** Run Experiment 1–2 in simulation; open-source code and baseline datasets; preprint results for community review.
- **Phase 2 (6–12 months):** Human pilot for Experiment 3 under IRB; release anonymized datasets and analysis scripts; produce safety/adversarial analysis.
- **Phase 3 (12–24 months):** Domain pilots (robotics, smart home) with external audits and governance reporting; refine Meta-Controller governance patterns for organizational deployment.

7.8 Funding, Community, and Research Agenda

Priority funding areas:

- **Core research:** hierarchical agent benchmarks and coherence metric standardization.
- **Measurement science:** validating ICS mappings across populations and contexts.
- **Safety research:** reward-shaping failure modes, adversarial robustness of ICS signals, and governance tooling.

Community building:

- Open dataset & baseline repo for coherence and Time-Integrated Coherence experiments.
- Incentivized hackathons & shared benchmarks to accelerate reproducibility.
- Collaborative pilots with human factors labs and AI alignment groups.

7.9 Concluding Practical Note

ConsciOS provides a layered architecture that is especially well-suited for hybrid human-agent systems where rapid, affective feedback and clear governance are necessary. The proposed ex-

periments are designed to deliver concrete evidence about whether coherence-based gating and affect-informed shaping improve adaptability, transfer, and alignment in hierarchical agents. The next step is to implement the simulation testbeds (Phase 0/1) and publish reproducible baselines that enable community scrutiny.

8. Becoming a Conscious Architect — Practical Implications for Design

8.1 Purpose and Scope

This section translates architecture into practice: design patterns, organizational applications, curricula for training human operators, and product prototypes that operationalize ConsciOS in real contexts. It emphasizes repeatability, instrumentation, and governance.

8.2 Design Patterns & Practices

- **Pattern: Nested Controller Decomposition** — adopt Embodied/Supervisory/Meta separation in system design; log policy families, selection traces, and coherence signals.
- **Pattern: Coherence-gated complexity** — use Time-Integrated Coherence thresholds to gate high-cost capabilities (compute, autonomy, privilege escalation).
- **Pattern: Rapid Feedback Loops** — implement ICS proxies or synthetic analogues that feed short-horizon controllers for real-time micro-adjustments.

8.3 Organizational Applications

- **Team design:** map Embodied Controller roles to operational teams, Supervisory Controller roles to coordination roles, Meta-Controller roles to governance/strategy; instrument team decision logs as policy traces.
- **Product design:** staged feature unlocking by Time-Integrated Coherence; adaptability modules for user interfaces based on measured ICS proxies.

8.4 Training & Curricula

- **Curriculum:** imagineer practices → coherence training → quality control drills. Translate book protocols into training modules (micro-exercises with measurement).
- **Operator tooling:** dashboards that show coherence, Time-Integrated Coherence balances, and option-availability metrics to support decision making.

8.5 Implementation Checklist

- Instrument short-horizon telemetry (Embodied Controller logs).
- Implement a coherence estimator and ICS proxy.
- Build policy library & gating logic (Time-Integrated Coherence).
- Design governance & safety overrides (Fallback Safety Controller).

Instrument, gate, and train around coherence as the operational lever; the checklist above provides a recommended implementation sequence for applied settings.

9. Discussion, Limitations & Future Work

9.1 Summary of Contributions

We formalized a nested control architecture for consciousness, introduced resonance and coherence metrics, operationalized affect as an internal feedback channel (ICS), and proposed an empirical roadmap spanning simulation and human trials.

