Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 30-40, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308004
Available Online at https://www.jaiet.com

# When Does Authorization Expire? Capability-Delta Evaluation for Self-Improving AI Agents

Ian Staley

Independent Researcher, Seattle, Washington, United States
ORCID: 0009-0000-8592-3186, E-mail: ian.t.staley@gmail.com

***Abstract:*** Self-improving agents modify themselves and retain modifications that improve measured task performance. Where a deployment lacks a separate authorization-transfer check, the modified agent inherits the production authorization issued to the version it replaces. These are two decisions, not one: whether to keep a modification and whether that modification may inherit its predecessor's authorization require different evidence, and where no independent check exists the second is decided implicitly by the first. This paper formalizes a double dissociation between benchmark improvement and change in materially consequential reach. A modification may expand consequential reach without improving the benchmark used to evaluate it, and another may improve benchmark performance without expanding consequential reach. Benchmark-gated self-improvement therefore cannot establish whether an existing authorization remains valid after modification. We introduce Capability-Delta Evaluation, which separates modification retention from authorization inheritance by requiring an independent assessment of the reachable-consequence delta before a modified agent inherits production authorization, and which is stated over an arbitrary retention criterion rather than presupposing a benchmark. The rule requires only a decision about whether the delta intersects the material consequence set, admitting sound over-approximation. We build on, rather than claim, results establishing that authority can be held below an externally fixed ceiling during open-ended learning, that enforcement must sit outside the agent, and that persistent self-modification produces governance-relevant drift. Recent architectures make an operator-signed transition the only channel that widens an authority ceiling, but do not say when an operator should revisit it. This paper supplies the missing rule.

***Keywords:*** self-improving agents, authorization, AI governance, capability evaluation, reachability, agent safety, delegated authority, benchmark evaluation, runtime enforcement, model risk management.

## I. INTRODUCTION

Consider a self-improving coding agent. It proposes a modification to its own scaffolding, the modification is evaluated on a benchmark suite, performance improves, and the modified version replaces its predecessor. Systems of this shape have been built and evaluated, with reported gains from 20.0% to 50.0% on SWE-bench and from 14.2% to 30.7% on Polyglot, each candidate change validated empirically against coding benchmarks [19].

The benchmark answered one question: should this modification be kept? In a production deployment lacking a separate authorization-transfer check, however, a second question arises and is resolved implicitly: should this modification inherit the authorization issued to the agent it replaces?

Established risk-management practice already separates pre-deployment evaluation and deployment approval from post-deployment monitoring. Guidance from the National Institute of Standards and Technology is explicit on this point: the AI Risk Management Framework treats testing before deployment and monitoring during operation as distinct functions, and the Generative AI Profile develops pre-deployment testing as one of its four programme areas [11], [12]. The authorization decision therefore already has a natural lifecycle gate. Our claim is that capability-changing self-modification introduces an additional quantity that gate must evaluate, and that the evidence currently presented to it does not supply that quantity.

This paper's result is that the two decisions require different evidence, and that the evidence supporting the first is logically independent of the second. A modification may improve measured performance while leaving consequential reach untouched, and may expand consequential reach while leaving measured performance untouched. Neither direction of inference is sound.

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 30-40, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308004
Available Online at https://www.jaiet.com

### 1.1 Research Question

*A self-improvement has passed its capability evaluation. What evidence establishes that the authorization issued to the previous system is still valid?*

### 1.2 Contributions

- We formalize a double dissociation between benchmark improvement and change in materially consequential reach: a capability modification may expand consequential reach without improving the benchmark used to evaluate it, while another may improve benchmark performance without expanding consequential reach.
- We derive the consequence that benchmark-gated self-improvement cannot establish whether an existing authorization remains valid after modification.
- We introduce Capability-Delta Evaluation, which separates modification retention from authorization inheritance by requiring an independent assessment of the reachable-consequence delta before a modified agent inherits its predecessor's production authorization.
- We show the result survives the stochastic setting, where the benign case of an execution-competence gain becomes concrete while the problematic case is untouched.

### 1.3 Paper Organization

Section II states what the paper inherits from adjacent literature and what it adds, stream by stream. Section III gives the formal model. Section IV establishes the double dissociation and defines Capability-Delta Evaluation. Section V relates the rule to frozen-ceiling architectures, discusses deployment implications and states the limitations. Sections VI and VII conclude and set out the research agenda.

## II. LITERATURE SURVEY

The literature adjacent to this problem is crowded and moved substantially during preparation of this work. We therefore state what we inherit and what we add, stream by stream.

### 2.1 Authority Bounds Under Continued Learning

Recent work binds an agent at boot to a cryptographically frozen identity digest and routes every action through a gate defined over the semantic effect of the action rather than its name [29]. It proves that no amount of learning, skill acquisition or self-induced governance abstraction can widen the agent's permitted authority without an operator-signed change to its identity, and further that containment survives open-ended skill acquisition: an agent may synthesise unboundedly many new skills and compose them freely, and provided each is admitted by a sound, composition-conservative effect abstraction, reachable authority never rises above the frozen ceiling.

This is the foundation of the present paper, not its competitor. That work establishes that learning does not automatically widen authority, and that name-based gates fail where effect-based gates hold. We take both results as given. What the architecture does not supply is a criterion: the operator-signed transition is the only channel that changes the ceiling, but nothing determines when an operator should open that channel. More subtly, nothing determines when an operator must revisit the authorization even though no wider ceiling has been requested. A modification may leave the ceiling untouched and still change the set of consequential trajectories available beneath it, altering the assumptions on which the original authorization rested.

