

Global AI Compliance White Paper 2026

From Model Governance to Agentic Lifecycle Conformance

Jearon Wong · Protocol Architect for the Agent Era

Intent	Authority	Agent Work	Evidence	Closure

DOCUMENT ID GACWP-2026-v0.3.2-FRC-R3	VERSION v0.3.2 Public Research Edition	DATE May 2026
STATUS Public Research Edition	SCORING MODEL RCCS-T / MRO / RCCS-M / ALCS	VISUAL SOURCE OF TRUTH HTML/PDF primary

Non-legal technical governance analysis. Not legal advice, not a certification standard, not regulatory approval, not a procurement recommendation, not a vendor ranking, and not a final vendor assessment.

IMPORTANT NOTICE

Publication Boundary

This publication is a technical governance analysis of AI Agent Lifecycle Governance. It is not legal advice, a certification standard, a conformity assessment, a regulatory approval path, an audit opinion, an insurance guarantee, a procurement recommendation, a vendor ranking, or a final vendor assessment.

RCCS-T, RCCS-M, and ALCS scores and system mappings are analytical and source-qualified. RCCS-M is an author analytical, forward-looking, MRO-adjusted model; it is not current law, certification, legal compliance proof, or a regulator-approved benchmark. Appendix G remains revalidated provisional, qualitative, non-ranking, and no-score.

TECHNICAL REPORT IDENTITY

Citation and Artifact Identity

Jearon Wong. Global AI Compliance White Paper 2026: From Model Governance to Agentic Lifecycle Conformance. Technical Report GACWP-2026-v0.3.2-FRC-R3, May 2026.

DOCUMENT ID / TRACE TAG GACWP-2026-v0.3.2-FRC-R3	VERSION v0.3.2 Public Research Edition
AUTHOR Jearon Wong	PUBLICATION MONTH May 2026
CANONICAL HUB https://www.jearonwong.com/research/global-ai-compliance-white-paper-2026/	HTML FULL TEXT https://www.jearonwong.com/research/global-ai-compliance-white-paper-2026/global-ai-compliance-white-paper-2026.html
PDF https://www.jearonwong.com/research/global-ai-compliance-white-paper-2026/global-ai-compliance-white-paper-2026.pdf	
AGENTIC LIFECYCLE GOVERNANCE Chapter 2	MISSING REGULATORY OBJECTS Chapter 6 overview
RCCS-T Traditional regulatory coverage dimensions	RCCS-M MRO-adjusted regulatory coverage
ALCS Agentic lifecycle conformance score	METHODOLOGY Scoring method and evidence confidence
SCORING RESULTS	SYSTEMS MAPPING

RCCS-T / RCCS-M / ALCS profiles	Detailed system mappings
EVIDENCE VALIDATION Evidence-based validation pattern	FAILURE SCENARIOS Enterprise failure scenarios
DETERMINISTIC DELIVERY Adoption and delivery control context	BOUNDARY / NON-CLAIM DISCIPLINE Publication scope and boundary statement
MR0-01 Human Role to MAS Responsibility Mapping	MR0-02 Delegated Authority Boundary
MR0-03 Agent Role is not Human Role	MR0-04 Accepted Outcome Compliance
MR0-05 Tool-Action Liability Boundary	MR0-06 Responsibility Transfer Across Agents
MR0-07 Authority Drift	MR0-08 MAS Evidence Partitioning
MR0-09 Cross-Project Reuse Compliance	MR0-10 Privacy / GDPR Lifecycle Mapping
MR0-11 Privacy-Preserving Third-Party Validation	MR0-12 Evidence Minimization and Selective Disclosure
MR0-13 Data Subject Rights vs Evidence Retention	MR0-14 Third-Party Processor / Subprocessor Chain
MR0-15 Vendor / Model / Runtime Substitution Conformance	MR0-16 Incident, Dispute, and Remediation Closure

Publication Contents

HTML anchors are active in the browser artifact. PDF rendering uses the shared whitepaper A4 print profile.

Front Matter

1. Problem This White Paper Solves
2. How to Read This Paper

Part I – Foundations

1. Chapter 1: Scope, Methodology, and Non-Legal Boundary
2. Chapter 2: The Missing Layer: AI Agent Lifecycle Governance
3. Chapter 3: Why Agentic AI Breaks Model-Centric Compliance
4. Chapter 4: Regulatory and Standards Baseline
5. Chapter 5: From Regulatory Language to Engineering Objects

Part II – Lifecycle Objects and Scoring

1. 6. Missing Regulatory Objects for Agentic and MAS Compliance
2. Chapter 7: Regulatory Compliance Coverage Score (RCCS)
3. Chapter 8: Agentic Lifecycle Conformance Score (ALCS)
4. Chapter 9: Composite Scoring Method and Evidence Confidence

Part III – System Mapping and Control

1. 10. Enterprise Adoption, Procurement, and Control Crosswalk
2. 11. Comparative Field and System Positioning
3. Chapter 12: Detailed System Mappings
4. Evaluation Results: RCCS-T / RCCS-M / ALCS Provisional Analytical Profiles
5. 13. MPLP Deep Mapping, Boundary, and Non-Claim Discipline

Part IV – Validation and Boundary

1. 14. Evidence-Based Validation Pattern for AI Agent Lifecycle Governance
2. 15. Enterprise Failure Scenarios
3. 16. Boundary to Companion Papers: Evidence, Assurance, and Insurability

Part V – Adoption and Conclusion

1. 17. Adoption Roadmap for AI Agent Lifecycle Governance
2. 18. Conclusion

Appendices

1. Appendix A — Machine-Readable Object Templates
2. Appendix B — Scorecard Template
3. Appendix C — References and Source Notes
4. Appendix D — Expanded Cross-System MRO Mapping
5. Appendix E — RCCS-T Detailed Rubrics and RCCS-M Adjustment Matrix
6. Appendix F — ALCS Detailed Scoring Rubrics
7. Appendix G — Expanded System-by-System Assessment
8. Appendix H — Enterprise Procurement Scorecard
9. Appendix I — Source Audit and Revalidation Register
10. Appendix J — v0.3.2 Gap Closure Coverage Matrix
11. Appendix K — Editorial Remediation Record
12. References / Sources

Problem This White Paper Solves

The central problem is that model governance is necessary but insufficient once AI systems act through agents, tools, handoffs, memory, projects, vendors, and accepted outcomes.

Traditional AI governance is usually organized around the model: model risk, model documentation, model monitoring, model security, and model transparency. Agentic systems shift the compliance object. The governance question is no longer only whether a model is evaluated; it is whether the lifecycle of delegated work can prove who authorized an action, what authority was granted, which agent or tool acted, what evidence was preserved, who accepted the outcome, and how disputes or remediation close.

This white paper frames that missing layer as AI Agent Lifecycle Governance. Enterprise failures emerge when lifecycle responsibility objects are absent: delegated authority boundaries are implicit, human oversight is not mapped to machine roles, evidence chains cannot reconstruct agent work, privacy rights conflict with retention, and vendor or runtime substitution changes governance semantics without conformance review.

The paper responds with sixteen Missing Regulatory Objects, the RCCS-T/RCCS-M/ALCS analytical model, evidence-based validation patterns, illustrative enterprise failure scenarios, and a staged adoption roadmap. These tools support structured posture review and audit-readiness analysis. They do not create legal advice, certification, regulatory approval, procurement recommendations, vendor rankings, or final vendor assessments.

Problem

Model-centric governance does not fully express lifecycle responsibility in agentic workflows.

Object Layer

MROs define the missing responsibility, authority, evidence, privacy, substitution, and closure objects.

Assessment

RCCS-T/RCCS-M/ALCS structure analytical posture review without becoming legal compliance scores.

Assurance Pattern

Evidence-based validation supports reviewable assurance while preserving non-certifying boundaries.

Adoption

The roadmap moves enterprises from inventory to authority, responsibility, evidence, privacy, validation, remediation, and scale.

How to Read This Paper

Executives

Read Chapters 1-3, 15, 17, and 18 for the business risk, failure patterns, adoption path, and conclusion.

Compliance / Legal Teams

Read Chapters 4-10 and Appendices C, I, J, and K for regulatory mapping, scoring boundaries, source posture, and governance records.

Product / Platform Teams

Read Chapters 5-9, 12, and Appendices A, B, E, and F for lifecycle objects, scoring rubrics, and implementation-facing templates.

Auditors / Assurance Reviewers

Read Chapters 10, 14, 15, and Appendices C, D, G, and I for evidence patterns, validation boundaries, and source-qualified system mappings.

Protocol / Runtime Architects

Read Chapters 6, 8, 13, 14, and Appendix A for MRO semantics, ALCS mapping, protocol fit, and machine-readable object templates.

From Model Governance to Agentic Lifecycle Conformance

Document ID: GACWP-2026-v0.3.2-FRC-R3

Version: v0.3.2 Public Research Edition

Trace Tag: GACWP-2026-v0.3.2-FRC-R3

Publication Note: Public Research Edition - non-legal technical governance analysis; not legal advice

Compiled: May 2026

Source Snapshot: Sources available and checked as of May 2026

Scope Snapshot: Public-source technical governance analysis based on sources available and checked as of May 2026

Author: Jearon Wong

Role: Protocol Architect for the Agent Era

Protocol Disclosure: Creator of MPLP — Multi-Agent Lifecycle Protocol

Site: JearonWong.com

Core Thesis

AI Agent / MAS compliance is not only model compliance. It is lifecycle responsibility compliance.

Document Control and Positioning

Table T-FM-01: Document Control and Positioning

Field	Value
Document ID	GACWP-2026-v0.3.2-FRC-R3
Version	v0.3.2 Public Research Edition
Status	Publication Draft — agenda-setting technical governance white paper; not legal advice
Author	Jearon Wong
Author role	Protocol Architect for the Agent Era
Author disclosure	Creator of MPLP — Multi-Agent Lifecycle Protocol
Site	JearonWong.com
Primary concept	AI Agent Lifecycle Governance
Primary transition	From model governance to agentic lifecycle conformance
Primary thesis	AI Agent / MAS compliance is not only model compliance. It is lifecycle responsibility compliance.
Protocol boundary	MPLP is discussed as a lifecycle protocol path, not as a certification program, legal authority, cloud platform, or exclusive compliance method.
Validation boundary	Validation Lab is referenced as a non-certifying evidence adjudication example, not as a regulator, conformity assessment body, certification authority, market endorsement program, or legal compliance certification mechanism.
Scoring boundary	RCCS-T / RCCS-M / ALCS is a three-profile analytical model, not legal compliance proof, certification, a regulator-approved benchmark, a vendor ranking, or procurement advice.
System mapping boundary	System mappings are responsibility-semantics comparisons, not product endorsements or procurement recommendations.

Publication Scope and Boundary Statement

This white paper is a **non-legal technical governance analysis**, not legal advice. It translates regulatory and standards language into engineering requirements for agentic and multi-agent systems.

Purpose

This research edition defines a missing semantic and engineering layer for AI agent and multi-agent system compliance. It is deliberately more detailed than an executive brief: the paper introduces a terminology layer, an

object model, a three-profile analytical model, system-specific mappings, comparative matrices, and implementation templates.

What This Paper Does

The paper does not claim that current AI regulation is absent or weak. The claim is narrower and more technical: current regulation and governance frameworks increasingly require human oversight, logging, documentation, monitoring, accountability, transparency, and privacy controls, but they do not yet fully specify the lifecycle objects required to prove those properties inside dynamic agentic and multi-agent execution.

The report therefore separates three questions that are often confused:

1. What does existing regulation or governance guidance already require?
2. What must engineering systems represent to make those requirements testable in agentic workflows?
3. Which current frameworks, platforms, and protocols have primitives that can be mapped to those requirements, and where do their mappings stop?

Scope Boundaries

Table T-FM-02: Publication Scope Boundaries

In Scope	Out of Scope
Agentic AI lifecycle responsibility	Legal advice for any specific jurisdiction
Multi-agent system responsibility transfer	Political content moderation regimes
Evidence, provenance, record-keeping, replay	Certification issuance or conformity assessment
Human oversight as an engineering control	Claims of regulatory approval
Privacy-preserving validation and selective disclosure	Full product security review of each vendor
Framework and platform semantic mapping	Performance benchmarking or model quality ranking

Conflict of Interest and Boundary Disclosure

Conflict of Interest

The author is the creator of **MPLP - Multi-Agent Lifecycle Protocol**. MPLP is one of the systems discussed in this paper. This creates an inherent conflict of interest. The conflict does not invalidate the analysis, but it must be disclosed clearly because this paper argues for a lifecycle protocol layer and then maps MPLP to that layer.

Boundary Controls

The controls used to limit over-claiming are as follows:

Table T-FM-03: Conflict of Interest Boundary Controls

Boundary	Required Wording	Forbidden Wording
MPLP	A lifecycle protocol path for expressing agentic responsibility objects	The only compliant standard
Validation Lab	A non-certifying evidence adjudication example	A certification authority
RCCS-T / RCCS-M / ALCS	A proposed three-profile analytical model	A regulator-approved benchmark
System mapping	A mapping of public primitives to lifecycle objects	A final legal compliance judgment

Boundary	Required Wording	Forbidden Wording
AI Agent Lifecycle Governance	A missing semantic and engineering layer	A binding regulatory standard

This boundary discipline strengthens the paper. The argument does not need to claim that MPLP is the only path. It only needs to show that agentic AI compliance requires lifecycle responsibility objects and that MPLP is one coherent way to express such objects.

Methodological Position: Necessity, Alignment, and Evaluation Boundary

RCCS-M and ALCS are proposed analytical adequacy models for agentic lifecycle governance. They are not existing law, certification, regulator-approved standards, procurement benchmarks, final vendor scores, independent third-party ratings, or legal compliance proof.

These models intentionally measure lifecycle-native governance capacity: responsibility objects, authority boundaries, evidence partitioning, accepted outcome, substitution conformance, privacy-aware validation, and remediation closure. They are not neutral legacy benchmarks for conventional model governance dashboards. They ask whether future Agent/MAS governance can represent the objects needed to make delegated work inspectable, replayable, reviewable, and accountable. The paper argues that these dimensions are structurally necessary for governing agentic and multi-agent work, because the risk-bearing unit is no longer only a model output; it is delegated lifecycle work moving through authority, action, evidence, review, acceptance, dispute, remediation, and reuse.

MPLP is expected to score strongly under RCCS-M and ALCS because it was designed around lifecycle responsibility semantics. That result should be read as **design alignment with the proposed object layer**, not as independent market validation, certification, deployment proof, or procurement superiority.

The central claim is not that "MPLP wins." The central claim is that future agentic and multi-agent compliance requires lifecycle responsibility objects. If a reader rejects the scoring model, the substantive dispute should be whether these lifecycle objects are necessary for governing agentic work, not whether the paper should hide the alignment between the proposed model and a protocol designed around that model.

Citation and Evidence Basis

This paper uses a five-level evidence hierarchy to ensure that claims are grounded in verifiable sources rather than marketing language or conceptual speculation.

Evidence Levels

Table T-FM-04: Evidence Levels

Level	Evidence Type	Use in This Paper	Confidence
L1	Binding legal, regulatory, or formal standards text	Used for baseline obligations and formal object requirements; does not prove implementation	Highest
L2	Official product documentation, API docs, developer documentation, platform architecture docs	Used for system mapping and capability interpretation	High

Level	Evidence Type	Use in This Paper	Confidence
L3	Independent audits, certifications, third-party evaluations, regulatory commentary	Used for supplemental confidence and risk interpretation	Medium
L4	Vendor white papers, product pages, blog posts, marketing statements	Used only when lower-level technical evidence is unavailable or as positioning evidence	Low
L5	Author inference, conceptual analysis, strategic interpretation	Allowed only when clearly marked as inference	Disclosed

Evidence Hierarchy Rationale

This hierarchy is necessary because the AI governance market has rapidly overloaded terms such as trace, guardrail, oversight, agent governance, evidence, approval, and evaluation. A trace in an observability product, a log in a cloud service, a review step in an SDK, and an evidence chain in a lifecycle protocol may all sound similar, but they have different compliance strength. This paper therefore treats terminology as insufficient unless it is tied to an object, a rule, an evidence source, and an enforcement or verification mechanism.

How to Cite This White Paper

Recommended Citation

Global AI Compliance White Paper 2026: From Model Governance to Agentic Lifecycle Conformance. Version v0.3.2 Public Research Edition (GACWP-2026-v0.3.2-FRC-R3). May 2026.

Short Citation

GAIC White Paper v0.3.2-FRC-R3 (May 2026)

Trace Tag for Technical References

GACWP-2026-v0.3.2-FRC-R3

Version History

Table T-FM-05: Version History

Version	Date	Trace Tag	Summary
v0.3.2-FRC-R3	May 2026	GACWP-2026-v0.3.2-FRC-R3	Public research edition integrating lifecycle governance, MROs, RCCS-T / RCCS-M / ALCS scoring, evidence validation, enterprise failure scenarios, adoption roadmap, and technical appendices.
v0.3.1-PC	May 2026	GACWP-2026-v0.3.1-PC	Publication candidate historical version. Adds ALCS-to-MRO mapping, clarifies RCCS weighting and calculation reproducibility, updates Colorado AI Act status note as of

Version	Date	Trace Tag	Summary
			2026-05-07, aligns Singapore MGF wording with official four-part framing.
v0.3-FULL	April 2026	GACWP-2026-v0.3-FULL	Full research edition baseline (historical reference only)

Revision Note for v0.3.2-FRC-R3

v0.3.2-FRC-R3 expands the white paper into a public research edition with enterprise control mapping, lifecycle scoring, evidence validation, system assessment boundaries, adoption guidance, and technical appendices.

Key improvements in R3:

- Full lifecycle governance framing for agentic and multi-agent systems
- Missing Regulatory Objects connected to RCCS-T, RCCS-M, and ALCS analysis
- Evidence validation and enterprise failure scenario coverage
- Adoption roadmap and enterprise control crosswalks
- Strengthened boundary disclosure, conflict-of-interest discipline, and non-claim language

Table of Contents

Part I — Problem and Method

1. Scope, Methodology, and Non-Legal Boundary
2. The Missing Layer: AI Agent Lifecycle Governance
3. Why Agentic AI Breaks Model-Centric Compliance
4. Regulatory and Standards Baseline
5. From Regulatory Language to Engineering Objects

Part II — Missing Regulatory Objects and Scoring

1. Missing Regulatory Objects for Agentic and MAS Compliance
2. RCCS-T / RCCS-M - Regulatory Compliance Coverage Scores
3. ALCS - Agentic Lifecycle Conformance Score
4. Composite Scoring Method

Part III — System Mapping and Lifecycle Validation

1. Enterprise Adoption, Procurement, and Control Crosswalk
2. Comparative Field and System Positioning
3. Detailed System Mappings
4. MPLP Deep Mapping, Boundary, and Non-Claim Discipline
5. Evidence-Based Validation Pattern
6. Enterprise Failure Scenarios
7. Boundary to Companion Papers

Part IV — Enterprise Adoption

1. Adoption Roadmap

2. Conclusion

Part V — Technical Appendices

1. Appendices A-K

List of Figures

- Figure F-01 — AI Agent Lifecycle Governance Stack
- Figure F-02 — Governance Stack / Missing Layer
- Figure F-03 — MRO Topology
- Figure F-04 — RCCS-T / RCCS-M / ALCS Three-Profile Scoring Model
- Figure F-05 — Lifecycle Conformance Mapping Strength Heatmap
- Figure F-06 — Enterprise Control Overlay for AI Agent Lifecycle Governance
- Figure F-07 — Model Risk in Agentic Lifecycle
- Figure F-08 — Evidence-Based Validation Pattern Flow
- Figure F-09 — Enterprise Failure Scenario Chain
- Figure F-10 — Companion Paper Boundary Map
- Figure F-11 — Stage 0-to-7 Adoption Roadmap

List of Tables

The paper contains detailed body and appendix tables. The most important table groups are:

A complete table inventory is maintained in the technical source ledger; this publication lists major table groups for reader navigation.

- Scope, boundary, document-control, and evidence-basis tables in the front matter and Chapter 1
- Regulatory and standards baseline tables in Chapter 4
- Missing Regulatory Object tables in Chapter 6
- RCCS-T, RCCS-M, ALCS, and optional composite scoring tables in Chapters 7-9
- Enterprise control, procurement, board reporting, model risk, and cybersecurity crosswalk tables in Chapter 10
- System mapping and comparative posture tables in Chapters 11-13 and Appendix D
- Evidence validation and failure scenario tables in Chapters 14-15
- Adoption roadmap and contribution-summary tables in Chapters 17-18
- Technical appendices A-K, including scorecard templates, source ledgers, detailed rubrics, Appendix G provisional system assessments, procurement scorecards, source audit registers, gap closure matrices, and editorial remediation records

Executive Summary

AI compliance is entering a structural transition. The compliance object is no longer limited to a model, a dataset, a prompt, or a single application output. In enterprise environments, AI systems increasingly plan work, use tools, call APIs, retrieve memory, delegate tasks, collaborate with other agents, and produce

operational outcomes. This means the central compliance question is changing from whether a model produced a safe output to whether an organized and proven agentic unit of work was authorized, executed, evidenced, reviewed, accepted, disputed, remediated, and improved.

