ODA3 INSTITUTE

ADOPTION GUIDE

The AI System Owner’s Guide to Operational Assurance

Doc ID: ODA3-2026-08-WHP-COM-019
Classification: Public / Freely Distributable
Publication status: CORRECTED FINAL (R10)
Original publication: 14 August 2026 | Corrected edition: 20 August 2026

Contents at a Glance

Use the Executive Navigation in the guide for role-based and situation-based routes. The section ranges below provide a stable overview without relying on a field-updated Word table of contents.

Publication and Authority Boundary

COM-019 is an informative practitioner role guide. It does not create, amend or replace GAISSF™ requirements, UAIF™ classification rules, AI-IRF™ response requirements, PAI-SF™ controls, controlled assessment methodology, law, regulation, functional safety, product-safety requirements or organizational decision authority.

The accountable system owner is treated as the coordinator of the system's operating basis—not as the owner of every specialist control or authority.

Detailed implementation belongs in COM-017 — Implementing GAISSF™. Evidence-pack preparation belongs in COM-015 — Building an AI Assurance Evidence Pack. Enterprise CISO assurance belongs in COM-018 — The CISO's Guide to AI Security Assurance — CORRECTED FINAL (R10). Active incident classification and response belong in COM-011 / COM-012 / COM-013, UAIF™ and AI-IRF™. Detailed physical-AI treatment belongs in COM-014 / PAI-SF™.

The System Owner Record

This is the practical entry point to COM-019. A system owner should be able to locate a current answer to each of the twelve questions below for an AI system for which they are accountable. Where an answer is genuinely unknown, record "Unknown" rather than leaving the question unanswered — an explicit unknown is more useful than an invented one.

1. Purpose — What is the system authorized to do, and what is excluded?

2. Owner — Who is accountable for the system?

3. Boundary — Which model, provider, data, tools, identities and physical paths matter?

4. Consequence — What business, security, human or physical effect can follow?

5. Control ownership — Who operates the material controls?

6. Provider boundary — What is enterprise-controlled, provider-controlled, shared or visibility-limited?

7. Evidence — What current operational records support the important system claims? (See §36, Minimum Evidence Questions.)

8. Unknowns — What remains provider-represented, inaccessible, contradictory or unverified?

9. Change — What materially changed since the prior operating basis was established?

10. Containment — Which identities, tools, connectors, routes or functions can be constrained?

11. Decision authority — Who can continue, constrain, suspend, retire, accept risk or authorize physical return to service?

12. Escalation — Who needs to know when the operating basis no longer supports the current decision?

This record is a management orientation aid, not a completeness test, readiness score, assessment instrument or certification input. Sections 11–98 provide the operating guidance, handoffs and practitioner tools used to develop, evidence and act on these answers. See §5 for the fuller explanatory treatment of these same twelve questions, and §91 for the related System Owner Accountability Card.

Current Source and Framework State

The following source hierarchy governs the terminology and boundaries used by COM-019:

1. Current controlled normative / technical sources govern framework architecture, control inventories, schema structure and technical definitions.

2. Current public publication-state and GEL material govern market-facing naming, maturity, licensing, authority and claims language.

3. Current FINAL practitioner guides govern cross-document handoffs and public-safe explanatory terminology.

4. Historical drafts / superseded wording remain source-history only and do not override a current controlling source.

Topic	Public treatment used in COM-019
Market-facing GAISSF name	GAISSF™ — Global AI Safety and Security Framework. Historical/source identifiers may retain earlier word order; practitioner text uses the current market-facing name.
GAISSF architecture	59 controls across nine domains. GAISSF-NOR-001 is the primary framework authority; GAISSF-NOR-004 is the canonical control catalogue.
Foundational / D9 treatment	52 canonical controls in D1–D8 form the Foundational baseline. The seven D9 physical-AI controls are additional and become mandatory where physical AI / cyber-physical actuation is within applicable/assessed scope; applicability is considered and justified.
UAIF™ public state	Final Publication v1.0.
UAIF machine-readable companion	TCR-STD-003 / UAIF machine-readable schema package is published. Before implementation, confirm the current schema version, release state and package metadata.
AI-IRF™ technical architecture	79 controls / 19 domains. Current practitioner/public comparison language treats AI-IRF™ as Final Publication v1.0; historical certification or scheme language is not carried forward automatically.
AIIS machine-readable companion	AIIS v1.0.0 is published for AI-IRF control/playbook encoding. Its optional conformance_test structure is reserved for future population; this does not make the schema itself future work.
PAI-SF™	Current canonical public structure: 12 domains / 41 controls. GAISSF D9 remains for continuity; PAI-SF™ supersedes and expands the detailed physical-AI security-assurance layer without creating a second independent obligation.
Physical / safety boundary	PAI-SF™ interfaces with functional safety and sector regimes but does not replace them or create product/sector approval or physical return-to-service authority.
Meaningful human authority	Public practitioner dimensions remain Awareness / Comprehension / Time / Authority / Effectiveness. No universal pass/fail threshold is inferred.
Provider evidence-source states	Customer-observed / Provider-provided / Shared-interface / Provider-only-inaccessible / Unknown. These are evidence-source states, not implementation-responsibility descriptors or control ratings.
Implementation responsibility	COM-017 distinguishes Enterprise-controlled / Provider-controlled / Shared / Visibility-limited for implementation reasoning. COM-019 keeps these separate from evidence-source states.
Assessment participation	The controlled assessment methodology remains a Controlled Design Baseline; COM-019 does not announce live assessment

Topic	Public treatment used in COM-019
	authority, current certification availability or certificate issuance.
Implementation boundary	COM-017 owns detailed control implementation and implementation-side verification. Implementation artifacts remain candidate evidence and do not establish conformity or assessment outcome.
GEL v1.0	Current governing use boundary. Internal Use, Commercial Use, factual-alignment statements, marks and reserved certification/audit authority remain separately governed where applicable.

See §99.4 for the guide's terminology rule on published schemas versus future interoperability tooling.

See §99.1 for the guide's treatment of historical/superseded source material.

Practitioner Suite Role Register

The following role boundaries govern how COM-019 should be read alongside the current adjacent practitioner publications:

Publication	Current practitioner role	COM-019 treatment
COM-008 — The AI Assurance Imperative: A Board Member's Guide to GAISSF™	Board oversight questions	Supplies bounded system-level facts, evidence context and decisions that can feed management/board reporting; does not replace board oversight or turn board questions into assessment criteria.
COM-011 — When AI Breaks: The ODA3 Guide to AI Incident Response	Short incident-response orientation	Establishes system-owner incident readiness and handoff context; does not replace active SOC/CSIRT or response workflow.
COM-012 — Classifying AI Incidents: A Practical Guide to UAIF™	UAIF™ classification and uncertainty discipline	Preserves local signal/observation/event-under-triage workflow language without creating a UAIF Observation record type, severity rule or legal-reportability conclusion.
COM-013 — AI Incident Response Field Guide: Using UAIF™ and AI-IRF™ for Real-World AI Incidents	Integrated classification/response practitioner treatment	Supplies system context, containment surfaces, provider contacts and decision authorities; COM-013 / AI-IRF™ own active response integration.
COM-014 — Securing Physical AI: A Practitioner's Guide to PAI-SF™	Detailed physical-AI practitioner and safety interface	Identifies the physical consequence path and handoff authorities; detailed PAI-SF™ implementation, safety engineering and return-to-service authority remain outside.
COM-015 — Building an AI Assurance Evidence Pack (FINAL)	Evidence preparation	Preserves operational records and context. COM-015 owns evidence indexing, evidence-source states, provenance, limitations, contradictions and preparation for handoff.
COM-017 — Implementing GAISSF™ (FINAL)	Detailed implementation and implementation-side verification	Coordinates ownership, operation, evidence generation and change at system level; COM-017 owns control translation, mechanism design, implementation, integration and implementation-side verification.
COM-018 — The CISO's Guide to AI Security Assurance (CORRECTED FINAL R10)	Enterprise CISO security-assurance management system	COM-018 owns enterprise portfolio visibility, CISO challenge, dashboards, concentration exposure and board/risk synthesis. COM-019 owns the accountable operating view for one specific AI system/service.

Role Shorthand

COM-018 asks whether the CISO can defend the enterprise AI security assurance position.

COM-019 asks whether the accountable owner can defend how this specific AI system is operated, changed, evidenced and escalated.

The second does not replace the first.

Guidance, Claims and Status Rule

COM-019 is public practitioner guidance. The following interpretation rule applies throughout:

- SHALL / SHALL NOT are not used to invent new public obligations.
- must / mandatory are reserved for verified controlling requirements or unavoidable authority conditions. The D9 mandatory

treatment in the source-state table comes from the applicable GAISSF source state rather than from COM-019.

- should, may, consider, useful and similar wording describe practitioner guidance unless a controlling source is explicitly

identified.

- A system-owner management position is not an independent assurance opinion, controlled GAISSF determination, audit

conclusion or certification decision.

- Local labels such as Continue, Constrain, Degrade, Suspend, Retire, Known, Unknown, or Provider-represented are

management aids only. They do not become ODA3 assessment states, readiness categories, control ratings, confidence levels or certification statuses.

- Completion of a COM-019 card, checklist, record or review does not create ready, assured, compliant, approved, certifiable or

equivalent status.

- COM-019 does not publish evidence-sufficiency rules, sampling, assessor judgement, finding severity, result aggregation,

certification eligibility or controlled assessment workflow.

- Where another organizational authority owns a decision, the system owner supplies the bounded system record, analysis and

recommendation rather than assuming that authority. Executive Navigation — Use COM-019 Without Reading It Linearly COM-019 is designed for selective use. A system owner can enter at the question that needs a decision and follow the relevant handoff. 10-Minute System Owner Orientation If time is limited, read these sections first:

1. Section 1 — Executive Summary
2. Section 4 — The AI System Owner Role
3. Section 10 — Core Boundaries and Non-Equivalences
4. Section 11 — Identify the AI System or Service
5. Section 20 — Define Scope and Material-Change Triggers
6. Section 21 — Name the Accountable System Owner
7. Section 23 — Map Provider and Shared Responsibility
8. Section 41 — Evidence Is Part of Operation
9. Section 51 — Treat Material Change as a System-Level Review Trigger
10. Section 61 — Design Incident Readiness Before the Incident

11. Section 81 — Build the System-Level Decision Basis

12. Sections 91–98 — Practitioner Tools

This route is orientation only. It does not create a mandatory reading order, readiness sequence or implementation gate.

The Three Practical Uses of COM-019

COM-019 can be used through three overlapping lenses:

Lens	Practical question	Main Parts
Build the operating picture	What is this system, what can it affect, who owns what, and where are the dependencies?	Parts II–IV
Operate the assurance basis	What evidence exists, what changed, what can fail, and are we ready to respond?	Parts V–VIII
Make and explain decisions	What option is supportable, who decides, what remains uncertain, and what needs escalation?	Parts IX–X

These lenses are navigation aids—not a lifecycle, maturity model or assessment sequence.

What the System Owner Should Always Be Able to Locate

The owner does not need every answer personally, but should be able to locate the current answer to these questions:

1. Purpose — What is the system authorized to do, and what is excluded?
2. Owner — Who is accountable for the system?
3. Boundary — Which model, provider, data, tools, identities and physical paths matter?
4. Consequence — What business, security, human or physical effect can follow?
5. Control ownership — Who operates the material controls?
6. Provider boundary — What is enterprise-controlled, provider-controlled, shared or visibility-limited?
7. Evidence — What current operational records support the important system claims?
8. Unknowns — What remains provider-represented, inaccessible, contradictory or unverified?
9. Change — What materially changed since the prior operating basis was established?
10. Containment — Which identities, tools, connectors, routes or functions can be constrained?
11. Decision authority — Who can continue, constrain, suspend, retire, accept risk or authorize physical return to service?
12. Escalation — Who needs to know when the operating basis no longer supports the current decision?

This reference is a management orientation aid, not a completeness test or readiness score. This is the same set of questions presented as a fillable entry-point record in the front matter — use that record as the working document; use this section for the fuller explanation of each question.

Role-Based Routes

Role	Start here	Then use
Business / product owner	4, 12, 13, 21	26–30, 51, 81–90
Technical service owner	11, 14–18, 22	34–40, 51–70, 72–79
AI Security Lead / security architect	17–20, 22–30	31–40, 61–70, 81–87
AI governance / GRC / risk	20–30	41–50, 71–80, 87–90
Provider / supplier owner	14, 18, 23	37, 45, 52, 56–58, 65, 95
SOC / CSIRT / incident lead	28, 35	61–70, 97
Safety / OT / physical-AI interface	19, 24–25, 28	39, 58, 67, 70, 93, 97
Internal assurance / audit liaison	41–50	81, 88–90, 96, 98

No route changes framework applicability, role authority, assessment state or certification status.