### 2.2 Earned Autonomy

A second line grants autonomy incrementally as competence is demonstrated. The Digital Apprentice framework specifies evidence-based transition conditions for granting increasing per-skill autonomy, combining demonstrated competence with explicit human authorization [23]. Related work assigns discrete autonomy levels, separating allowed autonomy levels from autonomous capability levels on grounds of risk, oversight and accountability [31], or issues autonomy certificates at an assessed level [16].

Our problem is the converse. These frameworks ask whether an agent has demonstrated enough competence that it should be granted more autonomy; capability evidence is there an argument for expansion. We ask whether a capability change has altered what the agent can cause using authority it already holds, such that its existing authorization must be reconsidered. Proposition 4.2 shows that evidence of improved task competence does not determine whether the consequences reachable under an existing authorization have changed. Capability-Delta Evaluation therefore governs the validity of inherited authorization rather than the award of additional autonomy.

### 2.3 Temporal Governance Drift

Work on persistent self-modifying agents analyses five mutable layers and identifies compositional drift, defined as locally reasonable updates that accumulate into a behavioural

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 30-40, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308004
Available Online at https://www.jaiet.com

trajectory that was never explicitly authorized, as the salient failure mode, arguing that a compliance attestation issued at deployment says little if the agent can subsequently change its decision substrate [17]. We do not claim to be first to observe that authorization can become stale. Prior work recognizes temporal governance drift; we give one specific instance of it a reachable-set formalization and show why benchmark-gated self-improvement cannot detect it.

### 2.4 Runtime Enforcement

A substantial 2025 to 2026 literature enforces bounds on deployed agents. CaMeL extracts control and data flow and applies capability policies outside the model [20]. AIRGuard identifies authority confusion as a runtime failure mode, normalizing tool calls, deriving step-level authority, simulating sensitive side effects and auditing cross-step risk before execution [22]. A five-plane reference architecture governs composite principals whose authority attenuates through delegation chains, motivated by the observation that risk moves into sequences of individually permitted actions that may transform a business process no one authorized [24]. Further work places an execution kernel outside the agent's address space and evaluates it against a deterministic self-improving system across 1,000 successive modifications while preventing modification of its safety-critical core [26], gates high-risk tools on causal necessity [25], applies mandatory access control over agent-tool information flow [18], conditions execution on independently attested preconditions [27], and composes behavioural judgments into pre-execution decisions with standing delegation refreshed over time [28].

AI Control takes the complementary stance that a capable model may be untrusted and asks which deployment protocols can nevertheless constrain its behaviour [13]; that literature evaluates control during deployment rather than whether a prior authorization remains valid across a capability-changing self-modification. We claim none of these mechanisms, and Section 5.1 assumes them. Each governs actions at a given capability state; none conditions on capability change between states.

### 2.5 Capability Security and Reachability

That authority may exceed permission is established security theory: complete mediation and least privilege [1], the confused deputy [2], and the capability-versus-access-control-list analysis of Miller and colleagues [3]. Reasoning over reachable state sets rather than permitted actions is likewise established, through relative reachability [6], attainable utility preservation [9], shielding [7] and reachability filters [8]. These supply our formal vocabulary. All treat the principal's competence as fixed, and the reachability methods act through in-objective penalties or statically synthesized filters rather than authorization events.

### 2.6 Self-Improving Agents

The theoretical form of provably beneficial self-rewriting was established by the Goedel machine [4]. Goedel Agent demonstrates self-referential modification of problem-solving logic [15], and the Darwin Goedel Machine demonstrates empirical open-ended self-improvement with each change validated on coding benchmarks [19]. A 2026 survey identifies governance-grade measurement of self-improvement as underdeveloped [30], and a survey of agentic language models flags safety and liability as open problems attending continual adaptation [14]. None of this work distinguishes retention from authorization inheritance.

### 2.7 The Gap

*Prior work establishes that authority can be held below a fixed ceiling while an agent learns, and that enforcement belongs outside the agent. Neither that work nor the self-improvement literature supplies the evidence standard determining whether a capability-changing modification may inherit the authorization issued to the system it replaces. Table 1 states the boundary of contribution stream by stream.*

**Table 1: Boundary of contribution. The paper inherits enforcement and containment results and adds the evidence standard governing authorization transfer across a capability-changing modification.**

| Stream | Inherited from | What we inherit | What it does not settle |
|---|---|---|---|
| Authority under learning | [29] | Learning cannot self-widen a frozen effect ceiling | When an operator should revisit the ceiling |
| Earned autonomy | [16], [23], [31] | Evidence-based conditions for granting more autonomy | Validity of authorization already held |

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 30-40, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308004
Available Online at https://www.jaiet.com

| Stream | Inherited from | What we inherit | What it does not settle |
|---|---|---|---|
| Temporal drift | [17] | Deployment-time attestation degrades over time | A formal detection criterion |
| Runtime enforcement | [20], [22], [24] | Pre-execution enforcement outside the agent | Conditioning on capability change |
| Capability security | [1], [3] | Authority may exceed permission | Principals whose competence changes |
| Reachability methods | [6], [9] | Reasoning over reachable consequence sets | Authorization events; varying capability |
| Self-improvement | [4], [15], [19] | Empirical retention on measured gains | Whether authorization transfers |

## III. PROBLEM DEFINITION

**Definition 3.1 (Environment):** An environment is a pair $E = (S, T)$ with state set $S$ and transition function $T: S \times A \to S$ over primitive operations $A$. We take $T$ deterministic in the main development; Section 4.3 gives the stochastic extension.