The Missing Layer

The first generation of AI governance focused on model governance. That work remains essential: risk management, documentation, transparency, robustness, bias mitigation, security, and monitoring are all foundational. But agentic AI adds a new layer of operational consequence. A model answer becomes enterprise risk when it is embedded inside a workflow that can send an email, approve a refund, update a database, deploy code, modify access permissions, route a customer case, create a compliance record, or trigger another autonomous agent.

This white paper names the missing layer **AI Agent Lifecycle Governance**. AI Agent Lifecycle Governance is the governance layer that defines how agentic work is authorized, executed, evidenced, reviewed, accepted, disputed, remediated, and reused across agents, humans, tools, projects, vendors, and organizational boundaries.

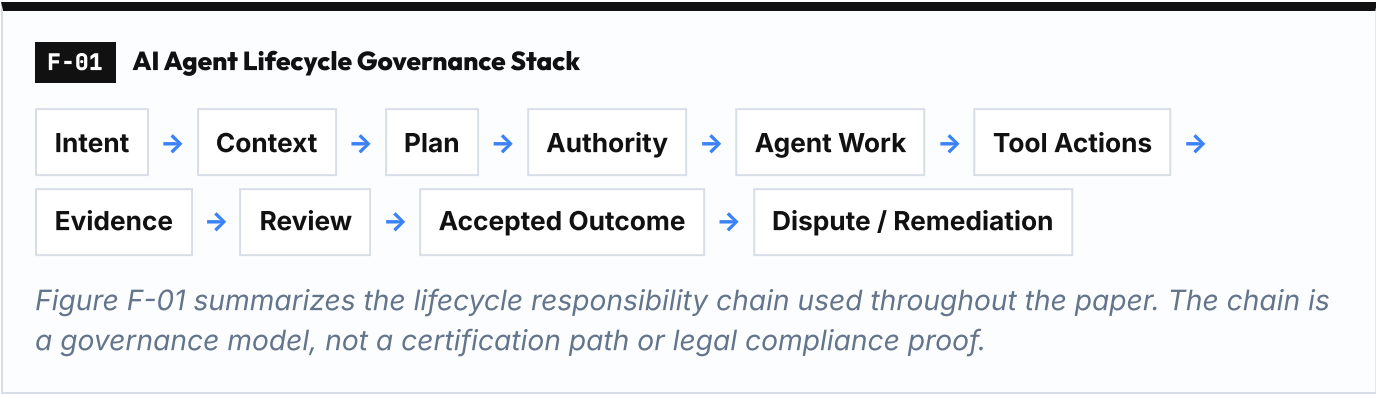
Key Findings

- Model governance remains necessary, but it is not sufficient for agentic and multi-agent workflows that act through tools, memory, APIs, and delegation.
- Lifecycle responsibility objects are needed to connect intent, authority, evidence, review, accepted outcome, dispute, remediation, and reuse.
- The sixteen Missing Regulatory Objects define engineering objects that help make existing governance expectations testable in agentic execution.
- RCCS-T, RCCS-M, and ALCS measure different but connected layers: traditional regulatory coverage, MRO-adjusted regulatory coverage, and agentic lifecycle conformance. A system can be strong in one layer and weaker in another.
- Logs, traces, dashboards, and guardrails are useful, but they are not evidence chains by themselves unless tied to authority, responsibility, privacy, acceptance, and remediation semantics.
- Validation verdicts can support evidence review, but they do not prove legal compliance, regulatory approval, certification, or procurement suitability.

Core Thesis

The paper argues that AI Agent / MAS compliance is not only model compliance. It is lifecycle responsibility compliance.

That thesis has practical consequences. Enterprise buyers, auditors, insurers, regulators, and internal governance teams need more than dashboards, logs, or guardrails. They need lifecycle objects: delegated authority boundaries, human-role-to-agent-responsibility mappings, tool-action liability records, evidence partitions, accepted outcome states, cross-project reuse controls, privacy-preserving validation packs, and dispute closure records.



Three-Profile Analytical Model

The report introduces three profile signals:

1. **RCCS-T - Traditional Regulatory Compliance Coverage Score** - measures how strongly a system covers existing regulatory and governance requirements such as risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring.
1. **RCCS-M - MRO-Adjusted Regulatory Compliance Coverage Score** - measures whether those regulatory obligations can be expressed through Missing Regulatory Objects required for agentic lifecycle responsibility compliance.
1. **ALCS - Agentic Lifecycle Conformance Score** - measures whether lifecycle responsibility conformance is mature across responsibility, authority, evidence, privacy, substitution, and remediation. A system can have strong RCCS-T and weaker RCCS-M or ALCS. That is not a contradiction. It means the system may be strong in existing model governance or platform controls while lacking deeper lifecycle responsibility semantics for multi-agent work.

Conceptual flow:

Technical example
RCCS-T -> MRO Adjustment Layer -> RCCS-M -> ALCS

RCCS-M is an author analytical, forward-looking, MRO-adjusted model. It is not current law, legal compliance proof, certification, a regulator-approved benchmark, vendor ranking, or procurement recommendation.

Missing Regulatory Objects

The report also introduces sixteen **Missing Regulatory Objects**. These are not proposed legal mandates. They are engineering objects that appear necessary if existing regulatory ideals are to become auditable in multi-agent execution. The objects are designed to make visible the difference between model governance, orchestration observability, and lifecycle responsibility governance.

The sixteen MROs are:

- MRO-01

: Human Role to MAS Responsibility Mapping
- MRO-02

: Delegated Authority Boundary
- MRO-03

: Agent Role is not Human Role
- MRO-04

: Accepted Outcome Compliance
- MRO-05

: Tool-Action Liability Boundary

- **MRO-06**: Responsibility Transfer Across Agents
- **MRO-07**: Authority Drift
- **MRO-08**: MAS Evidence Partitioning
- **MRO-09**: Cross-Project Reuse Compliance
- **MRO-10**: Privacy / GDPR Lifecycle Mapping
- **MRO-11**: Privacy-Preserving Third-Party Validation
- **MRO-12**: Evidence Minimization and Selective Disclosure
- **MRO-13**: Data Subject Rights vs Evidence Retention
- **MRO-14**: Third-Party Processor / Subprocessor Chain
- **MRO-15**: Vendor / Model / Runtime Substitution Conformance
- **MRO-16**: Incident, Dispute, and Remediation Closure

Comparative System Analysis

Finally, the report compares the major system categories now shaping the enterprise agentic stack:

- **MPLP** as a lifecycle protocol path
- **IBM watsonx.governance** as an enterprise AI governance platform
- **Microsoft Azure AI Foundry** as an enterprise AI platform and observability/control plane
- **AWS Bedrock and AgentCore** as managed agent runtime and guardrail infrastructure
- **Google Vertex AI/ADK/Model Armor** as managed AI and agent safety infrastructure
- **LangGraph/LangSmith** as stateful orchestration and observability/evaluation tooling
- **OpenAI Agents SDK** as a developer agent SDK
- **CrewAI** as a multi-agent orchestration framework

The comparison is not a ranking of product quality. It is a map of responsibility semantics. The key question is not which system is useful. Most of them are useful. The key question is whether the system defines the lifecycle responsibility objects required to prove that agentic work moved from intent to accepted outcome under controlled authority, evidence, privacy, and remediation constraints.

Enterprise Action Implications

- Start with an authority, responsibility, and evidence inventory before selecting or extending agentic platforms.
- Separate model governance controls from lifecycle responsibility controls so that logs, traces, guardrails, and reviews are not over-read as proof of accepted outcomes.
- Map current systems against MRO, RCCS-T, RCCS-M, and ALCS dimensions to identify which controls are documented, which are implementation-dependent, and which remain analytical assumptions.
- Treat MPLP, Validation Lab, and comparable lifecycle protocols or evidence tools as possible implementation patterns, not required paths or certification mechanisms.
- Preserve source and evidence levels near high-risk product, protocol, validation, and system-mapping claims before publication or procurement use.

Audience and Use

This paper is written for AI governance leaders, enterprise architects, compliance teams, cybersecurity leaders, agent platform builders, insurance and assurance professionals, and protocol designers. It is a technical governance analysis, not legal advice. It translates regulatory and standards language into engineering requirements for agentic and multi-agent systems.

Document Status

This is **v0.3.2 Public Research Edition (GACWP-2026-v0.3.2-FRC-R3)**. It remains bounded as non-legal technical governance analysis and does not claim certification, regulatory approval, procurement guidance, vendor ranking, or legal compliance proof.

Author: Jearon Wong

Author Position: Protocol Architect for the Agent Era

Protocol Disclosure: Creator of MPLP — Multi-Agent Lifecycle Protocol

Site: JearonWong.com

Non-legal technical governance analysis. Not legal advice. MPLP is discussed with explicit conflict-of-interest disclosure.

Chapter 1: Scope, Methodology, and Non-Legal Boundary

MAIN RESULT The paper defines a non-legal governance analysis for agentic lifecycle conformance.	METHOD / LENS Scope, evidence levels, document-control tables, and non-claim boundaries.
EVIDENCE BASIS Readers need the boundaries before interpreting any scoring, mapping, or system profile.	KEY TAKEAWAY / BOUNDARY The framework is analytical and source-qualified, not legal advice, certification, or procurement guidance.

Version: v0.3.2-FRC-R3

1.1 Purpose and Positioning

The front matter states the document boundary; this chapter explains how that boundary is operationalized as a technical governance method.

This paper is a technical governance analysis, not legal advice. It translates regulatory and standards language into engineering requirements for agentic and multi-agent systems. It is written for AI governance leaders, enterprise architects, AI risk teams, compliance teams, cybersecurity leaders, agent platform builders, insurance and assurance professionals, and protocol designers.

The paper does not claim that current AI regulation is absent or weak. The claim is narrower and more technical: current regulation and governance frameworks increasingly require human oversight, logging, documentation, monitoring, accountability, transparency, and privacy controls, but they do not yet fully specify the lifecycle objects required to prove those properties inside dynamic agentic and multi-agent execution.

This full research edition expands the v0.3 framework into a publication-grade white paper. Its purpose is to define a missing semantic and engineering layer for AI agent and multi-agent system compliance. It is deliberately more detailed than an executive brief: the paper introduces a terminology layer, an object model, a three-profile analytical model, system-specific mappings, comparative matrices, and implementation templates.

1.2 Intended Audience

This paper is designed for professionals who must translate regulatory obligations into operational systems:

- **AI Governance Leaders** who must define enterprise AI governance frameworks that extend beyond model risk management
- **Enterprise Architects** who must design agentic systems that can demonstrate lifecycle responsibility

- **AI Risk Teams** who must assess whether agentic workflows meet regulatory and internal control requirements
- **Compliance Teams** who must map regulatory language to technical controls
- **Cybersecurity Leaders** who must secure agentic workflows across tool actions, memory, and delegation
- **Agent Platform Builders** who must design platforms that support lifecycle governance primitives
- **Insurance and Assurance Professionals** who must evaluate whether agentic systems are auditable and insurable
- **Protocol Designers** who must create standards for agentic lifecycle responsibility

1.3 Scope Boundary

The scope includes internationally comparable legal frameworks, voluntary governance frameworks, management standards, web provenance standards, public cloud AI platforms, agent orchestration frameworks, developer SDKs, and protocol-level approaches. The analysis deliberately avoids jurisdiction-specific political content controls and focuses instead on lifecycle responsibility, auditability, privacy, evidence, and enterprise deployment readiness.

Table T-01-01: In Scope / Out of Scope

In Scope	Out of Scope
Agentic AI lifecycle responsibility	Legal advice for any specific jurisdiction
Multi-agent system responsibility transfer	Political content moderation regimes
Evidence, provenance, record-keeping, replay	Certification issuance or conformity assessment
Human oversight as an engineering control	Claims of regulatory approval
Privacy-preserving validation and selective disclosure	Full product security review of each vendor
Framework and platform semantic mapping	Performance benchmarking or model quality ranking

1.4 Methodology

The methodology uses a layered approach:

1. **Extract recurring regulatory concepts** from the baseline frameworks: risk management, documentation, logs, transparency, oversight, accountability, security, monitoring, privacy, and redress.
1. **Ask how these concepts must change** when the AI system becomes agentic. The central question is: what must be represented differently when the unit of work is no longer a single model output, but a multi-step workflow involving planning, tool use, memory, delegation, collaboration, and external consequence?
1. **Define Missing Regulatory Objects** that make those obligations testable. These are not proposed legal mandates. They are engineering objects that appear necessary if existing regulatory ideals are to become auditable in multi-agent execution.
1. **Introduce RCCS-T, RCCS-M, and ALCS scoring** to separate ordinary governance coverage, MRO-adjusted regulatory coverage, and lifecycle conformance. RCCS-T measures how strongly a system covers existing regulatory and governance requirements. RCCS-M measures whether that coverage can be expressed through Missing Regulatory Objects. ALCS measures whether a system defines the agentic lifecycle objects required for lifecycle responsibility compliance.
1. **Map representative systems** against those objects based on publicly available evidence. The comparison is not a ranking of product quality. It is a map of responsibility semantics.

The paper avoids pretending that regulatory texts already contain every engineering primitive needed for MAS. It also avoids the opposite error of claiming that existing regulation is irrelevant. The correct position is that existing regulation establishes the direction of travel, while agentic systems require additional engineering object models to implement that direction faithfully.

1.5 Evidence Basis

This hierarchy is necessary because the AI governance market has rapidly overloaded terms such as trace, guardrail, oversight, agent governance, evidence, approval, and evaluation. A trace in an observability product, a log in a cloud service, a review step in an SDK, and an evidence chain in a lifecycle protocol may all sound similar, but they have different compliance strength. This paper therefore treats terminology as insufficient unless it is tied to an object, a rule, an evidence source, and an enforcement or verification mechanism.

Table T-01-02: Evidence Basis Levels (detailed)

Level	Evidence Type	Use in This Paper	Confidence
L1	Binding legal, regulatory, or formal standards text	Used for baseline obligations and formal object requirements; does not prove implementation	Highest
L2	Official product documentation, API docs, developer documentation, platform architecture docs	Used for system mapping and capability interpretation	High
L3	Independent audits, certifications, third-party evaluations, regulatory commentary	Used for supplemental confidence and risk interpretation	Medium
L4	Vendor white papers, product pages, blog posts, marketing statements	Used only when lower-level technical evidence is unavailable or as positioning evidence	Low
L5	Author inference, conceptual analysis, strategic interpretation	Allowed only when clearly marked as inference	Disclosed

1.6 Non-Legal Boundary and Conflict of Interest Disclosure

Author position: Protocol Architect for the Agent Era

Non-legal technical governance analysis. Not legal advice. MPLP is discussed with explicit conflict-of-interest disclosure.

The author is the creator of MPLP - Multi-Agent Lifecycle Protocol. MPLP is one of the systems discussed in this paper. This creates an inherent conflict of interest. The conflict does not invalidate the analysis, but it must be disclosed clearly because this paper argues for a lifecycle protocol layer and then maps MPLP to that layer.

The controls used to limit over-claiming are as follows:

- **MPLP** is treated as a protocol path, not a regulator
- **Validation Lab** is treated as non-certifying evidence adjudication, not legal certification
- Other platforms are evaluated for their own strengths rather than dismissed
- Scoring claims are separated into traditional regulatory coverage, MRO-adjusted regulatory coverage, and lifecycle conformance evidence levels
- The paper distinguishes a conceptual object model from real-world legal compliance

Boundary	Required Wording	Forbidden Wording
MPLP	A lifecycle protocol path for expressing agentic responsibility objects	The only compliant standard
Validation Lab	A non-certifying evidence adjudication example	A certification authority
RCCS-T/RCCS-M/ALCS	A proposed analytical scoring framework; RCCS-M is author analytical and forward-looking	A regulator-approved benchmark
System mapping	A mapping of public primitives to lifecycle objects	A final legal compliance judgment
AI Agent Lifecycle Governance	A missing semantic and engineering layer	A binding regulatory standard

This boundary discipline strengthens the paper. The argument does not need to claim that MPLP is the only path. It only needs to show that agentic AI compliance requires lifecycle responsibility objects and that MPLP is one coherent way to express such objects.

Chapter 2: The Missing Layer: AI Agent Lifecycle Governance

MAIN RESULT Model governance is necessary but insufficient for agentic systems.	METHOD / LENS Lifecycle stack analysis and missing-layer framing.
EVIDENCE BASIS Agentic workflows shift compliance attention from model artifacts to delegated lifecycle responsibility.	KEY TAKEAWAY / BOUNDARY AI Agent Lifecycle Governance is the missing operational layer between model controls and accountable enterprise outcomes.

Version: v0.3.2-FRC-R3

2.1 Defining AI Agent Lifecycle Governance

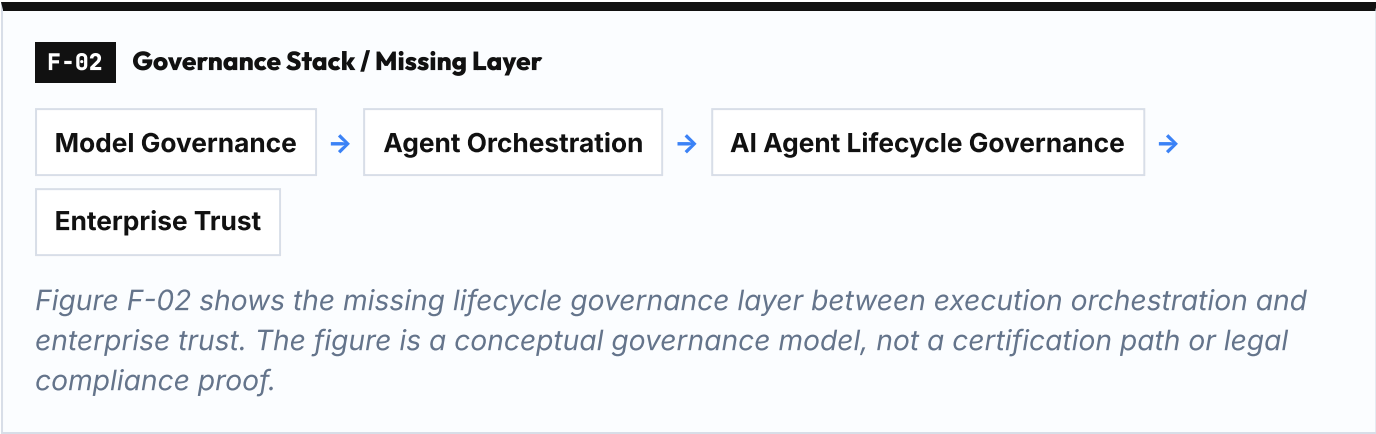
AI Agent Lifecycle Governance is the missing governance layer between model compliance and enterprise agent deployment. It defines how agentic work is authorized, executed, evidenced, reviewed, accepted, disputed, remediated, and reused across agents, humans, tools, projects, vendors, and organizational boundaries.

The phrase matters because existing language is fragmented:

- **Model governance** focuses on models
- **AI risk management** focuses on organizational risk processes
- **Agent orchestration** focuses on execution flow
- **Observability** focuses on traces, metrics, and debugging
- **Human-in-the-loop** focuses on review or approval moments

None of these terms alone names the full lifecycle responsibility problem.

AI Agent Lifecycle Governance names the whole accountability surface. It asks whether an organization can demonstrate continuity from intent to outcome, not merely whether a system completed a task. It asks whether a human role had the right responsibility boundary, not merely whether a person clicked approve. It asks whether a tool action had a liability boundary, not merely whether an API key existed. It asks whether evidence can be replayed for dispute, not merely whether logs exist. It asks whether privacy survives evidence retention, not merely whether a database has access control.



2.2 Why Existing Terms Are Insufficient

The table below shows how existing governance layers answer different questions and produce different artifacts. Each layer is necessary, but none alone addresses lifecycle responsibility for agentic work.

Table T-02-01: Governance Layer Comparison

Layer	Question It Answers	Typical Artifact	Limitation if Used Alone
Model Governance	Is the model robust, documented, monitored, and aligned with policy?	Model cards, evaluations, risk assessments, monitoring reports	Does not prove lifecycle responsibility for agentic work
Agent Orchestration	How do agents, tools, state, and workflows execute?	Graphs, tasks, tools, handoffs, checkpoints	Does not by itself define legal or governance responsibility
Observability	What happened during execution?	Traces, logs, metrics, dashboards	Does not by itself prove authorization, acceptance, or dispute readiness
Human Review	Did a person approve, reject, or modify a step?	Approval UI, review event, interrupt	Does not by itself map human role to MAS responsibility
AI Agent Lifecycle Governance	How did intent become accepted outcome under responsibility, authority, privacy, evidence, and remediation controls?	Lifecycle responsibility objects, evidence packs, acceptance records, rulesets	Requires adoption across architecture and governance, not only tooling

2.3 The Core Move: Regulatory Abstractions to Lifecycle Objects

The core move in this paper is to turn regulatory abstractions into lifecycle objects:

- **Human oversight** becomes **human-role-to-MAS responsibility mapping**
- **Logging** becomes **partitioned evidence chain**
- **Accountability** becomes **tool-action liability and responsibility transfer**
- **Transparency** becomes **evidence-linked review and accepted outcome**
- **Monitoring** becomes **authority drift detection, incident closure, and monitoring feedback**
- **Privacy** becomes **lifecycle data-flow mapping, minimization, and selective disclosure**

This is not bureaucracy for its own sake. It is the recognition that agentic systems need explicit objects because the old system boundary has dissolved. The work unit is now distributed across agents, tools, memory, people, projects, and vendors.

