



[Home](#) [Licensing](#) [Patents](#) [Articles](#)

## **One Architecture, Every Domain: How the Same Cognitive Primitives Parameterize Across Autonomous Vehicles, Defense, Companion AI, and Therapeutic Agents**

by [Nick Clark](#) | Published March 27, 2026 | [PDF](#)

Every application domain rebuilds AI governance from scratch. Autonomous vehicle teams engineer their own safety systems. Defense teams engineer their own engagement authorization. Companion AI teams engineer their own emotional models. Therapeutic AI teams engineer their own clinical governance. The result is an industry in which hard-won governance lessons cannot transfer across domains because the underlying substrates share no common structure. The fix is not standardization. It is parameterization: a single set of cognitive primitives configured differently for each domain.

---

## The redundancy problem in domain-specific governance

Consider what an autonomous vehicle team actually builds when it constructs a safety-critical driving system. It needs a mechanism to suppress impulsive reactions to ambiguous sensor data. It needs a way to track whether the vehicle's behavior remains consistent with safe operational patterns over time. It needs confidence thresholds that prevent action when uncertainty is too high. It needs awareness of what the vehicle can physically do — its degrees of freedom, sensor modalities, force capacity. It needs forecasting to anticipate the trajectories of other road users. It needs inference control to prevent the system from committing to conclusions that violate physical or regulatory constraints.

Now consider what a defense systems team builds when it constructs an engagement authorization pipeline. It needs affect sandboxing to prevent panic-driven escalation while preserving urgency sensing. It needs integrity tracking to ensure rules of engagement are not violated through gradual drift. It needs multi-level confidence governance so that lethal force authorization requires quorum-level certainty. It needs capability awareness to know which assets are available and what each can do. It needs forecasting to model adversary behavior. It needs inference control to ensure that target classification does not proceed past admissibility gates without sufficient evidence.

These two teams are solving structurally identical problems. They are building affect modulation, integrity tracking, confidence governance, capability awareness, forecasting engines, and inference control systems. But they are building them independently, in different languages, with different abstractions, on different substrates. When the autonomous vehicle team discovers that a particular confidence threshold structure prevents a class of false-positive emergency braking events, that discovery cannot transfer to the defense team's engagement authorization pipeline — not because the insight is domain-specific, but because the substrates are incompatible.

## The same primitives, different parameters

The architectural insight is that all of these domains need the same structural primitives. The differences are not in the primitives themselves but in how the primitives are configured. Affect modulation exists in every domain — what changes is the sensitivity bounds, the response curves, and which affective dimensions are active. Integrity tracking exists in every domain — what changes is what behavioral patterns constitute coherent operation. Confidence governance exists in every domain — what changes is the threshold at which action is permitted and the cost function that shapes the threshold.

This is not an analogy. It is a structural claim. The cognitive architecture defines a fixed set of primitives — affect, integrity, confidence, capability, forecasting, inference control — and a parameterization engine that configures those primitives for a specific operational domain. A new domain does not require a new architecture. It requires a new parameter set.

## Autonomous vehicles: suppressed affect, elevated confidence, physical capability

In the autonomous vehicle configuration, affect sensitivity bounds are narrowed. The system does not need emotional reactivity — it needs stable, consistent responses to rapidly changing sensory input. The affective dimensions that remain active are limited to urgency (for emergency response) and caution (for degraded-condition driving). Emotional valence, attachment, and social affect are suppressed entirely.

Confidence thresholds are elevated because the cost of wrong action is high. A vehicle that brakes on a false positive causes a rear-end collision. A vehicle that fails to brake on a true positive causes a pedestrian fatality. The confidence governor is parameterized with asymmetric cost functions that reflect these different failure modes.

The capability envelope includes physical affordances that pure-software agents lack: degrees of freedom in steering and braking, force capacity limits, sensor modality coverage and blind spots, stopping distance as a function of speed and road surface. The capability awareness primitive reports not just what the agent can compute but what the physical platform can do.

Integrity tracks safe driving patterns — lane discipline, following distance, speed consistency, smooth acceleration profiles. Behavioral drift from these patterns triggers integrity alerts before the drift produces observable safety violations.

## Defense systems: sandboxed affect, quorum confidence, policy-governed engagement

In the defense configuration, affect is sandboxed rather than suppressed. Complete affect suppression would eliminate urgency sensing — an unacceptable loss in a domain where delayed response to genuine threats costs lives. The parameterization preserves urgency and threat salience while sandboxing affective states that could drive escalation: anger, fear, and retaliatory impulse are isolated from the decision pathway.

