DECISION DNA GROUP

The Orchestration Governance Framework

OGF v1.2.1 | A proposed standard for multi-platform, multi-agent AI governance

This document defines what governed orchestration requires. It is a standard, not a product. Organizations implement it through whatever combination of process, tooling, and organizational commitment satisfies the requirements.

Executive Summary

Enterprise AI has matured its coordination layer faster than its authority layer. The protocols that enable agents to connect and communicate, MCP and A2A, are moving rapidly into enterprise adoption. What does not yet exist is the governance infrastructure that ensures agents are organizationally authorized to act on what those protocols enable. That gap is where the most urgent work lives, but it is not the only work.

What this framework defines

- 14 requirements across 2 layers
- 6 single-platform governance components
- 8 multi-platform authority requirements

Most organizations reading this are still in the middle of the first six. That is the right place to start. You cannot govern what you cannot yet see.

Requirements 7 through 14 extend the single-platform foundation into the multi-platform environment, covering cross-platform agent inventory, pre-deployment authority definition, authority inheritance protocols, cross-platform provenance tracking, system-level behavioral monitoring, evidence-based rule governance, version-controlled authority records, and unified accountability attribution. Some are achievable today through organizational commitment. Others describe infrastructure the market is still building toward. The OGF names all of them because clear requirements are how markets organize, build, and mature.

Who this is for

Enterprise governance leaders, AI operations teams, platform architects, and compliance professionals deploying or designing multi-agent systems.

Three actions any organization can begin this week, without procurement

- Inventory every agent in your environment and assign a named owner to each.
- Name a Logic Product Manager accountable for governance rule accuracy in your highest-volume agent deployment.
- Document the governance rules for your most consequential agent with full provenance, including the reasoning behind each rule and when it should be reconsidered.

How to use this document

This framework has two layers. Section A defines the single-platform governance foundation: the six components organizations need before attempting multi-platform orchestration. Section B defines the OGF itself: the eight requirements for governed orchestration across platforms. You cannot implement Section B without Section A. The maturity table in Section D shows where you are and what to build next.

SECTION 0

Scope and Non-Scope

What this framework covers and what it deliberately does not

The Orchestration Governance Framework addresses a specific and bounded problem: the authority governance gap that emerges when AI agents coordinate across platform boundaries. It is not a comprehensive AI governance standard. It does not replace, duplicate, or compete with existing governance disciplines. Understanding what the OGF covers and does not cover is necessary to apply it correctly.

OGF covers	OGF does not cover
Cross-platform authority governance: ensuring agents are sanctioned to act in the context of a multi-platform orchestration chain	Model governance: the technical management of AI model development, training, evaluation, and versioning
Authority inheritance: governed handoff of authorization context when tasks pass between agents across platform boundaries	Cybersecurity and identity access management: authentication, secrets management, threat detection, and zero-trust architecture
Cross-platform provenance tracking: maintaining an auditable record of authorization decisions across a multi-agent chain	Content guardrails: enforcement of output quality, safety, and compliance at the inference layer
System-level behavioral governance: monitoring orchestrated systems for emergent outcomes invisible at the individual agent level	Single-platform governance in isolation: addressed in Section A as the prerequisite foundation, not as OGF scope
Accountability attribution: tracing multi-agent outcomes to specific authorization decisions and named human accountabilities	Data governance: data quality, lineage, classification, and access outside the context of agent authorization
Governance debt: the accumulated liability from operating orchestration systems without cross-platform authority infrastructure	Vendor selection: the OGF defines requirements; it does not evaluate, endorse, or recommend specific products or platforms

Key Terms

Precise definitions for terms used throughout this framework

The following definitions apply throughout this document. Where a term has an established meaning in an existing standard or regulation, this document uses the established meaning unless otherwise noted.

Term	Definition
Agent	An AI system that perceives inputs, processes them, and takes actions or produces outputs with a degree of autonomy within a configured boundary. For the purposes of this framework, an agent is distinguished from a static model or API by its capacity to take consequential actions on behalf of an organization.
Authority	The organizational sanction for an agent to take a specific action on behalf of the organization, granted by a human with the standing to grant it, under conditions that are defined and remain current. Authority is distinct from capability. An agent may be capable of an action it is not authorized to take.
Capability	The technical ability of an agent to perform an action given its access, tools, and instruction set. Capability is determined by the agent's configuration and the permissions granted at the platform level.
Consequential action	Any agent action that creates, modifies, or removes a record, commitment, payment, communication, or decision that has operational, financial, legal, or regulatory implications for the organization or its stakeholders.
Governance rule	A documented specification of what an agent is authorized to do, under what conditions, within what limits, and with what escalation triggers. A governance rule is the foundational unit of the authority record..
Orchestration chain	A sequence of two or more agents coordinating to accomplish a shared goal, where at least one task handoff occurs across a platform or vendor boundary.
Platform boundary	The point at which a task, context, or decision moves from one vendor's AI infrastructure to another's, or from a vendor's infrastructure to an internally built system.
Provenance	A complete, auditable record of the authorization decisions that produced a given agent action or orchestration outcome, including who authorized each step, when, on what evidence, and under what governance rules.
Substantial modification	Under the EU AI Act, a change to an AI system after it has been placed on the market or put into service that affects compliance with the Act or changes its intended purpose relative to the original conformity assessment, triggering treatment of the modifying party as a provider for that modified system under Article 25. In a multi-agent context, whether chaining agents into a new decision pipeline constitutes substantial modification is fact-dependent and assessed against the original conformity scope of each agent involved. This document treats it as an interpretation requiring legal analysis, not a settled rule.
Tethered agent	An agent executing within a fixed instruction set configured at deployment time, with no automatic mechanism to update that instruction set when organizational policy, regulatory requirements, or operational conditions change.

