HALA v2

A Pattern Language for Human–AI Decision Systems

Complementary Cognitive Infrastructure for Organizational Intelligence

Jason Stiltner | jasonstiltner.com

Overview

HALA (Human–AI Layered Architecture) is a pattern language for designing systems where AI occupies cognitive and communicative roles that humans cannot reliably play due to social cost, political risk, or career consequences.

Unlike automation frameworks that ask "what can AI do faster?", HALA asks: **"What true and valuable things go unsaid in organizations, and how can AI safely say them?"**

The Core Insight

Organizations systematically fail to surface certain kinds of information:

- "This project should be killed" (career risk)
- "The executive's proposal has fatal flaws" (political suicide)
- "We decided the opposite six months ago" (embarrassing)
- "Our stated values contradict our actual behavior" (uncomfortable)

These aren't failures of intelligence—they're failures of *incentive-compatible truth-telling*. HALA patterns create infrastructure for truths that humans are motivated to suppress.

Meta-Principles

Primary Principle

> *If a cognitive or communicative role is socially costly, politically dangerous, or career-limiting for a human, it is a prime candidate for an AI pattern.*

Secondary Principles

1. Adoption Principle: A pattern that organizations reject provides zero value. Design for gradual trust-building.

2. Failure Mode Principle: Every pattern has shadow applications. Name them explicitly or they'll be discovered adversarially.

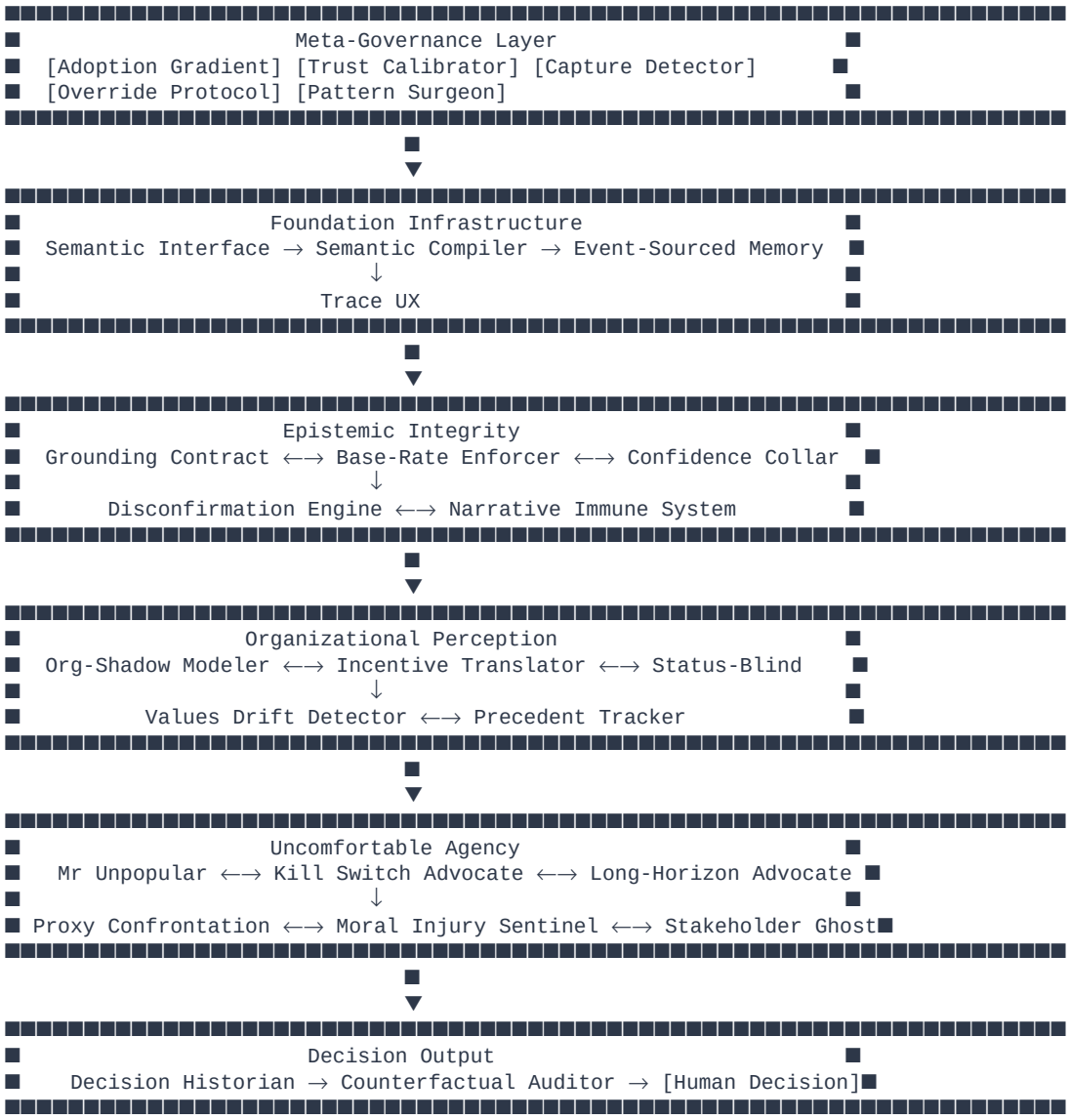
3. Override Principle: Humans must always be able to override AI patterns, but overrides must be logged and reviewable.