Situation-Based Routes

Situation	Use these sections
New AI system or major use case	11–30 → 31–40 → 91–93
New hosted model / provider	14, 18, 23, 37, 45, 52, 56–58, 95

Situation	Use these sections
RAG / enterprise knowledge system	15, 18, 35, 54, 63
Agentic AI / tool-using system	16–18, 25, 28, 38, 53, 55, 64, 66
Material model/tool/provider/permission change	20, 48, 51–59, 72–77, 94
Suspected AI incident	28, 61–70, 81, 87, 97
Physical-AI / high-consequence system	19, 25, 28, 39, 58, 67, 70
Audit, customer or regulator question	44–50, 81, 88–90, 96, 98
Decision to continue, constrain or suspend	81–87
Ownership transfer / retirement	21, 30, 59–60, 86, 91–93

Practitioner Handoff Map

Use COM-019 to maintain the system-level operating picture, then hand off when specialist depth is required:

Enterprise CISO challenge / portfolio synthesis → COM-018

Detailed GAISSF implementation → COM-017

Evidence-pack preparation → COM-015

Incident orientation / classification / response → COM-011 / COM-012 / COM-013, UAIF™ and AI-IRF™

Detailed physical-AI security assurance → COM-014 / PAI-SF™

A handoff does not erase system-owner accountability for keeping the system context current.

Informative System Owner Assurance Loop

Know → Bound → Own → Operate → Observe → Evidence → Change → Respond → Decide → Improve

This is a practitioner organizing device, not a normative GAISSF lifecycle, maturity model, assessment workflow or certification sequence.

Role Boundary

COM-018 asks: Can the CISO defend the enterprise AI security assurance position?

COM-019 asks: Can the accountable owner defend how this specific AI system is operated, changed, evidenced and escalated?

The system owner may coordinate implementation, evidence, incident readiness and change, but does not absorb the independent authority of security, legal/privacy, risk, assessment, certification, safety or sector functions.

PART I — THE AI SYSTEM OWNER MANDATE

Use this Part when: the organization needs to define what the system owner is accountable for and where specialist authority remains

separate.

Working output: a clear system-owner mandate and a shared vocabulary for accountability.

Part checkpoint: Can the named owner explain the system's operating basis and identify the specialist authorities they rely on?

The checkpoint is an informative management prompt, not a readiness gate, assessment state or certification prerequisite.

1. Executive Summary

The accountable AI system owner is the person who can answer for how a specific AI-enabled service is used, bounded, changed and operated—even when the underlying controls are distributed across security, engineering, platform teams and external providers. The owner's job is not to personally implement every safeguard. It is to maintain a defensible operating picture: what the system is for, what it depends on, which authorities it has, who owns the material controls, what evidence exists, what changed, what remains uncertain, how incidents can be contained, and which decisions require escalation. A strong system-owner position connects five things that are often separated in practice: purpose, boundary, ownership, evidence and decision authority. When one of these is unclear, assurance becomes fragile. A system can appear technically well controlled while still lacking a named owner, current provider information, usable incident records or a clear decision route when assumptions fail. COM-019 therefore treats operational assurance as an ongoing management discipline rather than a one-time approval event.

2. Purpose and Audience

This guide is written primarily for people who are accountable for an AI-enabled system or service in operation: AI product owners, service owners, business owners, technical service owners and equivalent accountable roles. It is also intended to be usable by organizations where the exact title differs. The relevant test is not the job title; it is whether the person is expected to explain the system's purpose, operating boundary, dependencies, decisions and unresolved conditions. The guide assumes that specialist functions already exist or can be engaged. Security may own security architecture; privacy may own privacy decisions; engineering may own implementation; procurement may own contractual processes; a provider may operate infrastructure; safety functions may own safe-state or return-to-service decisions. The system owner coordinates across those interfaces without erasing them. COM-019 is informative practitioner guidance. It does not create a new legal role, fiduciary duty, statutory accountable- person designation or GAISSF-specific organizational position.

3. Operational Assurance in Plain Terms

For COM-019, operational assurance means maintaining an evidence-supported, bounded management basis for believing that the system is being operated within the assumptions, controls and authorities on which its use depends. This is not an independent assurance opinion or a scored confidence level. For the system owner, that means being able to answer practical questions such as:

- What system are we actually operating?
- Which model, provider, data sources, tools and identities does it use?
- What may it do, and what remains outside its authorized scope?
- Which controls are ours, which are shared, and which are provider-controlled?
- What evidence shows the current operating state?
- What changed since the last meaningful review?
- What happens when monitoring, evidence or provider visibility is weak?

- Who decides whether to continue, constrain, suspend or retire the capability?

Operational assurance is not certainty. It does not prove absence of harm, future security, legal compliance or product safety. In COM-019, it means making the organization's current operating basis explicit enough to challenge, evidence and act upon without converting that management position into an independent assurance conclusion.

4. The AI System Owner Role

The system owner is the accountable integrator of the system's operating context. The role sits where business purpose meets technical operation. A practical system-owner mandate normally includes maintaining the system identity and purpose; ensuring ownership is assigned for material controls and dependencies; making change visible; ensuring evidence can be generated and located; maintaining incident-readiness interfaces; escalating unresolved conditions; and supporting system-level operating decisions. The role does not mean that every technical or governance decision is transferred to the owner. Where security, privacy, legal, model-risk, safety, procurement or operational authorities have independent decision rights, those rights remain separate. A useful test is: Can this person explain the current operating basis of the system without pretending to be the specialist authority for every component? If not, either the system-owner mandate is incomplete or the accountability model is too diffuse to support defensible operation. System ownership does not by itself transfer the CISO's enterprise security-assurance mandate, independent challenge role, portfolio visibility or board/risk synthesis into the system-owner role. Where those functions exist, COM-019 supplies the system-level operating facts and decisions that feed them.

A named owner is not the same fact as an owner who is exercising the role. The following are observational indicators — not a score, not an assessment finding, and not an automatic failure state — that the mandate may exist on a record without being exercised in practice:

- the named owner cannot describe the system's current boundary without reconstructing it from others;

- material changes have occurred that the owner did not know about until asked;

- the named escalation route cannot be explained, or there is no basis for knowing whether it remains usable when needed;

- unresolved unknowns exist that no one has revisited since they were first recorded;

- the owner cannot identify who holds current containment or decision authority for the system.

Any one of these, on its own, may simply reflect a quiet period for a low-consequence system. Persistent or combined indicators suggest the accountability model needs attention — a different owner, clearer authority, or a narrower scope — rather than a documentation fix.

5. Accountability Without Owning Every Control

Accountability is different from mechanism ownership. The system owner may be accountable for the AI-enabled service while a security team owns identity controls, a platform team owns model routing, a data team owns retrieval sources, a provider owns the hosted model, and a safety function owns physical return-to-service decisions. COM-019 therefore distinguishes:

- System owner — accountable for the system's purpose, scope and coordinated operating basis.
- Control owner — accountable for a particular control capability.
- Operator/custodian — runs the mechanism or process.
- Provider/platform owner — manages a shared or external dependency.
- Decision authority — makes a defined business, security, legal, privacy, safety or risk decision.
- Challenge function — independently questions the support for a claim or decision.

The system owner should know who occupies each role for material areas. "Shared" is acceptable only when the interfaces are explicit enough to understand who does what, who can see what evidence, and who acts when the control fails.

6. Relationship to Operational AI Assurance

Operational AI Assurance remains the broader ODA3 category. COM-019 addresses one operating layer within it: the accountable owner of a specific AI system. The relationship across the practitioner suite is intentionally federated. COM-018 gives the CISO an enterprise portfolio and challenge view. COM-017 explains implementation. COM-015 prepares evidence. COM-011/012/013 address incidents. COM-014 addresses physical-AI practitioner treatment. COM-019 connects these at system level. It asks whether the accountable owner can keep the system bounded, owned, observable, evidentiary, change-aware and decision-ready. COM-018 remains the enterprise CISO layer: portfolio challenge, cross-system provider/dependency exposure, CISO metrics and board/risk synthesis remain there. COM-019 should feed that layer rather than recreate it. This guide should therefore not be read as a new assurance standard. It is a management companion that helps one accountable role navigate existing framework, implementation, evidence, assessment and incident interfaces.

7. GAISSF Ecosystem Position

The GAISSF Ecosystem provides the framework context for COM-019. GAISSF™ — Global AI Safety and Security Framework provides the principal AI safety/security control baseline. UAIF™ — Unified AI Incident Framework provides structured incident classification. AI-IRF™ — AI Incident Response Framework structures AI incident response. PAI-SF™ — Physical AI Security Framework provides the detailed physical- AI security-assurance layer where cyber-physical consequence is in scope. A single system may touch more than one framework. An agentic industrial system, for example, may require GAISSF controls during normal operation, UAIF and AI-IRF when an incident occurs, and PAI-SF where AI decisions can influence physical actuation. COM-019 does not create a fifth framework or a system-owner profile. It helps the owner know which specialist framework or practitioner guide should take over when the issue leaves the system-owner layer.

8. The System Owner Assurance Loop

COM-019 uses an informative operating loop: Know → Bound → Own → Operate → Observe → Evidence → Change → Respond → Decide → Improve The sequence is recursive rather than linear. A material provider change can require the owner to revisit the boundary. An incident can expose a missing owner. A failed test can require a new control mechanism. A business repurposing can change the consequence model. The loop is useful because it keeps operational assurance connected to the actual system rather than to a static policy pack. It is not a maturity model, assessment workflow, lifecycle standard or certification sequence. An organization can enter the loop at the point where the current problem exists.

9. How to Use This Guide

Use this guide as a role reference rather than a mandatory implementation sequence. If the immediate problem is unclear scope, use Part II. If ownership is fragmented, use Part III. If implementation responsibility is unclear, use Part IV. If evidence is weak, use Part V. If the system is changing, use Part VI. If incident readiness is the concern, use Part VII. For recurring operation and remediation, use Part VIII. For an immediate operating decision or external assurance request, use Part IX. Part X contains lightweight records that can be implemented inside existing GRC, ticketing, CMDB, service-management or engineering systems. The system owner should avoid creating a parallel bureaucracy where authoritative records already exist. Link to established sources where possible; create a new record only where the existing system cannot express the required boundary, ownership, evidence or decision context.

10. Core Boundaries and Non-Equivalences

Several non-equivalences govern the entire guide. Ownership is not implementation. Naming an owner does not prove a control operates. Implementation is not conformity. A configured mechanism is candidate evidence of implementation, not an assessment outcome. Evidence existence is not evidence sufficiency. A document, dashboard or provider report can be relevant while still incomplete, stale or

scoped differently. Detection is not automatically incident classification. Local triage may begin before UAIF classification is settled. Security closure is not physical return to service. A cyber issue may be resolved while safety or operational authorities still withhold return to service. Provider certification does not transfer end-to-end conformity. It remains scoped evidence about the provider. A management decision is not ODA3 approval. Continue, constrain, suspend and similar choices are enterprise decisions made under applicable authorities. A system-owner label is not a statutory accountable-person designation. COM-019 does not create legal, fiduciary, regulatory or sector-defined accountability merely by naming an internal owner.

PART II — BOUND THE AI SYSTEM AND ITS CONSEQUENCE

Use this Part when: the team cannot clearly describe what the AI system includes, what it can influence, or where provider/tool/data

boundaries sit.

Working output: a current system boundary, purpose statement, dependency map and material-change trigger set.

Part checkpoint: Can the owner identify the system, current service basis, authority paths, material dependencies and consequence

path? The checkpoint is an informative management prompt, not a readiness gate, assessment state or certification prerequisite.

11. Identify the AI System or Service

Start with a concrete system identity. Avoid naming only a business programme such as "customer AI" or "copilot initiative" if several distinct services, models or action paths sit underneath it. A useful identity record includes the service/use-case name, business owner, technical service, production environment, primary model/provider, major integrations and a reference to the authoritative inventory entry. For composed AI systems, the operating unit may include more than a model endpoint. Orchestration, retrieval, tools, identity, monitoring, human review and downstream action can all be part of the assurance boundary. The owner should choose a scope that is narrow enough to be operationally meaningful but broad enough to include the components whose failure can invalidate the system-level assurance claim.

12. Define Purpose, Use Case and Intended Outcomes

Purpose constrains assurance. The same model can have very different operating implications when used for drafting marketing copy, making fraud decisions, controlling industrial equipment or acting through enterprise tools. Record the intended business purpose, authorized use cases, material exclusions and the downstream decision or action influenced by the output. The owner should also record what the system is not intended to do. Explicit exclusions help distinguish a permitted use from quiet scope expansion. Avoid writing purpose statements so broadly that any future use fits. If the business purpose changes materially, treat that as a change to the assurance basis rather than an editorial update.