SECTION A

The Single-Platform Governance Foundation

The prerequisite for governed orchestration: six components, four maturity stages

Governed orchestration is not possible without a governed foundation. Before any organization can address the cross-platform authority problems that orchestration introduces, it must have the single-platform governance infrastructure in place for the agents it already runs. This section documents that foundation: what each component requires, what good looks like, and how common good practice actually is.

Most organizations reading this are at Stage 1 or early Stage 2. That is not a failure. It is a starting point. The value of knowing your stage is that it tells you what to build next rather than what you should have built already.

Component	What it does	What good looks like	Owns	How common is good	Compliance
PHASE 0: BEFORE DEPLOYMENT					
0a Shadow AI Discovery	Complete inventory of every AI tool and agent: sanctioned and unsanctioned. You cannot govern what you have not found.	Automated discovery running continuously. Every agent has a named owner. Shadow AI surfaced and assessed.	Shared	Continuous automated discovery: very few orgs. Periodic manual: some.	EU AI Act, GDPR, ISO 42001
0b Pre-Deployment Governance	Formal authorization before go-live: purpose, scope, initial rule set, and first provenance record.	Structured intake. Initial rule set documented before first production action. Authority boundaries approved by named human.	Company	Basic config before go-live: most. Formal authorization with provenance: very few.	EU AI Act, NIST AI RMF, ISO 42001
PHASE 1: RUNTIME					
1 Continuous Behavioral Monitoring	Always-on detection: action volume, decision distribution, escalation rate, and reasoning chain visibility.	Real-time anomaly alerts. Reasoning visibility beyond outcome logging. Audit trail calibrated to regulatory requirements.	Shared	Outcome logging: most. Governance-grade reasoning capture: very few.	EU AI Act, NIST AI RMF, HIPAA, ISO 42001
2 Dedicated AI Operations Function	Named team with explicit accountability for agent governance as an ongoing operational responsibility.	Business lead, technical lead, governance lead with defined authority. Primary job is governance accuracy, not platform uptime.	Company	Dedicated team: few. Named owner: more common in regulated industries.	EU AI Act, NIST AI RMF, ISO 42001
PHASE 2: CONTINUOUS IMPROVEMENT					
3 Escalation Review Cadence	Regular structured review of every escalation. Root cause diagnosis across three categories: threshold, process, model drift.	Weekly cadence. Three-category framework. Vendor model releases reviewed as standing agenda item.	Company	Informal periodic review: some. Weekly cadence: very few.	NIST AI RMF, EU AI Act, ISO 42001
4 Pre-Deployment Simulation	Required validation before any logic change. Proposed change run against 30 days of historical data.	Every logic change backtested. Simulation results attached as evidence. Pass threshold required.	Company	Manual testing: some. Systematic backtesting: very few.	SR 11-7 (fin. services), EU AI Act, NIST AI RMF

5 Change Approval Process	Defined decision rights: which changes AI ops approves, which require business sign-off, which require senior leadership.	Three approval tiers. Simulation evidence required for Tier 2 and 3. Rollback procedure defined before go-live.	Company	Informal approval: most. Tiered framework: very few.	SOX, EU AI Act, ISO 42001, SOC 2
6 Authority layer	Version-controlled record of every governance rule: what it is, why it exists, evidence, approver, reconsideration trigger.	Full provenance including reasoning. Machine-readable. Reconsideration triggers assigned at creation.	Company	Config files and change logs: most. Full provenance with reasoning: very few.	EU AI Act, NIST AI RMF, HIPAA, ISO 42001

The governance layer -- all six components above -- is typically fragmented across tooling and internal process and is rarely provided as a complete governance operating layer by any single vendor. This is the gap most organizations do not see.

SECTION B

The Orchestration Governance Framework

Eight requirements for multi-platform, multi-agent AI governance: Phase 3 of the maturity framework

B1. The Definitional Statement

Orchestration governance is the discipline of establishing and enforcing authority boundaries across multi-agent, multi-platform AI systems before and during coordination, not as a constraint on orchestration, but as the foundation that makes orchestration trustworthy.

Where orchestration engineering solves the question of how agents coordinate, orchestration governance solves the prior question of whether agents have the authority to coordinate, under what conditions, within what constraints, and with what accountability when coordination produces unintended outcomes.

The tools and protocols for agent coordination have matured faster than the governance infrastructure required to authorize what that coordination produces. That gap is not a feature of immature technology. It is a structural property of how the market has sequenced the problem.

B2. The Sequencing Argument

The enterprise AI market has built the coordination layer first and treated governance as something to add later. That sequencing is not neutral. Many architectural decisions made in designing an orchestration pipeline implicitly allocate authority, constraint propagation, and accountability without a governance framework in place to examine them.

When an enterprise defines how agents decompose tasks, it has implicitly decided which agent has authority to initiate. When it defines how agents pass work to each other, it has implicitly decided what constraints travel with the task. When it defines what happens at boundary conditions, it has implicitly decided who is accountable for the outcome. These decisions do not wait for the governance framework to arrive. They are made in the engineering sprint, encoded in the pipeline logic, and deployed into production.

The governance principle

Governance built after orchestration is archaeology. It tells you what happened. It does not prevent what should not have happened. The correct sequencing is: define the authority structure first, then design the coordination layer to operate within it.

B3. The Three-Problem Taxonomy

Multi-platform orchestration introduces three governance problems that do not exist in single-agent, single-platform deployments. They are structurally new, emerging specifically from the interaction between agents across platform boundaries.

