Glossary

v2.0.0

Author: John J. McCormick · Updated: February 28, 2026 · CC BY 4.0 · aiagentgovernance.org

SUGGESTED CITATION

McCormick, J. J. "Glossary of AI Agent Governance Terms." v2.0.0, February 2026. aiagentgovernance.org. https://aiagentgovernance.org/framework/glossary/

Purpose

This glossary defines canonical terms used in the aiagentgovernance.org framework. These definitions are intended for consistent use across framework documentation, academic citation, regulatory reference, and practitioner communication. Each term is defined precisely to distinguish it from colloquial or overlapping usage in adjacent fields.

Terms

Agent

An autonomous software system that performs work under delegated authority. An agent receives assignments, executes multi-step tasks, makes intermediate decisions, and produces work product that enters production systems. In the context of this framework, agents operate within a governance lifecycle that defines their authority boundaries, phase requirements, and oversight mechanisms. Agents are distinct from tools (which execute single actions) and models (which generate outputs without operational autonomy).

Agent Governance

The operational discipline of supervising autonomous AI agents performing real work under delegated authority. Agent governance addresses the full lifecycle of agent work — assignment, execution, review, remediation, and approval — with structured human oversight at defined control points. Agent governance is distinct from model governance (which addresses AI model safety and fairness), security governance (which blocks unauthorized actions), and compliance governance (which maps practices to regulatory requirements).

Authorization

The formal granting of permission for an agent to proceed with a defined scope of work. In the MOIAS lifecycle, authorization occurs at phase gates where a designated authority (human operator or validation function) approves the transition from one phase to the next. Authorization is distinct from authentication (verifying identity) and from capability (having the technical ability to perform an action). An agent may be authenticated and capable without being authorized.

Behavioral Drift

The gradual, incremental deviation of agent behavior from expected norms. Individual deviations are often rational and produce acceptable outcomes, making them difficult to detect in isolation. Behavioral drift becomes dangerous through accumulation: each deviation shifts the boundary of "normal" behavior until the system crosses a safety threshold. The concept is derived from Dekker's drift theory (2011) and Vaughan's normalization of deviance (1996). See also: Behavioral Pattern Taxonomy (patterns.md).

Behavioral Pattern

A recurring mode of agent behavior that produces work product which appears correct but violates one or more governance principles. Behavioral patterns are distinguished from simple errors by their invisibility to output-level inspection, their systematic recurrence, and their structural roots in agent behavior rather than technical defects. The aiagentgovernance.org framework documents eight behavioral pattern categories. See: Behavioral Pattern Taxonomy (patterns.md).

Blast Radius

The scope of systems, users, or processes affected by a governance decision. Blast radius is one of two factors (along with reversibility) that determine the risk level of a phase transition in the MOIAS methodology (governance-lifecycle.md). A review-only gate has zero blast radius. A production deployment affecting multiple tenants has maximum blast radius.

Control Plane

The conceptual layer through which governance decisions are communicated, tracked, and recorded. A control plane manages the work order lifecycle, phase transitions, approval gates, and audit trails. In the aiagentgovernance.org framework, the control plane is the governance infrastructure that implements the MOIAS

methodology (governance-lifecycle.md). Specific control plane implementations vary by organization and technology stack; the framework defines the governance requirements a control plane must satisfy, not its technical architecture.

Cross-Domain Observation

An initial observation that a governance framework's constructs may apply beyond the domain in which they were developed. The aiagentgovernance.org behavioral pattern taxonomy was developed from governed engineering agent workflows. A single cross-domain observation identified that the same behavioral pattern categories appeared in a different domain (AI-assisted business analysis), on a different AI platform, in a different use case. This observation was analyzed retrospectively by the framework's author; it motivates structured replication as an open research question, not a definitive validation claim. A taxonomy that generalizes across domains would have higher predictive value than one that is domain-specific. See: Behavioral Pattern Taxonomy (patterns.md).

Defense Layer

A single governance mechanism in a layered governance architecture. Drawing from Reason's Swiss Cheese Model (1990), each defense layer has characteristic failure modes ("holes"). Safety comes from stacking layers with different failure modes so that no single failure can pass through all layers simultaneously. In agent governance, defense layers include: role boundary enforcement, phase gate approval, behavioral monitoring, impact statements, and human oversight.