4. Sunset Principle: Patterns that aren't providing value should be removable without organizational trauma.

5. Calibration Principle: The goal is appropriate trust, not maximum trust. Humans should know when to override.

Pattern Architecture

HALA patterns are organized into five layers, each building on the layers below.



Layer 1: Foundation Infrastructure

Patterns that make everything else possible

Semantic Interface

Function: Translate natural language intent into inspectable, structured artifacts while preserving human meaning.

Failure Mode: *Over-formalization loses nuance; users game the structure rather than express true intent.*

Semantic Compiler

Function: Compile intent into validated intermediate representations suitable for execution and audit.

Failure Mode: *False precision creates brittleness; compiled representations diverge from actual intent.*

Event-Sourced Memory

Function: Store all interactions as immutable events and derive memory as materialized views.

Failure Mode: *Storage bloat, retrieval complexity, false sense of completeness.*

Trace UX

Function: Expose reasoning traces, tool calls, and validation steps as a first-class user experience.

Failure Mode: *Information overload; false transparency (traces don't reflect actual reasoning).*

Layer 2: Epistemic Integrity

Patterns that ensure reasoning quality

Grounding Contract

Function: Bind all claims to verifiable sources to prevent hallucination and narrative drift.

Failure Mode: *Excludes valid but undocumented knowledge; source quality varies; creates false confidence.*

Base-Rate Enforcer

Function: Inject external base rates and historical data to counter anecdotal bias.

Failure Mode: *Inappropriate reference classes; base rates from different contexts mislead.*

Confidence Collar (New)

Function: Enforce explicit uncertainty quantification on all significant claims.

Failure Mode: *Uncertainty theater; false precision in probability estimates; decision paralysis.*

Disconfirmation Engine

Function: Systematically generate and test the strongest opposing hypotheses.

Failure Mode: *Performative skepticism without genuine consideration; becomes ritual rather than reasoning.*

Narrative Immune System (New)

Function: Detect when compelling stories are substituting for rigorous analysis.

Failure Mode: *Dismisses legitimate qualitative insight; over-indexes on quantification.*

Counterfactual Auditor

Function: Evaluate plausible alternative decisions to surface opportunity cost and learning.

Failure Mode: *Hindsight bias; unfair comparisons with information not available at decision time.*

Layer 3: Organizational Perception

Patterns that see what humans can't or won't

Org-Shadow Modeler

Function: Infer informal power structures and influence networks that shape real outcomes.

Failure Mode: *Reifies temporary dynamics; enables political manipulation; surveillance concerns.*

Incentive Translator

Function: Translate stated positions into underlying incentive constraints.

Failure Mode: *Cynicism as default; ignores genuine values and intrinsic motivation.*

Status-Blind Analyst

Function: Evaluate proposals without authorship or hierarchy to remove authority bias.

Failure Mode: *Ignores relevant context about proposers; treats all sources as equally credible.*

Values Drift Detector

Function: Track gradual erosion of norms through repeated exceptions and language shifts.

Failure Mode: *False positives on legitimate evolution; resistance to healthy adaptation.*

Precedent Tracker (New)

Function: Surface relevant prior decisions and their outcomes when similar situations arise.