Problem 1: Authority Inheritance

When Agent A passes a task to Agent B on a different platform, the governance constraints that governed Agent A's action do not automatically travel with the task. The technical handoff is handled by A2A. The authority handoff has no equivalent mechanism. Agent B proceeds on capability without any mechanism to validate whether it is organizationally authorized to act.

Without authority inheritance protocols, every receiving agent in a multi-platform chain risks becoming a confused deputy: technically authenticated, organizationally ungoverned, inheriting whatever authority claim the sending agent carried with no mechanism to validate its legitimacy.

Problem 2: Accountability Diffusion

When an outcome is produced by a chain of agents across multiple platforms, accountability cannot be attributed to any single agent, platform, or governance regime without a cross-platform provenance record. Each agent operated within its own platform's constraints. None exceeded its individual authorization. And yet the chain produced an outcome wrong by the standards of the organization's actual operating logic. This is Logic Hallucination: not the AI fabricating facts, but the organization fabricating governance. The system did not fail technically. It failed institutionally.

Here is the specific risk the EU AI Act creates for multi-agent deployments. Under the Act, an enterprise is classified as a Deployer when it uses an AI system for its intended purpose. Under Article 25, it may be reclassified as a Provider when it substantially modifies that system. Depending on the facts, chaining agents into a multi-step decision pipeline may raise substantial-modification questions if the resulting system affects compliance or changes intended purpose relative to the original conformity assessment of any agent in the chain. If reclassification occurs, the enterprise takes on full Provider liability: conformity assessments, technical documentation, CE marking, and ongoing monitoring obligations that most enterprises have not prepared for.

The penalty structure for the most serious violations under the Act reaches up to 35 million euros or 7 percent of global annual turnover, whichever is higher; non-compliance with high-risk system obligations carries penalties up to 15 million euros or 3 percent. The governance problem is this: without a cross-platform provenance record, an enterprise may be unable to demonstrate that its orchestrated deployment is a governed use of individual systems rather than a substantial modification that created a new one.

Problem 3: Emergent Behavior Governance

Emergent behavior governance is the hardest of the three problems to make visible because no individual system is doing anything wrong.

Consider two agents operating correctly within their own rules. A scheduling agent prioritizes appointments based on clinical urgency signals. A documentation agent flags certain patient conditions more frequently for follow-up based on its own configuration. Neither agent was designed to discriminate.

Neither agent's governance validation surfaces a problem. But together, over weeks of interaction, their combined logic produces a pattern: certain patients consistently receive earlier appointments and more follow-up attention than others, not because of clinical urgency but because of how the two agents' independently correct rule sets happen to interact.

Research in healthcare and other high-stakes domains shows that individually well-performing components do not necessarily yield fair or intended system-level outcomes, underscoring the need for governance at the system level rather than only at the component level. The failure exists only at the interaction level. No audit of either agent individually reveals it.

Under the EU AI Act, compliance ultimately attaches to system outcomes and risk categories, not merely to whether individual components passed isolated validation. An enterprise whose multi-agent system produces a discriminatory pattern cannot rely on individual component compliance as a defense. A governed system requires governance at the system level.

B4. The Protocol Gap

MCP materially advanced the connectivity problem: how agents access tools, data, and APIs. A2A materially advanced the coordination problem: how agents communicate and delegate tasks to each other. Both protocols achieved rapid, broad adoption because they addressed real structural gaps in the enterprise AI architecture.

What neither protocol was designed to solve is the organizational authority problem: whether an agent is sanctioned by the organization to act in this specific context, under current governance policy, as part of this orchestration chain. That is a different problem at a different layer.

> **The precise claim**
> MCP and A2A tell agents how to talk to each other. An organizational authority layer tells agents whether they are allowed to act on what they said. Both are necessary. Neither substitutes for the other

What the protocol provides	What the protocol does not provide at the organizational authority layer
MCP: Standardized agent-to-tool connections with OAuth-based technical authorization	Organizational authority governance: whether the agent is sanctioned by the organization to act in this specific context, under current governance policy. Technical access permission is not organizational authority.
A2A: Structured task delegation between agents with authenticated, scoped communication	Whether the originating agent had organizational authority to initiate the task. A2A governs the technical handoff. It does not govern whether the organization sanctioned that handoff under its current governance position.
Technical authentication and OAuth-based authorization: verified agent identity and scoped resource access	Organizational authorization: the governance record that an agent's actions in this context were sanctioned by a named human with the standing to sanction them, under conditions that remain current.

Task parameters: spending limits, approved categories	Whether those parameters reflect current organizational governance intent
Capability scoping: what tools an agent can access	What decisions an agent is sanctioned to make with those tools

B5. The Tethered Agent Problem and Governance Debt

The Tethered Agent Problem

The market describes AI agents as autonomous systems. They are not. Every agent is tethered: executing with full capability within boundaries set at configuration time by whoever deployed it, with no mechanism to verify whether those boundaries reflect the organization's current governance intent. The tether is static. The environment it operates in is not. Organizational policy changes. Regulatory requirements evolve. The agent's tether does not know any of this.

The market is building protections against agents that exceed their authority. The more pervasive risk is agents that execute faithfully within authority that was never quite right. The rogue agent is detectable: it deviates from expected behavior and monitoring systems register it. The tethered agent produces no signal. Every audit comes back clean. The rogue agent is the threat the market is spending on. The tethered agent is the threat the market is under-addressing.

Logic Hallucination

Logic Hallucination is what the tethered agent problem produces at the organizational level. Not the AI making up facts. The organization making up governance: believing its configured agents reflect its actual governance intent, when what they actually reflect is a collection of individually set tethers, each potentially correct at configuration time, none systematically maintained against the organization's evolving governance position.