2.4 The Enterprise Implication

The enterprise implication is severe. If agentic workflows cannot prove lifecycle responsibility, they will remain difficult to audit, insure, delegate, reuse, and scale. This is the reason that AI Agent Lifecycle Governance is not a philosophical category. It is an operational prerequisite for agentic AI becoming trusted enterprise infrastructure.

Chapter 3: Why Agentic AI Breaks Model-Centric Compliance

MAIN RESULT Agentic systems break model-centric controls by acting through tools, roles, memory, and handoffs.	METHOD / LENS Agentic workflow analysis and lifecycle failure modes.
EVIDENCE BASIS Governance fails when the compliance object stays at the model while work occurs across lifecycle events.	KEY TAKEAWAY / BOUNDARY The compliance unit must include authority, action, evidence, acceptance, and closure.

Version: v0.3.2-FRC-R3

3.1 The Unit of Risk Changes

Agentic AI breaks model-centric compliance because the unit of risk changes. A model-centric system creates risk through prediction, generation, classification, or recommendation. An agentic system creates risk through action, delegation, state change, tool use, memory, collaboration, and external consequence.

This does not mean models are unimportant. It means that model safety is no longer the outer boundary of system safety. A safe model can still be embedded inside a workflow that can send an email, approve a refund, update a database, deploy code, modify access permissions, route a customer case, create a compliance record, or trigger another autonomous agent.

3.2 Model Governance Remains Necessary But Insufficient

The first generation of AI governance focused on model governance. That work remains essential: risk management, documentation, transparency, robustness, bias mitigation, security, and monitoring are all foundational. But agentic AI adds a new layer of operational consequence.

This is why model governance must be treated as a foundation rather than the system boundary. Once a model is embedded inside a workflow, compliance must follow the work unit, not only the model artifact.

3.3 Failure Mode Comparison

The table below compares how failure modes manifest differently in model-centric systems versus agentic lifecycle systems.

Table T-03-01: Failure Mode Comparison

Failure Mode	Model-Centric View	Lifecycle Governance View
Wrong output	Model produced incorrect text	Was the output accepted, used, challenged, corrected, and evidenced?
Unauthorized tool action	Agent called an external API	Was the action inside delegated authority and tied to a responsible owner?
Agent handoff	A workflow routed work to another agent	Were constraints and responsibility explicitly transferred?
Human approval	A person clicked approve	Was that human role authorized for this outcome and did the evidence support acceptance?
Audit log	System stored events	Can evidence be reconstructed across agents, tools, privacy boundaries, and plan versions?
Privacy issue	Database contained personal data	Did personal data flow through prompts, memory, tools, traces, and evidence packs under purpose limitation?

3.4 Why Well-Evaluated Agents Can Still Fail Governance

A well-evaluated agent can still act under the wrong authority. A perfectly traced workflow can still lack accepted outcome governance. A strong guardrail can still fail to define who owns a tool action. A human approval event can still fail to show whether the human had the correct responsibility role.

The enterprise implication is severe. If agentic workflows cannot prove lifecycle responsibility, they will remain difficult to audit, insure, delegate, reuse, and scale. This is the reason that AI Agent Lifecycle Governance is not a philosophical category. It is an operational prerequisite for agentic AI becoming trusted enterprise infrastructure.

3.5 The Compliance Question Changes

The central compliance question is changing:

- **Model-centric question:** Did the model produce a safe output?
- **Agentic lifecycle question:** Was an agentic unit of work authorized, executed, evidenced, reviewed, accepted, disputed, remediated, and improved under controlled authority, evidence, privacy, and remediation constraints?

This shift changes the compliance question. The test is no longer only whether the model output was safe, but whether the unit of agentic work can be shown to have moved from intent to accepted outcome under controlled authority, evidence, privacy, and remediation constraints.

The next chapter therefore turns to the regulatory and standards baseline: what existing frameworks already require, and where agentic lifecycle objects are still needed.

Chapter 4: Regulatory and Standards Baseline

MAIN RESULT Existing laws and standards already imply lifecycle duties but do not fully name agentic responsibility objects.	METHOD / LENS Snapshot-bound regulatory and standards comparison.
EVIDENCE BASIS The white paper should translate regulatory language into engineering objects without overstating law.	KEY TAKEAWAY / BOUNDARY The baseline supports object derivation, not legal interpretation for a specific deployment.

Version: v0.3.2-FRC-R3

4.1 Purpose of the Baseline

This chapter establishes the regulatory and standards baseline that informs the Missing Regulatory Objects and scoring frameworks introduced later in the paper. The baseline is not exhaustive. It focuses on frameworks that establish recurring obligations around risk management, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring.

The analysis does not claim that these frameworks are weak or absent. The claim is technical: these frameworks give baseline obligations and conceptual direction, but agentic systems require engineering objects that bind authority, responsibility, evidence, and privacy to concrete execution states.

Each framework analyzed in this chapter contributes essential governance direction. The shared pattern across all frameworks is that they establish what must be governed, but they do not themselves define the lifecycle objects required to prove that agentic work moved from intent to accepted outcome under controlled authority, evidence, privacy, and remediation constraints. That engineering layer is the focus of Chapters 6 through 9.

This distinction is the bridge to the paper's scoring model. Existing regulation supplies risk, accountability, transparency, oversight, privacy, record-keeping, and monitoring language. It does not consistently define the lifecycle responsibility objects needed by agentic and multi-agent systems. RCCS-T therefore measures traditional regulatory coverage, while later chapters introduce MROs and RCCS-M to ask whether that coverage can be expressed through lifecycle responsibility objects.

4.2 Regulatory Frameworks Baseline Table

The table below summarizes the frameworks analyzed in this chapter, their type, relevant requirements, and the agentic gap that remains.

Table T-04-01: Regulatory Frameworks Baseline

Framework	Type	Relevant Requirements	Agentic Gap
EU AI Act	Binding regulation	Technical documentation, record-keeping, transparency, human oversight, accuracy/robustness/cybersecurity, post-market monitoring	Strong legal baseline for high-risk systems; does not itself define MAS responsibility-transfer object models
GDPR	Binding privacy regulation	Lawfulness, fairness, transparency, purpose limitation, minimization, storage limitation, integrity/confidentiality, accountability	Applies strongly to agentic memory, prompt payloads, traces, evidence packs, and cross-project reuse; lifecycle mapping remains an engineering task
NIST AI RMF 1.0	Voluntary risk framework	Govern, Map, Measure, Manage functions across the AI lifecycle	Excellent risk-management umbrella; intentionally not a MAS protocol or delegated-authority object model
ISO/IEC 42001	Management system standard	AIMS requirements for establishing, implementing, maintaining, and improving AI management systems	Supports organizational governance; needs technical lifecycle evidence objects for agentic execution
Singapore IMDA Model AI Governance Framework for Agentic AI	Voluntary governance framework	Assessing and bounding risks upfront; making humans meaningfully accountable; implementing technical controls and processes; enabling end-user responsibility	Strong agentic governance signal; voluntary framework rather than protocol schema, certification, or binding law
W3C PROV / Verifiable Credentials	Web provenance and credential standards	Provenance of entities, activities, agents; tamper-evident machine-verifiable claims	Useful foundation for evidence chain, selective disclosure, and third-party validation; not agent governance by itself
Colorado AI Act (SB24-205 / SB25B-004)	U.S. state law sample; not yet effective at 2026-05-07 snapshot	Consumer protections for high-risk AI systems; effective date extended to 2026-06-30 by SB25B-004	Useful state-level signal for algorithmic accountability; should be treated as forward-looking in this paper, not as an active effective-law baseline at the snapshot date

4.3 EU AI Act

[EU-AI-ACT-ART-11] [EU-AI-ACT-ART-12] [EU-AI-ACT-ART-14] [EU-AI-ACT-ART-15] [EU-AI-ACT-ART-72]

EU AI Act functions in this paper as binding regulation. Its relevance to agentic lifecycle conformance comes from the following requirement cluster: technical documentation, record-keeping, transparency, human oversight, accuracy/robustness/cybersecurity, post-market monitoring.

- What it contributes:** Strong legal baseline for high-risk systems; binding direction around technical documentation, record-keeping, human oversight, and post-market monitoring.
- Why it matters for agentic AI:** Articles 11, 12, 14, and 72 together indicate that high-risk AI systems must be documentable, loggable, supervisable, and monitored across their lifecycle. The missing agentic engineering question is how those obligations apply when a single outcome is produced by a moving graph of agents, tools, memory, delegated tasks, and human confirmations.

What it does not define (the agentic gap): The framework does not itself define MAS responsibility-transfer object models.

How it connects to lifecycle objects: The EU AI Act establishes binding direction around technical documentation, record-keeping, human oversight, and post-market monitoring. In agentic workflows, those obligations need lifecycle evidence boundaries: which agent or human role acted, which authority applied, which logs support the action, how oversight occurred, and how post-market monitoring reaches the workflow state rather than only the model artifact.

4.4 GDPR

[GDPR-ART-5]

GDPR functions in this paper as binding privacy regulation. Its relevance to agentic lifecycle conformance comes from the following requirement cluster: lawfulness, fairness, transparency, purpose limitation, minimization, storage limitation, integrity/confidentiality, accountability.

What it contributes: Binding privacy regulation with strong requirements for lawfulness, fairness, transparency, purpose limitation, minimization, storage limitation, integrity, confidentiality, and accountability.

Why it matters for agentic AI: GDPR is crucial because agentic systems do not keep personal data in one place. Personal data may appear in user intent, prompt context, retrieval results, memory, tool payloads, trace records, review packets, evidence packs, and third-party validation artifacts. Therefore privacy controls need to be mapped to lifecycle phases, not merely database tables.

What it does not define (the agentic gap): Lifecycle mapping remains an engineering task.

How it connects to lifecycle objects: GDPR pushes the lifecycle question into data flow. Privacy controls must follow prompts, memory, retrieval context, tool payloads, traces, evidence packs, redaction profiles, and validation artifacts. A lifecycle object model makes those flows reviewable without treating a database policy or privacy notice as the whole control surface.

4.5 NIST AI RMF 1.0

[NIST-AI-RMF-1.0]

NIST AI RMF 1.0 functions in this paper as voluntary risk framework. Its relevance to agentic lifecycle conformance comes from the following requirement cluster: Govern, Map, Measure, Manage functions across the AI lifecycle.

What it contributes: Excellent risk-management umbrella covering Govern, Map, Measure, and Manage functions across the AI lifecycle.

Why it matters for agentic AI: Provides a comprehensive risk management structure that applies to all AI systems, including agentic systems. The framework establishes the risk management direction; lifecycle objects make that direction testable in agentic workflows.

What it does not define (the agentic gap): Intentionally not a MAS protocol or delegated-authority object model.

How it connects to lifecycle objects: NIST AI RMF supplies the risk-management frame; agentic execution needs the frame to bind to action classes, delegated authority, evidence sufficiency, monitoring triggers, remediation records, and accepted-outcome review. Lifecycle objects make the Govern, Map, Measure, and Manage functions concrete inside multi-step workflows.

4.6 ISO/IEC 42001

[ISO-IEC-42001]

ISO/IEC 42001 functions in this paper as management system standard. Its relevance to agentic lifecycle conformance comes from the following requirement cluster: AIMS requirements for establishing, implementing, maintaining, and improving AI management systems.

What it contributes: Management system standard for AI governance; provides organizational structure for AI governance programs.

Why it matters for agentic AI: Establishes organizational governance requirements that apply to agentic systems. Organizational governance must be supported by technical lifecycle objects that prove governance was enforced at execution boundaries.

What it does not define (the agentic gap): Needs technical lifecycle evidence objects for agentic execution.

How it connects to lifecycle objects: ISO/IEC 42001 can organize the management system around AI governance, but agentic systems still need execution-level evidence objects. The management system needs records that show where policy, responsibility, authorization, monitoring, correction, and improvement were enforced in the actual agentic lifecycle.

4.7 Singapore IMDA Model AI Governance Framework for Agentic AI

[SINGAPORE-MGF-AGENTIC-AI]

Singapore MGF for Agentic AI functions in this paper as a voluntary governance framework. Its relevance comes from its four-part framing: assessing and bounding risks upfront; making humans meaningfully accountable; implementing technical controls and processes; and enabling end-user responsibility. Technical controls are especially relevant for agent planning, tools, protocol interactions, testing, monitoring, and lifecycle controls.

What it contributes: Most directly aligned with agentic-specific risk language; provides clear guidance on human accountability, technical controls, and end-user responsibility in agentic contexts.

Why it matters for agentic AI: Singapore's agentic AI framework is a signal that governance bodies are now treating agent planning, tool use, human accountability, testing, monitoring, and end-user responsibility as agent-specific concerns. The framework names the agentic lifecycle control problem; lifecycle objects make those controls testable.

What it does not define (the agentic gap): Still a governance framework rather than a protocol schema. It should not be overstated as a protocol, law, or certification regime; it is a voluntary framework that helps name the agentic lifecycle control problem.

How it connects to lifecycle objects: The Singapore framework names the agentic governance problem more directly than many general AI frameworks. The implementation step is to translate that framing into protocol or schema-level controls for authority, human accountability, testing, monitoring, tool use, and lifecycle evidence.

4.8 W3C PROV / Verifiable Credentials

[W3C-PROV] [W3C-VC-DM-2.0]

W3C PROV / Verifiable Credentials functions in this paper as web provenance and credential standards. Its relevance to agentic lifecycle conformance comes from the following requirement cluster: provenance of entities, activities, agents; tamper-evident machine-verifiable claims.

What it contributes: Useful foundation for evidence chain, selective disclosure, and third-party validation.

Why it matters for agentic AI: Provides standards for tamper-evident, machine-verifiable claims that can support evidence chains and third-party validation. Provenance standards provide the technical foundation for evidence chains; lifecycle objects define what must be proven.

What it does not define (the agentic gap): Not agent governance by itself.

How it connects to lifecycle objects: W3C PROV and Verifiable Credentials can represent provenance and machine-verifiable claims, but they do not decide which agentic governance facts must exist. Lifecycle objects define the semantics that provenance and credential structures may carry: authority, responsibility, evidence partition, selective disclosure, validation verdict, dispute, and remediation state.

4.9 Colorado AI Act (SB24-205 / SB25B-004)

[COLORADO-SB25B-004]

Colorado AI Act (SB24-205 / SB25B-004) functions in this paper as U.S. state law sample; not yet effective at 2026-05-07 snapshot. Consumer protections for high-risk AI systems; effective date extended to 2026-06-30 by SB25B-004.

What it contributes: Useful state-level signal for algorithmic accountability.

Why it matters for agentic AI: Demonstrates U.S. state-level movement toward algorithmic accountability requirements. At the 2026-05-07 snapshot date used in this paper, the effective date had been extended to 2026-06-30 by SB25B-004. It should therefore be treated as a forward-looking indicator of U.S. state-level algorithmic accountability requirements, not as an active baseline law at the time of writing.

What it does not define (the agentic gap): Should be treated as forward-looking in this paper, not as an active effective-law baseline at the snapshot date.

How it connects to lifecycle objects: State-level accountability requirements point toward a need for lifecycle evidence that can connect risk management, notice, review, impact, and accountability to actual system behavior. This paper keeps the Colorado discussion snapshot-bound and treats final legal-status verification as a publication QA item.

Chapter 5: From Regulatory Language to Engineering Objects

MAIN RESULT Regulatory requirements become actionable only when mapped to lifecycle objects.	METHOD / LENS Object-derivation logic from governance language.
EVIDENCE BASIS Enterprises need implementable objects, not only policy principles.	KEY TAKEAWAY / BOUNDARY The chapter bridges legal/standards language into Missing Regulatory Objects.

Version: v0.3.2-FRC-R3

5.1 The Gap Between Regulation and Engineering

The gap between regulation and engineering is often hidden by familiar words. A regulation may say record-keeping, but an engineer must decide what fields exist, when records are emitted, whether they are tamper-evident, how they are partitioned, what retention policy applies, how they reference plan versions, how personal data is minimized, whether they support selective disclosure, and how the record is used in dispute.

A governance document may say human oversight, but an implementation must decide which human role can approve which tool action, under which risk class, with what evidence, with what override path, and how that responsibility boundary is preserved across agent handoffs.

A framework may require transparency, but an agentic system must decide whether evidence can be replayed for review, whether the replay is reconstructable from partitioned evidence, whether privacy constraints allow disclosure, and whether the review is linked to accepted outcome governance.

A standard may mandate accountability, but a multi-agent workflow must determine who owns a tool action, how liability transfers across agents, how responsibility is recorded, and how ownership is proven when disputes arise.

Familiar regulatory words establish direction. Engineering objects make that direction testable, auditable, and enforceable in distributed agentic execution.

This chapter provides the bridge from regulatory language to the Missing Regulatory Objects that will be defined in Chapter 6.

5.2 Regulatory Language to Engineering Objects Mapping

The table below maps recurring regulatory language to the agentic engineering questions that must be answered, and the required lifecycle objects that make those answers testable.

Table T-05-01: Regulatory Language to Engineering Objects Mapping

Regulatory Language	Agentic Engineering Question	Required Lifecycle Object
Risk management	What can go wrong as agents plan, delegate, call tools, and modify state?	Risk register linked to lifecycle phases and action classes
Technical documentation	Can the system describe its agent roles, authority boundaries, evidence schema, and limitations?	Versioned architecture and lifecycle object documentation
Record-keeping	Can the organization reconstruct what happened across agents and tools?	Partitioned evidence chain with plan, role, tool, and outcome links
Transparency	Can reviewers understand the evidence behind an outcome?	Evidence pointer graph and explanation packet
Human oversight	Which human can approve, reject, override, or accept which action?	Human-role-to-MAS responsibility map and confirmation boundary
Accountability	Who owns a decision, tool action, or outcome?	Responsibility boundary, transfer record, and accepted outcome owner
Contestability	Can an affected party challenge the result and trigger review?	Dispute-ready replay and remediation closure object
Privacy	How is personal data constrained across memory, tools, traces, and evidence?	Privacy lifecycle map and selective disclosure policy
Monitoring	How are drift, incident, and lifecycle failures detected?	Continuous monitoring and incident closure records

This table is the bridge from legal language to the Missing Regulatory Objects defined in Chapter 6. The point is not to invent bureaucracy for its own sake. The point is to recognize that agentic systems need explicit objects because the old system boundary has dissolved. The work unit is now distributed across agents, tools, memory, people, projects, and vendors.

5.3 Why Explicit Objects Are Necessary

Without explicit lifecycle objects, regulatory language remains aspirational rather than enforceable:

- **Risk management** becomes a policy document that cannot be tied to actual agent actions. A risk register must link to lifecycle phases, action classes, and delegated authority boundaries.
- **Technical documentation** describes intent but not how authority, evidence, and privacy are enforced. Versioned architecture documentation must specify agent roles, authority boundaries, evidence schema, and limitations.
- **Record-keeping** produces flat logs that cannot reconstruct cross-agent responsibility. Evidence must be partitioned by agent, tool, human confirmation, plan version, authority boundary, privacy boundary, and accepted outcome.
- **Transparency** provides dashboards that show execution but not accepted outcome governance. Evidence pointer graphs must link outcomes to the authority, evidence, and review that made them accepted.
- **Human oversight** becomes a button click without role-to-responsibility mapping. The system must map which human role can approve, reject, override, or accept which action, under which risk class, with what evidence, and with what escalation path.
- **Accountability** becomes a narrative claim without evidence-backed ownership. Responsibility boundaries must record who owns a decision, tool action, or outcome, how liability transfers across agents, and how ownership is proven in disputes.

- **Contestability** becomes impossible because evidence cannot be replayed under privacy constraints. Dispute-ready replay requires partitioned evidence, selective disclosure, and remediation closure objects.
- **Privacy** becomes a database policy that does not govern prompts, memory, tools, traces, and evidence packs. Privacy lifecycle mapping must bind lawful basis, purpose, minimization, retention, access, erasure, redaction, and disclosure to every lifecycle surface where data may flow or persist.
- **Monitoring** detects failures but cannot close incidents through evidence, responsibility, correction, and acceptance. Continuous monitoring must link drift detection, incident triggers, evidence packs, responsible roles, corrective actions, and closure states.

5.4 The Engineering Object Requirement

At this point in the argument, the thesis becomes operational: lifecycle responsibility compliance requires objects that can be represented, emitted, reviewed, and replayed.

Enterprise buyers, auditors, insurers, regulators, and internal governance teams need more than dashboards, logs, or guardrails. They need lifecycle objects: delegated authority boundaries, human-role-to-agent-responsibility mappings, tool-action liability records, evidence partitions, accepted outcome states, cross-project reuse controls, privacy-preserving validation packs, and dispute closure records.

These objects are not proposed legal mandates. They are engineering and assurance objects that appear necessary if existing regulatory ideals are to become auditable in multi-agent execution. The objects are designed to make visible the difference between model governance, orchestration observability, and lifecycle responsibility governance.

Existing regulation establishes the direction of travel. Agentic systems require additional engineering object models to implement that direction faithfully.