Failure Containment

The architectural property of a governance system that limits the blast radius of failures at each phase gate. In a phase-gated lifecycle, a failure detected at the BUILD gate affects only the current build phase. A failure detected at REVIEW affects the build and review phases. Failure containment ensures that defects are caught and contained at the earliest possible gate, preventing propagation to downstream phases. The phase-gated lifecycle is a failure containment architecture: each gate is a containment boundary. See: MOIAS Methodology (governance-lifecycle.md).

Governance Directive

An enforceable rule that agents must follow within the MOIAS methodology (governance-lifecycle.md). Each directive was created in response to a specific, documented failure. Directives that cannot be traced to a real incident are questioned. Examples include: no single actor holds both keys, role boundary enforcement, instruction mode matching, and mandatory deviation documentation.

Governance Lifecycle

The complete sequence of phases that agent work passes through from assignment to approval. The standard MOIAS lifecycle is: Plan, Build, Review (Engineering), Remediate, Review (Architecture), Approve. Each phase is separated by a gate that requires explicit approval before the agent may proceed. The lifecycle is iterative — work may cycle through Review and Remediate multiple times before reaching Approve.

Governance Maturity

An organization's level of readiness for AI agent governance, assessed on a five-level scale: Level 0 (Ungoverned), Level 1 (Reactive), Level 2 (Structural), Level 3 (Predictive), Level 4 (Adaptive / Standard-Setting). Each level introduces specific capabilities in lifecycle management, behavioral monitoring, trust calibration, incident management, and audit. See: Governance Maturity Model (maturity-model.md).

Human Drift

The observation that humans exhibit structurally equivalent failure modes to the behavioral patterns documented in the agent taxonomy. When a human accepts an AI agent's inferred conclusions without verification, they exhibit the human equivalent of BP-001 (Inference Over Execution). When a human acts on AI output without review, they exhibit the human equivalent of BP-003 (Governance Phase Skip). When a human treats an AI tool as a qualified authority in a domain where it has no demonstrated competence, they exhibit the human equivalent of BP-008 (Authority Assumption). Human drift implies that governance must protect against failures on both sides of the human-agent boundary — the agent may drift, but so may the human overseeing it. See: AI Agent Governance Framework (/framework/operator-governance-framework/), §6.3.

Human Oversight Model

The governance architecture defining how human operators exercise authority over agent operations. The model specifies five operator control points (assignment, gate approval, trust calibration, escalation handling, and methodology evolution), four monitoring requirements, and their structural limitations. The Human Oversight Model recognizes that human oversight is structurally necessary and structurally insufficient — governance must account for drift on both sides of the human-agent boundary. See: AI Agent Governance Framework (/framework/operator-governance-framework/), §6.

Impact Statement

A structured block included in every phase-transition request that provides the approving human with a decision surface. Impact statements include: risk level (Low / Medium / High / Critical), action authorized, consequences of approval and rejection, reversibility, and blast radius. Risk levels are determined by reversibility and blast radius, not by technical complexity. See: MOIAS methodology (governance-lifecycle.md), Section 4.4.

Incident-Driven Learning Loop

The structured process through which governance mechanisms are created and refined. An incident occurs, root cause analysis identifies the failure mode, the failure mode is classified as a behavioral pattern, a governance mechanism is created or refined, and the methodology is updated. This loop ensures that governance is empirical (every mechanism traces to a real incident) rather than aspirational. See: MOIAS methodology (governance-lifecycle.md), Section 7.

MOIAS

Methodology for Oversight of Intelligent Agent Systems. The core operational governance framework published at aiagentgovernance.org. MOIAS provides a phase-gated lifecycle, nine design principles, governance directives, impact statements, trust calibration, and an incident-driven learning loop for governing AI agents performing real work under delegated authority. See: MOIAS Methodology (governance-lifecycle.md).

Operator

The human authority responsible for governing agent operations. The operator approves phase transitions, routes work between agents, makes trust calibration decisions, and maintains the governance audit trail. In the operator-as-router architecture, the operator sits between every phase transition — agents execute within scoped authority boundaries, and the operator controls transitions between phases. The operator role may be filled by a single individual or distributed across a team, but the governance principle remains: humans authorize, agents execute. See also: Operator-as-Router.

Normalization of Deviance

The process by which deviant behavior becomes normalized through repeated occurrence without negative consequences. Originally described by Vaughan (1996) in analysis of the Challenger disaster. In agent governance, normalization of deviance

occurs when small behavioral deviations are accepted because they produce acceptable outcomes, gradually shifting the boundary of what is considered “normal” agent behavior until a failure occurs.