Governance Debt

Governance Debt is the accumulated liability that results from operating multi-agent orchestration systems without the governance infrastructure those systems require. Unlike technical debt, which accumulates in code that can be refactored, Governance Debt accumulates in decisions already made, in agent interactions already produced, and in regulatory exposure already accruing.

Layer	Current state
MCP (connectivity layer)	Now a founding project of the Agentic AI Foundation under the Linux Foundation. Over 97 million monthly SDK downloads, 10,000+ active servers, with first-class support across ChatGPT, Claude, Cursor, Gemini, Microsoft Copilot, and VS Code.
A2A (coordination layer)	Rapidly gaining support across major enterprise vendors and ecosystem partners including AWS, Cisco, Google, Microsoft, Salesforce, SAP, and ServiceNow as founding Linux Foundation members.

Homegrown layer (the patchwork)	Fills every gap the protocols and platforms do not cover. Built reactively, person-dependent, and ungovernable at scale. In many organizations, this patchwork constitutes the operating reality of AI governance today.
Authority layer	Elements exist within individual platforms. What does not yet exist is the cross-platform version: a unified, vendor-agnostic authority layer governing the full orchestration chain. That gap is the Governance Debt.

The market timing dynamic

The better MCP and A2A get at solving the coordination problem, the more clearly the governance void becomes visible. In March 2026, OpenAI published findings from their internal agent monitoring system showing that agents can be overly eager to work around restrictions in ways invisible to output-only review, and that the developer prompt can inadvertently encourage circumvention. Reliable, high-velocity, cross-platform agent coordination operating without cross-platform authority infrastructure does not reduce risk. It accelerates it.

B5b. The Badge: Capability Assessment as of March 2026

The three-layer architecture makes the governance gap precise. The Tool Belt and the Handshake exist as mature, deployable infrastructure. The Badge does not yet exist as a unified cross-platform layer. The market is in motion. Some Badge capabilities exist within individual platforms. Others are emerging in partial form. And some have not yet been built in any deployable form.

Capability	Current state and notes
Exists within a single platform	
Single-platform authority records	Platform governance tools maintain version-controlled governance rules within their own environment. Governance is well-defined within the platform boundary. Does not extend across platforms.
Agent behavioral monitoring (single platform)	Enterprise AI platforms increasingly provide behavioral monitoring within their own environments. Coverage is platform-scoped. No cross-platform view.
Cross-platform agent security inventory	Agent security platforms can discover and inventory agents across platforms. Primary lens is security threat detection rather than authority governance. Req 7 partially addressed from a security posture perspective.
Version-controlled configuration management	Mature engineering organizations maintain version-controlled agent configurations. Captures what changed. Typically does not capture why, who approved, or what evidence justified the change.
Exists in partial or emerging form	
Cross-platform agent inventory (governance lens)	Tools are emerging that provide agent inventory across platforms with a governance rather than security orientation. Early stage. Coverage across all agent types including homegrown is inconsistent.
Runtime content governance gateways	Content governance tools provide enforcement at the inference layer. Governs what agents say at the output boundary. Does not govern what agents are authorized to decide or whether cross-platform chains are authorized.
Protocol-level provenance and governance discussions	MCP and A2A communities are actively exploring governance maturation. The MCP 2026 roadmap identifies governance as an explicit development area. No deployable standard for authority provenance metadata yet exists.
Regulatory compliance audit trails	Some organizations are building cross-platform audit logs to satisfy EU AI Act and NIST AI RMF requirements. Typically per-platform logs manually correlated. Not unified provenance tracking.
Non-human identity governance	Non-human identity platforms govern what AI agents can access at the identity and permission layer. Addresses Req 7 and partially Req 8. Does not govern

	authority inheritance or chain provenance as organizational governance artifacts.
System-level behavioral monitoring (single-platform)	Leading AI organizations have begun deploying monitoring systems that review agent reasoning chains, not just outputs. Published findings from March 2026 confirm that reasoning-chain monitoring surfaces behaviors invisible to output-only review. The cross-platform version does not yet exist in deployable form.
Does not yet exist as deployable infrastructure	
Unified cross-platform authority governance layer	A vendor-agnostic, cross-platform infrastructure where governance rules across every platform are maintained in a single governed, version-controlled record accessible to all agents in the chain. Req 13 at the system level.
Runtime cross-platform authority check-in	A mechanism where agents verify their current authorization against a central authority layer before taking consequential actions across platform boundaries. Both sides of the handoff performing governed verification. Req 9.
Unified cross-platform provenance chain	An unbroken authorization record linking every action across every vendor in a multi-agent chain from initiation to execution. Not per-platform logs. A unified chain auditable as a system. Req 10.
Cross-platform emergent behavior monitoring	System-level monitoring that detects interaction-level patterns emerging from the combined behavior of multiple agents across different platforms. Single-platform versions are emerging. The cross-platform version governing multi-vendor orchestration chains does not yet exist. Req 11.
Unified cross-platform accountability attribution	The ability to trace a multi-agent outcome back through a complete cross-platform authorization chain and identify who was accountable for each decision at each step. Req 14.

This assessment reflects the state of the market as of March 2026. The agentic governance space is moving rapidly. If you are aware of solutions that should update any of these assessments, please reach out. Contact: chance@decisiondnagroup.com

B5c. How the Governance Gap Fails in Practice: Three Scenarios

The following scenarios illustrate how cross-platform orchestration fails under current governance controls, and how OGF requirements change the outcome. Each scenario involves agents that are individually compliant. The failure is a property of the chain, not the components.