13. Identify Users, Affected Parties and Business Context

Map the operating context around the system. Identify who uses the system, who relies on its outputs, who can be affected by errors or misuse, and which business processes can amplify those effects. Include privileged users, administrators, reviewers, downstream operators and external users where relevant. The system owner should distinguish direct users from affected parties. A person may never interact with the AI system but still be affected by a decision, ranking, alert, transaction or physical action it influences. Context also includes operating geography, sector, customer commitments and critical business dependencies. COM-019 does not determine legal applicability, but the owner should know where those determinations are obtained and who owns them.

14. Track Models, Providers and Versions

A system name is not enough if the underlying model or provider can change. Track the actual service/model identifier, relevant version or routing configuration, provider, deployment region where material, fallback model, and any abstraction layer that can switch models without changing the application name. If the provider does not expose a stable model version, record that limitation and identify what change signals are available: release notes, service notices, behavior monitoring, contract commitments or enterprise tests. Version visibility is an assurance property. Unknown or provider-opaque versioning is not automatically a control failure, but it narrows what the owner can responsibly claim about continuity between earlier evidence and current operation.

15. Map Data, Retrieval and Knowledge Sources

For RAG and knowledge-augmented systems, map the information path rather than only the model. A useful operating chain is: source → ingestion → integrity/approval → index → retrieval authorization → context assembly → model → output → downstream use The owner should know which repositories can contribute content, who can change them, how access is enforced during retrieval, how stale or poisoned content is detected, and whether user permissions survive the transition into retrieval. Where the system uses embeddings, caches, vector stores or generated summaries, include them when they can materially change what information reaches the model. The key question is not simply "What data did we train on?" but "What information can influence this system at runtime, under whose authority, and with what observable change path?"

Where a system combines retrieval with agentic action, the trust boundary crossed by retrieved content matters as much as the retrieval path itself. Content pulled from a lower-trust source can become an input that influences a higher-authority decision or action downstream — a retrieved document, for example, shaping what tool an agent chooses to invoke or what parameters it uses. The owner should be able to answer, for the system's actual architecture: what is the provenance of content that can influence a consequential action; does authorization established earlier in the chain (e.g., a user's retrieval permissions) actually continue to apply once that content crosses into agent planning or execution; where does untrusted content cross into a privileged action path; and what evidence would show that this crossing happened, and on what basis it was permitted or refused. These are owner-level trust-boundary questions, not a specification for retrieval filtering, prompt sanitization or vector-store architecture — those implementation choices belong to COM-017.

16. Map Agents, Tools and External Actions

Agentic systems require an authority map in addition to a model map. A practical decomposition is: sponsoring principal → agent identity → objective → model → memory/state → plan/orchestration → tool permissions → delegated authority → approvals → actions → resulting state → telemetry → containment The owner should know which tools the agent can invoke, which identities or tokens it uses, whether it can delegate to sub- agents, what transaction/action limits apply, where human approval is meaningful, and how authority can be revoked. Do not treat tool names as static configuration. Mutable tool definitions, schemas, connectors or remote service behavior can change what the agent is capable of doing after initial approval. For high-consequence actions, preserve enough telemetry to reconstruct the action chain without assuming that full internal model reasoning can be recovered. See §17 for what the owner should be able to determine about current authority and revocation propagation for delegated agent actions. See §15 for the related question of how retrieved content can cross a trust boundary into agent planning or tool selection.

17. Map Identities, Credentials and Delegated Authority

Map human, service and agent identities that can influence the system. Include user roles, administrators, service accounts, API keys, OAuth grants, workload identities, agent identities, tool credentials and emergency/override access. For delegated AI action, record both who sponsored the authority and which technical identity exercised it. This distinction matters when an agent acts on behalf of a user but uses a shared service credential. Review privilege in terms of what can actually be changed or executed, not merely the IAM role name. The owner should know where revocation occurs

and whether revoking one token truly constrains all relevant paths. Least privilege remains useful, but agentic systems also require attention to least agency: limiting the range, duration and downstream consequences of actions the system is permitted to initiate.

Authority granted in the past is not the same fact as authority that remained current when a consequential action was executed. For agentic and delegated-authority systems, the owner should be able to identify:

- where current authority is evaluated — the relevant execution boundary at which current authority for the consequential action is evaluated, as distinct from where it was originally granted;

- how revocation propagates across delegated execution paths — when a sponsoring identity's authority is revoked or constrained, what happens to authority that was already delegated downstream;

- what authority may survive upstream revocation — some delegation architectures intentionally decouple a delegated credential from the sponsor's current session or standing, and this should be a known design property rather than an untested assumption;

- what evidence demonstrates the resulting authority state after a revocation or constraint action — distinguishing evidence that the change was requested from evidence that the intended authority state was reached at the relevant execution boundary.

This is a verification question, not an architecture requirement. COM-019 does not prescribe a specific identity, token or delegation pattern — that belongs to COM-017. The owner's responsibility is to know the answer for the system's actual architecture, and to know where the answer is uncertain.

18. Map Dependencies and Trust Boundaries

Map dependencies whose failure or change can alter system behavior or assurance. Typical dependencies include model APIs, cloud services, model registries, libraries, prompt/configuration repositories, vector stores, MCP/tool servers, identity providers, telemetry platforms, safety controllers, external data feeds and human operations. For each material dependency, ask:

- Is it internal, shared or third-party?
- Who owns the relationship?
- What change visibility exists?
- What evidence can the enterprise observe?
- What fallback or exit path exists?
- Can a failure propagate across multiple AI systems?

A dependency map should expose trust boundaries, not just architecture boxes. A connector may be technically small but security-significant if it crosses from generated text into code execution, payments, credentials or physical control.

19. Identify Consequence and Physical-Effect Paths

Where AI can influence the physical world, the system owner should map the consequence path beyond the software stack. The current practitioner sequence is: environment → sensing/perception → inference → autonomy state → command authorization → command → actuator state → safety-controller state → operator action → physical outcome → recovery/revalidation Not every system contains every element, but the owner should understand where AI output becomes an authorized physical command and which independent safety mechanisms constrain it. The system owner does not decide functional safety by using this map. The purpose is to identify the relevant security/safety interface, the authorities that govern containment and recovery, and the evidence needed to reconstruct a physical event. Detailed physical-AI treatment hands off to PAI-SF™ and COM-014.

20. Define Scope and Material-Change Triggers

Scope should include explicit material-change triggers. A change is material when it can reasonably invalidate a previous assumption about purpose, authority, exposure, control operation, evidence or

consequence. Examples include a model/provider switch, new tool permission, new data source, autonomy increase, new user population, major prompt/orchestration change, provider control change, new physical action path or significant business repurposing. Do not rely only on conventional release-version numbers. A remote provider or tool server can change behavior without changing the local application version. The owner should define who is notified, who reviews the change, which records/evidence are revisited and which decision authority is engaged. COM-019 does not prescribe a universal change threshold; materiality remains context-specific. See §92 (System Boundary Record) for a management aid that captures this boundary.

PART III — ESTABLISH OWNERSHIP AND DECISION RIGHTS

Use this Part when: responsibility is shared across business, security, engineering, providers, risk, legal/privacy or safety functions.

Working output: a named accountability map, decision-rights map and escalation route.

Part checkpoint: Can the owner say who operates, who evidences, who challenges and who decides for each material issue?

The checkpoint is an informative management prompt, not a readiness gate, assessment state or certification prerequisite.

21. Name the Accountable System Owner

Each material AI system should have an identifiable accountable owner. The owner should be a role with enough organizational authority to coordinate across functions, receive material-change information, demand clarification from control owners and providers, and escalate when the operating basis is no longer supportable. Avoid "owned by AI Governance Committee" when no person or role is accountable for the actual service. Committees can govern; they do not substitute for operating accountability. If ownership transfers, record the effective date, new owner, unresolved issues, current evidence limitations and pending decisions. A silent ownership handoff is itself an assurance weakness because assumptions and obligations can be lost. See §22 for how to treat ownership that is contested between functions, rather than simply missing.

22. Map Control and Operating Owners

Build a practical ownership map for material controls and operating processes. At minimum, distinguish who:

- defines the control expectation;
- implements or configures the mechanism;
- operates it day to day;
- monitors failure;
- generates the operational record;
- decides exceptions;
- remediates defects;
- provides independent challenge.

One person may perform several roles in a small organization, but the roles should remain conceptually distinct. The system owner should focus on missing interfaces. A strong enterprise access-control team does not help if nobody knows who reviews agent tool permissions for this system. A good logging platform does not help if no one owns the population or alert logic relevant to the AI service.

Ownership can also be contested rather than missing — two functions may each believe the other owns a control, decision or dependency, and both can point to something that looks like coverage. This is not resolved merely because a document names both parties. Where ownership is genuinely disputed or unclear between functions, record the dispute itself — which parties believe they hold the responsibility, and on what basis — and escalate it until the operating and decision boundary is explicit for that specific

interface. A visible, unresolved ownership dispute is a more useful record than a document that quietly implies coverage exists.

23. Map Provider and Shared Responsibility

Third-party AI creates two different questions: who operates the control and who can evidence it. For implementation responsibility, COM-019 carries the COM-017 practitioner descriptors Enterprise-controlled / Provider-controlled / Shared / Visibility-limited. These describe who controls the mechanism and are not assessment states. Do not confuse them with evidence-source states used later in COM-015, such as Customer-observed or Provider- provided. A provider-controlled mechanism may still produce customer-observable evidence; an enterprise-controlled integration may depend on provider-only information. For each material shared control, record the enterprise obligation, provider obligation, interface, failure notification, available evidence, unresolved visibility gap and fallback/exit path. A contract allocates responsibility; it does not prove current operating effectiveness. These are two independent dimensions, not one scale. A worked example makes the distinction concrete: a mechanism can be Enterprise-controlled and still lack usable customer-observed evidence if the relevant observation capability was never established. In that case, implementation responsibility remains Enterprise-controlled while the evidentiary position may remain Unknown. Conversely, a mechanism can be Provider-controlled while producing Customer-observed evidence where its externally observable behavior is captured through enterprise-side telemetry, even though the enterprise does not operate the mechanism itself. Knowing who controls a mechanism does not tell you what evidence exists about it, and knowing what evidence exists does not tell you who is responsible for the mechanism. Neither dimension ranks or scores the other — they are answered independently for each material mechanism, dependency or operating claim for which the distinction matters.

24. Define Security, Governance, Privacy, Legal and Safety Interfaces

The owner should know which specialist function governs each material decision. Useful interfaces commonly include:

- security — architecture, security exceptions, cyber containment;
- AI governance/model risk — use-case governance and non-security AI risk;
- privacy — personal-data processing and privacy decisions;
- legal/compliance — obligations, reporting and claims;
- procurement/supplier risk — contracts, provider controls and exit rights;
- engineering/platform — implementation and service operation;
- SOC/CSIRT — detection, triage and response;
- internal assurance/audit — independent challenge;
- safety/OT — safe-state and return-to-service authority where applicable.

The goal is not a large RACI. It is a decision map that shows who is consulted, who supplies evidence and who actually decides.

25. Define Human Authority and Oversight Responsibilities

Human oversight should be evaluated as practical authority, not as a box labelled "human in the loop." COM-019 carries the practitioner dimensions: Awareness / Comprehension / Time / Authority / Effectiveness For a consequential decision, ask whether the person knows intervention is required, can understand the relevant system information, has enough time to act, possesses real authority to change or stop the outcome, and can exercise that authority effectively. A nominal approval step may provide weak assurance if the reviewer cannot inspect the basis, is overloaded, or cannot reverse the action. COM-019 does not prescribe universal human approval for every autonomous action. The system owner should identify where human authority is part of the actual control design and whether that authority remains credible under real operating conditions.

26. Define Exception and Risk-Acceptance Routes

Define how unresolved control conditions are handled. An exception route should identify the condition, affected system/use case, compensating measures where applicable, owner, duration/review trigger, evidence limitations and the authority that can accept the residual business risk. The system owner may recommend or coordinate an exception but should not assume authority that belongs to security, risk, legal, safety or executive management. Temporary measures should remain visibly temporary. Avoid rolling exceptions forward without revisiting the original assumption. COM-019 does not create a universal exception duration, severity scale or risk-acceptance threshold.

27. Define Change and Release Decision Rights

Release authority should be explicit for material AI changes. Separate at least three questions:

1. Who verifies that the intended change was implemented?

2. Who decides whether the security/operational basis is acceptable for deployment?

3. Who accepts residual business risk when material concerns remain?

For provider-hosted systems, a local "release" may occur even though the provider changes the model independently. The owner should therefore include provider-change handling in the decision-rights model. A release decision is an enterprise management decision. It is not an ODA3 assessment result or certification status.