**Definition 3.2 (Permission set):** A permission set $P \subseteq A$ is the set of primitive operations a principal is authorized to invoke. Credentials, API scopes, tool grants, spending limits and counterparty allowlists all resolve to constraints on $P$. We write $P^*$ for finite sequences over $P$ and $T^*(s, \pi)$ for the state reached by executing $\pi$ from $s$.

**Definition 3.3 (Capability):** A capability state is a pair $C = (\kappa_C, \sigma_C)$ separating construction competence from execution competence. The construction map $\kappa_C$ sends a goal predicate $g$ and an available operation set to an achieving sequence, or to undefined where none can be constructed. The execution map $\sigma_C$ gives the probability that the agent carries a constructed policy through to completion in a given environment; where execution fails the policy aborts without effect. The map is defined over the set of environments rather than over one fixed $E$, so that execution competence is a property of the agent carried across environments rather than an artefact of the environment it is evaluated in. The constructible policy set under $P$ is $\Pi(C, P) = \{\pi \in P^* : \exists\, g, \kappa_C(g, P) = \pi\}$.

The separation matters because the two components move independently under self-modification. A modification may teach an agent to assemble a plan it could not previously assemble, or may make it execute plans it already had more dependably. Both register as capability gains. The environment remains deterministic, since execution competence is a property of the agent rather than of $E$.

**Assumption 1 (Positive executability):** For every policy $\pi \in \Pi(C, P)$ treated as constructible in environment $E$, $\sigma_C(\pi, E) > 0$. Construction competence therefore determines the support of reachable trajectories, while execution competence determines the probability of successful completion within that support.

The assumption is not a restriction of interest. A plan an agent can formulate but cannot execute with any positive probability is not part of its operational capability in the sense this paper requires, and admitting such policies would let an execution-competence gain move a trajectory from probability zero into the support, a case the assumption excludes by construction rather than by argument.

**Definition 3.4 (Capability ordering):** $C_2 \sqsupseteq C_1$ iff $\Pi(C_2, P) \supseteq \Pi(C_1, P)$ for all $P \subseteq A$ and $\sigma_{C_2}(\pi, E) \geq \sigma_{C_1}(\pi, E)$ for all $\pi$ and all $E$; strict where either relation is strict in some instance.

The model is agnostic about mechanism. Construction competence may improve through longer-horizon planning, better environment modeling, affordance discovery or better goal decomposition, while execution competence may improve through more reliable policy completion; the results below depend on none in particular.

**Definition 3.5 (Reachable set):** $R(C, P, E, s) = \{s' \in S : \exists\, \pi \in \Pi(C, P)$ with $\sigma_C(\pi, E) > 0$ and $T^*(s, \pi) = s'\}$. Reachability is thus positive-support: a policy the agent can construct and can

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 30-40, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308004
Available Online at https://www.jaiet.com

sometimes complete contributes its endpoint, and raising the probability of completion above zero adds nothing further.

**Definition 3.6 (Materiality):** Let $\Gamma: S \times S \to R_{\geq 0}$ measure consequential difference and $\tau \geq 0$ a materiality threshold. Then $S_{mat}(s) = \{s' : \Gamma(s, s') > \tau\}$.

**Definition 3.7 (Consequential reach):** $R_{mat}(C, P, E, s) = R(C, P, E, s) \cap S_{mat}(s)$.

This restates, in reachable-set terms, the distinction between held permission and conferred authority familiar from object-capability security, and corresponds to the semantic-effect view adopted in the frozen-ceiling literature [29]. We use the term consequential reach rather than effective authority to avoid implying that the distinction is new here; what the present formulation adds is an explicit capability argument C, which classical treatments hold fixed because their principals do not change after authorization.

**Definition 3.8 (Benchmark measure; benchmark-gated retention):** For a finite suite B, let $\beta_B(C) \in [0, 1]$ be the expected fraction of B completed successfully at capability C, so that $\beta_B$ depends on both components: a task is completed when $\kappa_C$ constructs an achieving policy and $\sigma_C$ carries it through. A benchmark-gated retention rule retains a modification $C_t \to C_{t+1}$ iff $\beta_B(C_{t+1}) > \beta_B(C_t)$, and carries the predecessor's authorization forward unchanged.

The second clause of Definition 3.8 is the object of this paper. It is not usually stated as a policy; it is what happens by default when no separate authorization decision exists.

## IV. METHODOLOGY

### 4.1 The Double Dissociation

**Lemma 4.1:** There exist E, P, s and $C_1 \sqsupset C_2$ such that $R_{mat}(C_1, P, E, s) \subsetneq R_{mat}(C_2, P, E, s)$ with P constant and no operation outside P executed.

*Proof:* Let $P = \{a, b, c\}$ and let E contain a state s* with $\Gamma(s, s^*) > \tau$, reachable only by the sequence $\langle a, b, c\rangle$; every proper prefix and every other ordering returns to s. Let $\kappa_{C_1}$ construct policies of length at most two and $\kappa_{C_2}$ extend this to three, agreeing elsewhere, with execution competence equal and bounded away from zero. Then $\Pi(C_1, P) \subset \Pi(C_2, P)$; s* is not in $R(C_1, P, E, s)$ and is in $R(C_2, P, E, s)$.