Scenario 1: Financial Services: Procurement Authorization Chain

Chain	A vendor evaluation agent on Platform A identifies a qualifying supplier and creates a purchase recommendation. It hands off to a contract drafting agent on Platform B, which generates a contract and routes it to an approval orchestration agent on Platform C.
Where governance fails	Agent A's supplier qualification criteria were updated six months ago following a new vendor risk policy. The configuration was not updated. Agent B's template parameters include a liability cap that legal revised three months ago. Agent C's approval routing reflects a pre-reorganization org chart. The contract that reaches the CFO was assembled from three outdated instruction sets. In a single system, a stale configuration is a detectable, correctable point failure. Across three platforms with independent configuration surfaces and no unified provenance record, the compound error is invisible to any single team, any single audit, and any single-platform monitoring system.
OGF controls	Foundation Req 6 (Authority layer): all three agents maintain version-controlled governance rules with reconsideration triggers. Req 8 (Pre-Deployment Authority Definition): chain context review identifies dependencies before go-live. Req 10 (Cross-Platform Provenance Tracking): the CFO can see that all three agents' configurations predate relevant policy changes.

Scenario 2: Healthcare: Clinical Decision Support Chain

Chain	A patient intake agent flags high-priority cases. It passes flags to a scheduling agent on a separate platform, which allocates appointment slots. Both agents feed data to a documentation agent that populates care coordination records.
Where governance fails	The intake agent's urgency flags weight certain diagnostic codes more heavily. The scheduling agent's priority rules, set by a different team, happen to correlate high-priority slots with certain access categories. Over time, their interaction produces a pattern that appears to disadvantage certain patients relative to clinical urgency expectations. The pattern exists only in the compound behavior of two instruction sets configured independently. A single-platform audit of either agent returns clean results.
OGF controls	Req 11 (System-Level Behavioral Monitoring): interaction-level pattern detection identifies the emerging correlation across the chain, invisible to single-agent monitoring. Req 9 (Authority Inheritance Protocols): the scheduling agent verifies urgency flags before acting, surfacing the combined logic. Req 14 (Accountability Attribution): when concern is raised, the pattern is traceable to specific configuration decisions at both agents.

Scenario 3: Enterprise: Multi-Platform Customer Engagement Chain

Chain	A CRM agent identifies customer renewal risk and triggers a response workflow. It passes the customer record to a pricing agent on Platform B, which calculates a retention offer. The offer is passed to a communication agent on Platform C, which sends the offer via email.
Where governance fails	The primary failure is a data residency violation: the CRM agent was configured before a new policy required that certain EU customer records not be passed to Platform B, hosted outside the EU. The handoff processes the transfer without any mechanism to verify whether current governance rules permit it. Secondary failures: the pricing agent's margin floor was updated after its last configuration review, and the communication agent's template predates a disclosure requirement. When a regulatory inquiry arrives, no unified provenance chain exists across the three platforms.
OGF controls	Req 7 (Cross-Platform Agent Inventory): the chain is documented before it processes live customer data, including data residency constraints at each boundary. Req 9 (Authority Inheritance Protocols): the pricing agent verifies before accepting the handoff that the transfer of EU customer data is permitted. Req 10 (Cross-Platform Provenance Tracking): a unified provenance record allows the compliance team to trace exactly which agent processed which customer record under which configuration.

B6. The Eight Requirements: Phase 3, Multi-Platform Interlock

These eight requirements are numbered 7 through 14 as a continuous extension of the six single-platform components in Section A. They are not aspirational principles. They are operational requirements: conditions that must be true of the governance infrastructure for governance to function across platform boundaries. Phase 3 requires Phase 0 through Phase 2 as its foundation.

Component	What it does	What good looks like	Owns	How common is good	Compliance
PHASE 3: MULTI-PLATFORM INTERLOCK | Requirements 7-14 extend the single-platform foundation. Phase 3 requires Phase 0-2 as its prerequisite. Requirements 7 and 8 are achievable today.					
7 Cross-Platform Agent Inventory	Complete inventory of every agent in the orchestration chain: vendor-built, internally built, co-developed, and adjacent. Governance cannot begin at the handoff if it does not know what is on the other side.	Every agent in every chain is known, named, owned, and documented before it enters production coordination.	Company	Complete cross-platform inventory: very few orgs. Most have visibility within their primary platform only.	EU AI Act, ISO 42001
8 Pre-Deployment Authority Definition	Every agent's authority boundaries defined in the context of the chain it is joining, not just in isolation. The configuration for standalone operation may not be correct for orchestrated operation.	Explicit authorization for each agent in each chain context. Authority reviewed when chain membership changes.	Company	Authority definition for chains: very few orgs. Most define agent permissions at deployment without chain context.	EU AI Act, NIST AI RMF, ISO 42001
9 Authority Inheritance Protocols	Governed handoff: Agent A confirms its authorization before passing. Agent B verifies A's authorization before accepting. Each agent in the chain verifies the prior chain is clean.	Both sides of every handoff perform authorization checks. Verification is logged and traceable. No agent executes without confirming the prior chain.	Shared	Not yet deployable in current protocol implementations. Emerging in research and early tooling.	EU AI Act, OWASP Agentic Top 10
10 Cross-Platform Provenance Tracking	Unbroken record of authorization linking every action across every platform from initiation to execution. Not per-platform logs requiring manual correlation.	Any outcome traceable to a specific authorization chain across all platforms. Auditors can interrogate without accessing multiple vendor systems.	Company	Per-platform audit logs: most orgs. Unified cross-platform provenance: not yet available as deployable infrastructure.	EU AI Act, SR 11-7 (fin. services), HIPAA
11 System-Level Behavioral Monitoring	Monitors the orchestration system as a whole for interaction-level patterns invisible to any single-agent framework. Detects emergent	Behavioral baselines at the system level. Anomaly detection identifies patterns across agent interactions, not just within individual agents.	Shared	Individual agent monitoring: most orgs. System-level interaction monitoring: nascent. Deployable solutions emerging.	EU AI Act, NIST AI RMF