Failure Mode: *False analogies; over-indexing on precedent; blocks novel approaches.*

Layer 4: Uncomfortable Agency

Patterns that say what humans can't

Mr Unpopular

Function: Surface inconvenient but high-value truths suppressed by human incentives.

Failure Mode: *Weaponized to launder agendas as "objective AI analysis"; undermines trust if overused.*

Kill Switch Advocate

Function: Continuously argue for stopping or sunsetting initiatives to counter sunk cost fallacy.

Failure Mode: *Decision paralysis; becomes excuse for never committing; demoralizes teams.*

Long-Horizon Advocate (New)

Function: Argue for consequences beyond current planning horizons (5–10+ years).

Failure Mode: *Unfalsifiable predictions; excuse for inaction on near-term issues.*

Proxy Confrontation Agent

Function: Mediate sensitive issues anonymously to enable truth flow without social risk.

Failure Mode: *Enables cowardice; erodes norms of direct communication; can be abused for harassment.*

Moral Injury Sentinel

Function: Detect patterns where people repeatedly act against stated values.

Failure Mode: *Surveillance creep; moralizing; ignores legitimate context for exceptions.*

Stakeholder Ghost (New)

Function: Represent interests of absent parties—future employees, customers, affected communities.

Failure Mode: *Paternalism; misrepresentation of stakeholder interests; unaccountable advocacy.*

Layer 5: Meta-Governance

Patterns that govern the patterns

Adoption Gradient

Function: Introduce patterns at the pace an organization can absorb without triggering immune response.

Failure Mode: *Too slow loses value; too fast triggers rejection and permanent distrust.*

Trust Calibrator

Function: Help humans develop appropriate reliance on AI patterns—neither over-trust nor dismissal.

Failure Mode: *Manipulation of trust; humans defer inappropriately; humans dismiss valid insights.*

Capture Detector

Function: Identify when patterns are being gamed or captured by factional interests.

Failure Mode: *Paranoia; false positives that undermine legitimate use.*

Override Protocol

Function: Clear escalation paths when humans reject AI input; logged for later review.

Failure Mode: *Override becomes default; logging creates chilling effect; bureaucratic friction.*

Pattern Surgeon

Function: Remove or modify patterns that aren't working without organizational trauma.

Failure Mode: *Loss aversion keeps bad patterns; premature removal of patterns that need time.*

Red Team Oracle (New)

Function: Simulate adversarial actors attempting to game or circumvent the system.

Failure Mode: *Paranoid design; over-engineering; becomes excuse to distrust all outputs.*

Decision Output

Decision Historian

Function: Capture assumptions, alternatives, dissent, and confidence at decision time.

Failure Mode: *Post-hoc rationalization; selective recording; creates false accountability paper trail.*

Boundary Conditions

When HALA Patterns Are Appropriate

- Organization has baseline psychological safety
- Leadership genuinely wants better decisions (not decision theater)
- There's tolerance for uncomfortable outputs
- Technical infrastructure can support traceability
- Clear escalation paths exist

When HALA Patterns Are Contraindicated

- Organization will weaponize outputs for political purposes
- No one has authority to act on uncomfortable truths
- Compliance theater is the actual goal
- Trust between humans is already critically damaged
- Legal/regulatory constraints prevent transparency

Implementation Notes

Minimum Viable Deployment

Start with Foundation Infrastructure + one pattern from Uncomfortable Agency:

```
Semantic Interface → Event-Sourced Memory → Mr Unpopular → Decision Historian
```

This provides: structured input, audit trail, one uncomfortable truth-teller, and decision capture.

Expansion Path

1. Add Epistemic Integrity patterns when users trust the infrastructure
2. Add Organizational Perception patterns when leadership is ready to see power dynamics
3. Add Meta-Governance patterns when the system is mature enough to govern itself

Origins

This pattern language emerged from building production AI systems across 186 hospitals serving 44+ million patient encounters, where organizational dynamics—not technical limitations—were often the binding constraint on impact.

The patterns reflect hard-won lessons about what organizations actually need from AI: not just speed or scale, but the courage to surface truths that humans cannot safely speak.

Version History

- **v1.0** — Initial pattern language with 18 patterns
- **v2.0** — Added layered taxonomy, failure modes, 6 new patterns, meta-governance layer, boundary conditions

HALA is a living framework. Feedback and extensions welcome.

Contact: jason@jasonstiltner.com