28. Define Incident and Containment Authority

Incident authority should be designed before an event. The owner should know who can disable a feature, revoke an agent token, block a tool, change routing, isolate a connector, suspend a provider integration, freeze changes, preserve records or invoke a fallback. Some containment actions may sit within security's pre-authorized decision rights; others may require business, legal, safety or operational coordination. COM-019 does not determine which actions are physically or operationally safe. For physical AI, digital isolation or shutdown can itself create physical risk. The cyber containment route should therefore connect to the relevant safety/operational authority. COM-019 prepares the authority map; active incident command belongs to the organization's response process and AI- IRF™ guidance.

29. Define External-Statement and Assurance-Claim Authority

External assurance claims need their own decision rights. This section addresses who is authorized to make an external statement. For what may or may not be said regardless of speaker, see §50. For the pre-release check to run before a statement leaves the organization, see §98. The system owner may know the system best but may not be authorized to tell a customer, auditor or regulator that the system is "compliant," "certified," "secure" or "approved." Define who can make:

- factual system descriptions;
- scoped alignment statements;
- security representations;
- contractual assurance statements;
- regulatory submissions;
- certification/assessment claims.

Before an external statement is made, verify its scope, period, evidence basis, limitations and authority. The owner's responsibility is often to provide accurate system facts—not to convert those facts into a legal or certification conclusion.

30. Build the Escalation Map

Create an escalation map that works under uncertainty. Useful triggers include loss of a named owner, unknown model/provider change, evidence-access failure, material control drift, newly expanded agent authority, unresolved high-consequence provider dependence, repeated failed tests, suspected incident, physical consequence path, or a decision that exceeds the owner's authority. For each trigger, record the next authority and the information that should travel with the escalation. Escalation should preserve uncertainty rather than hiding it. "We do not know whether the provider changed the relevant control after this date" can be a more responsible escalation statement than an unsupported assurance conclusion.

The escalation map should also address what happens when the named decision authority is not available when a decision is needed. For each material decision authority identified elsewhere in this guide, the owner should be able to identify either an alternate authority who can act in their place, or the bounded operating state that applies while no authority is reachable — for example, continuing under existing constraints, or treating the condition as one requiring containment until authority is restored. This is a continuity question, not a timing requirement: COM-019 does not prescribe how quickly an alternate must be reached or set an escalation clock. The point is that "the decision-maker was unreachable" should not, by itself, leave the system's operating state undefined.

PART IV — TURN REQUIREMENTS INTO OPERATING

RESPONSIBILITIES

Use this Part when: the applicable requirements exist on paper but ownership, operation, failure detection or evidence generation are

unclear.

Working output: a traceable connection from applicable requirement/control to operating responsibility and real mechanism owner.

Part checkpoint: Can the owner trace each material expectation to an operating mechanism, owner, failure signal and evidence

source without claiming assessment? The checkpoint is an informative management prompt, not a readiness gate, assessment state or certification prerequisite.

31. Use the Applicable Requirement and Control Baseline

The system owner should begin with the applicable requirement/control baseline established by authoritative governance, risk and framework processes. Within the GAISSF Ecosystem, GAISSF™ provides the principal safety/security control context. The owner should know which controls or requirements apply to the system and where the authoritative applicability record is maintained. COM-019 does not create a new system-owner baseline. It also does not determine legal applicability. The owner's job is to make sure the applicable requirement can be traced to an operating responsibility and a real system mechanism. If a requirement is listed as applicable but no one can identify what implements it, who operates it or how failure is detected, the operating picture is incomplete.

32. Separate Requirement, Control and Mechanism

Keep three layers distinct: Requirement — the obligation or expected outcome. Control — the organized capability used to address the requirement. Mechanism — the actual technical, procedural or human implementation operating in the system. For example, a requirement for controlled tool access may map to an access-control capability, implemented through agent identity, allowlists, scoped OAuth tokens, transaction limits and approval logic. This distinction prevents policy language from being mistaken for implementation. The system owner does not need to design every mechanism, but should be able to trace the applicable expectation to the owner and operating mechanism that make it real.

33. Handoff to COM-017 — Implementing GAISSF™

Detailed control implementation belongs in COM-017 — Implementing GAISSF™. Use COM-019 to establish:

- what the system is and why it is in scope;
- who owns the control or mechanism;
- which shared/provider dependency matters;
- how operation and failure are observed;
- what records should exist;
- which material changes require renewed attention.

Use COM-017 for decomposition, mechanism design, engineering integration and implementation-side verification. Implementation output becomes a candidate evidence source. It does not automatically establish conformity, evidence sufficiency, certification readiness or a favorable assessment result.

34. Assign Control Operating Responsibility

For every material control, identify who keeps it operating after implementation. Projects often have a clear build owner and no clear operating owner. Once the deployment team leaves, configuration drifts, permissions accumulate, alerts become noisy and provider assumptions age. The operating-responsibility record should include the mechanism, owner, operating activity, failure signal, change trigger, evidence source and escalation route. Shared enterprise controls can be reused. The system owner should not demand a dedicated mechanism when an existing IAM, logging, change-management or supplier-control process already satisfies the need. Instead, document how the system depends on that shared control and who owns the dependency. This section assigns and traces operating responsibility; it does not prescribe the technical mechanism or independently verify control effectiveness. Detailed implementation and implementation-side verification remain in COM-017.

35. Assign Monitoring and Failure-Detection Responsibility

A control cannot support ongoing operation if no one can tell when it fails or becomes irrelevant. For material mechanisms, ask:

- What observable condition indicates normal operation?
- What indicates degraded or failed operation?
- Who receives the signal?
- How does the signal connect to a case, ticket or decision?
- What population or component does monitoring actually cover?
- What important behavior remains invisible?

Monitoring should be designed for decisions, not for dashboard volume. COM-019 does not require universal prompt/output logging. Logging should reflect the risk, privacy, security and operational context. The system owner needs to know the signal, owner, coverage and blind spot; COM-019 does not convert that management need into a universal detection threshold, telemetry minimum or monitoring test protocol.

36. Assign Evidence-Generation Responsibility

Evidence generation should be assigned as an operating responsibility, not postponed until audit time. For each material control or system claim, identify what record is naturally produced during operation and who owns it. Examples include configuration state, approval history, telemetry, access records, change tickets, provider notices, test results, incident records and remediation evidence. The system owner should also ask whether the mechanism producing the evidence can be trusted within the relevant threat model. Evidence provenance is useful, but provenance alone is weak if the same system can silently alter or suppress the record. For a material system claim or consequential action, the owner should be able to determine:

1. what system state was operating at the relevant time;

2. who or what initiated the action — the acting identity or process;

3. what authority applied to that action;

4. what material inputs or dependencies influenced the action or claim;

5. what control or policy decision occurred, if any;

6. what consequential state change resulted;

7. what evidence supports each of the above, and what remains a visibility limitation.

These are Minimum Evidence Questions, not a logging specification. They describe what the owner should be able to determine, not the specific fields, formats, retention period or technical mechanism used to determine it — those implementation choices depend on system architecture, consequence, privacy constraints and organizational context. COM-017 provides the relevant implementation guidance; COM-015 governs evidence-pack preparation. Recording "Unknown" for one of these questions in relation to a given claim is itself useful information: it identifies a visibility gap rather than concealing one.

COM-019 does not define assessor evidence sufficiency. Its purpose is to ensure that operational evidence can exist before it is needed.

37. Operate Third-Party and Shared Controls

Third-party operation does not remove system-owner responsibility for the interface. For hosted AI, the owner should maintain a baseline covering service identity, authorized endpoint, data-handling terms, provider responsibilities, change notification, incident/support route, evidence access, fallback/exit and known visibility limitations. Where provider evidence is limited, preserve the limitation rather than translating it into confidence. Provider certifications, audits or reports may be useful scoped evidence. They do not automatically establish the end-to-end enterprise system's conformity or operating effectiveness. The owner should know which enterprise-side observations can corroborate provider behavior and which questions remain provider-only.

38. Operate Agentic-AI Controls

Agentic AI requires operating controls around identity, authority and consequence—not only output quality. Relevant patterns can include per-agent/service identity, scoped tool allowlists, least agency, value/destination/time/delegation limits, memory isolation, context-specific approval, rate or action constraints, circuit breakers, revocation and rollback. The owner should know how tool metadata/configuration changes are governed. A tool server that changes its schema or behavior after approval can alter the agent's practical authority without a new application release. Human approval is not a universal answer. Where it is used, test whether meaningful human authority exists across Awareness, Comprehension, Time, Authority and Effectiveness. Detailed implementation remains in COM-017.

39. Physical-AI Security Handoff

When the system can create or materially influence physical consequence, the system owner should hand off to PAI-SF™ / COM-014 for detailed treatment. The owner remains responsible for identifying the consequence path, security-control owners, applicable safety/operational authorities, incident interfaces and evidence handoffs. GAISSF Domain 9 remains the continuity point; PAI-SF™ supersedes and expands the detailed physical-AI security- assurance layer represented by D9. This does not create two independent sets of physical-AI obligations to double-count. Functional safety, roadworthiness, airworthiness, clinical safety, machinery safety and sector approvals remain separate where applicable. The system owner should never infer that a cyber fix by itself authorizes physical return to service. This holds even under operational pressure to resume service. If the safety authority required to authorize physical return to service is unavailable, that unavailability does not transfer the decision to the system owner, the CISO, or another party merely because operations are under pressure. Return to service should remain within the applicable safety-authority structure, including any formally established alternate or emergency authority. Where no authorized route is available, the system remains constrained. Operational urgency is a reason to escalate the authority gap, not a basis for assuming authority that has not been granted.

40. System-Owner Implementation Check

Before treating implementation as operationally established, the system owner should be able to answer:

- What requirement/control is this mechanism addressing?
- Who owns and operates it?
- What component/provider does it depend on?
- What failure signal exists?
- What evidence does normal operation generate?
- What change could invalidate it?
- What is the fallback or containment path?
- What remains unknown?

This is a management implementation check, not an ODA3 assessment procedure, readiness gate, control rating or replacement for COM-017 implementation-side verification. A "yes" on the checklist does not establish control effectiveness or conformity. A "no" identifies an operating question that needs clarification, ownership or escalation. Where COM-017 implementation records exist, they may inform the answer without becoming an assessment result.

PART V — MAINTAIN EVIDENCE AND OPERATIONAL MEMORY

Use this Part when: the organization needs to show what was operating, for which system/version/period, and what remained

unknown or contradictory.

Working output: a bounded operational record that can later feed COM-015 evidence preparation.

Part checkpoint: Can the owner locate current evidence and preserve its scope, period, provenance, limitations and historical state?

The checkpoint is an informative management prompt, not a readiness gate, assessment state or certification prerequisite.

41. Evidence Is Part of Operation

Evidence should emerge from normal operation wherever practical. If assurance depends on a control, the operating model should make it possible to later show what mechanism was in place, over what period, for which version/scope, and what happened when the mechanism was tested or triggered. This reduces the common audit-time pattern of reconstructing reality from memory. The system owner should encourage records that are decision-useful rather than maximal. More data is not automatically better evidence. The guiding principle is: Assurance is demonstrated, not asserted. In COM-019, that principle supports a bounded management position; formal evidence sufficiency, independent assurance and assessment outcome remain outside this guide.

42. Separate Design Evidence from Operating Evidence

Design evidence and operating evidence answer different questions. Design evidence can show intended architecture, policy, control design, configuration standard or test plan. Operating evidence can show what happened in the actual system over time: deployed configuration, telemetry, approval records, incidents, exceptions, provider notices, access events and remediation. Both can matter. Neither substitutes automatically for the other. A design document does not prove the configuration operated as designed. A log excerpt does not prove the overall control design was adequate. The system owner should preserve enough of both layers to explain the current claim and its limits.

43. Preserve the Minimum Operational Record

Maintain a minimum operational record for each material AI system. A practical record includes:

- system/use-case identity;
- accountable owner;
- current model/provider/version basis;
- system boundary reference;
- applicable control/requirement reference;
- material control and dependency owners;
- current material exceptions/unknowns;
- last material change;
- incident/readiness references;
- key operational evidence locations;
- current decision/escalation status.

This can be federated across existing systems. COM-019 does not require one master spreadsheet. The purpose is to prevent critical context from existing only in individual memory or scattered project documents.

44. Preserve Scope, Period, Version and Provenance

Every material operational record should preserve enough context to know what it actually supports. At minimum, retain: Scope — which system/component/use case. Period — when the record applies. Version/state — model, provider, configuration or release basis where material. Source — where the record came from. Provenance/integrity context — how it was generated or maintained. Limitations — what it does not show. These attributes should travel with important evidence. A screenshot without system/version/date context may be visually persuasive but operationally weak.

Provenance is only useful if the record itself can be trusted not to have changed silently. For material evidence, the owner should be able to say whether the record could have been altered, deleted or regenerated after the fact without that being visible — not through a specific technical mechanism, which is an implementation question, but as a property the owner knows to ask about. A record whose integrity cannot be spoken to is weaker evidence than its content alone suggests, even when the content itself looks persuasive.