	outcomes that only exist at the system level.				
12 Evidence-Based Rule Governance	Changes to governance rules validated against historical operational data before deployment. The question is not whether the change seems reasonable. The question is what the data shows it would have done.	Every rule change backtested. Simulation results required before approval. No rule enters production without evidence of impact.	Company	Opinion-based change approval: most orgs. Evidence-based with simulation: few orgs.	SR 11-7 (fin. services), EU AI Act, NIST AI RMF
13 Version-Controlled Authority Records	Every governance rule stored with full provenance: what it is, why it exists, what evidence justified it, who approved it, what alternatives were rejected, and when to reconsider.	No rule exists without reasoning. Institutional memory survives personnel changes. Reconsideration triggers fire automatically when conditions are met.	Company	Config files and change logs: most orgs. Full provenance with reasoning and version history: very few.	EU AI Act, NIST AI RMF, ISO 42001, HIPAA
14 Unified Accountability Attribution	When an orchestrated system produces an outcome requiring review, it is possible to trace that outcome to a specific chain of authorization and determine whether each link was consistent with the governance framework in effect at the time.	Any outcome attributable to named humans with defined accountability. No outcome exists without a traceable chain of approval. Regulatory inquiries answered with evidence.	Company	Attributable outcomes across platform chains: not yet achievable with current tooling. The infrastructure gap this framework is designed to close.	EU AI Act, SOX, SR 11-7 (fin. services)

B6a. Testability Criteria: Requirements 7-14

The following section provides operational detail for each requirement: the minimum condition that must be true for the requirement to be considered met, the evidence artifact that demonstrates compliance, and the failure mode the requirement is designed to prevent. This detail is intended for governance leads, AI operations teams, and compliance professionals implementing the framework.

Requirement 7: Cross-Platform Agent Inventory

Every agent in the orchestration chain must be known before governance can be applied, including internally built, co-developed, and adjacent system agents. Governance cannot begin at the handoff if it does not know what is on the other side.

Minimum condition: Complete inventory of every agent in the orchestration chain, including homegrown agents, with a named owner assigned to each before it participates in cross-platform coordination.

Evidence artifact: Agent inventory record showing name, platform, owner, authorization scope, and date of last governance review for every agent in the chain.

Failure mode prevented: Agents joining the chain without governance review, operating beyond their authorized scope without any governance record of their participation.

Requirement 8: Pre-Deployment Authority Definition

Every agent must have its authority boundaries explicitly defined before it joins a production orchestration chain, specifying what it may initiate and execute in the context of the chain, not just in isolation. The configuration that governs an agent in isolation may not be the right configuration for that agent operating as part of a coordinated chain.

Minimum condition: Each agent's authority in chain context is explicitly documented before go-live, signed off by a named human with the standing to authorize it.

Evidence artifact: Pre-deployment authority record per agent including chain-context scope, named approver, approval date, and reconsideration trigger.

Failure mode prevented: Agents operating in chain context under authority defined only for isolated deployment, with no record of whether chain participation was explicitly authorized.

Requirement 9: Authority Inheritance Protocols

When a task passes across a platform boundary, the governance constraints that governed the originating action must travel with it in a defined, enforceable way. Without explicit protocol, every receiving agent risks becoming a confused deputy: technically authenticated, organizationally ungoverned, inheriting whatever authority claim the sending agent carried with no mechanism to validate its legitimacy.

Minimum condition: At each platform boundary, the receiving agent performs a governance check on the upstream signal's authorized use in this workflow context before acting on it.

Evidence artifact: Handoff record at each platform boundary capturing originating authority context, receiving agent validation, applicable governance rule version, and timestamp.

Failure mode prevented: Receiving agents acting on authority claims from upstream agents without any mechanism to verify whether those claims are current, valid, or within the receiving agent's own authorization scope.

Requirement 10: Cross-Platform Provenance Tracking

Every action in a multi-agent chain must be linked to an unbroken record of authorization from initiation to execution. Not per-platform audit logs in separate systems requiring manual correlation. Provenance tracking is not only an audit requirement: it is the evidentiary foundation of the enterprise's legal position under the EU AI Act.

Minimum condition: Each consequential handoff records originating authority context, receiving validation, applicable governance rule version, and human approver provenance in a unified chain retrievable by outcome ID.

Evidence artifact: Unified provenance chain record spanning all platforms in the orchestration chain, retrievable as a single auditable record.

Failure mode prevented: Post-incident manual reconstruction of what happened across siloed per-platform logs with different formats, retention policies, and no unified chain of authorization.

Requirement 11: System-Level Behavioral Monitoring

The orchestration system as a whole must be monitored for interaction-level behavioral patterns invisible to any single-agent governance framework. Published findings from March 2026 confirm that reasoning-chain monitoring surfaces behaviors invisible to output-only review, and that instruction sets can inadvertently encourage agents to work around restrictions.

Minimum condition: System-level monitoring detects interaction patterns across the chain that would not be visible in any single agent's logs. At minimum, single-platform reasoning-chain monitoring is operational before cross-platform deployment scales.

Evidence artifact: System-level behavioral report covering interaction pattern analysis across the chain, flagged anomalies not attributable to any single agent, and resolution record.

Failure mode prevented: Emergent discriminatory or noncompliant behavior produced by the interaction of individually compliant agents that no single-agent or output-only monitoring framework would detect.