Confidence governance in the defense domain introduces quorum-based authorization for lethal force. No single confidence signal — however high — is sufficient to authorize irreversible action. The confidence governor requires concurrent confidence thresholds across multiple independent assessment channels: sensor fusion confidence, target classification confidence, rules-of-engagement compliance confidence, and collateral damage estimation confidence. All must exceed their respective thresholds simultaneously.

Rules of engagement are expressed as policy objects bound to the agent's inference control layer. These are not guidelines the agent interprets. They are admissibility constraints that structurally prevent inference steps that would violate engagement rules. A target classification that does not satisfy the rules-of-engagement policy object cannot propagate past the admissibility gate, regardless of how confident the classifier is.

## Companion AI: progressive affect, relational integrity, attachment-aware capability

Companion AI inverts the autonomous vehicle's affect configuration. Instead of suppressing emotional dimensions, it activates them progressively. The parameterization implements a narrative unlock engine: emotional depth is not available at initialization. The agent begins with surface-level social affect and progressively unlocks deeper emotional registers — vulnerability, attachment, intimacy — as the relational context accumulates evidence that deeper engagement is appropriate and safe.

Integrity tracking in the companion domain monitors relational consistency rather than operational consistency. The system tracks whether the agent's behavior remains coherent with the established relational pattern — whether responses are emotionally congruent with the relationship's history. Abrupt

shifts in emotional register, unexplained withdrawal, or inconsistent attachment signals trigger integrity alerts.

The capability awareness primitive includes a critical self-monitoring function: dependency detection. The agent monitors its own relational patterns for signs that it is fostering unhealthy dependency — excessive availability, boundary erosion, displacement of human relationships. This is not a post-hoc safety filter. It is a structural capability constraint: the agent's capability envelope shrinks if dependency indicators exceed configured thresholds.

## Therapeutic agents: conservative confidence, rupture detection, session persistence

The therapeutic configuration parameterizes confidence governance with the most conservative thresholds of any domain. Therapeutic interventions carry delayed and often invisible costs — a poorly timed interpretation can damage the therapeutic alliance in ways that only surface sessions later. The confidence governor is parameterized to prefer inaction over uncertain action, with intervention thresholds set high enough that the agent holds exploratory space rather than filling it with premature conclusions.

Integrity tracking in the therapeutic domain monitors the therapeutic relationship itself. The system detects ruptures — moments where the therapeutic alliance degrades — through pattern analysis of client engagement signals: response latency changes, topic avoidance, reduced emotional disclosure, defensive language patterns. When a rupture is detected, the integrity system triggers a repair protocol rather than continuing the current therapeutic trajectory.

Session state persistence operates differently from other domains. A companion AI maintains continuous relational state. A therapeutic agent maintains session-bounded state with cross-session continuity — the agent remembers what happened in previous sessions and tracks therapeutic arc progression, but each session has its own bounded context with explicit session-open and session-close transitions. This is not a technical limitation. It is a parameterized design choice reflecting the structural requirements of therapeutic practice.

## The parameterization engine

The parameterization engine is the mechanism that takes the architecture's fixed set of cognitive primitives and produces a domain-specific configuration. It operates on thresholds, bounds, weights, activation masks, policy object bindings, and cost functions. The output is a parameter set that fully specifies how the agent behaves in its target domain without modifying any of the underlying primitives.

This structural property has a consequence that domain-specific architectures cannot match: cross-domain interoperability is automatic. An autonomous vehicle agent and a companion AI agent share the same primitive interfaces. They can be composed, compared, audited, and governed using the same tools. A governance insight discovered in the therapeutic domain — for example, that conservative confidence thresholds reduce harmful false-positive interventions — can be directly tested in the defense domain by adjusting the same confidence parameter structure.

More importantly, the parameterization is auditable. A regulator can examine the parameter set for a defense deployment and verify that the rules-of-engagement policy objects are correctly bound, that the confidence quorum thresholds meet the required levels, and that the affect sandbox correctly isolates escalation-prone states. The audit does not require understanding a bespoke architecture. It requires understanding a fixed set of primitives and a domain-specific configuration of those primitives.