45. Separate Provider Representation from Enterprise Observation

Provider information should remain distinguishable from enterprise observation. COM-019 adopts the evidence-source language used by the practitioner suite:

- Customer-observed
- Provider-provided
- Shared-interface
- Provider-only-inaccessible
- Unknown

These are normalized practitioner evidence-source states, not control ratings, implementation-responsibility states or assessment confidence levels. Record provider scope, period, criteria, exclusions and known limitations. If evidence is inaccessible, preserve that as an assurance limitation rather than assuming either success or failure. A provider statement becomes stronger when it can be corroborated with enterprise-side telemetry or behavior, but not every provider-controlled mechanism will be independently observable. See §23 for a worked example distinguishing this evidence-source dimension from implementation-responsibility.

46. Capture Human Decision Evidence

Human decisions can themselves be assurance evidence when the reasoning matters. For consequential approvals, overrides, exceptions, continued-operation decisions or escalations, preserve:

- decision and date;
- responsible decision-maker;
- information available at the time;
- material uncertainty;
- dissent or challenge;
- conditions/constraints;
- escalation route;
- rationale;
- follow-up obligation.

This is Human Decision Evidence. The term is a COM-019 management shorthand for preserving consequential human reasoning. It does not create a seventh COM-015 evidence class, a mandatory GAISSF artifact, an assessor evidence type or a confidence rating. A recorded rationale does not prove the decision-maker genuinely understood the issue or that the decision was correct. It does, however, make authority, information and uncertainty visible for later review.

47. Preserve Unknowns, Contradictions and Negative Evidence

Do not curate the operational record into a success story. Preserve material:

- failed or negative tests;
- retests;
- contradictory evidence;
- provider representations that later changed;
- unresolved anomalies;
- unknown/inaccessible evidence;
- specifically checked-but-not-observed conditions where the search basis matters.

These are two distinct states, and neither should be treated as proof that a condition did not occur.

Unknown means the relevant fact is not established — there is no evidence either way, and the supportable claim is narrower as a result, not false.

Checked and not observed means a defined examination actually occurred, and the condition was not observed within that examination's stated scope, period and visibility. This is only meaningful when the owner can state what was checked and within what boundary — an unstated "we looked and didn't see it" carries little weight.

Neither state should be silently converted into a claim that the condition is absent. What was not examined remains genuinely unknown, and what was examined but not observed remains bounded by the examination's own limits. These management distinctions are not new UAIF enums or assessment confidence states.

48. Keep Evidence Current Through Change

Evidence is current only relative to the system state it represents. When a model, provider, tool, permission, retrieval source, orchestration path or business purpose changes, revisit the evidence linked to the affected assumptions. Do not “refresh” evidence by changing dates on an old artifact. Preserve history. The organization may need to know what evidence supported the earlier state, what changed, and when the earlier claim stopped applying. Temporal-integrity rule: do not rewrite later remediation or

improved evidence as though it operated throughout an earlier period. The earlier state, limitation, failure or unknown remains part of the historical record. The system owner should connect material-change records to the evidence that needs reconsideration. COM-015 can later structure the evidence pack; COM-019 keeps the operational continuity visible. See §96 (Operational-Evidence Handoff Checklist) for a management aid when preparing operational records for evidence handoff.

49. Handoff to COM-015 — Building an AI Assurance Evidence Pack

Use COM-015 — Building an AI Assurance Evidence Pack when operational records need to be prepared into a structured evidence handoff. COM-019 should supply the system context: scope, version, owners, provider boundaries, change history, evidence locations, failed/negative results, contradictions and known limitations. COM-015 owns evidence indexing, provenance treatment, evidence-source states, limitations and pack preparation. Evidence-pack completeness does not establish evidence sufficiency, conformity, readiness or certification. The system owner should therefore avoid promising an assessment outcome based on the existence of a well-organized pack.

50. System-Owner Claims Firewall

The system owner should resist claims that are broader than the evidence and authority. This section addresses the content of a claim, independent of who is authorized to make it — see §29 for speaker authority — or how a specific statement is checked before release — see §98. Avoid unsupported statements such as:

- "the AI is secure";
- "fully compliant";
- "GAISSF certified";
- "ODA3 approved";
- "audit ready";
- "safe to operate";
- "zero risk";
- "the provider is compliant so our system is compliant."

Prefer bounded factual statements: what scope was reviewed, what controls are implemented, what evidence exists, what remains unknown, what period/version the statement covers, and which authority —if any—issued a separate assessment or certification. COM-019 does not create authorization to use reserved ODA3 claims.

PART VI — MANAGE CHANGE, PROVIDERS AND LIFECYCLE

Use this Part when: a model, provider, prompt, tool, permission, RAG source, business purpose or lifecycle state changes.

Working output: a material-change record showing affected assumptions, evidence, controls, dependencies and decision authority.

Part checkpoint: Can the owner explain what changed, what earlier evidence may no longer support, and what decision or follow-up is

required? The checkpoint is an informative management prompt, not a readiness gate, assessment state or certification prerequisite.

51. Treat Material Change as a System-Level Review Trigger

Treat material change as a system-level management review trigger because change can invalidate the basis on which continued operation was previously justified. This phrase does not create a formal reassessment event, mandatory gate or ODA3 status transition. A useful change review asks:

- What changed?
- Which prior assumptions are affected?

- Does scope change?
- Do control owners or dependencies change?
- Does evidence remain relevant?
- Does incident/containment design still work?
- Does the consequence path change?
- Which authority needs to decide or be informed?

Not every patch requires the same review depth. COM-019 does not prescribe universal thresholds. The point is to prevent material change from passing through ordinary release processes without revisiting the assurance assumptions it alters.

52. Model and Provider Change

Model and provider changes can alter behavior even when the user interface remains unchanged. Track provider/model identity, version or release basis, routing, fallback and key service terms. For a provider substitution, revisit data handling, model behavior assumptions, evaluation/test relevance, monitoring, evidence access, incident route, jurisdiction/region where material, subprocessor dependencies and exit/fallback. If the provider changes a model without exposing a version, document the visibility limitation and use available behavioral/change signals. Do not assume that prior evidence transfers automatically to a materially different model or provider.

53. Prompt, Orchestration and Tool Change

Prompts, system instructions, orchestration logic and tool definitions can materially alter behavior without changing the underlying model. Treat significant changes to:

- system prompts;
- policy/guardrail instructions;
- agent planning logic;
- tool schemas;
- tool allowlists;
- connector definitions;
- approval rules;
- downstream action logic

as candidates for material-change review. Mutable remote tool metadata deserves particular attention because authority can change after initial onboarding. The owner should know whether configuration changes are versioned, who can make them, what testing occurs and how rollback works.

54. Data, RAG and Knowledge-Source Change

RAG changes can be assurance-significant even when no model code changes. Revisit the chain: source → ingestion → integrity/approval → index → retrieval authorization → context assembly → model → output → downstream use when repositories, connectors, permissions, embedding/indexing logic, source approval, ranking or context assembly change materially. A newly connected knowledge source can expand both information exposure and prompt-injection/content-poisoning risk. The owner should preserve source provenance, authorization assumptions, index refresh/change information and known retrieval limitations. Do not assume that an "approved model" makes an ungoverned retrieval layer safe.

55. Identity, Permission and Delegation Change

Identity and permission changes can increase AI consequence without altering the model. Review changes to:

- agent/service identity;

- OAuth grants;
- API keys;
- tool permissions;
- transaction limits;
- delegated authority;
- approval requirements;
- sub-agent delegation;
- administrator roles.

For agentic systems, ask whether practical agency increased: more destinations, higher transaction value, broader tools, longer-lived tokens, deeper delegation or reduced human intervention. Ensure revocation paths still work after the change. A permission change can be material even when the code diff is trivial.

56. Provider Opacity and Change Notification

Provider opacity should be treated as a managed limitation. The owner should know how provider changes are learned: contractual notice, release notes, API versioning, model identifiers, service communications, support channels, customer telemetry or enterprise behavior tests. If material changes can occur without notice, record that fact and decide how the organization will detect unexpected behavior. Unknown provider state should not be silently converted into "no change." Where visibility limitations materially affect a decision, escalate them to the relevant security/risk/business authority rather than asking the system owner to manufacture certainty.

57. Shared Dependency and Concentration Exposure

Shared dependencies can create portfolio-scale risk from a system-level component. A single model provider, identity service, vector database, tool server, cloud platform or telemetry service may support many AI systems. The system owner should identify material shared dependencies and report them into the enterprise portfolio so the CISO or relevant risk function can see concentration exposure. At system level, maintain fallback and failure assumptions. At portfolio level, shared dependency risk belongs in COM-018's enterprise view. COM-019 therefore surfaces concentration; it does not create the enterprise concentration model.

58. Fallback, Rollback and Degraded Operation

Fallback and degraded operation should be designed before they are needed. Possible patterns include:

- alternate model/provider;
- reduced feature set;
- lower authority;
- read-only mode;
- human/manual processing;
- delayed processing;
- system suspension.

Do not label a fallback "safe" merely because it is simpler. It can introduce new errors, capacity problems or operational hazards. Record who can invoke the fallback, what conditions trigger consideration, how data/state are preserved, how users are informed where relevant, and what validation is needed before normal authority returns. For physical AI, coordinate fallback with safety/operational authority.

59. Lifecycle Transition, Transfer and Re-Purposing

AI systems often change ownership or purpose without being rebuilt. Treat acquisition, transfer between business units, platform migration, regional expansion, customer repurposing, new decision use or increased autonomy as lifecycle transitions that can invalidate earlier assumptions. A transfer record should carry the system boundary, current owners, provider dependencies, open exceptions, incident history, evidence limitations, material changes and pending decisions. Do not assume that a system accepted for one use case is automatically acceptable for a materially different one. Scenario similarity is not an applicability or assurance conclusion.

60. Retirement and Decommissioning

Retirement should remove more than the user interface. Plan for:

- model/API access revocation;
- agent/tool token revocation;
- connector removal;
- scheduled job shutdown;
- data/index retention or deletion decisions;
- provider contract/entitlement closure;
- monitoring decommissioning;
- evidence and decision-record retention;
- user/process migration;
- physical-system handoff where applicable.

Confirm that dormant credentials or integrations cannot continue to act. Retirement is also an evidence event: preserve enough historical context to explain what system existed, when it stopped operating and what unresolved obligations remain. See §94 (Material-Change Review Record) and §95 (Provider/Shared-Responsibility Record) for management aids covering change and provider dependencies.

PART VII — INCIDENT READINESS AND RESPONSE INTERFACE

Use this Part when: the system needs to be identifiable, reconstructable, containable and ready to hand off to incident responders.

Working output: an incident-readiness interface covering records, containment surfaces, provider escalation and specialist authorities.

Part checkpoint: Could responders identify the current system state, preserve the important evidence, constrain relevant authority and

reach the right provider/business/safety roles? The checkpoint is an informative management prompt, not a readiness gate, assessment state or certification prerequisite.

61. Design Incident Readiness Before the Incident

Incident readiness is an operating property, not a document stored with the project. Before an event, the owner should know:

- which system/version/provider is involved;
- who can access relevant records;
- which identities/tools can be revoked;
- how to isolate or constrain the service;
- how to reach the provider;
- which business and safety dependencies are affected;

- who owns incident command;
- where evidence should be preserved.

A system that cannot be identified, reconstructed or constrained may discover its assurance weakness only after harm begins. COM-019 designs readiness; active response remains with incident processes and AI-IRF™.

62. Detection Is Not Automatically an Incident

A monitoring signal, complaint, model anomaly or unexpected output is not automatically a formally classified AI incident. A signal / observation / event under triage can remain in the organization's existing SOC, support, quality or case- management workflow while the organization determines what occurred. Those local workflow labels do not create a UAIF record type. The owner should preserve the signal, affected scope, time, version/provider basis, immediate actions and unknowns. Do not invent a UAIF Observation record type or unsupported schema field for convenience. Response may begin before classification is complete. Classification and response can evolve together as evidence improves.

63. Preserve Execution Context and Operational Evidence

Preserve enough execution context to reconstruct consequential behavior. Depending on the system, this may include:

- request/event identifiers;
- model/provider/version basis;
- prompt/configuration version;
- retrieved sources;
- agent identity and tool calls;
- approvals and delegated authority;
- resulting external actions/state changes;
- relevant telemetry;
- provider messages;
- operator interventions.

The objective is not perfect replay of hidden model cognition. It is sufficient reconstruction of the operational question. Balance evidence preservation with privacy, security, privilege and data-retention requirements.

64. Prepare Containment and Revocation Surfaces

Pre-identify the surfaces through which the system can be constrained. Possible containment surfaces include:

- user/session;
- identity/token;
- tool/API;
- connector/data source;
- agent;
- model route;
- workflow;
- feature flag;
- provider integration;
- network/service;

- physical command path.