9.2 Limitations

- **Measurement validity:** physiological proxies (HRV, EEG) are noisy and context-sensitive.
- **Operational complexity:** multi-modal coherence computation is computationally nontrivial.
- **Ethical concerns:** privacy, signal manipulation, and reward-hacking risks in human-in-the-loop settings.
- **Novelty limits:** several constructs are integrative—must avoid overclaiming novelty where existing work overlaps.
- **Domain scope:** Our empirical validation focuses primarily on hierarchical reinforcement learning agents in structured environments. The framework’s applicability to other AI architectures (large language models, transformers, diffusion models) and to more open-ended, real-world domains remains to be demonstrated. For high-dimensional spaces (e.g., LLM latent states), we suggest a “sparse coherence” approach where resonance is evaluated only on key decision nodes to manage computational latency. Similarly, while we propose human-subjects protocols, large-scale validation across diverse populations and contexts has not yet been conducted.
- **Alignment properties:** While our agent benchmarks demonstrate improved task performance through coherence-based selection, we have not yet systematically evaluated whether ConsciOS-based architectures exhibit superior alignment properties—such as robustness to

distributional shift, resistance to reward hacking, or long-term goal stability—compared to alternative approaches. These alignment-specific evaluations represent crucial future work.

We view these limitations not as fundamental weaknesses but as invitations for future research. The v1.0 designation of this paper reflects our expectation that the framework will evolve through continued empirical validation, community engagement, and iterative refinement based on real-world applications.

9.3 Implications for AI Alignment

The AI alignment problem—ensuring AI systems pursue goals aligned with human values—remains critical [18], [19]. Current approaches (RLHF, constitutional AI, reward modeling) address surface-level behaviors without reforming the fundamental architecture of goal formation and value selection. ConsciOS proposes an alternative: coherence-based control architectures where AI systems optimize alignment with long-term values through their own internal coherence signals.

Potential advantages: ConsciOS-based architectures offer three potential alignment advantages:

1. **Natural multi-objective optimization:** The Resonance Engine balances task performance, value coherence, and resource efficiency without hand-tuned weighting schemes. The coherence metric ($C(F; S)$) evaluates policy frames against meta-level priors, enabling robust value alignment across contexts and timescales.
2. **Systematic distinction between coherent and reactive behavior:** ICS and Time-Integrated Coherence mechanisms operationalize the difference between impulsive actions and value-aligned choices. This enables reward functions that incentivize coherent behavior based on the agent’s own internal coherence signals, and evaluation frameworks that assess genuine value alignment versus superficial policy matching. We explicitly note the risk of “coherent but evil” systems; benevolence does not emerge solely from internal consistency but requires hard-coded human-compatible priors in the Meta-Controller (e.g., via ethical constraints or active inference discovery).
3. **Hierarchical value alignment:** The nested architecture (Meta-Controller → Supervisory Controller → Embodied Controller) embeds values at multiple timescales. Long-term ethical principles encode as Meta-Controller priors; mid-horizon policy selection evaluates actions for coherence with these principles. This addresses long-term alignment challenges while avoiding brittleness of single-level approaches.

Research needs: These implications require substantial empirical validation. Critical questions: Do ConsciOS-based architectures exhibit superior alignment properties in complex environments? Can coherence metrics capture human values robustly? How do they scale computationally? What are failure modes in adversarial settings?

We position this as initial formalization and call for the AI safety community to empirically evaluate whether ConsciOS-based control offers practical advantages for building aligned artificial intelli-

gence.

9.4 Future Work and Roadmap

- Formal benchmarks for coherence metrics.
- Large-scale, cross-population validation of ICS mappings.
- Hierarchical agent competitions with standardized Time-Integrated Coherence gating.
- Governance patterns & audit tooling for high-impact deployments.
- Publish code, datasets, and pre-registrations to accelerate independent validation.

This section offers an honest appraisal of current limitations, discusses implications for AI alignment research, and sketches the empirical roadmap ahead, prioritizing measurements, benchmarks, and safeguards required to evaluate the model rigorously.

10. Conclusion

This paper presents ConsciOS as a unified engineering program for consciousness as a designable system—one that can be modeled, instrumented, and improved through rigorous experimentation. We contribute three core elements: (i) a nested three-layer control architecture (Embodied Controller, Supervisory Controller, Meta-Controller) that decomposes conscious agency into testable subsystems grounded in viable systems theory, hierarchical reinforcement learning, and active inference; (ii) coherence-based selection mechanisms (Resonance Engine, ICS, Time-Integrated Coherence) that operationalize affect and interoceptive feedback as measurable control signals; and (iii) a comprehensive empirical roadmap with detailed experimental protocols, canonical terminology mappings, and reproducible code references.