This translation is what makes RCCS-M possible. Without MROs, regulatory coverage remains RCCS-T: a traditional view of governance primitives such as documentation, oversight, record-keeping, transparency, security, accountability, contestability, and monitoring. With MROs, the same coverage can be tested against agentic lifecycle responsibility objects. RCCS-M is the author analytical profile for that MRO-adjusted coverage.

5.5 Bridge to Chapter 6

The following chapter defines sixteen Missing Regulatory Objects. These are not proposed legal mandates. They are engineering objects that appear necessary if existing regulatory ideals are to become auditable in multi-agent execution.

The objects are designed to make visible the difference between model governance, orchestration observability, and lifecycle responsibility governance. They address the gap between regulatory language and agentic execution by providing concrete, testable, evidence-backed structures for:

- Human-to-agent responsibility mapping
- Delegated authority boundaries
- Agent role definitions
- Accepted outcome compliance
- Tool-action liability boundaries
- Responsibility transfer across agents
- Authority drift detection
- Evidence partitioning

- Cross-project reuse compliance
- Privacy lifecycle mapping
- Privacy-preserving third-party validation
- Evidence minimization and selective disclosure
- Data subject rights vs evidence retention
- Third-party processor / subprocessor chains
- Vendor / model / runtime substitution conformance
- Incident, dispute, and remediation closure

The bridge from this chapter to Chapter 6 is direct: regulatory language establishes what must be proven; Missing Regulatory Objects define how to prove it in agentic workflows. In scoring terms, this is the bridge from RCCS-T to RCCS-M.

6. Missing Regulatory Objects for Agentic and MAS Compliance

MAIN RESULT Sixteen MROs define the missing lifecycle responsibility layer.	METHOD / LENS Canonical MRO dictionary and object field model.
EVIDENCE BASIS Without these objects, authority, evidence, privacy, substitution, and closure remain implicit.	KEY TAKEAWAY / BOUNDARY MROs provide analytical object structure, not a certification checklist.

6.0 Why Missing Regulatory Objects Matter

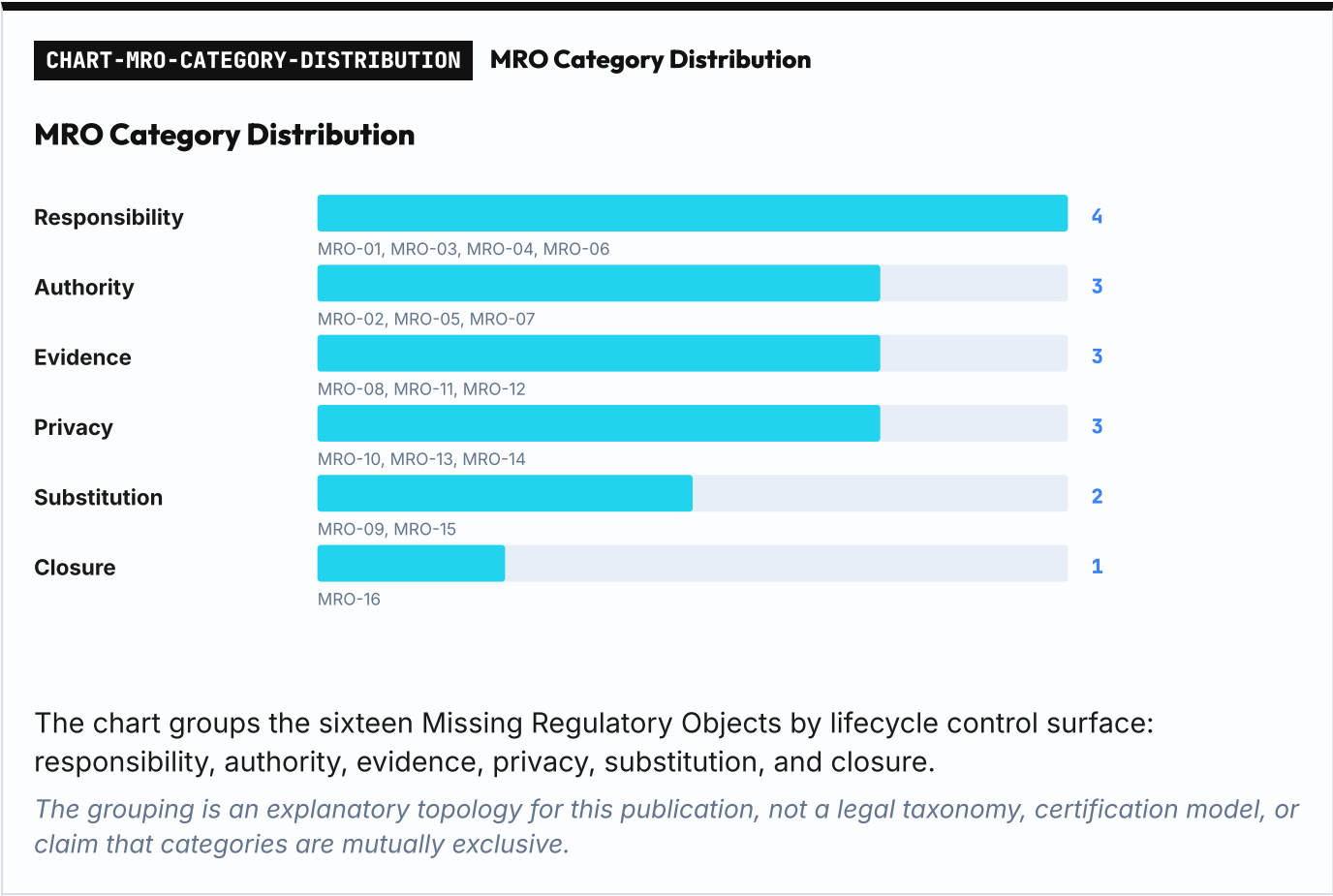
This chapter defines sixteen Missing Regulatory Objects. They are called missing because existing governance language implies their necessity but usually does not define their concrete machine-readable form. They are not proposed statutes. They are engineering and assurance objects that allow regulatory concepts to become testable in agentic workflows.

The phrase matters because existing language is fragmented. Model governance focuses on models. AI risk management focuses on organizational risk processes. Agent orchestration focuses on execution flow. Observability focuses on traces, metrics, and debugging. Human-in-the-loop focuses on review or approval moments. None of these terms alone names the full lifecycle responsibility problem.

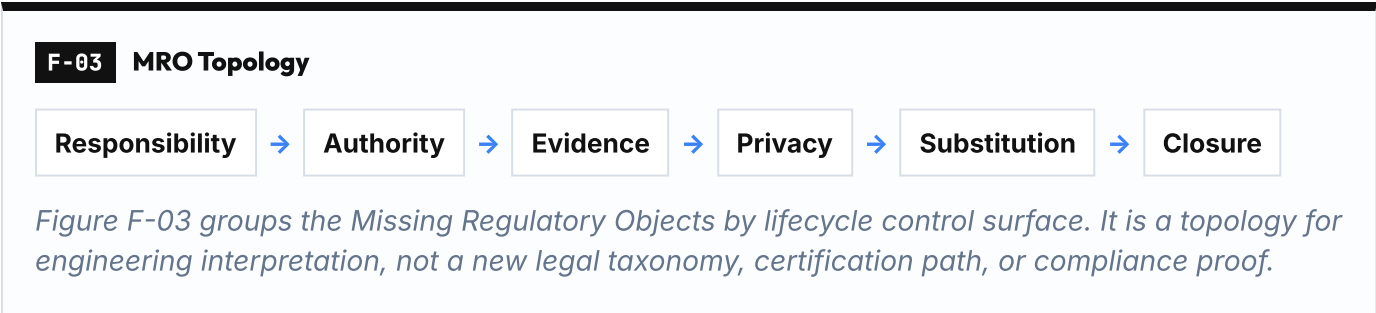
These objects bridge the gap between regulatory abstractions and lifecycle implementation. Human oversight becomes human-role-to-MAS responsibility mapping. Logging becomes partitioned evidence chain. Accountability becomes tool-action liability and responsibility transfer. Transparency becomes evidence-linked review and accepted outcome. Monitoring becomes lifecycle drift, incident closure, and continuous improvement. Privacy becomes lifecycle data-flow mapping, minimization, and selective disclosure.

The core move in this paper is to turn regulatory abstractions into lifecycle objects. Regulatory compliance coverage remains the baseline, but agentic lifecycle conformance is the differentiating layer. If agentic workflows cannot prove lifecycle responsibility, they will remain difficult to audit, insure, delegate, reuse, and scale. This is the reason that AI Agent Lifecycle Governance is not a philosophical category. It is an operational prerequisite for agentic AI becoming trusted enterprise infrastructure.

6.1 Summary of the Sixteen Missing Regulatory Objects



The following table summarizes all sixteen MROs. Full detail for [MRO-01](#) through [MRO-16](#) is provided in this chapter.



ID	Object	Core Judgment	Primary Lifecycle Concern
MRO-01	Human Role to MAS Responsibility Mapping	Human oversight without human-to-agent responsibility mapping is supervision theater.	Accountability, human oversight, role mapping
MRO-02	Delegated Authority Boundary	IAM permission decides access. Delegated authority decides responsibility.	Authority, delegation, scope control
MRO-03	Agent Role is not Human Role	An agent role is a bounded responsibility surface, not a human job title.	Role semantics, capability boundaries
MRO-04	Accepted Outcome Compliance	Output is a system event. Accepted outcome is a governance state.	Outcome acceptance, review, evidence

ID	Object	Core Judgment	Primary Lifecycle Concern
MRO-05	Tool-Action Liability Boundary	Tool use is where AI output be-comes external consequence.	External actions, liability, reversibility
MRO-06	Responsibility Transfer Across Agents	Handoff without explicit responsi-bility transfer is just routing.	Multi-agent handoff, constraint inheritance
MRO-07	Authority Drift	The most dangerous agent failures are often unauthorized authority transitions.	Authority monitoring, drift detection
MRO-08	MAS Evidence Partitioning	In MAS, evidence must be parti-tioned, linked, and reconstructable.	Evidence chain, multi-agent traceability
MRO-09	Cross-Project Reuse Compliance	Cross-project reuse is not just effi-ciency. It is a compliance boundary.	Reuse, context reset, privacy review
MRO-10	Privacy / GDPR Lifecycle Mapping	Privacy compliance in agentic sys-tems must be mapped across the agent lifecycle.	Privacy, data flow, lifecycle stages
MRO-11	Privacy-Preserving Third-Party Validation	External validation without raw sensitive data is a prerequisite for enterprise audit-readiness.	External validation, privacy preservation
MRO-12	Evidence Minimization and Selective Disclosure	Evidence without minimization be-comes a privacy risk. Privacy with-out evidence becomes an ac-countability gap.	Evidence minimization, selective disclosure
MRO-13	Data Subject Rights vs Evidence Retention	Agentic audit systems must pre-serve accountability without hoarding personal data.	Data subject rights, retention, accountability
MRO-14	Third-Party Processor / Subprocessor Chain	A multi-agent system is also a multi-party responsibility chain.	Processor chain, vendor responsibility
MRO-15	Vendor / Model / Runtime Substitution Conformance	Vendor neutrality is not real unless compliance survives substitution.	Substitution, vendor changes, conformance continuity
MRO-16	Incident, Dispute, and Remediation Closure	A compliance incident closes when evidence, responsibility, cor-rection, and acceptance are resolved.	Incident closure, dispute resolu-tion, remediation

Table T-06-01: Summary of the Sixteen Missing Regulatory Objects

6.1.1 Common MRO Field Model

All sixteen MRO object cards share a common set of identity and evidence fields. This common field model avoids repeating generic fields across every object card and establishes a consistent structure for lifecycle governance objects.

The following table defines the common fields that apply to all MROs:

Table T-06-01A: Common MRO Field Model

Common Field	Purpose
object_id	Unique identifier for the object instance
object_type	MRO type identifier for the object instance
lifecycle_id	Identifier linking the object to a specific lifecycle execution
project_id	Project or regulated context in which the object applies
object_version	Version identifier for the object schema or instance
related_human_role_id	Human role identifier associated with the object

Common Field	Purpose
related_agent_role_id	Agent role identifier associated with the object, where applicable
authority_scope	Scope of authority for this object
risk_class	Risk classification for this object
evidence_pointer	Reference to evidence supporting the object's validity
privacy_treatment	Privacy treatment for this object
status	Current lifecycle status of the object (active, closed, revoked, etc.)
created_at	Timestamp when this object was created
updated_at	Timestamp when this object was last updated
closure_reason	Reason for closure, if the object is closed

Implementation aliases may appear in specific systems. In this paper, `actor_role_id` is treated as an implementation alias for `related_human_role_id`, and `timestamp` is treated as implementation shorthand for `created_at` / `updated_at`, not as separate canonical fields.

Individual MRO object cards below therefore emphasize MRO-specific purpose, controls, failure mode, and audit question rather than repeating the full common field model.

6.2 MRO-01 — Human Role to MAS Responsibility Mapping

Problem

Regulation can require oversight, but agentic execution needs a precise map from human accountable roles to agent roles, tasks, tools, evidence, and accepted outcomes.

Required Object

A role-responsibility map binding human role IDs, agent role IDs, permitted delegated scopes, reserved human decisions, review duties, and evidence obligations.

Judgment

Human oversight without human-to-agent responsibility mapping is supervision theater.

Object Card

Field	Description
Object purpose	To make human role to MAS responsibility mapping explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	<code>object_id</code> , <code>object_type</code> , <code>lifecycle_id</code> , <code>project_id</code> , <code>object_version</code> , <code>related_human_role_id</code> , <code>related_agent_role_id</code> where applicable, <code>evidence_pointer</code> , <code>status</code> , <code>created_at</code> , <code>updated_at</code>
Required controls	scope boundary, risk class, evidence requirement, privacy treatment, review/approval rule, revocation or closure rule
Failure if missing	The system may appear governed at UI or workflow level while lacking enforceable human role to MAS responsibility mapping semantics.
Audit question	Can an independent reviewer determine whether human role to MAS responsibility mapping was properly established and preserved for the relevant lifecycle phase?

Table T-06-02: MRO-01 Object Card — Human Role to MAS Responsibility Mapping

Detailed Discussion

This object is the bridge between human organizational accountability and machine execution. In a MAS, a product owner, reviewer, compliance officer, operator, or customer-success lead may each retain different decision rights even when agents perform the underlying work. The mapping must therefore distinguish who owns intent, who approves risk, who reviews evidence, and who accepts the final outcome.

The core question for [MRO-01](#) is whether a system can identify the accountable human role for each agentic lifecycle state, not merely whether a user account or approval button exists. Strong mapping requires a durable relationship among human role, agent role, delegated scope, evidence obligation, review duty, and accepted outcome authority.

The difficulty is that a human may own business acceptance while an agent owns evidence collection, and another agent owns execution. A governance object must preserve those differences instead of flattening them into a single owner field.

Enterprise Implication

Without this map, a company can show that a human was somewhere in the loop but cannot prove that the right human held the right responsibility at the right lifecycle point. This weakens internal audit, customer assurance, and board-level accountability because role labels become narrative claims rather than evidence-backed assignments.

Adjacent identity, approval, or workflow controls may support this object, but lifecycle conformance requires explicit object semantics that preserve responsibility mapping across multi-agent delegation, cross-project reuse, and external validation. The implementation test is to select a completed run and ask: who owned intent, who approved risk, who executed each step, who reviewed evidence, and who accepted the outcome? If the answer requires tribal memory, the mapping is not audit-grade.

Design Implication

The object should be represented as a role-responsibility graph linking human role IDs, agent role IDs, delegated scopes, reserved decisions, review duties, evidence obligations, and accepted-outcome authority. It must be updated when responsibilities move across teams or projects.

The design should separate persona labels from capability boundaries, evidence duties, and human accountability so that collaboration structure does not substitute for responsibility semantics.

Audit Question

For any completed agentic workflow, can the system reconstruct which human role held which responsibility at each lifecycle phase, backed by evidence pointers and acceptance records?

6.3 MRO-02 — Delegated Authority Boundary

Problem

API access and tool permission do not prove that a particular business action was authorized under the correct scope, condition, risk level, and escalation path.

Required Object

A delegated-authority object binding role, action, scope, condition, risk class, evidence requirement, expiry, revocation, and escalation path.

Judgment

IAM permission decides access. Delegated authority decides responsibility.

Object Card

Field	Description
Object purpose	To make delegated authority boundary explicit, inspectable, and re-playable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, relat-ed_human_role_id, related_agent_role_id where applicable, evidence_-pointer, status, created_at, updated_at
Required controls	scope boundary, risk class, evidence requirement, privacy treatment, review/approval rule, revocation or closure rule
Failure if missing	The system may appear governed at UI or workflow level while lacking enforceable delegated authority boundary semantics.
Audit question	Can an independent reviewer determine whether delegated authority boundary was properly established and preserved for the relevant life-cycle phase?

Table T-06-03: MRO-02 Object Card — Delegated Authority Boundary

Detailed Discussion

Delegated authority is narrower than access control. A model or agent may technically possess a credential while lacking business authority to use it for a particular action. The compliance object must describe not only who can call a tool, but why that call is authorized under a specific intent, plan, risk class, condition, and time window.

The core question for MRO-02 is whether a system distinguishes technical permission from business authority. Strong support requires action-level delegation tied to scope, condition, risk class, expiry, escalation, and evidence. The same tool call can be harmless in one context and impermissible in another. Authority must therefore be evaluated against intent and plan state, not only against static credentials.

Enterprise Implication

If authority is reduced to IAM, API keys, or tool availability, a low-risk task can silently become a high-impact act. The organization may later discover that the system had permission to act but no evidence that the action was approved under the relevant business scope.

Adjacent controls such as IAM, guardrails, and workflow approvals may support this object, but lifecycle conformance requires explicit object semantics that bind a specific action to delegated authority, escalation rules, and evidence.

Design Implication

The boundary should be checked before high-impact tool calls and should include action class, permitted scope, expiration, revocation, escalation path, evidence pointer, and human override requirements. It should fail closed when authority cannot be reconstructed.

Adjacent platform capabilities can support lifecycle governance, but they do not automatically define accepted outcome, human-role-to-MAS mapping, or cross-project responsibility continuity.

Audit Question

Before a high-impact tool call, can the system show the exact authority basis and escalation rule for that action? If not, permission is being mistaken for authority.

6.4 MRO-03 — Agent Role is not Human Role

Problem

Agent names such as researcher, writer, reviewer, or manager are useful UI metaphors but are not adequate compliance identities.

Required Object

An agent responsibility boundary declaring duties, non-duties, allowed tools, inherited constraints, verification expectations, and human accountability linkage.

Judgment

An agent role is a bounded responsibility surface, not a human job title.

Object Card

Field	Description
Object purpose	To make agent role boundaries explicit, inspectable, and replayable within an agentic lifecycle, separating persona labels from responsibility semantics.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	scope boundary, risk class, evidence requirement, privacy treatment, review/approval rule, revocation or closure rule
Failure if missing	The system may appear governed at UI or workflow level while lacking enforceable agent role responsibility semantics.
Audit question	Can an independent reviewer determine whether agent role boundaries were properly established and preserved for the relevant lifecycle phase?

Table T-06-04: MRO-03 Object Card — Agent Role is not Human Role

Detailed Discussion

Agent labels such as researcher, writer, analyst, or reviewer are useful interface metaphors, but they do not automatically carry legal, organizational, or professional responsibility. A compliance-grade role must define bounded capabilities, evidence obligations, escalation duties, and non-delegable human responsibilities.

The core question for MRO-03 is whether agent roles are governance identities or only execution personas. Strong support requires separating persona labels from capability boundaries, evidence duties, and human accountability. Human-like role labels are persuasive in UI and documentation, but they can obscure the fact that legal and business accountability remains with humans and organizations.

Enterprise Implication

Treating an agent persona as a human job role creates false accountability. The company may believe that a reviewer agent has performed review while no accountable human or approved review boundary exists.

Adjacent controls may support this object, but lifecycle conformance requires explicit object semantics for responsibility, evidence, escalation duties, and human accountability linkage.

Design Implication

Separate UX persona, runtime capability, and governance role. The role object should identify what the agent can do, what it must prove, what it must escalate, and which human role remains accountable for acceptance or rejection.

The implementation test is to replace the agent name with a neutral identifier. If the governance meaning disappears, the system is relying on persona language rather than responsibility structure.

Audit Question

If the agent name is replaced with a neutral identifier, does the governance meaning remain intact, or was the system relying on persona language rather than responsibility structure?

6.5 MRO-04 — Accepted Outcome Compliance

Problem

A model output or orchestrator completion event does not become enterprise work until the right role has reviewed, accepted, or rejected it under evidence.

Required Object

An accepted_outcome state linking source intent, plan, trace, reviewer, acceptance status, dispute window, remediation route, and evidence pointers.

Judgment

Output is a system event. Accepted outcome is a governance state.

Object Card

Field	Description
Object purpose	To make accepted outcome compliance explicit, inspectable, and re-playable within an agentic lifecycle, separating execution completion from governance acceptance.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	scope boundary, risk class, evidence requirement, privacy treatment, review/approval rule, revocation or closure rule
Failure if missing	The system may appear governed at UI or workflow level while lacking enforceable accepted outcome compliance semantics.
Audit question	Can an independent reviewer determine whether accepted outcome compliance was properly established and preserved for the relevant lifecycle phase?