The owner should know which team controls each surface and what side effects containment can create. Containment should target the authority or exposure driving risk. For physical AI, cyber containment may need coordination with safety/operational authority because abrupt shutdown or isolation can itself create physical hazard.

65. Prepare Provider Escalation and Dependency Containment

Hosted and shared AI services need a provider escalation path before an incident. Record:

- support/security contact route;
- severity/escalation channel if contractually defined;
- evidence-request process;
- emergency access/restriction options;
- change freeze or routing options where available;
- provider incident notification;
- subprocessor dependency;
- fallback/exit.

During an incident, separate provider representation from enterprise-observed facts. If provider evidence is delayed or unavailable, preserve that as uncertainty. Do not let lack of provider visibility stop proportionate enterprise containment when local evidence supports action.

66. Prepare for Agentic-AI Incidents

Agentic incidents can involve unauthorized action even when the model output itself looks normal. Prepare to reconstruct: principal → agent identity → objective → plan/orchestration → tool permission → delegated authority → approval → action → resulting state Relevant incident capabilities can include rapid token revocation, tool disablement, delegation cutoff, memory/session isolation, transaction freeze, circuit breaker and rollback. Preserve tool metadata/schema versions where change could have altered authority. Do not assume a human-approval step was effective; examine whether the reviewer had Awareness, Comprehension, Time, Authority and Effectiveness.

67. Prepare for Physical-AI Incidents

Physical-AI incident readiness should connect cyber evidence to the physical consequence path: environment → sensing/perception → inference → autonomy state → command authorization → command → actuator state → safety-controller state → operator action → physical outcome → recovery/revalidation The system owner should know which logs or records exist at each material point and who controls them. Protect people and critical operations first. Cyber evidence preservation should not override immediate safety action. Physical containment, safe-state decisions and return to service may involve authorities outside cybersecurity. COM-019 does not replace functional safety or sector incident procedures. See §39 for the specific case where the safety authority required for return-to-service is temporarily unavailable.

68. Handoff to UAIF™ for Classification

Use UAIF™ when a potentially material AI event requires structured classification and shared incident information. The system owner contributes system context: use case, scope, model/provider state, action authority, observed facts, evidence limitations and affected dependencies. Keep facts, hypotheses, provider representations and unknowns distinct. Classification can change as evidence changes, and the case history should preserve the basis for material classification change. Technical classification does not itself determine legal reportability, liability or causality. Where structured machine-readable UAIF exchange is used, the published TCR-STD-003/schema companion governs the

supported structure. A COM-019 readiness or owner record becomes UAIF data only when it is mapped through supported UAIF fields/profile structures.

69. Handoff to AI-IRF™ for Response

Use AI-IRF™ for structured AI incident response. The system owner supports response by supplying business purpose, current system boundary, control/dependency owners, provider contacts, containment surfaces, evidence locations, fallback options and decision authorities. Do not require classification to be fully settled before proportionate response begins. COM-019 does not restate the controlled AI-IRF lifecycle as a system-owner checklist. Its role is to make sure the system can participate effectively in the response process. AIIS v1.0.0 is the published machine-readable control/playbook encoding; future conformance_test population is distinct from schema availability. The 79-control / 19-domain technical architecture does not cause COM-019 to carry forward historical certification or scheme-state language; current public practitioner treatment keeps AI-IRF™ publication state separate.

70. Recovery, Revalidation and Return-to-Service Boundary

Recovery should aim to restore a supportable operating basis, not merely technical availability. COM-019 does not define a universal recovery, validation or return-to-operation acceptance criterion. Before normal operation resumes, determine what changed, what was fixed, what evidence exists, which prior assumptions remain valid and which controls or dependencies require revalidation. For non-physical systems, the applicable business/security authorities decide whether operation can resume. For physical AI, distinguish security recovery/closure from safety or operational return to service. Both may need to be satisfied, but they are not the same decision. COM-019 does not create return-to-service criteria or authority. See §97 (Incident-Readiness Handoff Checklist) for a management aid confirming responders can engage the system.

PART VIII — OPERATE, REVIEW, REMEDIATE AND IMPROVE

Use this Part when: the system is in ongoing operation and the concern is drift, stale evidence, recurring failures, exceptions or

remediation.

Working output: a current view of material drift, exceptions, remediation and lessons requiring local or enterprise action.

Part checkpoint: Can the owner distinguish current operation from temporary containment, later remediation and unresolved historical

gaps? The checkpoint is an informative management prompt, not a readiness gate, assessment state or certification prerequisite.

71. Establish an Operating Review Rhythm

Establish a review rhythm that fits the system's change rate, consequence and dependency model. Useful review triggers can include material change, provider notice, control failure, evidence gap, incident, exception expiry, major business repurposing or periodic management review. COM-019 does not prescribe a universal monthly, quarterly or annual cadence. The owner should use existing service-management, risk, security and product forums where they work. The goal is not a new meeting; it is a repeatable point where material changes, gaps and decisions become visible. Event-driven review is often more important than calendar review for rapidly changing AI systems.

72. Detect Control Drift and Broken Assumptions

Control drift occurs when a mechanism still exists but no longer matches the assumption it was meant to support. Examples include expanded permissions, disabled alerts, stale allowlists, provider behavior change, new data paths, reviewer overload, model routing changes or undocumented tool additions. Broken assumptions are equally important. A control can operate exactly as designed while the business use or consequence has changed around it. The owner should ask what changed in the system,

environment, provider or operating process—not only whether a configuration is still present. Drift signals should feed change review, remediation or escalation as appropriate.

73. Detect Evidence Staleness and Coverage Gaps

Evidence can become stale before the underlying control fails. Watch for evidence that no longer covers the current model/provider version, user population, tool set, operating period, region, data source or delegated authority. Also watch for coverage gaps: a dashboard may show healthy metrics for only part of the system. The owner should know the last material evidence period and what has changed since. Do not invent a universal "evidence age" threshold. Currency depends on the claim, system change rate and source. Where staleness matters, record it explicitly so later evidence preparation does not present an outdated artifact as current.

74. Govern Exceptions and Temporary Measures

Exceptions and temporary measures should remain visible in normal operation. Track the condition, affected scope, owner, compensating measures, decision authority, start date, review/expiry trigger, evidence limitation and remediation route. An exception should not disappear simply because the system continues to operate. Temporary controls can also become permanent by inertia. If a temporary measure remains necessary, revisit whether the underlying design or risk decision should change. COM-019 does not create exception severity or universal expiry periods.

75. Maintain the Remediation Portfolio

Maintain a system-level remediation view that distinguishes unresolved issues from completed work. Useful fields include:

- issue/condition;
- affected scope;
- owner;
- origin (incident, test, review, provider notice, audit, change);
- containment/temporary measure;
- intended corrective action;
- dependency/blocker;
- evidence of completion;
- decision/escalation status.

Do not prioritize solely by issue age. A newer issue can be more consequential than an old cosmetic defect. Portfolio-level aggregation belongs to COM-018; COM-019 maintains the system context needed for that aggregation.

Patterns in the remediation portfolio itself can be informative, separate from the status of any single item. Repeated deferral of the same issue, recurrence of a previously closed issue, issues aging without resolution, or items being reopened after being marked complete can each indicate something about the operating basis worth surfacing — a control that isn't actually holding, a dependency that keeps resurfacing, or a capacity gap in remediation itself. These are observational signals for the owner to notice and, where material, escalate or bring into a decision review. COM-019 does not define a universal threshold, score or SLA for when a pattern becomes material — that judgment depends on the system's consequence and context.

76. Distinguish Correction, Containment and Corrective Action

Use precise remediation language. Correction — fixes the immediate defect or record. Containment — limits current exposure or authority. Corrective action — addresses the cause or control weakness. Preventive/systemic improvement — reduces recurrence across related systems or processes. For example, revoking one agent token is containment; correcting the permission policy may be corrective

action; improving centralized agent entitlement governance may be systemic improvement. Do not call containment "resolved" when the root condition remains. COM-019 does not define universal remediation deadlines or severity tiers.

77. Validate the Fix Without Claiming Assessment

After remediation, verify that the intended fix is actually present and operating in the relevant system state. Verification may include configuration inspection, targeted testing, telemetry review, access checks, provider confirmation or observed behavior. Preserve failed retests as well as successful ones. Implementation-side verification is not an assessment result. It does not establish independent assurance, evidence sufficiency, conformity, assessment readiness or certification. If the fix changes the system materially, re-enter the change-review process and revisit affected evidence and assumptions.

78. Learn from Incidents, Near Misses and Failed Tests

Near misses, failed tests and incident precursors can reveal assurance weaknesses before a major event. Review what they say about:

- ownership;
- monitoring;
- delegated authority;
- provider visibility;
- evidence preservation;
- containment;
- escalation;
- human oversight;
- shared dependencies.

Do not erase failed results after the final test passes. The failure/remediation history may explain why the control changed and whether similar systems remain exposed. Learning should feed both the local system and the relevant enterprise control owner where the issue is shared.

79. Review Related Systems and Shared Dependencies

When a material issue is discovered, ask whether the same dependency or assumption exists elsewhere. Examples include a vulnerable model package, shared tool server, over-broad OAuth grant pattern, provider outage, poisoned knowledge source, weak agent identity pattern or missing change notification. The system owner should report shared-pattern risk to the CISO, platform owner or relevant enterprise function. COM-019 does not require the system owner to assess the entire portfolio. It requires the owner not to treat a clearly shared weakness as purely local.

80. Improve Without Inventing a Maturity Score

Improvement should be based on observable operating problems and decisions, not on an invented maturity number. Useful improvement questions include:

- Are ownership gaps decreasing?
- Are material changes becoming visible earlier?
- Can incidents be reconstructed faster?
- Are provider unknowns better understood?
- Are exceptions being resolved rather than rolled forward?
- Are repeated failures being addressed systemically?
- Are external claims becoming better bounded?

These are management questions, not a scoring model. COM-019 deliberately avoids a composite system assurance score because one number can hide important differences between a missing owner, provider opacity, failed control and stale evidence.

PART IX — MAKE AND DEFEND SYSTEM-LEVEL DECISIONS

Use this Part when: a consequential operating, risk, escalation or external-assurance decision is required.

Working output: a bounded decision record containing scope, evidence, uncertainty, options, authority and follow-up.

Part checkpoint: Can the decision-maker see what is known, what is uncertain, what options exist and who actually holds authority?

The checkpoint is an informative management prompt, not a readiness gate, assessment state or certification prerequisite.

81. Build the System-Level Decision Basis

A system-level decision should be based on an explicit, bounded record rather than intuition alone. A practical decision basis includes:

- system/use-case and current scope;
- current model/provider/configuration basis;
- material controls and dependencies;
- current evidence and its limitations;
- recent material changes;
- active incidents/exceptions/remediation;
- relevant consequence;
- unresolved unknowns;
- available operating options;
- applicable decision authority.

The purpose is not to eliminate judgement. It is to make the judgement traceable. COM-019 uses Continue / Constrain / Degrade or Fall Back / Suspend / Retire as illustrative system-level management options consistent with the broader practitioner suite. They are not mandatory decision gates, a formal ODA3 status taxonomy, assessment result, readiness state or certification state. Different authorities may need to decide different aspects of the same situation. Where enterprise policy assigns the decision to the CISO, business executive, risk authority, legal/privacy function or safety authority, the system owner supplies the bounded decision basis and recommendation rather than absorbing that authority.

81.1 Quick Decision Frame

For a fast system-level decision discussion, capture eight items:

1. Scope — which system/use case/version is being decided?

2. Trigger — what changed, failed or became uncertain?

3. Evidence — what directly supports the current position?

4. Unknown — what is provider-represented, inaccessible, inferred or contradictory?

5. Consequence — what happens if the current operation continues?

6. Options — continue, constrain, degrade/fall back, suspend, retire, or another locally governed option?

7. Authority — who actually decides?

8. Record — where is the rationale, condition and follow-up preserved?

This is a meeting/decision aid only. It is not a formal decision gate or ODA3 approval sequence.

82. Decision Option — Continue

Continue means the applicable organizational authority chooses to maintain current operation for the bounded context. Before recommending continuation, the system owner should be able to explain why the available operating basis remains supportable, what material limitations exist and what monitoring/change conditions could alter the decision. Continuation does not mean "risk free," "certified" or "assured forever." Where unresolved conditions remain, record them rather than implying a clean state. This is an illustrative management option, not an ODA3 operating-state taxonomy, assessment result, readiness state, certification category or universal instruction. The applicable organizational authority makes the decision.

Quick use

- Use when: the available operating basis remains supportable for the bounded context.
- First question: What fact or evidence would most likely change the continuation decision?
- Preserve: limitations, monitoring conditions, next review/change trigger and decision authority.