ConsciOS reframes consciousness as an engineering challenge—testable, instrumentable, and governable. The architecture offers practical affordances for AI alignment through interpretable hierarchical decomposition, affect-informed policy selection, and built-in safety mechanisms. By providing formal, testable hypotheses about consciousness as hierarchical control, we bridge contemplative traditions, systems science, and modern computational frameworks—translating millennia of systematic observation into falsifiable hypotheses that are practically applicable to the urgent challenge of building aligned artificial intelligence. A key implication is that **alignment is an architectural property, not a training outcome**: coherence must be designed into the control structure before deployment, not patched post-hoc.

We invite researchers, engineers, and practitioners to implement, test, and critique the proposed models and to collaborate on open benchmarks and datasets.

Appendix A — Experimental protocols

Experimental Protocols (full templates)

- **A.1** H1: Structure-Change Leverage RCT (human) — objectives, sample sizes, randomization, outcome metrics, analysis plan, preregistration template.
- **A.2** H2: Feedback Coherence → Option-Availability — ecological sampling + lab micro-tasks.
- **A.3** H3: Nested Controller Benchmark (simulations) — environment specs, seeds, agent code skeleton, logging format.
- **A.4** H4: ICS as RLHF shaping (pilot human trials) — consent forms, pre-screening, safety checks, adversarial monitoring.
- **A.5** Toy ablation (simulation demo) — **Purpose:** verify telemetry and selector sensitivity. **Setup:** episodic context shifts; hierarchical agent with coherence-weighted selection (b , a sweeps). **Outputs:** selection traces and aggregated heatmaps (reward, alignment rate, position-match proxy). **Code:** repository code/ directory (env, agents, plots) [7]; figures are illustrative only. Note: Figure A1 is an illustrative schematic; full benchmarks with non-flat performance landscapes are reserved for future empirical work.
- **A.6** H5: Somatic Resonance Validation (human) — **Purpose:** test whether subjective thoracic expansion/contraction correlates with physiological coherence and predicts frame selection. **Design:** within-subject time-series; collect HRV (time/frequency indices), optional EEG coherence, and rapid subjective reports of somatic feelings and ICS ladder; induce small local perturbations and log subsequent frame selection. **Analysis:** mixed models with lagged predictors; test added predictive value over utility and baseline affect.



Fig. A1. Toy ablation heatmaps across $b \times a$ (reward, alignment rate, position-match). Axes: $x =$ coherence weight b (low→high), $y =$ utility weight a (low→high); g fixed. Illustrative demo; not a benchmark result. Credit: ConsciOS demo (this work).

Each template includes the stepwise procedure, required hardware/software, analysis scripts skeleton, expected effect sizes, and a power-calculations placeholder.

Appendix B — Measurement Instruments & Analysis Pipelines

- **B.1** HRV measurement spec (sensor types, sampling rates, preprocessing).

- **B.2** EEG coherence pipeline (preprocessing, epoching, Phase-Locking Value (PLV) / Inter-Subject Correlation (ISC) metrics).
- **B.3** Coherence computation code (KLD, log-evidence, embedding cosine examples) — code repository [7]
- **B.4** Policy logging schema & Supervisory Controller selection trace format (JavaScript Object Notation (JSON) schema).
- **B.5** Statistical analysis pipelines (time-series mixed models, Granger causality / vector autoregression (VAR), causal estimation approach).

Appendix C — Public Translation & Operationalization

Table 2. Operational Definitions & Design Metaphors — for cross-disciplinary application (e.g., organizational design), this architecture maps formal control variables to operational metaphors used in the author’s applied handbooks:

Canonical Term	Applied Metaphor	Operational Definition / Measures
Embodied Controller	Echo-Self	Local actor subsystem executing fast closed-loop control. Measures: reaction latency, action entropy, short-horizon task performance.
Supervisory Controller	Super-Self	Aggregates feedback and selects among policy families. Measures: policy selection latency, switch frequency, selection accuracy.
Meta-Controller	Meta-Self	Encodes long-horizon priors and the generative space of policies. Measures: prior concentration, transfer/meta-learning performance.
Central Coherence Estimator	Kernel	Focal interoceptive/state-confidence signal. Measures: HRV, interoceptive accuracy; estimator precision.