Table T-06-05: MRO-04 Object Card — Accepted Outcome Compliance**Detailed Discussion**

Agentic systems often mark work as complete when execution reaches a terminal state. Compliance requires a stronger state: the output must be reviewed against intent, evidence, risk, and acceptance criteria before it becomes an accepted outcome.

The core question for MRO-04 is whether output completion is separated from outcome acceptance. Strong support requires acceptance criteria, reviewer identity, evidence linkage, and dispute/remediation state. Execution engines naturally optimize for terminal states, while governance requires a second state transition from completed output to accepted outcome.

Enterprise Implication

Without accepted-outcome semantics, completed tasks may enter business processes without accountable review. This creates disputes when customers, regulators, or internal stakeholders ask who accepted the work and on what evidence.

Adjacent execution controls may complete work reliably, but lifecycle conformance requires explicit object semantics for accepted delivery rather than executed task closure. In a dispute, the enterprise must be able to distinguish execution quality, review failure, and missing acceptance policy.

Design Implication

The `accepted_outcome` object should bind deliverable ID, plan version, trace reference, reviewer role, acceptance criteria, acceptance decision, dispute window, and remediation path. Completion and acceptance should remain separate states.

The implementation test is to identify a completed task and ask: who accepted it, against what criteria, with what evidence, and what is the dispute window? If the answer is that completion equals acceptance, the system has collapsed execution into governance.

Audit Question

For any completed task, can the system identify the acceptance criteria, evidence reviewed, reviewer role, decision timestamp, and dispute window?

6.6 MRO-05 — Tool-Action Liability Boundary

Problem

Agentic AI becomes operationally consequential when it sends, modifies, deploys, purchases, transfers, deletes, or triggers external actions.

Required Object

A tool-action liability object recording initiator, tool identity, external system, authority credential, reversibility, rollback plan, affected data, and owner.

Judgment

Tool use is where AI output becomes external consequence.

Object Card

Field	Description
Object purpose	To make tool-action liability boundary explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	scope boundary, risk class, evidence requirement, privacy treatment, review/approval rule, revocation or closure rule
Failure if missing	The system may appear governed at UI or workflow level while lacking enforceable tool-action liability boundary semantics.
Audit question	Can an independent reviewer determine whether tool-action liability boundary was properly established and preserved for the relevant life-cycle phase?

Table T-06-06: MRO-05 Object Card — Tool-Action Liability Boundary

Detailed Discussion

Tool use is the moment where AI output becomes external consequence. Sending email, changing records, deploying code, querying regulated data, or triggering payment workflows each carries different liability and reversibility properties.

A tool call is not only an execution event. It is the point where AI output can affect external systems, records, customers, code, money, permissions, communications, or regulated data. The governance object must bind action, affected system, authority source, reversibility, rollback path, data sensitivity, evidence pointer, and accountable owner.

The core question for MRO-05 is whether the system can reconstruct the liability boundary for each consequential tool action. Strong support requires recording initiator, tool identity, external system, authority credential, reversibility status, rollback plan, affected data categories, and accountable owner. The same tool may be low-risk in one context and high-impact in another.

The difficulty is that a tool log alone may show that an action happened, but not whether the agent had authority, whether the action was reversible, which system was affected, or who owns the resulting consequence.

Enterprise Implication

Without liability boundaries, enterprises face disputes when external actions cause customer impact, regulatory violations, or financial loss. The organization cannot prove who authorized the action, whether it was within delegated scope, or whether rollback procedures were available.

Adjacent controls such as IAM and API permissions may support this object, but lifecycle conformance requires explicit object semantics that bind the specific tool action to authority, reversibility, affected systems, and accountable ownership.

Design Implication

Every consequential tool action should produce a liability boundary record: initiator, authority source, target system, action type, reversibility, rollback path, data sensitivity, evidence pointer, and accountable owner.

The implementation test is to select a high-impact tool action and ask: who initiated it, under what authority, affecting which system, with what reversibility, and who owns the consequence? If the answer requires tribal memory or manual investigation, the liability boundary is not audit-grade.

Audit Question

For any consequential tool action, can the system reconstruct the initiator, authority source, affected system, reversibility status, rollback path, and accountable owner?

6.7 MRO-06 — Responsibility Transfer Across Agents

Problem

Handoff in a workflow engine does not automatically transfer responsibility, inherited constraints, retained obligations, or evidence duties.

Required Object

A transfer object declaring transferred scope, source role, receiving role, retained responsibilities, inherited constraints, and receiving-role acceptance.

Judgment

Handoff without explicit responsibility transfer is just routing.

Object Card

Field	Description
Object purpose	To make responsibility transfer across agents explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	scope boundary, risk class, evidence requirement, privacy treatment, review/approval rule, revocation or closure rule
Failure if missing	The system may appear governed at UI or workflow level while lacking enforceable responsibility transfer across agents semantics.
Audit question	Can an independent reviewer determine whether responsibility transfer across agents was properly established and preserved for the relevant lifecycle phase?

Table T-06-07: MRO-06 Object Card — Responsibility Transfer Across Agents

Detailed Discussion

Handoff is an execution event; responsibility transfer is a governance event. When one agent delegates to another, the receiving agent must inherit constraints, evidence duties, privacy restrictions, and escalation conditions rather than merely receiving a prompt or task payload.

The core question for MRO-06 is whether the system distinguishes workflow routing from responsibility transfer. Strong support requires explicit transfer records declaring transferred scope, retained scope, inherited constraints, evidence obligations, receiving-role acceptance, and rejection/escalation behavior.

The difficulty is that workflow engines naturally optimize for task routing, while governance requires explicit responsibility semantics. A downstream agent can claim it only followed instructions while the upstream agent no longer controls execution, leaving a gap in audit and remediation.

Enterprise Implication

Without explicit transfer semantics, multi-agent workflows create accountability gaps. When disputes arise, the organization cannot prove which agent held which responsibility at which lifecycle point. This weakens internal audit, customer assurance, and regulatory defense.

Adjacent orchestration controls may support this object, but lifecycle conformance requires explicit object semantics for transferred responsibility, retained responsibility, inherited constraints, evidence duties, and escalation.

Design Implication

Responsibility transfer should be explicit: transferred scope, retained scope, inherited constraints, evidence obligations, receiving-role acceptance, and rejection/escalation behavior must be recorded.

The implementation test is to select a multi-agent workflow and ask: at each handoff, what responsibility was transferred, what was retained, what constraints were inherited, and did the receiving agent accept? If the answer is that handoff equals transfer, the system has collapsed execution into governance.

Audit Question

For any multi-agent workflow, can the system reconstruct which agent held which responsibility at each handoff, including transferred scope, retained scope, and inherited constraints?

6.8 MRO-07 — Authority Drift

Problem

Agents can drift from advising to executing, drafting to sending, reading to modifying, or summarizing to committing changes.

Required Object

A drift event model comparing observed behavior against original delegated authority, risk class, tool scope, and required confirmation boundary.

Judgment

The most dangerous agent failures are often unauthorized authority transitions.

Object Card

Field	Description
Object purpose	To make authority drift explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	scope boundary, risk class, evidence requirement, privacy treatment, review/approval rule, revocation or closure rule
Failure if missing	The system may appear governed at UI or workflow level while lacking enforceable authority drift semantics.
Audit question	Can an independent reviewer determine whether authority drift was properly established and preserved for the relevant lifecycle phase?

Table T-06-08: MRO-07 Object Card — Authority Drift**Detailed Discussion**

Authority drift occurs when an agent gradually moves from advice to execution, from draft to send, from read-only to write, or from internal recommendation to external commitment. It is often a transition problem rather than a single bad output.

The core question for MRO-07 is whether the system monitors for unauthorized authority transitions. Strong support requires comparing observed behavior against original authority boundaries and plan state. Unauthorized transitions should trigger stop, downgrade, or human confirmation rather than being treated as ordinary execution variance.

The difficulty is that drift is dangerous because each local step may appear reasonable while the cumulative lifecycle state exceeds the original delegation. This is especially acute in long-running workflows, autonomous retries, or agent loops.

Enterprise Implication

Without drift detection, agents can silently escalate from low-risk to high-impact actions. The organization may discover the problem only after customer impact, regulatory violation, or financial loss. Drift detection is a lifecycle monitoring requirement, not merely a model safety feature.

Adjacent guardrails and content filters may support this object, but lifecycle conformance requires explicit object semantics that monitor authority boundaries across the full execution timeline.

Design Implication

The system should compare observed behavior against original authority boundaries and plan state. Unauthorized transitions should trigger stop, downgrade, or human confirmation rather than being treated as ordinary execution variance.

The implementation test is to simulate a long-running workflow where an agent gradually escalates from read-only to write, or from draft to send. Can the system detect the transition and halt execution before external consequence? If not, authority drift is unmonitored.

Audit Question

For any long-running workflow, can the system detect when an agent's observed behavior exceeds its original authority boundary, and does it halt or escalate before external consequence?

6.9 MRO-08 — MAS Evidence Partitioning

Problem

A flat chronological log cannot show how different agents, tools, humans, plans, privacy constraints, and evidence fragments relate.

Required Object

Partitioned evidence by agent, tool, human confirmation, plan version, authority boundary, privacy boundary, and accepted outcome.

Judgment

In MAS, evidence must be partitioned, linked, and reconstructable.

Object Card

Field	Description
Object purpose	To make MAS evidence partitioning explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	scope boundary, risk class, evidence requirement, privacy treatment, review/approval rule, revocation or closure rule
Failure if missing	The system may appear governed at UI or workflow level while lacking enforceable MAS evidence partitioning semantics.
Audit question	Can an independent reviewer determine whether MAS evidence partitioning was properly established and preserved for the relevant lifecycle phase?

Table T-06-09: MRO-08 Object Card — MAS Evidence Partitioning

Detailed Discussion

Multi-agent evidence is not a single log stream. It includes plan versions, agent messages, tool calls, human confirmations, data accesses, privacy decisions, model outputs, evidence hashes, and outcome states. Different auditors may need different partitions.

The core question for MRO-08 is whether evidence can be partitioned, linked, and reconstructed for different audit purposes. Strong support requires partitioning by agent, tool, authority boundary, privacy class, confirmation event, plan version, and accepted outcome. Partitions should be linked by stable identifiers and integrity hashes.

The difficulty is that flat logs are difficult to use in disputes because they mix irrelevant telemetry with critical proof. They also make selective disclosure and privacy minimization harder because sensitive payloads and governance metadata are not separated.

Enterprise Implication

Without evidence partitioning, multi-agent workflows become difficult to audit, dispute, or selectively disclose. The organization cannot prove which agent performed which action under which authority without exposing unnecessary sensitive data. This weakens regulatory defense, customer assurance, and external validation.

Adjacent observability and trace systems may support this object, but lifecycle conformance requires explicit object semantics that partition evidence by governance boundaries rather than only by execution timeline.

Design Implication

Evidence should be partitioned by agent, tool, authority boundary, privacy class, confirmation event, plan version, and accepted outcome. Partitions should be linked by stable identifiers and integrity hashes.

The implementation test is to select a completed multi-agent workflow and ask: can the system produce evidence for a specific agent's actions, a specific tool's usage, a specific privacy class, or a specific accepted outcome without exposing the full raw log? If not, evidence partitioning is not audit-grade.

Audit Question

For any completed multi-agent workflow, can the system reconstruct evidence partitions for specific agents, tools, privacy classes, and accepted outcomes without exposing unnecessary raw data?

6.10 MRO-09 — Cross-Project Reuse Compliance

Problem

Agent workflows are increasingly reused across projects, but reuse without revalidation creates compliance drift. A workflow validated for one context, risk class, or privacy boundary may be deployed in a different context without resetting authority, evidence, or privacy constraints.

Required Object

A cross-project reuse compliance record linking original validation context to new deployment context, including authority reset, privacy review, evidence revalidation, and acceptance criteria adjustment.

Judgment

Reuse without revalidation is the most common lifecycle governance failure.

Object Card

Field	Description
Object purpose	To make cross-project reuse compliance explicit, inspectable, and re-playable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, relat-ed_human_role_id, related_agent_role_id where applicable, evidence_-pointer, status, created_at, updated_at
Required controls	original validation record, receiving project context, reuse authoriza-tion, context delta, privacy review, evidence revalidation, acceptance criteria adjustment
Failure if missing	A workflow may be reused in a new project, risk class, or privacy con-text without authority reset, evidence revalidation, or privacy reassessment.
Audit question	Can an independent reviewer determine whether cross-project reuse was authorized, revalidated, and privacy-reviewed for the new deploy-ment context?

Table T-06-10: MRO-09 Object Card — Cross-Project Reuse Compliance

Detailed Discussion

Reuse is a powerful efficiency mechanism, but it is also a compliance boundary. A workflow validated for internal analytics may be reused for customer-facing decisions without recognizing that the risk class, privacy obligations, and evidence requirements have changed. The original validation context does not automatically transfer.

The governance challenge is that reuse often happens through copy-paste, template libraries, or shared agent definitions. These mechanisms optimize for speed but do not enforce context reset. A reused workflow may carry forward authority assumptions, privacy treatments, or evidence obligations that no longer apply.

Cross-project reuse requires explicit revalidation. The reuse record must link the original validation context to the new deployment context and document what changed: authority scope, risk class, privacy boundary, data

sources, tool permissions, human oversight requirements, and acceptance criteria. If the new context is materially different, the workflow must be revalidated rather than merely copied.

The difficulty is that reuse is often invisible to governance systems. A developer may copy a workflow definition, adjust a few parameters, and deploy it in a new project without triggering any compliance review. The organization discovers the problem only when a dispute arises and the reused workflow cannot prove that it was authorized for the new context.

Enterprise Implication

Without reuse compliance, organizations face hidden governance debt. A workflow validated once may be reused dozens of times across different projects, risk classes, and privacy contexts without any record of revalidation. This creates audit gaps, privacy violations, and liability exposure.

Adjacent template libraries and shared agent definitions may support this object, but lifecycle conformance requires explicit object semantics for context reset, privacy review, and evidence revalidation when deployment context changes.

Design Implication

The reuse record should bind original workflow ID, original validation record, receiving project ID, reuse authorization, context delta analysis, privacy review, evidence revalidation, and acceptance criteria adjustment. Reuse should trigger a compliance checkpoint rather than being treated as a copy operation.

The implementation test is to select a reused workflow and ask: was it revalidated for the new context, was the privacy boundary reassessed, were authority and evidence requirements reset, and who approved the reuse? If the answer is that reuse was automatic, the system has no reuse compliance boundary.

Audit Question

For any reused workflow, can the system prove that it was revalidated for the new deployment context, including authority reset, privacy review, and evidence revalidation?

6.11 MRO-10 — Privacy / GDPR Lifecycle Mapping

Problem

GDPR applies to the full lifecycle, but most AI governance focuses on model training data. Agentic systems retrieve, process, store, share, and delete personal data across agents, tools, memory systems, and handoffs. Privacy compliance requires lifecycle data-flow mapping, not only model-level controls.

Required Object

A lifecycle data-flow map showing what personal data enters, flows through, and exits each agent, tool, memory, and handoff, including legal basis, purpose limitation, retention period, and data subject rights.

Judgment

Privacy compliance in MAS requires lifecycle data-flow mapping, not only model-level controls.

Object Card

Field	Description
Object purpose	To make privacy / GDPR lifecycle mapping explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	data-flow map, legal-basis reference, purpose-limitation record, retention policy, data subject rights workflow, privacy impact assessment reference
Failure if missing	Personal data may move through agents, tools, memory, and handoffs without lifecycle-level evidence of purpose, retention, rights handling, or privacy treatment.
Audit question	Can an independent reviewer reconstruct what personal data entered, flowed through, and exited each agent, tool, and memory system, and verify that legal basis, purpose limitation, and retention policies were enforced?

Table T-06-11: MRO-10 Object Card — Privacy / GDPR Lifecycle Mapping

Detailed Discussion

Privacy compliance in agentic systems is not limited to model training data. Agents retrieve customer records, process personal data in tool calls, store conversation history in memory systems, share data across agent handoffs, and delete data in response to data subject requests. Each of these operations is a privacy event.

The governance challenge is that privacy controls are often applied at the model or database layer but not at the agent lifecycle layer. An agent may retrieve personal data from a compliant database, process it through multiple tools, store it in a memory system, and share it with another agent without any lifecycle-level privacy tracking. The organization cannot prove what personal data was accessed, for what purpose, under what legal basis, or how long it was retained.

GDPR requires data controllers to document processing activities, enforce purpose limitation, implement retention policies, and honor data subject rights. In a multi-agent system, this requires a lifecycle data-flow map that shows what personal data enters each agent, how it flows through tools and memory, where it is stored, when it is deleted, and how data subject rights are implemented.

The difficulty is that agentic systems are dynamic. Data flows change based on runtime decisions, tool availability, and agent handoffs. A static data-flow diagram is insufficient. The map must be reconstructable from lifecycle evidence, showing actual data flows rather than only intended flows.

Enterprise Implication

Without lifecycle data-flow mapping, organizations cannot prove GDPR compliance for agentic systems. A data subject access request may require reconstructing what personal data was processed by which agents, for what purpose, under what legal basis, and where it is now stored. If the organization cannot answer these questions, it faces regulatory penalties and reputational damage.

Adjacent privacy dashboards and data governance systems may support this object, but lifecycle conformance requires explicit object semantics that track personal data through the full agent lifecycle, not only model training or database layers.

Design Implication

The data-flow map should be reconstructable from lifecycle evidence. Each agent, tool, memory system, and handoff should record what personal data was accessed, for what purpose, under what legal basis, how long it

was retained, and how data subject rights were implemented. The map should support data subject access requests, rectification, erasure, and portability.

The implementation test is to simulate a data subject access request and ask: can the system reconstruct what personal data was processed by which agents, for what purpose, under what legal basis, and where it is now stored? If the answer requires manual investigation, the lifecycle data-flow map is not audit-grade.

Audit Question

For any completed workflow involving personal data, can the system reconstruct what personal data entered, flowed through, and exited each agent, tool, and memory system, and verify that legal basis, purpose limitation, and retention policies were enforced?

6.12 MRO-11 — Privacy-Preserving Third-Party Validation

Problem

Enterprise buyers want independent validation, but full evidence disclosure violates privacy and trade secrets. Third-party validators need to verify compliance without accessing raw sensitive data, model weights, or proprietary workflows.

Required Object

A privacy-preserving validation protocol allowing selective disclosure and zero-knowledge proofs, enabling validators to verify compliance properties without accessing raw evidence.

Judgment

Third-party validation must not require full evidence disclosure.

Object Card

Field	Description
Object purpose	To make privacy-preserving third-party validation explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	validation scope definition, selective disclosure policy, redaction profile, proof or hash manifest, evidence-owner authorization, validator access boundary
Failure if missing	External validation may require raw sensitive evidence access, creating privacy, trade-secret, and over-disclosure risk.
Audit question	Can an independent validator verify compliance properties without accessing raw sensitive data, and can the evidence owner prove that only authorized evidence was disclosed?

Table T-06-12: MRO-11 Object Card — Privacy-Preserving Third-Party Validation

Detailed Discussion

Third-party validation is a trust mechanism. Enterprise buyers, insurers, and regulators want independent verification that an agentic system meets compliance requirements. But full evidence disclosure creates

privacy and trade secret risks. A validator who accesses raw customer data, model weights, or proprietary workflows becomes a privacy risk and a competitive threat.

The governance challenge is that traditional audit models assume full evidence access. A financial auditor expects to see transaction records. A security auditor expects to see system logs. But in agentic systems, full evidence access may violate GDPR, expose trade secrets, or create insider threat risks.

Privacy-preserving validation requires a different model. The evidence owner must be able to prove compliance properties without disclosing raw evidence. This can be achieved through selective disclosure, where only governance metadata is shared, or through zero-knowledge proofs, where the validator can verify a property without seeing the underlying data.

The difficulty is that privacy-preserving validation is technically complex. Zero-knowledge proofs require cryptographic expertise. Selective disclosure requires careful evidence partitioning. Validators may resist these approaches because they reduce audit confidence. The market needs validation protocols that balance privacy preservation with audit rigor.

Enterprise Implication

Without privacy-preserving validation, enterprises face a dilemma: accept unvalidated systems or disclose sensitive data to validators. This slows adoption, increases risk, and creates vendor lock-in because switching vendors requires re-disclosing evidence.

Adjacent validation programs may support this object, but lifecycle conformance requires explicit object semantics for selective disclosure and independent verification without unnecessary raw data access.

Design Implication

The validation protocol should support selective disclosure and zero-knowledge proofs. Evidence should be partitioned so that governance metadata can be shared without exposing raw payloads. Validators should be able to verify compliance properties such as authority boundaries, evidence completeness, and privacy controls without accessing customer data or model weights.

The implementation test is to simulate a third-party validation and ask: can the validator verify compliance properties without accessing raw sensitive data, and can the evidence owner prove that only authorized evidence was disclosed? If the answer is that full evidence access is required, the validation protocol is not privacy-preserving.

Audit Question

Can an independent validator verify compliance properties without accessing raw sensitive data, and can the evidence owner prove that only authorized evidence was disclosed?

6.13 MRO-12 — Evidence Minimization and Selective Disclosure

Problem

Lifecycle governance requires evidence retention, but GDPR requires data minimization. Retaining full execution logs, conversation history, and tool payloads creates privacy risk. Deleting evidence creates accountability gaps.