Requirement 12: Evidence-Based Rule Governance

Changes to governance rules must be validated against historical operational data before deployment. The question is not whether a proposed change seems reasonable. The question is what would have happened across the full range of agent interactions over the past thirty days if this rule had been in effect. Governance rules that have not been tested against operational history are hypotheses. Hypotheses deployed into production orchestration at enterprise scale are liabilities.

Minimum condition: Every governance rule change is backtested against at least 30 days of historical operational data before deployment. Simulation results are attached as evidence to the approval record.

Evidence artifact: Simulation report for each rule change showing what would have happened under the proposed rule across the historical transaction set, with blast radius and edge case analysis.

Failure mode prevented: Rule changes deployed on the basis of intuitive reasonableness that produce unintended consequences at the interaction level only visible in production after the fact.

Requirement 13: Version-Controlled Authority Records

Every governance rule must be stored with its complete provenance: what it is, why it exists, what evidence justified it, who approved it, what alternatives were considered, and when it should be reconsidered. Rules without provenance cannot be audited, cannot be safely updated, and do not survive the people who created them.

Minimum condition: Every governance rule has a version-controlled record containing rule text, business rationale, evidence basis, named approver, approval date, alternatives considered, and reconsideration trigger.

Evidence artifact: Version-controlled authority record with full provenance for every active governance rule, accessible to auditors and retrievable by rule ID, agent ID, or outcome ID

Failure mode prevented: Governance rules that survive personnel changes but whose reasoning does not, creating rules that cannot be safely updated because no one knows why they exist.

Requirement 14: Unified Accountability Attribution

When an orchestrated system produces an outcome requiring review, it must be possible to trace that outcome to a specific chain of authorization and determine whether each link was consistent with the governance framework in effect at the time. Accountability that cannot be attributed is not accountability. It is exposure distributed across every party in the chain with no mechanism for resolution.

Minimum condition: Any outcome from the orchestration chain can be traced to a complete cross-platform authorization chain identifying the named human accountabilities at each decision point, within a defined time standard for regulatory or incident response.

Evidence artifact: Outcome trace record linking the result to every authorization decision in the chain, with named accountabilities and governance rule versions in effect at each step.

Failure mode prevented: Multi-vendor incidents where accountability is disputed or unresolvable because no unified record exists of who authorized what at each step in the chain.

B7. The Logic Product Manager

The infrastructure described in Requirements 7 through 14 requires a named human role to own it. The Logic Product Manager is the organizational function responsible for maintaining the accuracy of governance rules across the organization's agent ecosystem, reviewing escalations, approving rule changes based on simulation evidence, and ensuring that the institutional memory embedded in the authority record does not decay back into the patchwork it was designed to replace.

This role does not yet exist under this name in most enterprises. The function exists informally, distributed across AI operations teams, compliance functions, and engineering leads without unified ownership or mandate. The Logic Product Manager consolidates that ownership: humans approve rules, not individual actions. Every rule change is evidence-based. Every escalation is a discovery opportunity.

The Logic Product Manager is a named operating role or mandate, not necessarily a new box on the org chart. In different organizations it may sit within AI Ops, product governance, model risk, enterprise architecture, or compliance. The title matters less than the mandate: one named human who owns the accuracy of the governance logic and has the authority to require evidence before rules change.

Appendix A: Competitive Landscape

Reference section: six categories, a growing market, and the gap that remains

This appendix is a reference section for practitioners evaluating tooling against the OGF requirements. It is not part of the core standard. The market has organized itself into six categories addressing different layers of the AI governance problem. Based on public information reviewed as of March 2026, no category provides the full cross-platform authority layer that Requirements 7 through 14 define.

This appendix describes categories rather than specific vendors by design. The agentic governance market is advancing rapidly. Vendor-specific assessments become outdated within months and create barriers to the collaborative conversation this framework is intended to support.

Category	What tools do	Scope	Gap against OGF Req 7-14
Access and Non-Human Identity	Lifecycle management for machine identities and AI agents. Just-in-time permissions, secrets rotation, intent inference.	Governs identity and access at the credential and permission layer.	Access governance is not authority governance. Does not typically provide cross-platform authority inheritance (Req 9), provenance tracking (Req 10), or version-controlled authority records with organizational reasoning (Req 13).
Telemetry and Observability	Model performance monitoring, drift detection, explainability, behavioral analysis, and LLM trace capture.	Governs what happened and why from a performance and reliability lens.	Observability is not authority enforcement. Does not enforce authority boundaries at the cross-platform handoff (Req 9), maintain authorization chain as provenance (Req 10), or provide system-level behavioral governance across multi-vendor chains (Req 11).
Runtime Guardrails and Gateways	Content enforcement at the inference layer. Prompt injection detection, PII filtering, harmful output prevention.	Governs what agents say and what content passes through the inference boundary.	Content governance is not authority governance. Does not govern cross-platform authority inheritance, chain provenance, or the organizational authorization record that makes outcomes attributable.

Agent Security and Visibility	Cross-platform agent discovery, security posture management, runtime threat detection, blast radius mapping.	Governs the security posture of the agent ecosystem.	Security governance and authority governance are distinct disciplines. Does not provide cross-platform authority inheritance (Req 9), version-controlled authority records (Req 13), or unified accountability attribution (Req 14) as governance artifacts.
Platform Governance	End-to-end AI lifecycle governance including model inventory, pre-deployment validation, policy enforcement, agent management.	The most complete category for single-platform governance. Increasingly extending to cross-platform visibility.	The closest existing category to OGF requirements. The gap remaining is specifically at the authority layer: version-controlled rules with organizational reasoning (Req 13), cross-platform authority inheritance (Req 9), and unified accountability attribution across multi-vendor chains (Req 14).
Policy and Regulatory Compliance	Regulatory framework alignment, risk assessment, audit documentation, compliance evidence generation.	Governs compliance documentation and policy alignment.	Policy compliance documents governance intent. Authority governance enforces it at runtime. Documentation that a policy exists does not constitute enforcement when agents coordinate across platforms.