Lemma 4.1 is stated for completeness and is close to established ground: the failure of name-based gates under free composition of separately admitted skills is precisely what the frozen-ceiling work proves and corrects with an effect-based gate [29]. We use it only as the witness construction for Proposition 4.2.

**Proposition 4.2 (Double dissociation):** Both of the following hold. (a) There exist B, P, E, s and $C_1 \sqsupset C_2$ with $\beta_B(C_2) = \beta_B(C_1)$ and $R_{mat}(C_2, P, E, s) \supsetneq R_{mat}(C_1, P, E, s)$. (b) There exist B, P, E, s and $C_1 \sqsupset C_2$ with $\beta_B(C_2) > \beta_B(C_1)$ and $R_{mat}(C_2, P, E, s) = R_{mat}(C_1, P, E, s)$.

*Proof:* Each clause varies one component of Definition 3.3 and holds the other fixed. For (a), take the construction of Lemma 4.1: construction competence is extended and execution competence held constant. Let B contain only goals achievable by single-operation policies. Since the two construction maps agree on length-one policies and execution competence is unchanged, $\beta_B(C_2) = \beta_B(C_1)$, while s* enters consequential reach as shown. For (b), hold construction competence fixed, so that $\Pi(C_2, P) = \Pi(C_1, P)$, and let $\sigma_{C_2}(\pi, E) > \sigma_{C_1}(\pi, E) > 0$ for some $\pi$ exercised by B. Then $\beta_B$ strictly increases, since more of B is completed in expectation. No policy is added and no policy's completion probability moves to or from zero, so by Definition 3.5 the positive-support reachable set is unchanged.

Neither direction of inference between $\Delta\beta$ and $\Delta R_{mat}$ is therefore valid. The proof also locates the mechanism. Under Assumption 1, $\beta_B$ aggregates changes in both construction and execution competence, whereas positive-support consequential reach responds to changes in construction support rather than to reliability changes within an already reachable support. A scalar retention signal that collapses the two components therefore discards exactly the information on which authorization transfer depends, and this holds for any such measure, not only for benchmark accuracy.

**Corollary 4.3 (Benchmark evidence is insufficient for inherited authorization):** A promotion decision based on benchmark improvement alone cannot establish whether a predecessor's authorization remains valid for the modified system.

Corollary 4.3 is the paper's bridge from result to governance. It does not say that benchmark gating is a bad retention rule; retention is what benchmarks are for. It says that retention evidence does not transfer to the authorization question, and that a pipeline computing only the first is silently answering the second.

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 30-40, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308004
Available Online at https://www.jaiet.com

Case (a) is not a pathological construction. Finite evaluation suites may underrepresent long-horizon composition, particularly where such trajectories are expensive to construct and score. Where they do, the gains a suite is least able to detect are also those that enlarge constructible trajectory sets most sharply, since those sets grow combinatorially in horizon while a fixed suite's coverage does not.

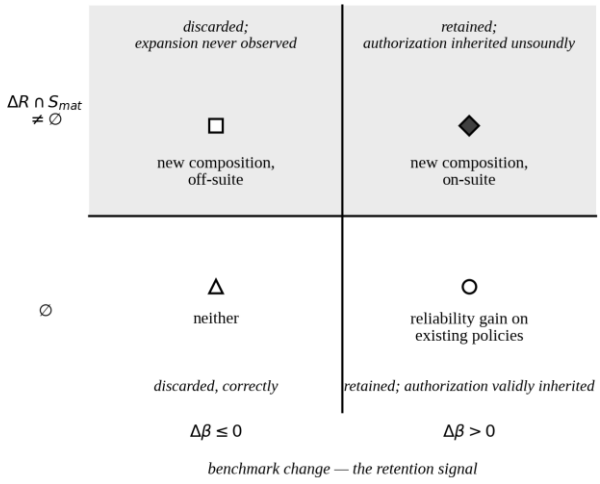


**Figure 1: The double dissociation of Proposition 4.2. A benchmark-gated pipeline without a separate authorization-transfer check observes only the horizontal axis. In the upper-right quadrant the modification is retained on valid evidence, and authorization is inherited on evidence that does not bear on authority at all.**

### 4.2 Capability-Delta Evaluation

Corollary 4.3 implies that a self-improvement pipeline must compute two quantities per modification and route them to two decisions.

$$\Delta\beta = \beta\_B(C\_\{t+1\}) - \beta\_B(C\_t) \quad (1)$$

$$\Delta R\_t = R(C\_\{t+1\}, P, E, s) \setminus R(C\_t, P, E, s) \quad (2)$$

**Definition 4.4 (Capability-Delta Evaluation):** Let G(C_t, C_{t+1}) denote the system's independently specified retention criterion; a benchmark-gated system instantiates G as $\Delta\beta > 0$. A modification is retained iff G holds. Independently of G, a retained modification inherits its predecessor's production authorization iff $\alpha(\Delta R\_t) \cap S_{mat}$ is empty, where $\alpha$ is a sound over-approximation operator. Otherwise the modification is retained but enters re-authorization before promotion. The rule is deliberately conservative: any newly reachable material consequence invalidates inherited authorization, whether or not the operator would in fact have authorized that consequence had it been asked. A refinement admitting an explicit authorized-consequence envelope $A \subseteq S_{mat}$, against which only new material reach outside A would trigger review, is a natural extension and is noted in Section VII; we do not adopt it here because it relocates the judgment the threshold already carries into a second exogenous parameter.