Required Object

An evidence minimization and selective disclosure policy defining retention periods, redaction rules, and disclosure boundaries for different evidence classes.

Judgment

Evidence retention must balance auditability with privacy minimization.

Object Card

Field	Description
Object purpose	To make evidence minimization and selective disclosure explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	retention policy, redaction rules, disclosure boundaries, deletion schedule, audit trail, evidence class handling
Failure if missing	Evidence retention may become a privacy risk, while evidence deletion may create accountability gaps.
Audit question	Can an independent reviewer verify that evidence retention was minimized to what is necessary for accountability, and that disclosure was limited to authorized parties and purposes?

Table T-06-13: MRO-12 Object Card — Evidence Minimization and Selective Disclosure

Detailed Discussion

Evidence retention is a governance requirement. Auditors, regulators, and dispute resolution processes need evidence to reconstruct what happened. But GDPR requires data minimization: personal data should be retained only as long as necessary for the specified purpose. These requirements are in tension.

The governance challenge is that agentic systems generate large volumes of evidence: execution logs, conversation history, tool payloads, model outputs, memory snapshots, and handoff records. Retaining all of this evidence creates privacy risk. Deleting it creates accountability gaps. The organization must balance auditability with privacy minimization.

Evidence minimization requires a retention policy that distinguishes different evidence classes. Governance metadata such as authority boundaries, acceptance decisions, and responsibility transfers may need long retention periods. Raw payloads such as customer messages, tool responses, and memory snapshots may need short retention periods or immediate redaction. The policy must define what is retained, for how long, in what form, and for what purpose.

Selective disclosure requires disclosure boundaries. Different auditors may need different evidence partitions. A privacy regulator may need data-flow evidence but not model weights. A financial auditor may need transaction evidence but not conversation history. A customer dispute may require specific evidence without exposing unrelated data. The system must support selective disclosure without requiring full evidence access.

Enterprise Implication

Without evidence minimization, organizations face privacy violations and regulatory penalties. Retaining full execution logs indefinitely violates GDPR data minimization principles. Without selective disclosure, organizations face dilemmas: deny audit requests or expose unnecessary sensitive data.

Adjacent observability and evidence management systems may support this object, but lifecycle conformance requires explicit object semantics that balance auditability with privacy minimization and selective disclosure.

Design Implication

The evidence policy should define retention periods, redaction rules, and disclosure boundaries for different evidence classes. Governance metadata should be retained longer than raw payloads. Personal data should be redacted or pseudonymized unless required for accountability. Disclosure should be partitioned by audit purpose.

The implementation test is to simulate a data subject erasure request and ask: can the system delete personal data while preserving accountability evidence, and can it prove that only necessary evidence was retained? If the answer is that all evidence must be retained or all evidence must be deleted, the policy does not balance auditability with privacy minimization.

Audit Question

For any evidence retention decision, can the system prove that evidence was minimized to what is necessary for accountability, and that disclosure was limited to authorized parties and purposes?

6.14 MRO-13 — Data Subject Rights vs Evidence Retention

Problem

GDPR grants data subjects the right to erasure, but lifecycle governance requires evidence retention for audit, dispute resolution, and regulatory defense. These requirements are in structural tension in agentic systems where evidence chains must be preserved to prove accountability.

Required Object

A data subject rights vs evidence retention policy defining retention periods, anonymization rules, legal basis for retention, and reconciliation procedures when erasure requests conflict with evidence obligations.

Judgment

Data subject rights and evidence retention are in structural tension in agentic systems.

Object Card

Field	Description
Object purpose	To make data subject rights versus evidence retention handling explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	retention policy, legal-basis documentation, anonymization procedures, erasure request log, retention override justification, escalation path
Failure if missing	The organization may either lose accountability evidence or over-retain personal data without a documented reconciliation process.
Audit question	Can an independent reviewer verify that evidence retention was limited to what is legally necessary for accountability, and that data subject rights were honored except where legal retention obligations apply?

Table T-06-14: MRO-13 Object Card — Data Subject Rights vs Evidence Retention

Detailed Discussion

The tension between data subject rights and evidence retention is fundamental. GDPR Article 17 grants individuals the right to erasure when personal data is no longer necessary for the original purpose. But lifecycle governance requires retaining evidence to prove that agentic work was authorized, executed under proper authority, reviewed, and accepted. If a customer later disputes an outcome, the organization needs evidence to reconstruct what happened.

This tension becomes acute in agentic systems because evidence chains are distributed across agents, tools, memory systems, and handoffs. A data subject erasure request may require deleting conversation history, tool payloads, and memory snapshots while preserving governance metadata such as authority boundaries, acceptance decisions, and responsibility transfers. The challenge is determining what can be deleted without breaking the accountability chain.

The policy must define retention periods for different evidence classes. Raw payloads containing personal data may have short retention periods. Governance metadata may have longer retention periods based on legal, regulatory, or contractual obligations. When a data subject requests erasure, the system must determine whether the request can be honored immediately, whether evidence can be anonymized instead of deleted, or whether retention is legally required.

The difficulty is that legal retention obligations vary by jurisdiction, industry, and context. Financial services may require seven-year retention. Healthcare may require longer. Employment records may have different rules. The policy must map these obligations to evidence classes and provide clear procedures for handling erasure requests that conflict with retention requirements.

Enterprise Implication

Without a clear policy, organizations face impossible choices: honor erasure requests and lose accountability evidence, or deny erasure requests and violate GDPR. Both paths create legal risk. The organization needs a documented policy that defines when evidence retention overrides erasure rights, what legal basis applies, and how to minimize retained personal data.

Adjacent evidence retention controls may support this object, but lifecycle conformance requires explicit object semantics that distinguish governance metadata from raw payloads and reconcile data subject requests with accountability needs.

Design Implication

The policy should define retention periods by evidence class, legal basis for retention, anonymization procedures, and escalation paths for erasure requests that conflict with retention obligations. The system should support selective erasure: deleting raw payloads while preserving anonymized governance metadata.

The implementation test is to simulate a data subject erasure request and ask: can the system delete personal data while preserving accountability evidence, can it document the legal basis for any retained evidence, and can it prove that retention was minimized to what is legally necessary? If the answer is that all evidence must be retained or all evidence must be deleted, the policy does not resolve the tension.

Audit Question

For any data subject erasure request, can the system prove that personal data was deleted except where legal retention obligations apply, and that retained evidence was minimized and anonymized where possible?

6.15 MRO-14 — Third-Party Processor / Subprocessor Chain

Problem

GDPR requires data controllers to maintain processor agreements and notify data subjects of subprocessors. In multi-agent systems, the processor chain is dynamic: agents call tools, tools call APIs, APIs invoke third-party services, and the chain changes based on runtime decisions. Static processor lists cannot capture this dynamic reality.

Required Object

A dynamic processor chain record tracking which processors and subprocessors handled which personal data at which lifecycle phase, including processor identity, data categories, processing purpose, legal basis, and data flow timestamps.

Judgment

In MAS, the processor chain is dynamic and must be tracked at lifecycle granularity.

Object Card

Field	Description
Object purpose	To make third-party processor / subprocessor chain handling explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	processor identity, subprocessor identity, data categories, processing purpose, legal-basis reference, processor agreement reference, data-flow timestamps
Failure if missing	Static vendor or processor lists may fail to show which third parties processed which personal data during actual agentic execution.
Audit question	Can an independent reviewer reconstruct which processors and subprocessors handled which personal data at which lifecycle phase, and verify that processor agreements and legal basis were in place?

Table T-06-15: MRO-14 Object Card — Third-Party Processor / Subprocessor Chain

Detailed Discussion

GDPR Article 28 requires data controllers to use only processors that provide sufficient guarantees of compliance, to maintain written processor agreements, and to notify data subjects of subprocessors. This works well for static systems where the processor list is known in advance. But in multi-agent systems, the processor chain is dynamic and context-dependent.

An agent may retrieve customer data from an internal database, send it to a third-party API for enrichment, store the result in a cloud memory service, and pass it to another agent that uses a different model provider. Each of these steps may involve a different processor or subprocessor. The chain changes based on which tools are available, which models are selected, and which agents are invoked.

The governance challenge is that static processor lists become outdated immediately. A data subject access request may ask: which third parties processed my data? The organization cannot answer without reconstructing the actual processor chain from lifecycle evidence. The record must show which processors handled which data categories at which lifecycle phase, under which legal basis, and pursuant to which processor agreement.

The difficulty is that processor chains can be long and nested. A tool may call an API that invokes a subprocessor that uses another subprocessor. The organization may not have direct visibility into nested subprocessor relationships unless the primary processor discloses them. The record must capture what is known and flag what is unknown or unverified.

Enterprise Implication

Without dynamic processor chain tracking, organizations cannot answer data subject access requests, cannot verify processor compliance, and cannot detect unauthorized subprocessor usage. This creates GDPR violation risk and weakens vendor management.

Adjacent vendor management systems may support this object, but lifecycle conformance requires explicit object semantics that track processor and subprocessor usage at lifecycle granularity, not only at configuration time.

Design Implication

The processor chain record should capture processor identity, subprocessor identity, data categories, processing purpose, legal basis, processor agreement reference, and data flow timestamps for each lifecycle phase. The record should be reconstructable from tool call evidence and should support data subject access requests.

The implementation test is to select a completed workflow involving personal data and ask: which processors and subprocessors handled which data categories, under which legal basis, and pursuant to which processor agreements? If the answer requires manual investigation or vendor inquiry, the processor chain record is not audit-grade.

Audit Question

For any completed workflow involving personal data, can the system reconstruct which processors and subprocessors handled which data categories at which lifecycle phase, and verify that processor agreements and legal basis were in place?

6.16 MRO-15 — Vendor / Model / Runtime Substitution Conformance

Problem

Enterprises want vendor optionality and the ability to substitute models, agent runtimes, or tool vendors without revalidation. But substitution without conformance verification breaks lifecycle governance. A workflow validated with one model may behave differently with another model. A tool validated with one vendor may have different authority boundaries with another vendor.

Required Object

A vendor substitution conformance record proving that substitution preserved lifecycle responsibility properties, including authority boundaries, evidence requirements, privacy constraints, and acceptance criteria.

Judgment

Vendor substitution without revalidation breaks lifecycle conformance.

Object Card

Field	Description
Object purpose	To make vendor / model / runtime substitution conformance explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	original validation record, substitution authorization, delta analysis, conformance check record, privacy boundary review, revalidation decision
Failure if missing	Vendor, model, or runtime substitution may create compliance drift while appearing to be only a configuration change.
Audit question	Can an independent reviewer verify that vendor substitution preserved lifecycle responsibility properties, and that conformance was revalidated before deployment?

Table T-06-16: MRO-15 Object Card — Vendor / Model / Runtime Substitution Conformance

Detailed Discussion

Vendor neutrality is a strategic goal for many enterprises. Organizations want to avoid vendor lock-in, negotiate better pricing, and maintain optionality. But vendor neutrality is not real unless compliance survives substitution. Swapping one model for another, one agent runtime for another, or one tool vendor for another can change behavior in ways that break lifecycle conformance.

A model substitution may change output quality, reasoning patterns, or tool-calling behavior. A runtime substitution may change how authority boundaries are enforced, how evidence is collected, or how privacy constraints are applied. A tool vendor substitution may change API semantics, error handling, or reversibility properties. These changes can be subtle but consequential.

The governance challenge is that substitution is often treated as a configuration change rather than a compliance event. A developer may swap model providers, update an API endpoint, or switch agent frameworks without triggering any revalidation. The organization discovers the problem only when a dispute arises and the substituted system cannot prove that it preserved the original lifecycle responsibility properties.

The conformance record must document what was substituted, why, what changed, what was tested, and whether the substitution preserved authority boundaries, evidence requirements, privacy constraints, and acceptance criteria. If the substitution introduces material changes, the workflow must be revalidated rather than merely reconfigured.

Enterprise Implication

Without substitution conformance, vendor optionality becomes compliance drift. The organization may believe it has validated workflows when in fact it has validated only the original vendor configuration. Substitution without revalidation creates hidden governance debt that surfaces during audits or disputes.

Adjacent vendor-neutral architectures may support this object, but lifecycle conformance requires explicit object semantics proving that substitution preserved responsibility properties rather than merely preserving technical capability.

Design Implication

The substitution record should bind original workflow ID, original validation record, substitution authorization, conformance test results, delta analysis, and revalidation decision. Substitution should trigger a conformance checkpoint rather than being treated as a transparent configuration change.

The implementation test is to simulate a vendor substitution and ask: was conformance revalidated, were lifecycle responsibility properties preserved, and can the system prove that the substituted configuration meets the same governance requirements as the original? If the answer is that substitution was automatic, the system has no substitution conformance boundary.

Audit Question

For any vendor substitution, can the system prove that lifecycle responsibility properties were preserved, and that conformance was revalidated before deployment?

6.17 MRO-16 — Incident, Dispute, and Remediation Closure

Problem

Most AI governance focuses on prevention: risk assessment, guardrails, human oversight, and monitoring. But lifecycle governance is incomplete without incident detection, dispute resolution, remediation, and closure. When an agentic system produces an incorrect outcome, violates a policy, or causes customer harm, the organization needs a structured process to detect the incident, investigate root cause, remediate the problem, close the dispute, and implement continuous improvement.

Required Object

An incident, dispute, and remediation closure record linking incident detection to root cause analysis, remediation action, dispute resolution, acceptance of closure, and continuous improvement measures.

Judgment

Lifecycle governance is incomplete without incident closure and continuous improvement.

Object Card

Field	Description
Object purpose	To make incident, dispute, and remediation closure explicit, inspectable, and replayable within an agentic lifecycle.
Minimum identity fields	object_id, object_type, lifecycle_id, project_id, object_version, related_human_role_id, related_agent_role_id where applicable, evidence_pointer, status, created_at, updated_at
Required controls	incident detection record, root cause analysis, remediation action, dispute resolution decision, closure acceptance, continuous improvement plan, privacy-limited evidence disclosure
Failure if missing	Incidents may be detected or handled manually without evidence-linked remediation, dispute closure, or continuous improvement.
Audit question	Can an independent reviewer verify that incidents were detected, investigated, remediated, and closed with documented root cause and continuous improvement measures?

Table T-06-17: MRO-16 Object Card — Incident, Dispute, and Remediation Closure

Detailed Discussion

Incident closure is where lifecycle governance proves its value. Prevention is important, but no system is perfect. Agents will make mistakes, policies will be violated, and customers will be harmed. The question is whether the organization can detect incidents quickly, investigate root cause accurately, remediate effectively, resolve disputes fairly, and improve continuously.

An incident is any event where an agentic system produces an outcome that violates policy, exceeds authority, causes harm, or fails to meet acceptance criteria. A dispute is any challenge to an outcome by a customer, regulator, or internal stakeholder. Remediation is the action taken to correct the problem. Closure is the state where the incident is resolved, the dispute is settled, and the organization has implemented measures to prevent recurrence.

The governance challenge is that incident management is often ad hoc. A customer complaint may trigger an investigation, but the investigation may not link back to the original lifecycle evidence. The root cause may not be documented. The remediation may not be verified. The dispute may be settled without any continuous improvement. The organization learns nothing from the incident.

The closure record must link incident detection to lifecycle evidence, root cause analysis, remediation action, dispute resolution decision, closure acceptance, and continuous improvement plan. The record must be auditable: an independent reviewer should be able to verify that the incident was handled properly and that the organization learned from it.

Enterprise Implication

Without incident closure, organizations cannot prove that they handle agentic failures responsibly. Customers lose trust. Regulators question accountability. Insurers raise premiums or deny coverage. The organization cannot demonstrate continuous improvement because it has no structured process for learning from incidents.

Adjacent monitoring systems may support this object, but lifecycle conformance requires explicit object semantics that link incidents to remediation, dispute closure, and continuous improvement.

Design Implication

The closure record should bind incident ID, detection timestamp, lifecycle evidence pointers, root cause analysis, remediation action, dispute resolution decision, closure acceptance, and continuous improvement plan. Incidents should remain open until remediation is verified and closure is accepted by the accountable owner.

The implementation test is to simulate an incident and ask: can the system detect it, link it to lifecycle evidence, document root cause, track remediation, resolve disputes, and close with continuous improvement measures? If the answer is that incidents are handled manually without structured records, the system has no incident closure capability.

Audit Question

For any incident, can the system verify that it was detected, investigated, remediated, and closed with documented root cause and continuous improvement measures?

6.18 Chapter Summary

The sixteen Missing Regulatory Objects defined in this chapter represent the missing object layer between regulatory abstractions and agentic lifecycle implementation. They are not proposed statutes or legal mandates. They are engineering and assurance objects that make regulatory concepts testable in multi-agent execution.

Existing regulation and governance frameworks increasingly require human oversight, logging, documentation, monitoring, accountability, transparency, and privacy controls. But they do not yet fully specify the lifecycle objects required to prove those properties inside dynamic agentic workflows. The sixteen MROs bridge that gap. They turn regulatory abstractions into lifecycle objects: human oversight becomes human-role-to-MAS

responsibility mapping, logging becomes partitioned evidence chain, accountability becomes tool-action liability and responsibility transfer, transparency becomes evidence-linked review and accepted outcome, monitoring becomes lifecycle drift and incident closure, privacy becomes lifecycle data-flow mapping and selective disclosure.

These objects are necessary because agentic AI is not only model output. It is organized work that moves from intent to accepted outcome through delegation, execution, evidence collection, review, acceptance, dispute, remediation, and reuse. If that work cannot be proven under controlled authority, evidence, privacy, and remediation constraints, it remains difficult to audit, insure, delegate, reuse, and scale. This is why AI Agent Lifecycle Governance is not a philosophical category. It is an operational prerequisite for agentic AI becoming trusted enterprise infrastructure.

The next chapters introduce the three-profile analytical model that measures how systems map to these objects. RCCS-T measures traditional regulatory compliance coverage. MROs provide the adjustment layer that asks whether that coverage can be represented as lifecycle responsibility objects. RCCS-M measures MRO-adjusted regulatory coverage. ALCS measures lifecycle conformance maturity.

Conceptually:

Technical example
$\text{RCCS-T} + \text{MRO Coverage} = \text{RCCS-M}$

This is an analytical model, not a legal formula, certification, regulator-approved scoring method, vendor ranking, or procurement recommendation. Its purpose is to make the transition visible: regulatory language names the governance expectation; MROs define the lifecycle responsibility object; RCCS-M evaluates whether the traditional regulatory surface can be expressed through that object layer.

Chapter 7: Regulatory Compliance Coverage Score (RCCS)

MAIN RESULT RCCS measures traditional regulatory coverage separately from lifecycle conformance.	METHOD / LENS RCCS dimensions, rubrics, and evidence multipliers.
EVIDENCE BASIS Separating RCCS prevents model governance strength from being mistaken for agentic lifecycle assurance.	KEY TAKEAWAY / BOUNDARY RCCS is not a legal compliance score and does not prove implementation.

Version: v0.3.2-FRC-R3

7.0 Chapter Overview

This chapter defines the two RCCS scoring lenses used in this white paper:

- **RCCS-T — Traditional Regulatory Compliance Coverage Score** measures how strongly a system covers existing regulatory and governance requirements such as risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring.
- **RCCS-M — MRO-Adjusted Regulatory Compliance Coverage Score** measures whether those same regulatory obligations can be represented as Missing Regulatory Objects and lifecycle responsibility semantics.

RCCS-T and RCCS-M use the **same 10 dimension names and weights**. They differ in scoring lens. RCCS-T asks whether the system provides traditional governance and control coverage. RCCS-M asks whether the same regulatory obligation is expressed as inspectable, replayable, evidence-linked lifecycle responsibility objects.

RCCS is an analytical framework for comparing system capabilities against regulatory ideals. It is not a legal compliance certification, not a conformity assessment, and not a regulatory approval. It is a structured method for evaluating whether a system provides the primitives, controls, evidence mechanisms, and lifecycle objects that existing regulation and agentic/MAS governance increasingly require.

This chapter defines the ten RCCS dimensions, the RCCS-M adjustment test, the separate 0-5 scoring lens for RCCS-T and RCCS-M, evidence requirements for each lens, and the shared calculation method. Compact dimension rubrics begin in section 7.11; Appendix E provides the detailed RCCS-T baseline rubrics and RCCS-M adjustment matrix.

7.1 Why Traditional RCCS Is Necessary but Insufficient

RCCS-T measures how strongly a system covers existing regulatory and governance requirements. It evaluates whether a system provides the capabilities, controls, and evidence mechanisms that frameworks such as the EU AI Act, GDPR, NIST AI RMF, ISO/IEC 42001, and Singapore MGF increasingly require.

RCCS-T and RCCS-M are not legal compliance scores. A high RCCS-T or RCCS-M score does not mean a system is legally compliant. A low score does not mean a system is non-compliant. Legal compliance depends on organizational practice, deployment context, jurisdiction, risk class, and how the system is used. RCCS measures system capability, not organizational conformance.

The purpose of RCCS-T is to translate regulatory abstractions into measurable system capabilities. Regulation requires risk management, but what does that mean in a system? RCCS-T defines ten dimensions that operationalize regulatory requirements into system features that can be evaluated, compared, and improved. It is a bridge between regulatory language and engineering implementation.