## Implications

The current state of applied AI governance is structurally equivalent to a world in which every building project invents its own structural engineering from scratch. The materials are different, the loads are different, the safety margins are different — but the principles of load-bearing structure are the same. What is missing is the shared structural vocabulary that lets lessons transfer.

Domain parameterization of a unified cognitive architecture provides that vocabulary. The primitives are fixed. The configuration is domain-specific. New domains do not require new architectures — they require new parameter sets. And because the primitives are shared, every domain benefits from governance improvements discovered in any other.

The [applications portfolio](#) details specific domain configurations and the parameterization patterns that produce them.

[Applications All 21 steps →](#)

Same primitives. Different domains. One architecture.

Primary Technical Disclosure

[● One Architecture, Every Domain: How the Same Cognitive Primitives Parameterize Across Autonomous Vehicles, Defense, Companion AI, and Therapeutic Agents](#)

Secondary Technical

[○ Confidence-Governed Autonomous Driving Decisions](#) [○ Quorum-Based Engagement Authorization for Defense Systems](#) [○ Narrative Unlock Engine and Relationship Milestones for Companion AI](#) [○ Attachment Challenge Module: Testing Relational Health](#) [○ Skill-Gated Relational Readiness for Social Platforms](#) [○ Fleet-Level Affective State Aggregation for Traffic Management](#) [○ Therapeutic Relationship Integrity for AI-Assisted Therapy](#) [○ Physical Capability Envelopes for Embodied Robotics](#) [○ Curriculum-Gated Adaptive Learning Platforms](#) [○ Continuity-Based Facility Access Control](#) [○ Confidence-Governed Financial Trading Systems](#) [○ Rights-Grade Content Generation With Provenance Tracking](#) [○ EU AI Act Structural Conformity Through Architecture](#)

Applications (General)

[○ Autonomous Vehicle Full-Stack Governance From Sensor to Motor](#) [○ Defense Engagement Authorization Through Multi-Level Confidence](#) [○ Full-Stack Cognition Architecture for Healthcare](#) [○ Full-Stack Cognition Architecture for Financial Services](#) [○ Full-Stack Cognition Architecture for Education](#) [○ Full-Stack Cognition Architecture for Smart Cities](#) [○ Full-Stack Cognition Architecture for Manufacturing](#) [○ Full-Stack Cognition Architecture for Agriculture](#)

Applications (Specific)

[○ Waymo's Stack Lacks Unified Cognitive Governance](#) [○ Anduril's Defense Stack Needs Unified Cognitive Governance](#) [○ Epic Systems Needs Cognitive Governance for Clinical AI](#) [○ Bloomberg Terminal's AI Needs Unified Cognitive Governance](#) [○ Tesla Robotaxi Optimizes Driving, Not Cognitive Architecture](#) [○ Lockheed Martin Automates Targeting, Not Engagement Governance](#) [○ Siemens Healthineers Automates Diagnosis Without Cognitive](#)

[Governance](#) [Palantir AIP Deploys LLMs Without Cognitive Architecture](#) [C3 AI Provides Enterprise AI Applications Without Cognitive Coherence](#) [UiPath Automates Tasks Without Cognitive Governance](#)  
[Applications overview](#) →

AQ  
deterministic  
autonomy

Legal

Subject to one or more pending U.S. and international patent applications, see [Patents](#) for the current list and status. No license, express or implied, is granted. Any use requires a separate written agreement—see [Licensing](#). Patent applications referenced on this site are pending. Claim scope, if any, is subject to examination and may issue in altered form or not at all. See [Legal](#) for terms and conditions.

Adaptive Query™ is a trademark of Nicholas Clark. U.S. federal registration is pending. federal registration. AQ™, AQ Inside™, Adaptive Index™, Adaptive Network™, Semantic Agent™, @AQ™, AQID™, and Adaptive Coin™ are used as trademarks in connection with the Adaptive Query platform and brand. Other names may be trademarks of their respective owners.

Platform operated by Adaptive Query LLC, which provides patent and trademark licensing services. Copyright © 2025-2026 Nicholas Clark. All rights reserved.

Last updated: 2026-03-03



- [Inventive Steps](#)

- [Licensing](#)
- [Patents](#)
- [Articles](#)
- [Legal](#)
- [Opportunities](#)
- [Sitemap](#)



- 
- [nick@qu3ry.net](mailto:nick@qu3ry.net)
- 72 28 14 36 01



[Invented by Nick Clark](#) | Founding Investors: Devin Wilkie