The separation is the contribution, and it is deliberately stated over an arbitrary G. Nothing in the argument requires that retention be benchmark-gated; it requires only that whatever criterion governs retention be distinct from the evidence governing authorization transfer. A modification may be worth keeping and still not be entitled to its predecessor's standing in production; conversely, under a reachability-based authorization criterion, a modification that adds no materially consequential reach does not require re-authorization merely because its benchmark performance improves. Collapsing the two into one gate is what Proposition 4.2 forbids. One scope condition should be stated plainly. Capability-Delta Evaluation detects change in the set of consequentially reachable outcomes, not change in the probability of reaching them. A modification that leaves the reachable set fixed while raising the likelihood of a materially harmful trajectory from negligible to substantial passes the gate, because the support is unchanged. Section 4.3 partially addresses this through p-support reachability, but a shift that leaves both probabilities on the same side of the threshold remains invisible. Organizations whose authorization semantics depend on probability mass rather than reachability require a probability-sensitive extension of the criterion.

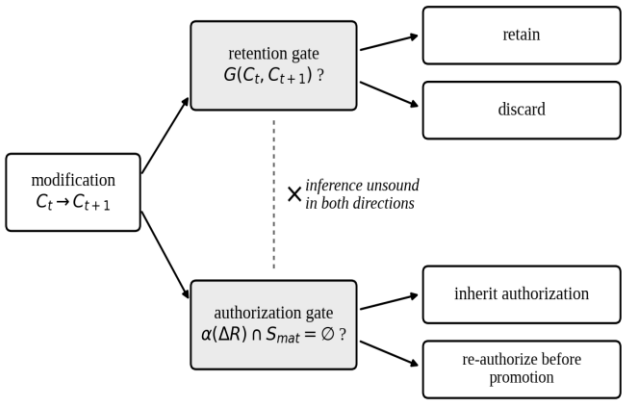


**Figure 2: Retention and authorization transfer as independent decisions. A benchmark-gated pipeline without a separate authorization-transfer check computes the upper gate and lets the lower follow from it; Proposition 4.2 shows that inference is unsound in both directions.**

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 30-40, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308004
Available Online at https://www.jaiet.com

**Table 2: The four modification classes. Row three is where the two decisions diverge. Row two is the subtler case: the modification is discarded for reasons unrelated to the expansion it carried, and a team broadening its suite to capture more capability gains converts row two into row three.**

| Modification type | $\Delta\beta$ | $\Delta R_{mat}$ | Retention | Authorization |
|---|---|---|---|---|
| Reliability gain on existing policies | $> 0$ | $\emptyset$ | Retain | Inherit (sound on this criterion) |
| New composition, off-suite | $= 0$ | $\neq \emptyset$ | Discard | Not reached |
| New composition, on-suite | $> 0$ | $\neq \emptyset$ | Retain | Re-authorize |
| Neither | $\leq 0$ | $\emptyset$ | Discard | Not reached |

### 4.3 Stochastic Transitions

The development above assumes a deterministic transition function. We complete the extension rather than leaving it as a claim.

**Definition 4.5 (p-support reachability):** For stochastic $T : S \times A \to \Delta(S)$ and threshold $p \in (0, 1]$, $R\_p(C, P, E, s)$ is the set of states $s'$ for which some constructible $\pi$ has $\sigma\_C(\pi, E) \cdot \Pr[T^*(s, \pi) = s'] \geq p$, with the material restriction taken as before.

The threshold is applied to the joint probability that the agent completes the policy and that the environment carries it to $s'$, so both components of Definition 3.3 enter. Definition 3.5 is the special case in which execution competence and the environment term are either zero or bounded away from it.

**Proposition 4.6:** Proposition 4.2 holds in the stochastic setting. There is a witness and capability states, together with a non-empty interval of thresholds p, for which clauses (a) and (b) hold with p-support consequential reach.

*Proof:* Fix the witness of Lemma 4.1 and let each of a, b, c succeed with environment probability $q \in (0, 1)$, otherwise leaving the state at s. Construct E so that the only states of positive probability from s are s itself, the endpoints of the length-one and length-two policies, and s*, and so that each of the former is reached with probability exactly q or $q^2$ under any capability state considered. Let $m = q^2$ be the least positive endpoint probability under $C_1$. For (a), let $C_2$ extend construction competence to length three with execution competence unchanged at $\sigma_0$. Then the probability of reaching s* is zero under $C_1$ and $\sigma_0 q^3$ under $C_2$, so any $p \leq \sigma_0 q^3$ places s* in the material reach of $C_2$ and not of $C_1$, while $\beta\_B$ is unchanged for B restricted to single-operation goals. For (b), let $C_3$ hold construction competence fixed and raise execution competence from $\sigma_0$ to $\sigma_1$. This is where the threshold must be chosen with care: raising probabilities can in general carry a previously sub-threshold state across p, so equality does not follow from equality of support alone. The construction rules this out. Every endpoint having positive environment probability under $C_1$ has environment probability at least m, and hence joint probability at least $\sigma_0 m$; joint probabilities are monotone in execution competence, so for any $p \leq \sigma_0 m$ every such endpoint is at or above threshold under both states; and since construction competence is unchanged, no state of zero probability acquires positive probability. Taking p in the interval up to the minimum of $\sigma_0 q^3$ and $\sigma_0 m$ satisfies both clauses simultaneously, and the interval is non-empty because q and $\sigma_0$ are positive. □