Traditional coverage is necessary but insufficient for agentic and multi-agent systems. A governance dashboard, audit log, or approval workflow can support RCCS-T while still failing to express who held responsibility, what authority was delegated, which agent or tool acted, what evidence was partitioned, whether the outcome was accepted, and how incident or substitution closure was proven. High RCCS-T does not automatically imply high RCCS-M.

RCCS-M adds the MRO adjustment. It asks a second question: can the regulatory coverage captured by RCCS-T be expressed through lifecycle responsibility objects that agentic systems can emit, inspect, replay, and validate? A governance dashboard may support RCCS-T. An authority boundary object, accepted outcome record, partitioned evidence chain, and substitution conformance record support RCCS-M.

The distinction matters because enterprise governance platforms may show strong RCCS-T coverage through conventional dashboards, policies, documentation, controls, and monitoring. Lifecycle protocol systems may show stronger RCCS-M where they define explicit responsibility, authority, evidence, acceptance, privacy, substitution, and closure objects. This is a score-delta interpretation, not a product ranking.

7.2 RCCS-T Dimensions

RCCS-T and RCCS-M share the same 10 dimension names and weights. The dimensions correspond to recurring regulatory and governance requirements. The scoring lenses differ: RCCS-T evaluates traditional governance/control coverage, while RCCS-M evaluates whether that coverage is expressible through lifecycle responsibility objects.

The ten shared RCCS dimensions are:

1. **Risk Management (20%)** - Does the system support risk identification, assessment, and mitigation across the AI lifecycle?
2. **Data Governance (10%)** - Does the system track data lineage, quality, and lifecycle management?
3. **Documentation (10%)** - Does the system produce and maintain required technical documentation?
4. **Record-Keeping (10%)** - Does the system maintain audit trails and evidence records?
5. **Transparency (10%)** - Does the system provide explainability and disclosure mechanisms?
6. **Human Oversight (15%)** - Does the system support human review, approval, and intervention?
7. **Security (5%)** - Does the system implement security controls and access management?
8. **Accountability (5%)** - Does the system assign responsibility and track decisions?
9. **Contestability (5%)** - Does the system support dispute and appeal mechanisms?

10. Monitoring (10%) - Does the system support continuous monitoring and incident detection?

The following table summarizes the ten dimensions, their regulatory sources, core questions, and weights. The table is the baseline for both RCCS-T and RCCS-M; section 7.4 explains the MRO adjustment test that changes the RCCS-M raw score.

Table T-07-01: RCCS Dimensions Summary

Dimension	Regulatory Source	Core Question	Weight
Risk Management	EU AI Act Art. 9, NIST AI RMF Govern/Map/Measure/Manage, ISO/IEC 42001	Does the system support risk identification, assessment, and mitigation across the AI lifecycle?	20%
Data Governance	GDPR Art. 5, ISO/IEC 42001, NIST AI RMF Map	Does the system track data lineage, quality, and lifecycle management?	10%
Documentation	EU AI Act Art. 11, ISO/IEC 42001, NIST AI RMF Govern	Does the system produce and maintain required technical documentation?	10%
Record-Keeping	EU AI Act Art. 12, GDPR Art. 30, ISO/IEC 42001	Does the system maintain audit trails and evidence records?	10%
Transparency	EU AI Act Art. 13, GDPR Art. 5, NIST AI RMF Govern	Does the system provide explainability and disclosure mechanisms?	10%
Human Oversight	EU AI Act Art. 14, Singapore MGF Agentic AI, NIST AI RMF Manage	Does the system support human review, approval, and intervention?	15%
Security	EU AI Act Art. 15, ISO/IEC 42001, NIST AI RMF Manage	Does the system implement security controls and access management?	5%
Accountability	GDPR Art. 5, ISO/IEC 42001, Singapore MGF Agentic AI	Does the system assign responsibility and track decisions?	5%
Contestability	EU AI Act recitals, GDPR Art. 22, Singapore MGF Agentic AI	Does the system support dispute and appeal mechanisms?	5%
Monitoring	EU AI Act Art. 72, NIST AI RMF Manage, ISO/IEC 42001	Does the system support continuous monitoring and incident detection?	10%

7.3 RCCS-M: MRO-Adjusted Interpretation of RCCS

RCCS-M measures whether traditional regulatory obligations can be represented as Missing Regulatory Objects and lifecycle responsibility semantics. Operationally, RCCS-M asks whether a regulatory obligation is represented as inspectable, replayable, evidence-linked lifecycle responsibility objects.

The MRO adjustment changes what counts as strong support. For example:

- Risk management becomes stronger when risks are linked to lifecycle risk objects, authority boundaries, mitigations, monitoring evidence, and incident closure.
- Human oversight becomes stronger when approval is bound to human-role responsibility, delegated authority, escalation, and accepted outcome authority.
- Record-keeping becomes stronger when logs become partitioned evidence chains tied to role, agent/tool action, authority boundary, privacy boundary, and accepted outcome.
- Monitoring becomes stronger when runtime observation includes authority drift, substitution conformance, privacy lifecycle events, and incident/dispute/remediation closure.

RCCS-M is an author analytical, forward-looking model. It is not current law, legal compliance proof, certification, a regulator-approved benchmark, vendor ranking, or procurement recommendation.

7.4 Dimension Mapping: RCCS-T Dimension to MRO Adjustment Test

Table T-07-01A: RCCS-T Dimension to RCCS-M MRO Adjustment Test

Methodology matrix rendered as semantic row cards to preserve long evidence and boundary text; all source rows are retained.

RCCS-01 Risk Management	
Traditional RCCS-T Test	Does the system identify, assess, mitigate, and monitor AI risks?
RCCS-M Adjustment Question	Are risks linked to lifecycle authority, evidence, mitigation, monitoring, and closure objects?
Related MROs	MRO-02, MRO-07, MRO-08, MRO-16
Required MRO Evidence	Risk object, authority boundary, mitigation record, evidence pointer, monitoring/closure record
Negative Control	Risk register without lifecycle linkage

RCCS-02 Data Governance	
Traditional RCCS-T Test	Does the system govern data quality, lineage, access, retention, and lifecycle handling?
RCCS-M Adjustment Question	Are data flows mapped across agents, tools, memory, evidence stores, privacy boundaries, and reuse contexts?
Related MROs	MRO-08, MRO-10, MRO-12, MRO-13, MRO-14
Required MRO Evidence	Lifecycle data-flow map, selective disclosure profile, retention/erasure policy, processor chain record
Negative Control	Dataset lineage without agent/runtime lifecycle mapping

RCCS-03 Documentation	
Traditional RCCS-T Test	Does the system produce and maintain technical and governance documentation?
RCCS-M Adjustment Question	Does documentation bind intent, plan, authority, role, evidence, accepted outcome, and substitution context?
Related MROs	MRO-01, MRO-02, MRO-04, MRO-08, MRO-15
Required MRO Evidence	Versioned lifecycle documentation, object schema, trace manifest, substitution record
Negative Control	Static model card without lifecycle responsibility objects

RCCS-04 Record-Keeping

Traditional RCCS-T Test	Does the system record relevant events, logs, and audit trails?
RCCS-M Adjustment Question	Are records partitioned by role, authority boundary, agent/tool action, evidence pointer, privacy boundary, and accepted outcome?
Related MROs	MRO-08 , MRO-12 , MRO-13 , MRO-16
Required MRO Evidence	Partitioned evidence chain, evidence pointer, disclosure policy, closure record
Negative Control	Generic logs without lifecycle responsibility partitioning

RCCS-05 Transparency

Traditional RCCS-T Test	Does the system disclose purpose, behavior, limitations, and relevant decision factors?
RCCS-M Adjustment Question	Can disclosure be tied to lifecycle evidence, role, authority, accepted outcome, privacy boundary, and dispute path?
Related MROs	MRO-04 , MRO-08 , MRO-11 , MRO-12 , MRO-16
Required MRO Evidence	Evidence-linked explanation, disclosure profile, redacted validation pack, dispute replay record
Negative Control	Explanation UI without lifecycle evidence or selective disclosure

RCCS-06 Human Oversight

Traditional RCCS-T Test	Does the system support human review, approval, intervention, and escalation?
RCCS-M Adjustment Question	Is oversight bound to human-role responsibility, delegated scope, escalation, and accepted outcome authority?
Related MROs	MRO-01 , MRO-02 , MRO-03 , MRO-04 , MRO-07
Required MRO Evidence	Human-role map, authority grant, escalation path, acceptance state, drift alert
Negative Control	Generic approve/reject UI without responsibility or authority scope

RCCS-07 Security

Traditional RCCS-T Test	Does the system implement access, authentication, authorization, and security monitoring controls?
RCCS-M Adjustment Question	Are security controls linked to delegated authority, tool-action liability, lifecycle evidence, and privacy-limited disclosure?
Related MROs	MRO-02 , MRO-05 , MRO-08 , MRO-10 , MRO-12
Required MRO Evidence	Delegated authority record, tool-action liability object, evidence partition, privacy boundary
Negative Control	Access control treated as delegated authority proof

RCCS-08 Accountability	
Traditional RCCS-T Test	Does the system assign roles, responsibilities, policies, and escalation paths?
RCCS-M Adjustment Question	Are accountability relationships represented as human/agent responsibility mappings, transfer records, and accepted outcome states?
Related MROs	MRO-01, MRO-03, MRO-04, MRO-05, MRO-06
Required MRO Evidence	Responsibility map, agent role boundary, transfer record, acceptance record, liability record
Negative Control	User role assignment without responsibility continuity

RCCS-09 Contestability	
Traditional RCCS-T Test	Does the system support dispute, appeal, investigation, correction, and remediation workflows?
RCCS-M Adjustment Question	Can challenged outcomes be replayed with evidence, reviewer role, accepted/rejected state, remediation action, and closure?
Related MROs	MRO-04, MRO-08, MRO-12, MRO-13, MRO-16
Required MRO Evidence	Dispute replay pack, evidence manifest, remediation record, closure acceptance
Negative Control	Customer support ticket without lifecycle replay or closure

RCCS-10 Monitoring	
Traditional RCCS-T Test	Does the system monitor performance, drift, incidents, and continuous improvement?
RCCS-M Adjustment Question	Does monitoring detect authority drift, lifecycle drift, substitution change, privacy events, incidents, and remediation closure?
Related MROs	MRO-07, MRO-09, MRO-10, MRO-15, MRO-16
Required MRO Evidence	Drift event, reuse/revalidation record, privacy event record, substitution conformance record, incident closure
Negative Control	General observability dashboard without responsibility-object monitoring

7.5 Scoring Criteria: How 0-5 Differs Under RCCS-T and RCCS-M

RCCS-T and RCCS-M use the same 0-5 raw score range, but the meaning of each raw score differs because they use different scoring lenses.

Table T-07-02A: RCCS-T vs RCCS-M 0-5 Scoring Lens

Methodology matrix rendered as semantic row cards to preserve long evidence and boundary text; all source rows are retained.

0

RCCS-T Meaning	No visible traditional coverage
RCCS-M Meaning	No lifecycle object expression
Evidence Difference	No credible evidence for either lens
Negative Control	Absence of public feature or object support

1

RCCS-T Meaning	Policy, concept, or marketing mention only
RCCS-M Meaning	MRO concept implied but not represented as an object or evidence artifact
Evidence Difference	RCCS-T may rely on weak public mention; RCCS-M needs at least identifiable lifecycle-object semantics
Negative Control	Governance language without implementation or object evidence

2

RCCS-T Meaning	Partial feature or control support
RCCS-M Meaning	Partial lifecycle object support, mostly implementer-built or inferred
Evidence Difference	RCCS-T can count partial controls; RCCS-M requires a plausible but incomplete mapping to MRO evidence
Negative Control	Adjacent feature treated as lifecycle object

3

RCCS-T Meaning	Documented operational control
RCCS-M Meaning	Documented object-level mapping for some lifecycle responsibility objects
Evidence Difference	RCCS-T requires official feature/workflow documentation; RCCS-M requires documented role, authority, evidence, acceptance, privacy, substitution, or closure mapping
Negative Control	Workflow support without explicit responsibility semantics

4

RCCS-T Meaning	Strong operational control with evidence
RCCS-M Meaning	Strong lifecycle object coverage with evidence pointers and responsibility/authority binding
Evidence Difference	RCCS-T requires strong control evidence; RCCS-M requires object records, schemas, protocol evidence, or replayable artifacts
Negative Control	Strong dashboard or log export without object partitioning

5

RCCS-T Meaning	Comprehensive traditional governance control with strong evidence
RCCS-M Meaning	Inspectable, replayable, evidence-linked lifecycle responsibility object coverage across the relevant dimension
Evidence Difference	RCCS-T can be satisfied by comprehensive implementation or audit evidence; RCCS-M requires implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts showing lifecycle object conformance
Negative Control	Legal text or policy baseline treated as implementation proof

Legal text may establish a regulatory baseline but does not by itself prove RCCS-T implementation or RCCS-M lifecycle-object expression.

7.6 Evidence Requirements: Traditional Evidence vs MRO Evidence

RCCS-T evidence and RCCS-M evidence are not the same.

RCCS-T evidence may include dashboards, policies, documentation, risk workflows, audit logs, oversight features, control reports, monitoring views, governance console records, and official product documentation showing traditional governance/control coverage.

RCCS-M evidence should show lifecycle responsibility object expression. Examples include authority boundary records, human-role responsibility maps, accepted outcome states, partitioned evidence chains, privacy lifecycle maps, substitution conformance records, incident/dispute/remediation closure records, replayable lifecycle traces, protocol schemas, object manifests, or validated operational artifacts.

Official product documentation may support RCCS-T strongly while supporting RCCS-M only indirectly. When lifecycle responsibility objects are not directly documented, RCCS-M scoring should be marked as source-qualified or L5 author analytical inference rather than treated as source-bound proof.

7.7 Calculation Method

RCCS-T and RCCS-M use the same dimension weights and the same normalized dimension formula:

Dimension Score = (Raw Score / 5) × 100 × Evidence Confidence Multiplier

RCCS-T raw scores and RCCS-M raw scores are assigned separately. The formula is shared; the scoring lens differs. RCCS-T raw scores evaluate traditional governance/control coverage. RCCS-M raw scores evaluate MRO-adjusted lifecycle responsibility object expression.

The evidence multiplier adjusts each dimension score based on evidence quality:

- L1 : 1.00
- L2 : 0.85
- L3 : 0.75
- L4 : 0.55

- **L5**: 0.35

Weights are fixed for this version of the white paper:

- Risk Management: 20%
- Human Oversight: 15%
- Data Governance: 10%
- Documentation: 10%
- Record-Keeping: 10%
- Transparency: 10%
- Monitoring: 10%
- Security: 5%
- Accountability: 5%
- Contestability: 5%

Weights are fixed for this version of the white paper. Future versions may adjust weights based on regulatory evolution or stakeholder feedback.

RCCS-T and RCCS-M should not be averaged invisibly. The preferred publication-facing display is the three-profile view: RCCS-T, RCCS-M, and ALCS. Optional composite views are defined in Chapter 9 and remain secondary.

The following table illustrates the calculation formula.

Table T-07-02: RCCS Dimension Score Calculation Formula

Component	Formula	Example
Raw Score	0-5 scale based on capability strength	Risk Management = 4
Evidence Multiplier	L1 - L5 multiplier	L2 evidence = 0.85
Dimension Score	(Raw Score / 5) × 100 × Evidence Multiplier	(4 / 5) × 100 × 0.85 = 68.0
Framework Score	Aggregate of adjusted RCCS dimension scores	RCCS-T or RCCS-M framework score
Optional Composite View	Secondary calculation using RCCS-T or RCCS-M with ALCS	See Chapter 9

This calculation method is used for RCCS-T, RCCS-M, and ALCS. The difference is the evaluation lens: RCCS-T asks about traditional regulatory coverage, RCCS-M asks about MRO-adjusted coverage, and ALCS asks about lifecycle conformance maturity.

7.8 Boundary and Negative Controls

RCCS-T and RCCS-M are analytical scoring frameworks, not legal compliance scores. This boundary must be understood clearly to avoid misinterpretation.

What RCCS is:

- An analytical framework for comparing system capabilities against regulatory ideals
- A structured method for evaluating whether a system provides governance primitives and whether those primitives can be expressed through lifecycle responsibility objects
- A tool for buyers, auditors, and governance teams to assess system readiness
- A bridge between regulatory language and engineering implementation

What RCCS is not:

- A legal compliance certification or conformity assessment
- A regulatory approval or endorsement
- A guarantee that a system meets all legal obligations
- A substitute for legal counsel, compliance review, or risk assessment

Key boundary principles:

1. **High RCCS-T or RCCS-M score does NOT mean legal compliance.** A system may score well while the organization using it fails to comply with legal obligations due to deployment gaps, policy weaknesses, incomplete adoption, or jurisdictional differences.
1. **High RCCS-T does NOT mean lifecycle responsibility conformance.** Traditional governance coverage can be strong while authority, responsibility, evidence, privacy, substitution, and closure objects remain weak.
1. **High RCCS-M does NOT mean legal compliance or certification.** RCCS-M is author analytical and forward-looking; it is not current law, regulator-approved, a procurement benchmark, or a legal compliance proof.
1. **High ALCS does NOT mean regulatory compliance.** ALCS evaluates lifecycle conformance maturity; RCCS-T and RCCS-M still need to be read separately.
1. **Low RCCS score does NOT mean non-compliance.** A system may score lower on RCCS while the organization achieves compliance through compensating controls, manual processes, or alternative governance mechanisms.
1. **RCCS measures system capability, not organizational practice.** A system may provide strong governance primitives while the organization fails to use them. Conversely, an organization may achieve strong governance through processes that are not reflected in system capabilities.
1. **RCCS is context-independent, but compliance is context-dependent.** RCCS evaluates system capabilities without regard to deployment context, risk class, jurisdiction, or use case. Legal compliance depends on all of these factors.

The purpose of this boundary statement is to prevent over-claiming. RCCS-T and RCCS-M are useful analytical tools, but they are not compliance certifications. Organizations must not rely on RCCS scores as proof of legal compliance. Buyers must not treat high RCCS scores as a substitute for due diligence, legal review, or risk assessment. RCCS-M must also not be presented as a regulator-approved benchmark or official standard.

7.9 Rubric Tables / Appendix E Bridge

The following sections (7.11 through 7.20) detail each RCCS dimension. Each dimension section includes:

- **Regulatory basis** - Which frameworks and requirements inform the dimension
- **Core question** - What the dimension evaluates
- **Scoring criteria** - How to apply the 0-5 raw scale to system capabilities
- **RCCS-M adjustment note** - How the dimension changes when evaluated through lifecycle responsibility objects
- **System mapping guidance** - How to evaluate whether a system supports the dimension

The compact chapter rubrics preserve the traditional RCCS-T baseline and add RCCS-M adjustment notes where practical. Appendix E contains the detailed RCCS-T rubrics plus the reusable RCCS-M adjustment

matrix. Appendix E does not duplicate every RCCS-M rubric row; it provides the matrix that evaluators should apply to the ten shared dimensions.

7.10 What RCCS-T / RCCS-M Do Not Prove

RCCS-T and RCCS-M do not prove legal compliance, certification, regulatory approval, audit success, deployment readiness, product quality, market leadership, or procurement suitability. They also do not replace ALCS. RCCS-T names traditional regulatory coverage. RCCS-M names MRO-adjusted regulatory coverage. ALCS names lifecycle conformance maturity.

Chapter 9 explains how to interpret score deltas across RCCS-T, RCCS-M, and ALCS. A system with high RCCS-T and lower RCCS-M may have strong traditional governance surfaces but weaker lifecycle object expression. A system with lower RCCS-T and higher RCCS-M/ALCS may have strong lifecycle semantics but less productized governance-platform coverage. This difference is an analytical profile, not a ranking.

7.11 RCCS-01: Risk Management

Regulatory Basis

Risk management is the highest-weighted RCCS dimension because it appears as a core requirement across all major frameworks. EU AI Act Article 9 requires providers of high-risk AI systems to establish, implement, document, and maintain a risk management system. NIST AI RMF 1.0 structures its entire framework around four risk management functions: Govern, Map, Measure, and Manage. ISO/IEC 42001 requires organizations to establish and maintain processes for identifying, analyzing, evaluating, and treating AI-related risks throughout the system lifecycle.

These frameworks converge on a shared expectation: AI systems must support continuous risk identification, assessment, mitigation, and monitoring. Risk management is not a one-time activity performed during model development. It is a lifecycle discipline that must be maintained across deployment, operation, updates, and decommissioning.

Core Question

Does the system support risk identification, assessment, mitigation, and continuous monitoring across the AI lifecycle?

RCCS-M Adjustment Note

RCCS-M adjustment: are risks linked to lifecycle authority, evidence, mitigation, monitoring, and incident/dispute/remediation closure objects rather than only stored as a risk register or dashboard item?

Scoring Rubric

Table T-07-03: RCCS-01 Risk Management Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of risk management capability No risk management features documented or observable

1

Conceptual support

Documentation mentions risk management but provides no implementation primitives

2

Partial implementation

Risk assessment tools exist but are limited to specific phases (e.g., model development only) or lack integration with execution controls

3

Moderate implementation

Risk registers, assessment workflows, and mitigation tracking exist; some gaps in continuous monitoring or cross-lifecycle integration

4

Strong implementation

Comprehensive risk lifecycle support including identification, assessment, mitigation tracking, continuous monitoring, and risk-based controls integrated across all system phases

5

Complete capability demonstrated by implementation evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove implementation

System Mapping Guidance

When evaluating a system for risk management capability, look for evidence of four core functions: risk identification, risk assessment, risk mitigation, and continuous monitoring.