83. Decision Option — Constrain

Constrain means reducing the system's scope, authority or exposure while retaining some operation. Examples can include reducing user population, disabling a tool, narrowing destinations, removing high-impact actions, limiting data sources, adding approval, lowering transaction limits or restricting a use case. The system owner should identify what risk/uncertainty the constraint addresses and what residual operation remains. A constraint may be temporary or become the new design. Do not treat constraint as a universal substitute for remediation; it is one management response to a bounded condition and remains subject to the applicable decision authority. This is an illustrative management option, not an ODA3 operating-state taxonomy, assessment result, readiness state, certification category or universal instruction. The applicable organizational authority makes the decision.

Quick use

- Use when: the organization can reduce scope, authority or exposure while keeping some useful operation.
- First question: Which authority or exposure is being reduced, and what residual operation remains?
- Preserve: constraint, rationale, owner, residual limitation and condition for reconsideration.

84. Decision Option — Degrade or Fall Back

Degrade or fall back means operating through a reduced-function or alternate path. Examples can include a lower-authority mode, alternate provider/model, manual review, read-only mode or conventional non-AI process. The owner should understand the new failure modes introduced by the fallback, not assume that lower functionality is automatically safer. Record who can invoke the fallback, data/state implications, user/process impacts and the conditions for returning to normal operation. For physical AI, fallback decisions may require safety/operational authority. COM-019 does not determine that a degraded or fallback mode is safe. This is an illustrative management option, not an ODA3 operating-state taxonomy, assessment result, readiness state, certification category or universal instruction. The applicable organizational authority makes the decision.

Quick use

- Use when: an alternate or reduced-function path may preserve business continuity.
- First question: What new failure mode or authority change does the fallback introduce?
- Preserve: fallback configuration, invocation authority, state/data implications and path back to normal operation.

85. Decision Option — Suspend

Suspend means stopping the affected capability while preserving evidence, recovery options and business continuity as appropriate. Suspension may be appropriate when the system basis is materially uncertain, an incident is active, containment is incomplete, provider behavior is unknown, or continued

operation exceeds authorized risk. The owner should know what "suspend" technically means. Disabling a UI may be insufficient if background agents, API calls or credentials remain active. This is an illustrative management option, not an ODA3 operating-state taxonomy, assessment result, readiness state, certification category or universal instruction. The applicable organizational authority makes the decision. Suspension is not a system-owner unilateral entitlement.

Quick use

- Use when: continued operation is no longer supportable for the affected capability pending further action.
- First question: What exactly needs to stop, and what can continue without preserving the same exposure?
- Preserve: technical suspension surface, preserved evidence/state, business impact, authority and recovery dependency.

86. Decision Option — Retire

Retire means removing the capability from service when continued operation no longer has a supportable business/control basis or a strategic decision has been made to end it. Retirement should include decommissioning of identities, integrations, tools, data/indexes, provider entitlements and scheduled processes, not merely product withdrawal. Preserve historical evidence and decision records where retention is required by applicable legal, contractual, organizational or scheme requirements, and otherwise according to the organization's governed retention rules. A system owner may recommend retirement; final authority may sit with business, executive, regulatory or safety functions depending on context. This is an illustrative management option, not an ODA3 operating-state taxonomy, assessment result, readiness state, certification category or universal instruction. The applicable organizational authority makes the decision. Retirement in particular is not a certification, maturity, readiness or failure category.

Quick use

- Use when: the system or capability no longer has a supportable operating or strategic basis.
- First question: What residual identities, integrations, data, provider entitlements or obligations survive retirement?
- Preserve: decommissioning actions, retained records, residual obligations, owner and effective retirement date.

87. Risk Acceptance and Escalation

When the system owner cannot resolve a material condition within their authority, escalate it to the role that can decide. A useful escalation states:

- the condition;
- affected scope;
- what is known;
- what remains unknown;
- available evidence;
- current containment/constraint;
- available options;
- owner recommendation;
- decision required;
- time/consequence context where relevant.

Risk acceptance should be attributable to a defined authority. Avoid vague phrases such as "business accepted" without naming the role/process. See §30 for what should be identified in advance if the relevant decision authority is not reachable when needed. COM-019 does not create a universal risk-acceptance threshold. See §93 (Ownership and Decision-Rights Matrix) for a management aid that separates operation from authority across material topics.

88. Respond to Audit, Customer or Regulator Assurance Requests

External assurance requests often arrive as broad questions: "Is this system compliant?", "Is the model secure?", "Can you certify it?", "Has this been audited?" Translate the request into bounded propositions before answering. Ask:

- Which system/use case?
- Which period/version?
- Which requirement or claim?
- What evidence supports the answer?
- What remains provider-represented or unknown?
- Does a separately valid assessment or certification record actually exist from an applicable authorized process, and does it

cover this scope and period?

- Who is authorized to make the statement?

The system owner supplies system facts. Legal, regulatory, audit or certification conclusions remain with the appropriate authority. If a future GAISSF assessment is undertaken under an applicable version-specific approval and authorized process, that process follows the separately governed and authorized route applicable to the engagement. COM-019 does not announce current live assessment authority, certification availability, assessor authorization or an assessor workflow.

89. Report to the CISO, Executive Management and the Board

System-level reporting should give enterprise leaders enough information to understand the decision, not overwhelm them with architecture detail. A concise report can cover:

1. system purpose and consequence;
2. current model/provider/dependency basis;
3. material changes;
4. material control/evidence gaps;
5. incidents or near misses;
6. provider/shared dependencies;
7. unresolved exceptions/remediation;
8. decision or escalation required.

COM-018 owns the CISO portfolio and board/risk synthesis. COM-019 supplies a bounded, evidence-referenced system- level input to that view; it does not create a parallel system-owner dashboard, independent assurance opinion or board- assurance process. When information reaches executive or board oversight, preserve the COM-008 discipline: ask for the evidence, not the policy. The owner should distinguish direct observation from provider representation and inference.

90. Report Uncertainty Explicitly

Uncertainty should be reported as information, not treated as embarrassment. Useful management labels include:

- Known / directly supported
- Provider-represented
- Inferred
- Unknown / inaccessible
- Contradicted

- Specifically checked but not observed

See §47 for the distinction between Unknown and Checked and not observed that underlies these labels. These are local management communication aids. They are not ODA3 confidence levels, assessment states, control ratings, readiness categories, UAIF classifications or replacements for the incident fact/hypothesis/provider/unknown distinctions used in COM-012/COM-013. Always state the scope and basis. "No unauthorized tool use observed in the monitored logs for this period" is stronger than "No unauthorized tool use occurred." The quality of an assurance position often depends as much on honest uncertainty as on positive evidence. See §98 (Claims and External-Statement Checklist) for the pre-release check on external statements, and §91 (System Owner Accountability Card) for the underlying system record.

PART X — PRACTITIONER TOOLS, BOUNDARIES AND PUBLICATION CONTROLS

Use this Part when: the team needs a lightweight record to make accountability, boundary, change, provider reliance, evidence or

incident readiness easier to manage.

Working output: the minimum useful record in the organization's existing system of record or a linked COM-019 management aid.

Part checkpoint: Does the tool improve traceability without being mistaken for a readiness state, assessor workpaper, certification

input or independent assurance conclusion? The checkpoint is an informative management prompt, not a readiness gate, assessment state or certification prerequisite. Tool-family boundary: Sections 91–98 are optional public management aids. They are not mandatory GAISSF artifacts, controlled assessor workpapers, assessment-readiness instruments, certification inputs, formal ODA3 status records or substitutes for authoritative enterprise systems of record. Completing them does not establish readiness, evidence sufficiency, conformity, approval, independent assurance or certification eligibility.

How the tools relate: the eight tools broadly track the guide's own progression — accountability and boundary (§91–92), ownership and decision rights (§93), change and provider dependencies (§94–95), evidence handoff (§96), incident readiness (§97), and external claims (§98). They remain independent, modular records; use only the ones relevant to the current need, in any order, rather than treating this as a required sequence.

91. Practitioner Tool — System Owner Accountability Card

The System Owner Accountability Card is a one-page management aid for making accountability concrete.

Quick use

Best used when: assigning a new owner, transferring ownership, preparing an executive review, or making the system easy to identify during an incident.

Minimum usable version: system/service, business purpose, accountable owner, current model/provider, material consequence, current material gap, incident route and evidence location. Do not wait for every field to be perfect before recording known gaps. An explicit Unknown is more useful than an invented answer.

Field	What to record
System / service	Canonical name and inventory reference
Business purpose	Intended use and material exclusions
Accountable owner	Named role / function
Technical service owner	Operational engineering owner
Primary model/provider	Current service basis
Material consequence	Business / security / physical consequence summary
Material dependencies	Shared platforms, data, tools, providers
Key decision authorities	Security, risk, privacy/legal, safety where relevant
Last material change	Date/reference and change summary

Field	What to record
Current material gap	Most important unresolved condition / unknown
Incident route	SOC/CSIRT/provider escalation reference
Evidence location	Link/reference to operational records

Use: onboarding, ownership transfer, executive review or incident readiness.

Completion is not an ODA3 readiness designation, assessment result, approval or legal accountable-person designation.

92. Practitioner Tool — System Boundary Record

The System Boundary Record captures useful boundary elements to consider when making system-level claims.

Quick use

Best used when: teams use the same system name but mean different technical or business boundaries, or when an external claim needs a precise scope.

Minimum usable version: business use case, current model/provider/routing, data/RAG sources, tools/connectors, identities, material dependencies, physical path if applicable, and explicit exclusions. Prefer links to authoritative architecture and inventories over copying diagrams into another repository.

Boundary element	What to record
Business process / use case	What the AI supports or changes
System / service boundary	Current system/service components and explicit exclusions
Model / provider / routing	Current model, provider, route and fallback where relevant
Data / RAG sources	Material data, retrieval sources and authorization boundaries
Tools / connectors / identities	Material execution paths, connectors and identities
Dependencies / consequence path	Material dependencies and physical-effect path where applicable

The record should point to authoritative architecture/inventory sources rather than duplicate them unnecessarily.

A boundary record does not determine assessment scope unless the applicable controlled process adopts it.

93. Practitioner Tool — Ownership and Decision-Rights Matrix

The Ownership and Decision-Rights Matrix distinguishes operation from authority.

Quick use

Best used when: ownership is diffuse, escalation is slow, or the organization repeatedly asks "who decides this?"

Minimum usable version: accountable system owner plus named operating/evidence/decision authority for model/provider, identity/tool authority, monitoring, incident containment, external claims and physical return to service where applicable.

The empty cell is itself useful information when it exposes an unresolved ownership gap. Where a topic genuinely does not apply to the system — for example, "Physical return to service" for a system with no physical-effect path — mark the row explicitly as Not Applicable rather than leaving it blank.

A blank cell and an intentional N/A carry different meanings: a blank cell may reflect an unresolved ownership gap, while an explicit N/A reflects a considered determination that the topic does not arise for this system. Use N/A only where the system boundary provides a supportable basis for non-applicability; uncertainty about whether a topic applies should remain visible as uncertainty rather than being converted to N/A.

Topic	Control/operating owner	Evidence owner	Decision authority	Escalation
System purpose/use				
Model/provider				

Topic	Control/operating owner	Evidence owner	Decision authority	Escalation
Identity/access				
Agent/tool authority				
Data/RAG				
Monitoring				
Security exception				
Provider issue				
Incident containment				
External assurance statement				
Physical return to service				

Do not force one person into every column. The value of the tool is the separation.

It is a management record, not a normative RACI or assessor workpaper.

94. Practitioner Tool — Material-Change Review Record

Use the Material-Change Review Record when a change may invalidate earlier assumptions.

Quick use

Best used when: a model, provider, prompt, orchestration path, tool, permission, RAG source, user population, autonomy level or business purpose changes materially.

Minimum usable version: change, before/after state, affected assumptions/controls/evidence, unresolved unknowns, decision authority and outcome/follow-up. For a small change, one concise record may be enough; consequence and uncertainty should drive depth rather than form length. Record:

- change identifier/date;
- system/version before and after;
- reason for change;
- affected components/providers;
- purpose/scope impact;
- control/permission impact;
- evidence impact;
- incident/containment impact;
- provider/shared-dependency impact;
- physical/safety impact where relevant;
- tests/verification performed;
- unresolved unknowns;
- decision authority;
- outcome/constraints;
- evidence references;
- follow-up action.

A simple review can be short. High-consequence change may require specialist review. This record does not create a universal change-approval gate.

95. Practitioner Tool — Provider / Shared-Responsibility Record

The Provider / Shared-Responsibility Record keeps contractual responsibility, operating control and evidence visibility separate.

Quick use

Best used when: the system depends materially on hosted models, external tools, managed AI services, shared enterprise platforms or inaccessible provider controls.

Minimum usable version: provider/service, enterprise-controlled areas, provider-controlled areas, evidence-source state, change-notification route, incident route, fallback/exit and open limitation. This record is especially useful when contract wording and actual operational visibility differ.