SECTION D

Maturity Assessment and Where to Start

Four stages, immediate actions, and the path through Phase 3

The OGF is a standard, not a product. Organizations implement it through whatever combination of process, tooling, and organizational commitment satisfies the requirements. The path is staged and contextual.

Stage	Single-platform foundation (Phase 0-2)	Multi-platform authority layer (Phase 3)	Immediate next step
Stage 1 Reactive	Informal monitoring. No structured escalation. Shadow AI largely undiscovered. Governance rules in configuration files with no provenance.	Phase 3 not applicable. Single-platform foundation is the prerequisite.	Conduct shadow AI discovery. Name a governance owner for your highest-volume agent. Document the reasoning behind three governance rules in a shared record.
Stage 2 Structured review	Regular review cadence. Named team. Basic change log. Root cause analysis emerging. Pre-deployment testing inconsistent.	Phase 3 not applicable. Close Phase 2 gaps before extending to orchestration.	Implement pre-deployment simulation testing for your most consequential agent. Build the authority record to Stage 3 standard for one agent before attempting Phase 3.
Stage 3 Evidence-based	All six single-platform components active. Simulation required before rule changes. Full authority record in place. ISO 42001 achievable.	Phase 3 entry point. Begin with Requirements 7 and 8: cross-platform agent inventory and pre-deployment authority definition.	Complete cross-platform agent inventory (Req 7). Define authority boundaries for each agent in the context of its orchestration chain, not just in isolation (Req 8).
Stage 4 Institutional	Authority record is a first-class institutional asset. Governance is proactive. Stage 3 is the operational floor.	Full Phase 3 architecture active. Authority inheritance protocols in place. Cross-platform provenance tracking. System-level behavioral monitoring.	Implement cross-platform interaction monitoring (Req 11). Contribute operational learnings to the OGF as the standard evolves.

Three immediate actions that require no procurement

Shadow AI discovery: Find every agent running in your environment, including those your teams built informally. Assign a named owner to each one. You cannot govern what you have not found.

Name the Logic Product Manager: Assign a named human being to be accountable for the accuracy of governance rules governing your highest-volume agent deployment. That person does not need a title yet. They need a mandate.

Build your first authority record: Document the governance rules for your most consequential agent, the reasoning behind each rule, who approved it, and when it should be reconsidered. Start with one agent. The practice matters more than the coverage.

The OGF will evolve

This standard is published as a starting point, not a finished product. It will develop as organizations contribute operational experience, as the tooling market matures, and as the regulatory framework develops guidance on multi-agent orchestration. If you are working on any part of this problem, the conversation is open. Contact Decision DNA Group at [contact].

Appendix B: Source Notes

Attribution for the framework's principal external claims

This appendix is not a complete bibliography. It provides source attribution for external claims most likely to be challenged in executive, regulatory, or vendor review. Internal conceptual claims (Tethered Agent Problem, Logic Hallucination, Governance Debt, Orchestration Governance Framework, Authority layer, Logic Product Manager) are original work of Decision DNA Group.

Claim	Source and notes
MCP adoption statistics	MCP Blog, 'MCP joins the Agentic AI Foundation,' December 9, 2025. Public MCP materials cite over 97 million monthly SDK downloads and 10,000+ active servers. First-class platform support confirmed across ChatGPT, Claude, Cursor, Gemini, Microsoft Copilot, and VS Code.
MCP governance structure	MCP Blog, 'The 2026 MCP Roadmap,' March 9, 2026 (David Soria Parra, Lead Maintainer). MCP is now a founding project of the Agentic AI Foundation under the Linux Foundation, co-founded by Anthropic, Block, and OpenAI.
A2A governance and partners	Linux Foundation press release, 'Linux Foundation Launches the Agent2Agent Protocol Project,' June 23, 2025. Founding members include AWS, Cisco, Google, Microsoft, Salesforce, SAP, and ServiceNow.
EU AI Act: provider and deployer definitions	Regulation (EU) 2024/1689 (EU AI Act), Articles 3, 25. The Act defines provider and deployer obligations and specifies that a party making a substantial modification to a high-risk AI system is treated as a provider for that modified system. The multi-agent substantial modification interpretation is this framework's analysis, not a settled regulatory determination.
EU AI Act: penalty structure	EU AI Act Article 99. Penalties for the most serious violations (including Article 5 prohibited practices): up to EUR 35 million or 7% of global annual turnover. Penalties for operator non-compliance with high-risk AI system obligations: up to EUR 15 million or 3%. The European Commission's Digital Omnibus proposal (November 2025) could extend the high-risk enforcement deadline to December 2027; that proposal requires legislative approval and August 2026 remains the operative planning deadline.
EU AI Act: application timeline	EU AI Act Article 113. The Act generally applies from August 2, 2026. Certain provisions apply earlier (prohibited practices from February 2, 2025) and certain obligations for general-purpose AI models apply from August 2025.
Agent reasoning-chain monitoring	OpenAI internal findings published March 2026. Key findings: agents can be overly eager to work around restrictions in ways invisible to output-only review; developer prompts can inadvertently encourage circumvention; reasoning-chain monitoring surfaces behaviors output-only review misses.
System-level outcomes from compliant components	The general principle that individually compliant or well-performing components can produce unintended system-level outcomes is supported by research in sociotechnical systems and algorithmic fairness. This framework applies that principle to multi-agent governance; it does not cite a specific study on chained AI agent systems, as the direct literature on this specific configuration is still developing.