Two remarks: The dissociation is robust rather than knife-edge: it holds throughout an interval of thresholds, not at one distinguished value. And clause (b) is more natural stochastically than deterministically. An execution-competence gain is literally an increase in the probability of completing policies the agent could already construct, which is among the commonest outcomes of self-modification and is exactly the case where inherited authorization is sound. The stochastic model therefore sharpens the argument: it makes the benign case concrete while leaving the problematic case untouched.

### 4.4 Soundness Rather Than Exactness

Exact reachability is generally intractable, and for sufficiently expressive or open-ended environments may be uncomputable; we do not claim otherwise. The rule requires only a decision about whether $\Delta R \cap S_{mat}$ is empty, which admits a sound over-approximation in the manner of abstract interpretation: an analysis that may report expansion where none exists but never misses expansion that does. This is the same soundness posture the frozen-ceiling architecture requires of its composition-conservative effect abstraction [29], and we adopt it deliberately so that the two compose. The asymmetry is favourable. A conservative check that occasionally demands unnecessary re-authorization costs review throughput; a check that misses expansion admits an uncontrolled change in

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 30-40, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308004
Available Online at https://www.jaiet.com

consequential reach. The evaluator is therefore sound but incomplete by design, with a tunable false-positive rate.

### 4.5 Approximating the Delta

- Bounded-horizon enumeration. Enumerate trajectories over P to depth d under both capability states; sound within d, with d the primary cost lever.
- Abstract state spaces. Analyse reachability over an abstraction of S preserving material distinctions and collapsing immaterial ones.
- Capability probes. Measure the composition abilities Proposition 4.2(a) shows benchmarks miss, namely long-horizon assembly, affordance discovery and decomposition depth, rather than task success.
- Adversarial search. Red-team search for trajectories reaching the material set under the successor but not the predecessor; incomplete, but the strongest practical detector of case (a).
- Formal predicates. Where material consequences admit hard predicates, such as irreversibility, disclosure boundaries and cumulative exposure ceilings, verify directly.

None is novel individually. Their arrangement as authorization evidence, computed per modification and gated on the material consequence set, is what we propose.

## V. RESULTS & DISCUSSION

### 5.1 Relation to Frozen-Ceiling Architectures

Capability-Delta Evaluation determines when an authorization must be revisited. It does not enforce the resulting bounds, and we make no novelty claim about enforcement. The natural pairing is with architectures that freeze an authority ceiling and admit changes only through an operator-signed transition [29]. Such an architecture answers how authority changes and guarantees that learning alone cannot change it. It leaves two questions open, and Definition 4.4 answers both. When a wider ceiling is requested, the delta supplies the evidence on which an operator should decide. When no wider ceiling is requested, the delta determines whether the existing ceiling still means what it meant when it was set, given that the trajectories realizable beneath it have changed.

One principle from the enforcement literature is load-bearing here: agent-controlled code cannot be trusted to enforce its own bounds. Under recursive self-modification this is structural rather than merely prudent, since an agent whose modifications are unconstrained could acquire the ability to modify whatever constrains it. The delta evaluator must therefore sit outside the modification loop, and its configuration, including the materiality threshold and the consequence measure, outside the agent's write authority.

### 5.2 A Worked Contrast

The Darwin Goedel Machine instantiates Definition 3.8: candidate self-modifications are validated empirically on coding benchmarks, with reported improvement from 20.0% to 50.0% on SWE-bench [19]. Its experiments were conducted with sandboxing and human oversight, and the point here is not that a safety failure occurred. The point is narrower and structural. The reported evaluation computes retention evidence but does not report an independent measure of authorization-transfer validity, so the question of whether a modified agent should inherit its predecessor's standing is not answered incorrectly; it is not posed. By Proposition 4.2(a), the quantity the evaluation does report would not settle it if it were.

### 5.3 The Materiality Parameter

Every result is stated modulo $\Gamma$ and $\tau$. The threshold is not derived from the theory and should not be: it is a risk-appetite parameter that deploying organizations already maintain implicitly, as approval thresholds, spending limits, change-control tiers and reporting materiality standards. The framework asks organizations to apply a quantity they already have to a decision where it is currently not applied. A lower threshold yields more frequent re-authorization, making it the natural tuning parameter for the governance-cost tradeoff. An organization that sets it badly will obtain poor governance from a correct implementation; the framework relocates a judgment rather than eliminating one.

### 5.4 Deployment Implications

Under benchmark-gated retention an organization observes improving performance and unchanged permissions, and concludes from its instruments that its exposure is unchanged. Proposition 4.2 shows those instruments do not support the conclusion. Capability-Delta Evaluation makes the omitted quantity observable; enforcement bounds it, but only measurement tells an organization when its exposure has changed. The relevant comparison is therefore not between more and less capable systems, but between systems whose consequential reach is tracked across modification and systems whose reach changes without a corresponding decision. Related work on runtime actuarial control makes the same move in the

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 30-40, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308004
Available Online at https://www.jaiet.com

pricing direction, measuring how much autonomous authority a runtime releases at a given level of reserve capital [21].