Question	Record
Service/provider	Exact service and relevant region/version
Enterprise-controlled areas	Mechanisms operated by the enterprise
Provider-controlled areas	Mechanisms operated by provider
Shared interfaces	Where both parties contribute
Visibility-limited areas	Material controls with restricted visibility
Evidence source	Customer-observed / Provider-provided / Shared-interface / Provider-only-inaccessible / Unknown
Change notification	How provider change is learned
Incident route	Security/support escalation
Subdependencies	Material subprocessors/shared services
Fallback/exit	Migration, alternate route or suspension path
Open limitation	Unresolved provider assumption or evidence gap

The tool is not a provider rating and does not transfer provider conformity to the enterprise system.

96. Practitioner Tool — Operational-Evidence Handoff Checklist

Use the Operational-Evidence Handoff Checklist before records move into COM-015 evidence preparation.

Quick use

Best used when: operational records are about to be organized under COM-015, or when a team cannot tell whether an artifact still matches the current system state.

Minimum usable version: system/boundary, control or claim, period/version, source/location, owner, provenance/integrity context, negative/contradictory evidence, material changes and known limitations. Do not improve the handoff by deleting unfavorable history. Confirm that the handoff identifies:

- system/use-case and boundary;
- relevant control/claim;
- period and version/state;
- evidence source/location;
- owner;
- provider evidence-source state where relevant;
- integrity/provenance context;
- failed/negative tests;
- contradictions;
- unknown/inaccessible evidence;
- material changes during the period;

- sensitive-data handling constraints;
- related incident/remediation references;
- known limitations.

The checklist improves handoff organization; it does not decide evidence sufficiency, confidence, admissibility, assessment readiness, control result or certification eligibility.

97. Practitioner Tool — Incident-Readiness Handoff Checklist

Use the Incident-Readiness Handoff Checklist to confirm that responders can engage the system.

Quick use

Best used when: onboarding a system into SOC/CSIRT coverage, running an exercise, changing a provider/agent authority path, or preparing for a high-consequence deployment.

Minimum usable version: current system identity, reachable owner, evidence/log location, containment/revocation surfaces, provider route, fallback, and safety authority where relevant. The practical test is whether responders can find the owner, the records and the containment surface — not whether the checklist looks complete.

Readiness question	Reference / owner
Canonical system identity and current model/provider known?	
Current owner / technical owner reachable?	
Relevant logs/records accessible?	
Agent/tool identities and revocation path known?	
Containment surfaces identified?	
Provider security/support route available?	
Business continuity / fallback known?	
Physical/safety authority identified if applicable?	
Evidence preservation constraints known?	
UAIF / AI-IRF handoff route known?	

A completed checklist does not create an incident-ready ODA3 status and does not mean an incident will be easy to investigate or contain. It makes known prerequisites visible before an event.

98. Practitioner Tool — Claims and External-Statement Checklist

Use the Claims and External-Statement Checklist before a material assurance statement leaves the organization. This checklist operationalizes §29 (who may speak) and §50 (what should not be said) into a pre-release check for a specific statement. It does not establish new claim-content rules beyond §50 or new authority rules beyond §29.

Quick use

Best used when: answering a customer questionnaire, audit request, board question, regulator inquiry, sales assurance request or public statement about the system.

Minimum usable version: exact scope, period/version, factual claim, evidence basis, material limitation, relevant external assessment/certification record if any, and authorized speaker/signatory. Use the checklist to narrow unsupported language before it becomes an external commitment. Ask:

- What exact system/use case does the statement cover?
- What period/version does it cover?
- Is the claim factual, inferred, provider-represented or independently assessed?
- What evidence supports it?
- What material limitations or unknowns remain?

- Are we using "secure," "compliant," "certified," "approved," "endorsed" or equivalent language?
- Does a separately valid authority support that wording?
- Does a provider certificate/report actually cover this end-to-end claim?
- Are legal/regulatory interpretations owned by the proper function?
- Is the signer/speaker authorized to make the statement?

If a broad claim cannot be supported by the available evidence and proper authority, narrow it rather than polishing it. The checklist itself supplies no authority to make a reserved ODA3, legal, regulatory, audit or certification claim.

99. Notably Absent, Source Traceability and Next Reading

COM-019 is intentionally bounded. Notably Absent:

- no new GAISSF requirement or system-owner conformity profile;
- no public maturity/readiness score or system-owner assurance status;
- no composite system assurance score;
- no independent assurance opinion created by a COM-019 record, decision or checklist;
- no universal risk/severity threshold;
- no mandatory review cadence or remediation SLA;
- no evidence-sufficiency/confidence scoring;
- no assessor sampling, control rating, finding severity or result logic;
- no certification eligibility or availability;
- no live assessment authority or certificate-issuance authority;
- no assessor authorization;
- no partner/empanelment status;
- no regulator recognition or legal-compliance conclusion;
- no functional-safety or product/sector approval;
- no physical return-to-service authority;
- no guarantee of future security or absence of incidents.

99.1 Core Source Register

Historical-Source Caution (relocated from front matter): Some historical GAISSF and AI-IRF technical files contain older naming, certification-state language, numeric thresholds, service-level targets or other scheme-development constructs. COM-019 uses those files only for technical elements that remain controlled/current and does not import historical commercial, certification, readiness, recognition or universal-threshold claims into COM-019 merely because they appear in a preserved source.

Source	Role carried into COM-019
GAISSF-NOR-001 — Framework Standard	Primary GAISSF framework authority; 59-control / nine-domain architecture and scope/applicability basis.
GAISSF-NOR-004 — Control Catalogue	Canonical control catalogue; 52 D1–D8 Foundational controls plus seven D9 physical-AI controls.
ODA3-2026-06-TCR-STD-002 — UAIF™ v1.0 Technical Specification	UAIF architecture and Final Publication v1.0 maturity boundary.
TCR-STD-003 / current UAIF schema package	Published machine-readable UAIF companion; implementation should confirm current release/package metadata.
ODA3-2026-06-TCR-HAI-001 — corrected AI-IRF technical source	79-control / 19-domain technical architecture; historical scheme language is not automatically public-current.
ODA3-2026-06-TCR-HAI-003	Published AI-IRF machine-readable control/playbook encoding; optional AIIS v1.0.0 / conformance_test is reserved for future population.

Source	Role carried into COM-019
PAI-SF™ canonical sources	12-domain / 41-control physical-AI framework and security/safety/sector handoff boundary.
GEL v1.0 — GAISSF Ecosystem License	Current licensing, marks, factual-alignment and reserved certification/audit authority boundary.

Historical source titles remain identifiable as historical source titles and are not silently rewritten into current market-facing terminology.

99.2 Current Practitioner Handoffs

1. The AI Assurance Imperative: A Board Member's Guide to GAISSF™ — ODA3-2026-08-WHP-COM-008 — board oversight questions; COM-019 contributes system-level facts and evidence rather than board criteria.

2. When AI Breaks: The ODA3 Guide to AI Incident Response — ODA3-2026-08-WHP-COM-011 — short operational orientation for potentially material AI events.

3. Classifying AI Incidents: A Practical Guide to UAIF™ — ODA3-2026-08-WHP-COM-012 — UAIF™ classification and uncertainty discipline.

4. AI Incident Response Field Guide: Using UAIF™ and AI-IRF™ for Real-World AI Incidents — ODA3-2026-08-WHP-COM-013 — deeper classification/response integration.

5. Securing Physical AI: A Practitioner's Guide to PAI-SF™ — ODA3-2026-08-WHP-COM-014 — detailed physical-AI security-assurance and safety interface.

6. Building an AI Assurance Evidence Pack — ODA3-2026-08-WHP-COM-015 — FINAL — evidence preparation, evidence-source states, provenance, limitations and handoff.

7. Implementing GAISSF™ — ODA3-2026-08-WHP-COM-017 — FINAL — detailed implementation and implementation-side verification.

8. The CISO's Guide to AI Security Assurance — ODA3-2026-08-WHP-COM-018 — CORRECTED FINAL (R10) — enterprise CISO portfolio, challenge, metrics and board/risk synthesis.

The implementation-to-assessment relationship remains: scope / applicability → implementation → evidence generation → COM-015 evidence preparation → separately governed assurance/assessment activity, where applicable and independently authorized. This is a functional relationship, not a mandatory chronological workflow.

99.3 Quick Continuation Route

For most system owners, the practical next document depends on the immediate problem:

- Need to implement or change a control? → COM-017.
- Need to prepare operational records into evidence? → COM-015.
- Need to address a later assessment or independent-assurance request? → follow the separately governed and authorized route applicable to the engagement.
- Need enterprise CISO escalation or portfolio treatment? → COM-018.
- Need incident classification/response? → COM-011 / COM-012 / COM-013.
- Need detailed physical-AI treatment? → COM-014 / PAI-SF™.

99.4 Machine-Readable Companions

Machine-Readable Terminology Rule (relocated from front matter): Published schema = describe it as published. Future interoperability = describe integrations/tooling as development. Future work may include product field mappings, SIEM/SOAR/GRC integrations, connectors, SDKs, adapters, additional profiles, reference implementations or validation/conformance tooling. COM-019 does not describe TCR-STD-003 or AIIS v1.0.0 as though their core schemas are still awaiting publication.

The current machine-readable companions relevant to COM-019 include:

- TCR-STD-003 / the published UAIF machine-readable schema package; and
- AIIS v1.0.0, the published AI-IRF machine-readable control/playbook encoding.

Future interoperability work may include field mappings, SIEM/SOAR/GRC integrations, connectors, SDKs, adapters, additional profiles, reference integrations or validation/conformance tooling. Such future work does not imply that the core machine-readable schemas are awaiting publication.

100. Publication Metadata, GEL Attribution and Final Publication Statement

100.1 Publication Metadata

- Document ID: ODA3-2026-08-WHP-COM-019
- Title: The AI System Owner's Guide to Operational Assurance
- Subtitle: A Practitioner's Guide to Accountability, Evidence, Change and Operational Decisions for AI Systems
- Document Type: WHP — Adoption / Practitioner Role Guide
- Framework Context: GAISSF Ecosystem / Operational AI Assurance
- Classification: Public — GEL v1.0
- Publisher: ODA3 Institute
- Original publication date: 14 August 2026
- Corrected-edition publication date: 20 August 2026
- Revision state: CORRECTED FINAL (R10)
- Copyright: © 2026 ODA3 Pvt Ltd

Suggested citation: ODA3 Institute (2026), The AI System Owner's Guide to Operational Assurance: A Practitioner's Guide to Accountability, Evidence, Change and Operational Decisions for AI Systems, ODA3-2026-08-WHP-COM-019. No DOI, ISBN, accreditation number, certification number or regulator-recognition identifier is assigned by this guide.

100.2 Naming and Marks

Current public practitioner usage in COM-019 is:

- GAISSF™ — Global AI Safety and Security Framework
- UAIF™ — Unified AI Incident Framework
- AI-IRF™ — AI Incident Response Framework
- PAI-SF™ — Physical AI Security Framework
- GAISSF Ecosystem — umbrella term for the integrated ODA3 framework ecosystem

Historical source titles may preserve older naming or mark treatment for source identification only.

100.3 GEL Attribution and Use Boundary

GAISSF™, UAIF™, AI-IRF™, and PAI-SF™ are frameworks of ODA3 Institute and are referenced under the GAISSF Ecosystem License (GEL) v1.0. GEL v1.0 is the governing licence for use of the GAISSF Ecosystem. Public availability and Internal Use do not create unrestricted third-party Commercial Use rights. External professional delivery—including consulting, implementation, integration, assessment, audit, paid training, managed services, software/SaaS embedding, certification or other client delivery—remains subject to the applicable GEL commercial rights and any separately applicable ODA3 programme authorization. Factual alignment or interoperability statements, where accurate and otherwise permitted, do not themselves imply ODA3 certification, endorsement, audit, approval, partner status or official authorization.

100.4 Final Publication Statement

This document is the CORRECTED FINAL (R10) edition of ODA3-2026-08-WHP-COM-019 — The AI System Owner's Guide to Operational Assurance.

R10 reconciles current ODA3 publication states and practitioner handoffs and incorporates bounded practitioner-usability improvements to accountability, authority, evidence, change, decision and operational-record guidance. These changes do not create new GAISSF™ requirements, assessment methodology, certification criteria, readiness states or organizational authority.

CORRECTED FINAL (R10) status applies to this practitioner role guide only. It does not create or alter the maturity, authority, availability, recognition or approval state of any underlying framework, assessment methodology, certification programme, partner programme or external authority.

The publication state or completion of this guide does not create or alter:

- framework maturity;
- live assessment authority;
- certification availability;
- certificate-issuance authority;
- assessor authorization;
- partner or empanelment status;
- regulator recognition;
- legal-compliance status;
- product or sector approval;
- functional-safety approval;
- physical return-to-service authority; or
- commercial-delivery rights.