Canonical Term	Applied Metaphor	Operational Definition / Measures
Interoceptive Control Signal (ICS)	Emotional Guidance Scale (EGS)	Laddered affect used as internal control signal. Measures: self-report ladder, HRV, EEG proxies.
Resonance Engine	Resonance Engine	Coherence-based selector. Measures: coherence score, selection confidence.
Time-Integrated Coherence	FREQ Coin	Time-integrated coherence resource. Measures: cumulative coherence over window, option-availability proxy.
Quality Control	Quality Control	Belief-surfacing / model revision. Measures: update frequency, belief entropy, error magnitude.
Policy/Frame Library	Policy/Frame Library	Library of precomputed policy frames/timelines for selection. Measures: policy diversity, match scores.
Fallback Safety Controller	Ego Autopilot	Low-variance default policy engaged under low confidence. Measures: reversion frequency, conservatism index.
Diagnostic Hierarchy	The Iceberg	Diagnostic hierarchy levels (Events/Patterns/Structures/Models). Measures: event/pattern/structure metrics.
System Flows	The 7 Flows	Flows (Input/Process/Output/Feedback/Actors). Measures: throughput, latency.

Detailed Terminology Mappings. The following entries provide expanded definitions, operational measures, and suggested citations for each ConsciOS term listed in Table 2. These detailed mappings support reproducible operationalization and citation tracking across experimental protocols.

Embodied Controller

- **ConsciOS Alias (applied metaphor):** Echo-Self
- **Operational definition / measures:** Local actor subsystem executing fast closed-loop control. Measures: reaction latency, action entropy, short-horizon task performance, sensorimotor noise.
- **Suggested citation & placement:** VSM S1–3 mapping; hierarchical RL; Section 4.1, Appendix B [16], [17].

Supervisory Controller

- **ConsciOS Alias (applied metaphor):** Super-Self
- **Operational definition / measures:** Aggregates feedback and selects among policy families. Measures: policy selection latency, switch frequency, selection accuracy under perturbation.
- **Suggested citation & placement:** Meta-RL & hierarchical RL; Sections 4.1–4.4 [16], [17].

Meta-Controller

- **ConsciOS Alias (applied metaphor):** Meta-Self
- **Operational definition / measures:** Encodes long-horizon priors and the generative space of policies. Measures: prior concentration, transfer/meta-learning performance.
- **Suggested citation & placement:** Meta-learning; Sections 4.1 & 7 [16].

Central Coherence Estimator

- **ConsciOS Alias (applied metaphor):** Kernel
- **Operational definition / measures:** Focal interoceptive/state-confidence signal. Human proxy: HRV, interoceptive accuracy. Agent proxy: estimator precision.
- **Suggested citation & placement:** Interoception literature; Methods/Appendix B [3].

Interoceptive Control Signal (ICS)

- **ConsciOS Alias (applied metaphor):** Emotional Guidance Scale (EGS)
- **Operational definition / measures:** Laddered affect used as internal control signal. Measures: self-report ladder, HRV, EEG proxies, affect classification.
- **Suggested citation & placement:** Affect & interoception reviews; Section 5.2 & Appendix A [3].

Resonance Engine

- **ConsciOS Alias (applied metaphor):** Resonance Engine
- **Operational definition / measures:** Chooses policy frame with maximal coherence; match-score or Bayesian evidence metric. Measures: coherence score, selection confidence.
- **Suggested citation & placement:** Active inference / predictive processing; Section 5.1 [14].

Time-Integrated Coherence (TIC)

- **ConsciOS Alias (applied metaphor):** FREQ Coin
- **Operational definition / measures:** Time-integrated coherence units (e.g., area under coherence curve). Measures: cumulative coherence over window, option-availability proxy.
- **Suggested citation & placement:** Section 5 & Appendix B.