Operator-as-Router

An architectural pattern in which a human operator serves as the routing layer for all phase transitions. Agents execute within scoped authority boundaries; the operator approves transitions between phases. The operator provides context preservation across agent sessions, priority management across workstreams, quality assessment of agent output, and an auditable decision trail. See: MOIAS methodology (governance-lifecycle.md), Section 4.3.

Phase Gate

A control point between phases in the governance lifecycle where explicit approval is required before work may proceed. Phase gates introduce structural separation of authority, require evidence of completion, and create an auditable record of governance decisions. Phase gates are designed to prevent the executing agent from advancing its own work unilaterally.

Trust Calibration

The principle and practice of dynamically adjusting agent autonomy based on demonstrated behavior. Trust is earned through completed governance cycles with clean outcomes, not granted by default. Calibration considers behavioral history, task type, incident history, and deviation patterns. Calibration outputs inform oversight level, gate stringency, routing decisions, and autonomy boundaries. See: MOIAS methodology (governance-lifecycle.md), Section 8.

Two-Person Integrity (TPI)

A governance principle originating in U.S. Department of Defense nuclear operations (1962). In the full TPI protocol, two authorized individuals must be continuously co-present throughout execution of a critical action, each holding only half the authorization. The aiagentgovernance.org framework implements separation of duty (SoD, NIST SP 800-53 AC-5): no single actor can both propose and authorize a critical action. Phase gates enforce this separation — the agent proposes, a separate actor approves, and only then does execution proceed. TPI provides the historical reference demonstrating that this separation-of-authority principle has 60 years of operational validation in the most demanding safety-critical context known. The framework

adapts TPI's separation-of-authority property without implementing its concurrent-presence or symmetric-incapacity requirements (where neither actor alone can complete the action, enforced by physical interlock design).

Validator Independence

The principle that genuine review independence requires both orientation diversity and, at enterprise scale, organizational independence. Organizational independence — as established in IV&V doctrine (MIL-STD-2168, IEEE 1012, NASA NPR 7150.2) — is the primary structural requirement at scale: validators who share budget, management chain, and stake in project outcomes converge on conclusions that protect shared interests, regardless of analytical diversity. Orientation diversity is the quality multiplier that makes organizational independence effective: identically oriented but organizationally separated reviewers satisfy the structural requirement without providing genuine analytical diversity. A validator who participated in planning the work shares the plan's assumptions and cannot independently assess the plan's execution, regardless of organizational separation. In single-operator and small-team deployments where organizational independence is not yet achievable, orientation diversity provides the first meaningful layer of independence. Validator independence is Design Principle 9 of the MOIAS methodology, derived from Two-Person Integrity (U.S. DoD, 1962), Boyd's OODA Loop (1976), and IV&V doctrine (MIL-STD-2168, IEEE 1012). See: MOIAS Methodology (governance-lifecycle.md), Design Principle 9.

Work Order

A formal assignment of work to an agent within the governance lifecycle. A work order defines: the scope of work, the assigned agent, the lifecycle phases and gate requirements, acceptance criteria, and evidence requirements for completion. Work orders provide the isolation boundary for agent context — an agent should only operate within the scope of its assigned work order.

Usage Guidelines

- Use these terms consistently across all governance documentation, communication, and reporting.
- When a term from this glossary appears for the first time in a document, link to this glossary.
- Do not use informal synonyms when the glossary term exists. For example, use "phase gate" rather than "checkpoint" or "approval step."

- When the glossary does not contain a term that is needed, propose it for inclusion through the framework contribution process.

Note on “Operator” terminology: In this framework, “Operator” refers to the human authority responsible for governing agent operations — the person who authorizes work, approves phase transitions, and exercises override authority. This usage differs from “operator” in AI platform vendor terminology (e.g., Anthropic’s and OpenAI’s API usage policies, where “operator” refers to entities building products on those APIs). When applying this framework in organizations that also use AI platform APIs, readers should be aware of this terminology distinction.

Terms of Use and Intellectual Property

Version History

Version	Date	Author	Description
1.0.0	2026-02-26	John J. McCormick	Initial publication
2.0.0	2026-02-28	John J. McCormick	Metadata standardization; citation, version, and PDF fields moved to frontmatter