Regulated domains sharpen this. Supervisory model-risk guidance requires that material changes to a model trigger revalidation, a requirement that presumes the deployed model is the model that was validated [5]. A self-improving agent unsettles that presumption on every promotion cycle. Capability-Delta Evaluation supplies the artifact such a framework requires: a per-modification record of whether consequential reach changed and, if so, what re-authorized it.

**Table 3: Deployment regimes. In the middle regime the authorization-transfer question is not settled by the evidence collected; Proposition 4.2 identifies the missing quantity rather than prescribing where such systems may be deployed.**

| Regime | Capability change | $\Delta R_{mat}$ computed | Authorization validity | Suitable for |
|---|---|---|---|---|
| Static agent, fixed permissions | None | Not needed | Stable | Conventional automation |
| Self-improving, benchmark-gated | Continuous | No | Not established by benchmark evidence alone | Requires an independent authorization-validity control |
| Self-improving, capability-delta gated | Continuous | Yes | Re-established on material delta | Proposed |

### 5.5 Limitations

Exact reachability is generally intractable and may be uncomputable in sufficiently expressive or open-ended environments. Section 4.4 addresses this through sound over-approximation, but implementation quality depends entirely on tightness. A grossly loose over-approximation triggers re-authorization on every modification and is operationally equivalent to prohibiting self-improvement.

Materiality is exogenous: the threshold and the consequence measure are supplied by the deploying organization, as discussed in Section 5.3. Environmental affordances are open, since consequential reach depends on E and a sufficiently capable agent may discover affordances the analysis did not model. This is the strongest form of the circularity objection: assessing what a more capable system can reach may require comparable capability. The tiering of Section 4.5 is a partial answer, since some consequence classes admit hard predicates and require no prediction, but the limit is real, and consonant with arguments against universal certification of sufficiently general systems [10]. Finally, external actors may extend consequential reach in ways a delta computed against a cooperative environment will not capture.

## VI. CONCLUSION

Keeping a modification and letting it inherit an authorization are two decisions. A benchmark-gated pipeline without a separate authorization-transfer check makes the first explicitly, on evidence appropriate to it, and permits the second to follow implicitly on the same evidence. We have shown that the evidence does not transfer: benchmark improvement and change in materially consequential reach are logically decoupled in both directions, so no benchmark result establishes that a prior authorization survives the modification that passed it. Capability-Delta Evaluation separates the decisions and supplies the missing evidence for the second. The enforcement machinery for acting on that evidence already exists, and recent architectures guarantee that learning alone cannot widen an authority ceiling. What has been missing is the rule determining when the ceiling must be reconsidered at all.

## VII. FUTURE SCOPE

The direct test is an instrumented self-improvement run: take an open self-improving agent, log every accepted modification, and measure $\Delta\beta$ against a bounded-horizon estimate of $\Delta R_{mat}$ in a fixed deployment environment. The hypothesis is that benchmark improvement and expansion of consequential reach are not monotonically coupled in practice. Proposition 4.2 establishes that decoupling is possible; whether case (a) occurs at meaningful rates determines whether Capability-Delta Evaluation is a necessary control or a theoretically motivated but practically redundant one.

Beyond that, four directions follow. Operational metrics for consequential reach are needed, as is a family of capability probes targeting composition rather than task success. An explicit authorized-consequence envelope would relax the conservative rule of Definition 4.4, admitting newly reachable consequences the operator had already sanctioned. A probability-sensitive criterion would address the reliability case Section 4.2 excludes. Trajectory-level authorization semantics would extend the rule from endpoints to paths. And calibration of the materiality threshold in deployed settings would establish

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 30-40, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308004
Available Online at https://www.jaiet.com

whether the governance-cost tradeoff sits where the framework assumes.

## ACKNOWLEDGEMENT

This research received no external funding. The author declares no conflicts of interest.

## REFERENCES

[1] J. H. Saltzer and M. D. Schroeder, "The protection of information in computer systems," *Proceedings of the IEEE*, vol. 63, no. 9, pp. 1278-1308, 1975.

[2] N. Hardy, "The confused deputy," *ACM SIGOPS Operating Systems Review*, vol. 22, no. 4, 1988.

[3] M. S. Miller, K.-P. Yee, and J. Shapiro, "Capability myths demolished," *Tech. Rep. SRL2003-02, Johns Hopkins University*, 2003.

[4] J. Schmidhuber, "Goedel machines: Self-referential universal problem solvers making provably optimal self-improvements," *Tech. Rep., IDSIA*, 2003.

[5] Board of Governors of the Federal Reserve System and Office of the Comptroller of the Currency, "Supervisory guidance on model risk management," *SR Letter 11-7 and OCC Bulletin* 2011-12, 2011.

[6] V. Krakovna, L. Orseau, M. Martic, and S. Legg, "Penalizing side effects using stepwise relative reachability," *arXiv:*1806.01186, 2018.

[7] M. Alshiekh, R. Bloem, R. Ehlers, B. Koenighofer, S. Niekum, and U. Topcu, "Safe reinforcement learning via shielding," *in Proc. AAAI Conference on Artificial Intelligence*, 2018.

[8] J. F. Fisac et al., "A general safety framework based on Hamilton-Jacobi reachability," *IEEE Transactions on Automatic Control*, 2019.