Quality Control

- **ConsciOS Alias (applied metaphor):** Quality Control
- **Operational definition / measures:** Frequency of internal model updates following coherence shifts. Measures: belief entropy, update rate, error magnitude.
- **Suggested citation & placement:** Active inference & Bayesian update; Section 5.4 [14].

Policy/Frame Library

- **ConsciOS Alias (applied metaphor):** Policy/Frame Library
- **Operational definition / measures:** Library of precomputed policy frames/timelines for selection. Measures: policy diversity, match scores, retrieval latency.
- **Suggested citation & placement:** Meta-RL & simulation; Sections 5.1 / 4.4 [18], [19].

Fallback Safety Controller

- **ConsciOS Alias (applied metaphor):** Ego Autopilot
- **Operational definition / measures:** Low-variance survival policy under low-coherence. Measures: reversion frequency, conservatism index.
- **Suggested citation & placement:** Systems thinking (Meadows; Senge); Section 2.1 [10], [12].

Diagnostic Hierarchy levels

- **ConsciOS Alias (applied metaphor):** The Iceberg (Events / Patterns / Structures / Models)
- **Operational definition / measures:** Diagnostic hierarchy levels. Measures: event/pattern/structure

metrics.

- **Suggested citation & placement:** Systems thinking (Meadows); Section 2.2 [10].

System Flows

- **ConsciOS Alias (applied metaphor):** The 7 Flows
- **Operational definition / measures:** Systems decomposition—each flow has standard metrics (throughput, latency, bottleneck).
- **Suggested citation & placement:** Systems engineering & cybernetics; Section 2.2 [11], [13].

Note:

For each term, Appendix B contains measurement protocols and recommended instruments (e.g., HRV measurement specs, EEG coherency pipeline, policy-switch logging format).

Appendix D — Viable System Model (VSM) Mapping

VSM Systems and ConsciOS alignment (concise):

- VSM S1–S3 (Operations and immediate control) → Embodied Controller (short-horizon perception–action loops; fast feedback).
- VSM S4 (Intelligence/adaptation/future planning) → Supervisory Controller (policy/frame selection; supervisory control).
- VSM S5 (Policy/identity/governance) → Meta-Controller (long-horizon priors; identity constraints; slow adaptation).

Notes: Echo corresponds to operational bandwidth and local control; Super aggregates feedback and selects among policy families; Meta encodes priors and identity constraints that shape policy space and slow updates. This mapping is heuristic but aligns with canonical VSM roles [11], [13].

AI Usage Declaration

The architectural design, system decomposition (Embodied/Supervisory/Meta), and control-theoretic logic are the original work of the author. Large Language Models were utilized strictly as typesetting, copy-editing, and formatting tools to accelerate the drafting of the manuscript.

References

- [1] B. Schoen, E. Nitishinskaya, M. Balesni, et al., “Stress Testing Deliberative Alignment for Anti-Scheming Training,” Apollo Research & OpenAI, 2025. arXiv:2509.15541.