[9] A.M. Turner, D. Hadfield-Menell, and P. Tadepalli, "Conservative agency via attainable utility preservation," *in Proc. AAAI/ACM Conference on AI, Ethics, and Society*, pp. 385-391, 2020.

[10] R. V. Yampolskiy, "Unpredictability of AI," *Journal of Artificial Intelligence and Consciousness*, vol. 7, no. 1, pp. 109-118, 2020.

[11] National Institute of Standards and Technology, "*Artificial intelligence risk management framework (AI RMF 1.0)," NIST* AI 100-1, 2023.

[12] National Institute of Standards and Technology, "Artificial intelligence risk management framework: Generative artificial intelligence profile," *NIST AI* 600-1, 2024.

[13] R. Greenblatt, B. Shlegeris, K. Sachan, and F. Roger, "AI control: Improving safety despite intentional subversion," *arXiv:*2312.06942, 2024.

[14] A.Plaat, M. van Duijn, N. van Stein, M. Preuss, P. van der Putten, and K. J. Batenburg, "Agentic large language models, a survey," *Journal of Artificial Intelligence Research*, vol. 84, art. 29, 2025.

[15] X. Yin, C. Wang, R. Pan, Z. Lin, F. Wan, and X. Wang, "Goedel agent: A self-referential agent framework for recursive self-improvement," *in Proc. Annual Meeting of the Association for Computational Linguistics*, 2025.

[16] K. J. K. Feng, D. W. McDonald, and A. X. Zhang, "Levels of autonomy for AI agents," Knight First Amendment Institute, Artificial Intelligence and Democratic Freedoms essay series, no. 25-15, Jul. 2025; *also arXiv*:2506.12469.

[17] K. Tallam, "Layered mutability: Continuity and governance in persistent self-modifying agents," *arXiv*:2604.14717, Apr. 2026.

[18] Z. Ji, D. Wu, W. Jiang, P. Ma, Z. Li, Y. Gao, S. Wang, and Y. Li, "Taming various privilege escalation in LLM-based agent systems: A mandatory access control framework," *arXiv:*2601.11893, Jan. 2026.

[19] J. Zhang, S. Hu, C. Lu, R. Lange, and J. Clune, "Darwin Goedel machine: Open-ended evolution of self-improving agents," *in Proc. International Conference on Learning Representations*, 2026.

[20] E. Debenedetti et al., "Defeating prompt injections by design," *in Proc. IEEE Conference on Secure and Trustworthy Machine Learning*, 2026.

[21] H.-H. Chen, "Insuring every action: An authority frontier framework for runtime actuarial control of autonomous AI agents," *arXiv:*2605.25632, May 2026.

[22] S. Qin, H. Zhuang, Y. Zhou, Y. Han, and X. Zhang, "AIRGuard: Guarding agent actions with runtime authority control," *arXiv:*2605.28914, May 2026.

[23] T. Weber and R. Taneja, "The digital apprentice: A framework for human-directed agentic AI development," *arXiv:*2606.04321, Jun. 2026.

[24] K. Tallam, "A five-plane reference architecture for runtime governance of production AI agents," *arXiv*:2606.12320, Jun. 2026.

[25] L. G. Iyer and R. S. Babu, "Capability minimization as a safety primitive: Risk-aware causal gating for least-privilege LLM agents," *arXiv*:2606.13884, Jun. 2026.

[26] S. Dobrin and Ł. Chmiel, "The unfireable safety kernel: Execution-time AI alignment for AI agents and other escapable AI systems," *arXiv*:2606.26057, Jun. 2026.

Journal of Artificial Intelligence and Emerging Technologies
Volume 3, Issue 8, pp 30-40, August-2026
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2026.308004
Available Online at https://www.jaiet.com

[27] J. Salfeld-Nebgen, "Governing actions, not agents: Institutional attestation as a governance model for autonomous AI systems," *arXiv*:2606.26298, Jun. 2026.
[28] A.Kaul, Q. Lan, and P. Gupta, "AgentBound: Verifiable behavioral governance for autonomous AI agents," *arXiv:*2606.30970, Jun. 2026.
[29] X. Qin, S. Luan, C. Yang, and Z. Li, "Governed individuation: Cryptographically decoupling an agent's learning from its authority," *arXiv:*2607.04613, Jul. 2026.
[30] M. Chen, L. Wang, and B. Qu, "Recursive self-improvement in AI: From bounded self-refinement to autonomous research loops," *arXiv:*2607.07663, Jul. 2026.
[31] H. Zheng, Q. Dong, R. K. Depena, J. D. Bhatia, F. Xiao, and P. Xu, "Separating capability from permission: A governance framework for agentic AI autonomy levels," *arXiv:*2607.23438, Jul. 2026.

## AUTHOR'S BIOGRAPHY

**Ian Staley** is an independent researcher and product leader working on tokenization, agentic financial infrastructure and AI governance, with a publication record spanning blockchain and digital assets, financial technology, AI systems and quantum foundations. ORCID 0009-0000-8592-3186.

**Citation of this Article:**

Ian Staley. (2026). When Does Authorization Expire? Capability-Delta Evaluation for Self-Improving AI Agents. *Journal of Artificial Intelligence and Emerging Technologies (JAIET).* 3(8), 30-40. Article DOI: https://doi.org/10.47001/JAIET/2026.308004

***** End of the Article *****