- [2] OpenAI, “Detecting and Reducing Scheming in AI Models,” 2025. Available at: <https://openai.com/index/detecting-and-reducing-scheming-in-ai-models/>
- [3] L. F. Barrett and W. K. Simmons, “Interoceptive predictions in the brain,” *Nature Reviews Neuroscience*, vol. 16, pp. 419–429, 2015. doi:10.1038/nrn3950.
- [4] S. W. Lazar, C. E. Kerr, R. H. Wasserman, et al., “Meditation experience is associated with increased cortical thickness,” *NeuroReport*, vol. 16, no. 17, pp. 1893–1897, 2005. doi:10.1097/01.wnr.0000186598.66243.19.
- [5] B. K. Holzel, U. Ott, T. Gard, H. Hempel, M. Weygandt, K. Morgen, and D. Vaitl, “Investigation of mindfulness meditation practitioners with voxel-based morphometry,” *Social Cognitive and Affective Neuroscience*, vol. 3, no. 1, pp. 55–61, 2008. doi:10.1093/scan/nsm038.
- [6] Y.-Y. Tang, R. Tang, and M. I. Posner, “Brief meditation training induces white matter changes in the anterior cingulate,” *Proceedings of the National Academy of Sciences (PNAS)*, vol. 107, no. 35, pp. 15649–15652, 2010. doi:10.1073/pnas.1011043107.
- [7] J. Riddle and J. W. Schooler, “Hierarchical consciousness: the Nested Observer Windows model,” *Neuroscience of Consciousness*, vol. 2024, no. 1, 2024, Art. niae010. doi:10.1093/nc/niae010.
- [8] J. Smallwood and J. W. Schooler, “The Science of Mind Wandering: Empirically Navigating the Stream of Consciousness,” *Annual Review of Psychology*, vol. 66, pp. 487–518, 2015. doi:10.1146/annurev-psych-010814-015331.
- [9] K. Kaynak, “ConsciOS v1.0 Code Repository,” Source code, 2025. Available: <https://github.com/Sistemist/consciOS-paper>.
- [10] P. M. Senge, “The Fifth Discipline: The Art and Practice of the Learning Organization,” Revised and Updated. Doubleday/Currency, 2006.
- [11] P. Checkland, “Systems Thinking, Systems Practice.” John Wiley & Sons, 1981.
- [12] D. H. Meadows, “Thinking in Systems: A Primer.” Chelsea Green Publishing, 2008.
- [13] S. Beer, “Brain of the Firm.” John Wiley & Sons, 1972.
- [14] K. Friston, “The free-energy principle: a unified brain theory?,” *Nature Reviews Neuroscience*, vol. 11, no. 2, pp. 127–138, 2010. doi:10.1038/nrn2787.
- [15] M. Albarracín, I. Hipólito, J. Ramstead, et al., “Designing Explainable Artificial Intelligence with Active Inference: A Framework for Transparent Introspection and Decision-Making,” in *Active Inference*, M. Biehl et al., Eds. Springer, 2024, pp. 123–144. arXiv:2408.06348.
- [16] K. Friston, L. Da Costa, D. Hafner, C. Hesp, and T. Parr, “Active Inference: The Free Energy Principle in Mind, Brain, and Behavior.” MIT Press, 2022. doi:10.7551/mitpress/12441.001.0001.
- [17] A. Darling, S. Denton, and K. Safron, “Relevance Realization through Active Inference and the Free Energy Principle,” arXiv:2501.09899, 2025.
- [18] R. S. Sutton and A. G. Barto, “Reinforcement Learning: An Introduction,” 2nd ed. MIT Press, 2018.

- [19] R. S. Sutton, D. Precup, and S. Singh, “Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning,” *Artificial Intelligence*, vol. 112, nos. 1–2, pp. 181 –211, 1999. doi:10.1016/S0004-3702(99)00052-1.
- [20] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané, “Concrete Problems in AI Safety.” arXiv:1606.06565, 2016. Available: <https://arxiv.org/abs/1606.06565>.
- [21] E. Hubinger, C. van Merwijk, V. Mikulik, J. Skalse, and S. Garrabrant, “Risks from Learned Optimization in Advanced Machine Learning Systems.” arXiv:1906.01820, 2019. Available: <https://arxiv.org/abs/1906.01820>.
- [22] P. Lanillos, C. Meo, C. Pezzato, et al., “Active Inference in Robotics and Artificial Agents: Survey and Challenges,” arXiv:2112.01871, 2021.
- [23] M. Barthet, A. Khalifa, A. Liapis, and G. N. Yannakakis, “Play with Emotion: Affect-Driven Reinforcement Learning,” in 2022 10th International Conference on Affective Computing and Intelligent Interaction (ACII), 2022. arXiv:2208.12622v1.
- [24] S. Pateria, B. Subagdja, A.-H. Tan, and C. Quek, “Hierarchical Reinforcement Learning: A Comprehensive Survey,” *ACM Computing Surveys*, 54(5), 1–35, 2021. doi:10.1145/3453160.
- [25] L. Ouyang, J. Wu, X. Jiang, et al., “Training language models to follow instructions with human feedback,” in Advances in Neural Information Processing Systems (NeurIPS), 2022. arXiv:2203.02155.