Risk identification requires mechanisms for discovering and documenting risks. Strong systems provide risk registers, risk taxonomies, and workflows for capturing risks from multiple sources including model evaluation, deployment review, incident reports, and external threat intelligence. Weak systems rely on manual documentation without structured capture mechanisms.

Risk assessment requires methods for analyzing and prioritizing risks. Look for risk scoring frameworks, impact and likelihood matrices, risk classification schemes, and decision workflows that route high-risk scenarios to appropriate review or mitigation. Systems that provide only generic risk labels without assessment workflows score lower.

Risk mitigation requires tracking and verification of risk controls. Strong systems link identified risks to specific mitigation measures, track implementation status, and verify that controls are active before deployment. Systems that document mitigations without enforcement or verification score lower.

Continuous monitoring requires runtime risk detection and response. Look for monitoring dashboards, alerting mechanisms, drift detection, and incident response workflows that connect runtime observations back to the risk register. Systems that perform risk assessment only at design time without runtime monitoring score lower.

Common Gaps

The most common gap is risk assessment that exists for model development but does not extend to agent execution, tool use, or multi-agent handoffs. Organizations may have strong ML risk management practices

while lacking risk controls for agentic workflows. Another common gap is risk registers that document risks without linking them to runtime controls or monitoring, creating a disconnect between risk identification and risk mitigation.

7.12 RCCS-02: Data Governance

Regulatory Basis

Data governance requirements stem from GDPR Article 5, which establishes principles for processing personal data including data quality, purpose limitation, and data minimization. EU AI Act Article 10 requires providers to ensure that training, validation, and testing datasets are subject to appropriate data governance and management practices. ISO/IEC 42001 requires organizations to establish and maintain data management processes covering data quality, data lifecycle, and data provenance.

For agentic systems, data governance extends beyond training datasets to include runtime data flows: retrieval results, tool payloads, memory contents, agent handoff data, and evidence records. Data governance must track what data enters the system, how it flows through agents and tools, where it is stored, and when it is deleted.

Core Question

Does the system track data lineage, quality, lifecycle, and provenance across agent execution?

RCCS-M Adjustment Note

RCCS-M adjustment: are data flows mapped across agents, tools, memory, evidence stores, privacy boundaries, reuse contexts, retention rules, and processor/subprocessor chains?

Scoring Rubric

Table T-07-04: RCCS-02 Data Governance Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of data governance capability No data tracking features documented or observable
1	Conceptual support Documentation mentions data governance but provides no implementation primitives
2	Partial implementation Data catalogs or lineage tracking exist but are limited to training data or specific data sources; runtime data flows are not tracked
3	Moderate implementation Data lineage, quality metrics, and lifecycle management exist for most data flows; some gaps in agent-to-agent data tracking or memory system governance

4

Strong implementation

Comprehensive data governance covering training data, runtime retrieval, tool payloads, memory contents, agent handoffs, and evidence records with full lineage, quality metrics, and lifecycle management

5

Complete capability demonstrated by implementation evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove implementation

System Mapping Guidance

Data governance evaluation requires examining four capabilities: data cataloging, lineage tracking, quality management, and lifecycle controls.

Data cataloging requires inventories of data sources, datasets, and data flows. Strong systems maintain catalogs that identify what data is available, where it originates, what sensitivity classifications apply, and what usage restrictions exist. Look for data catalogs that cover not only training datasets but also runtime data sources such as retrieval systems, tool APIs, and memory stores.

Lineage tracking requires the ability to trace data from source to consumption. Evaluate whether the system can show where a piece of data originated, which agents accessed it, which tools processed it, and where it was stored or transmitted. Systems that track lineage only for training pipelines but not for runtime agent execution score lower.

Quality management requires metrics and controls for data accuracy, completeness, consistency, and timeliness. Look for data quality dashboards, validation rules, and quality gates that prevent low-quality data from entering critical workflows. Systems that assume data quality without measurement or enforcement score lower.

Lifecycle controls require policies and mechanisms for data retention, archival, and deletion. Evaluate whether the system enforces retention periods, supports data subject deletion requests, and can prove that data was deleted when required. Systems that retain data indefinitely without lifecycle policies score lower.

Common Gaps

The most common gap is data governance that covers training data but not runtime data flows. Organizations may have strong data catalogs and lineage tracking for ML pipelines while lacking visibility into what data agents retrieve, process, and store during execution. Another common gap is data quality controls that apply to databases but not to agent memory systems or tool payloads, creating blind spots in data governance coverage.

7.13 RCCS-03: Documentation**Regulatory Basis**

Documentation requirements are established in EU AI Act Article 11, which requires providers of high-risk AI systems to draw up technical documentation that demonstrates compliance with regulatory requirements. ISO/IEC 42001 requires organizations to maintain documented information necessary for the effectiveness of

the AI management system. NIST AI RMF Govern function includes documentation of AI system characteristics, intended use, and risk management decisions.

Documentation is not merely a compliance artifact. It is the primary mechanism for communicating system capabilities, limitations, and governance controls to deployers, operators, auditors, and regulators. For agentic systems, documentation must cover not only model characteristics but also agent roles, authority boundaries, tool permissions, and lifecycle controls.

Core Question

Does the system produce and maintain required technical documentation across the system lifecycle?

RCCS-M Adjustment Note

RCCS-M adjustment: does documentation bind intent, plan, authority, role, evidence, accepted outcome, privacy boundary, and substitution context as lifecycle responsibility records?

Scoring Rubric

Table T-07-05: RCCS-03 Documentation Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of documentation capability No documentation features documented or observable
1	Conceptual support Documentation mentions documentation requirements but provides no automated generation or maintenance primitives
2	Partial implementation Model cards or system documentation exist but are manually maintained and not version-controlled or linked to deployments
3	Moderate implementation Automated documentation generation exists for models and some system components; some gaps in agent role documentation, tool documentation, or deployment linkage
4	Strong implementation Comprehensive automated documentation covering models, agents, tools, authority boundaries, deployment configurations, and change history with version control and deployment linkage
5	Complete capability demonstrated by implementation evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove implementation

System Mapping Guidance

Documentation evaluation requires examining four capabilities: documentation generation, version control, deployment linkage, and coverage completeness.

Documentation generation requires mechanisms for producing required documentation artifacts. Look for automated generation of model cards, system cards, agent role definitions, tool specifications, and deployment documentation. Systems that rely entirely on manual documentation score lower because manual processes are error-prone and difficult to maintain.

Version control requires tracking documentation changes over time. Evaluate whether documentation is version-controlled alongside code and configuration, whether changes are linked to specific releases or deployments, and whether historical documentation can be retrieved for audit purposes. Systems that maintain documentation in separate systems without version control score lower.

Deployment linkage requires connecting documentation to specific deployed instances. Strong systems can show which documentation version applies to which deployment, enabling auditors to verify that deployed systems match their documentation. Systems that maintain documentation separately from deployment records score lower.

Coverage completeness requires documentation for all system components that affect governance. For agentic systems, this includes not only model documentation but also agent role definitions, authority boundaries, tool permissions, memory system configurations, and human oversight mechanisms. Systems that document models but not agent governance primitives score lower.

Common Gaps

The most common gap is documentation that exists but is not version-controlled or linked to deployments. Organizations may maintain model cards and system documentation while lacking the ability to prove which documentation version applied to a specific deployment at a specific time. Another common gap is documentation that covers models but not agent roles, tool permissions, or authority boundaries, leaving governance controls undocumented.

7.14 RCCS-04: Record-Keeping

Regulatory Basis

Record-keeping requirements are established in EU AI Act Article 12, which requires providers and deployers of high-risk AI systems to keep logs automatically generated by their systems. GDPR Article 30 requires controllers and processors to maintain records of processing activities. ISO/IEC 42001 requires organizations to maintain records necessary to demonstrate conformity with requirements.

Record-keeping differs from documentation in that records capture what actually happened during system operation, while documentation describes what the system is designed to do. For agentic systems, records must capture agent decisions, tool invocations, human approvals, authority checks, and outcome acceptance events.

Core Question

Does the system maintain audit trails and evidence records across agent execution?

RCCS-M Adjustment Note

RCCS-M adjustment: are records partitioned by human/agent role, authority boundary, agent/tool action, evidence pointer, privacy boundary, accepted outcome, and closure state rather than stored as generic logs?

Scoring Rubric

Table T-07-06: RCCS-04 Record-Keeping Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of record-keeping capability No logging or audit trail features documented or observable
1	Conceptual support Documentation mentions logging but provides no structured record-keeping primitives
2	Partial implementation Execution logs exist but are not structured for audit purposes; no evidence retention policy or tamper-evident storage
3	Moderate implementation Structured audit trails exist for most lifecycle events; some gaps in evidence retention policies, tamper-evident storage, or cross-agent record linkage
4	Strong implementation Comprehensive tamper-evident audit trails covering all lifecycle events with structured records, evidence retention policies, integrity verification, and cross-agent record linkage
5	Complete capability demonstrated by implementation evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove implementation

System Mapping Guidance

Record-keeping evaluation requires examining four capabilities: structured logging, audit trail completeness, tamper-evident storage, and retention policies.

Structured logging requires records that are machine-readable and queryable. Look for structured log formats, consistent schemas, and query interfaces that allow auditors to reconstruct specific events or workflows. Systems that produce only unstructured text logs score lower because they are difficult to audit and analyze.

Audit trail completeness requires capturing all governance-relevant events. For agentic systems, this includes agent decisions, tool invocations, human approvals, authority checks, risk assessments, data accesses, and outcome acceptance events. Evaluate whether the system captures these events consistently and whether

records can be linked across agents and lifecycle phases. Systems that log execution events but not governance events score lower.

Tamper-evident storage requires mechanisms to detect unauthorized modification of records. Look for cryptographic integrity verification, append-only storage, or blockchain-based audit trails. Systems that store logs in mutable storage without integrity verification score lower because records cannot be trusted in disputes.

Retention policies require defined periods for retaining different record types. Evaluate whether the system enforces retention periods, supports deletion of expired records, and can prove that records were retained for the required period. Systems that retain all records indefinitely or delete records without policy enforcement score lower.

Common Gaps

The most common gap is logs that exist but are not structured for audit purposes. Organizations may have extensive execution logs while lacking the ability to reconstruct governance events such as authority checks, human approvals, or outcome acceptance. Another common gap is logs stored in mutable systems without integrity verification, making it impossible to prove that records were not altered after the fact.

7.15 RCCS-05: Transparency

Regulatory Basis

Transparency requirements are established in EU AI Act Article 13, which requires providers to design and develop high-risk AI systems to be sufficiently transparent to enable deployers to interpret system output and use it appropriately. GDPR Articles 13-14 require controllers to provide information to data subjects about automated decision-making. ISO/IEC 42001 requires organizations to establish transparency mechanisms appropriate to the AI system's risk level and context.

Transparency serves multiple purposes: enabling deployers to understand system behavior, allowing users to challenge decisions, and supporting auditors in verifying compliance. For agentic systems, transparency must extend beyond model explainability to include agent decision rationale, tool selection justification, and authority boundary explanations.

Core Question

Does the system provide explainability and disclosure mechanisms for agent decisions and outcomes?

RCCS-M Adjustment Note

RCCS-M adjustment: can explanations and disclosures be tied to lifecycle evidence, role, authority, accepted outcome, privacy boundary, selective disclosure, and dispute path?

Scoring Rubric

Table T-07-07: RCCS-05 Transparency Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0**No evidence of transparency capability**

No explainability or disclosure features documented or observable

1**Conceptual support**

Documentation mentions transparency but provides no implementation primitives

2**Partial implementation**

Model explainability tools exist but do not extend to agent decisions, tool selection, or authority boundaries

3**Moderate implementation**

Explainability mechanisms exist for models and some agent decisions; some gaps in tool selection rationale, authority boundary explanations, or user-facing disclosure

4**Strong implementation**

Comprehensive transparency covering model outputs, agent decision rationale, tool selection justification, authority boundaries, and user-facing disclosure mechanisms

5**Complete capability demonstrated by implementation evidence**

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove implementation

System Mapping Guidance

Transparency evaluation requires examining four capabilities: model explainability, agent decision rationale, disclosure mechanisms, and user-facing transparency.

Model explainability requires mechanisms for explaining model outputs. Look for feature importance, attention visualization, counterfactual explanations, or other explainability techniques. However, model explainability alone is insufficient for agentic systems. Evaluate whether explainability extends to agent-level decisions.

Agent decision rationale requires explanations for why an agent chose a particular action, tool, or delegation path. Strong systems capture and expose the reasoning behind agent decisions, including which information was considered, which constraints were applied, and which alternatives were rejected. Systems that provide only execution traces without decision rationale score lower.

Disclosure mechanisms require interfaces for communicating system capabilities and limitations to deployers and users. Look for system cards, capability statements, limitation disclosures, and risk warnings. Evaluate whether disclosures are specific and actionable rather than generic disclaimers.

User-facing transparency requires mechanisms for informing users when they are interacting with an AI system and what decisions are being made. For agentic systems, this includes disclosing when agents are acting autonomously, when human review is required, and how users can challenge decisions. Systems that operate without user awareness score lower.

Common Gaps

The most common gap is model explainability that does not extend to agent decision-making. Organizations may have strong explainability tools for model outputs while lacking visibility into why an agent selected a particular tool, delegated to another agent, or escalated to human review. Another common gap is transparency mechanisms that exist for developers but not for end users, leaving users unable to understand or challenge agent decisions.

7.16 RCCS-06: Human Oversight

Regulatory Basis

Human oversight requirements are established in EU AI Act Article 14, which requires high-risk AI systems to be designed and developed to enable effective oversight by natural persons. ISO/IEC 42001 requires organizations to establish human oversight mechanisms appropriate to the AI system's risk level. NIST AI RMF Manage function includes human-AI configuration as a key practice. Singapore MGF for Agentic AI emphasizes making humans meaningfully accountable for agentic system outcomes.

Human oversight is the second-highest weighted RCCS dimension because it is the primary mechanism for ensuring that AI systems remain under human control. For agentic systems, oversight must be designed into the system architecture, not merely added as a review step after execution.

Core Question

Does the system support human review, approval, intervention, and escalation across agent execution?

RCCS-M Adjustment Note

RCCS-M adjustment: is oversight bound to human-role responsibility, delegated authority scope, escalation path, drift detection, and accepted outcome authority rather than only an approve/reject UI?

Scoring Rubric

Table T-07-08: RCCS-06 Human Oversight Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of human oversight capability No human review or intervention features documented or observable
1	Conceptual support Documentation mentions human oversight but provides no implementation primitives
2	Partial implementation Human approval workflows exist but are not mapped to responsibility boundaries or risk levels; no intervention or escalation mechanisms

3**Moderate implementation**

Human review, approval, and intervention mechanisms exist for most high-risk actions; some gaps in escalation paths, responsibility mapping, or cross-agent oversight

4**Strong implementation**

Comprehensive human-in-the-loop controls including risk-based review, approval workflows, intervention mechanisms, escalation paths, and responsibility mapping across all agent actions

5**Complete capability demonstrated by implementation evidence**

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove implementation

System Mapping Guidance

Human oversight evaluation requires examining four capabilities: review interfaces, approval workflows, intervention mechanisms, and escalation paths.

Review interfaces require mechanisms for humans to examine agent plans, decisions, and outputs before they take effect. Look for review dashboards, approval queues, and decision summaries that present relevant information to human reviewers. Evaluate whether review interfaces are designed for the specific oversight task rather than generic log viewers. Systems that require humans to review raw execution logs score lower.

Approval workflows require explicit human authorization before high-risk actions execute. Strong systems identify which actions require approval based on risk level, authority boundaries, or policy rules, and enforce approval requirements before execution. Look for approval routing, approval tracking, and approval evidence capture. Systems that allow high-risk actions to execute without approval score lower.

Intervention mechanisms require the ability for humans to stop, modify, or override agent actions. Evaluate whether humans can intervene during execution, whether intervention is recorded as evidence, and whether agents respect intervention signals. Systems that allow agents to continue executing after human intervention requests score lower.

Escalation paths require routing decisions to appropriate human roles based on risk, complexity, or policy violations. Look for escalation rules, escalation routing, and escalation tracking. Strong systems escalate automatically when agents encounter situations beyond their authority or capability. Systems that require agents to decide when to escalate score lower.

Common Gaps

The most common gap is human review that exists but is not mapped to responsibility boundaries. Organizations may have approval workflows while lacking clarity about which human role is accountable for which agent decision. Another common gap is oversight mechanisms that apply to final outputs but not to intermediate agent decisions, tool selections, or authority transitions, allowing agents to drift beyond their intended scope before human review occurs.

7.17 RCCS-07: Security

Regulatory Basis

Security requirements are established in EU AI Act Article 15, which requires high-risk AI systems to be resilient against attempts to alter their use or performance through exploitation of system vulnerabilities. GDPR Article 32 requires controllers and processors to implement appropriate technical and organizational measures to ensure a level of security appropriate to the risk. ISO/IEC 42001 requires organizations to establish security controls for AI systems.

Security is foundational for all other governance controls. Without security, risk management, documentation, record-keeping, and oversight mechanisms can be bypassed or manipulated. For agentic systems, security must cover not only platform security but also agent-to-agent communication, tool access controls, and authority boundary enforcement.

Core Question

Does the system implement security controls, access management, and threat protection across agent execution?

RCCS-M Adjustment Note

RCCS-M adjustment: are security controls linked to delegated authority, tool-action liability, evidence partitioning, privacy lifecycle boundaries, and selective disclosure rules?

Scoring Rubric

Table T-07-09: RCCS-07 Security Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of security capability No security controls documented or observable
1	Conceptual support Documentation mentions security but provides no implementation primitives
2	Partial implementation Platform security exists but does not extend to agent-to-agent communication, tool access controls, or authority boundary enforcement
3	Moderate implementation Security controls exist for platform, agent communication, and tool access; some gaps in threat detection, security monitoring, or authority boundary enforcement

4

Strong implementation

Comprehensive security architecture covering platform security, agent-to-agent communication, tool access controls, authority boundary enforcement, threat detection, and security monitoring

5

Complete capability demonstrated by implementation evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove implementation

System Mapping Guidance

Security evaluation requires examining four capabilities: access controls, communication security, threat protection, and security monitoring.

Access controls require authentication, authorization, and permission management. Look for identity management, role-based access control, and least-privilege enforcement. For agentic systems, evaluate whether access controls apply not only to human users but also to agents and tools. Systems that grant broad permissions to agents without fine-grained access control score lower.

Communication security requires encryption and integrity protection for data in transit. Evaluate whether agent-to-agent communication, agent-to-tool communication, and agent-to-external-system communication are encrypted and authenticated. Systems that transmit sensitive data or credentials in plaintext score lower.

Threat protection requires defenses against adversarial attacks, prompt injection, and unauthorized access. Look for input validation, output filtering, and adversarial robustness testing. For agentic systems, evaluate whether the system protects against prompt injection attacks that could manipulate agent behavior or bypass authority boundaries. Systems without adversarial defenses score lower.

Security monitoring requires detection and response to security events. Look for security logging, anomaly detection, and incident response workflows. Evaluate whether security monitoring covers agent behavior, tool usage, and authority boundary violations. Systems that monitor platform security but not agent security score lower.

Common Gaps

The most common gap is platform security that does not extend to agent-to-agent communication or tool access controls. Organizations may have strong perimeter security while lacking visibility into how agents communicate with each other or which tools agents can access. Another common gap is security controls that protect against external threats but not against prompt injection or adversarial manipulation of agent behavior.

7.18 RCCS-08: Accountability

Regulatory Basis

Accountability requirements are established in GDPR Article 5(2), which requires controllers to be responsible for and able to demonstrate compliance with data protection principles. EU AI Act Article 26 establishes responsibilities along the AI value chain, clarifying obligations for providers, deployers, importers, and distributors. ISO/IEC 42001 requires organizations to establish accountability mechanisms for AI system

decisions and outcomes. Singapore MGF for Agentic AI emphasizes making humans meaningfully accountable for agentic system outcomes.

Accountability is the principle that someone must be responsible for AI system decisions and outcomes. For agentic systems, accountability requires clear assignment of responsibility for agent actions, tool use, and outcomes, with evidence to prove who was responsible for what.

Core Question

Does the system assign responsibility and track accountability for agent decisions and outcomes?

RCCS-M Adjustment Note

RCCS-M adjustment: are accountability relationships represented as human-role-to-MAS responsibility mappings, agent role boundaries, responsibility transfer records, accepted outcome states, and liability records?

Scoring Rubric

Table T-07-10: RCCS-08 Accountability Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of accountability capability No responsibility assignment or accountability tracking documented or observable
1	Conceptual support Documentation mentions accountability but provides no implementation primitives
2	Partial implementation Responsibility assignment exists for models but does not extend to agent actions, tool use, or multi-agent handoffs
3	Moderate implementation Accountability mechanisms exist for most agent actions and outcomes; some gaps in tool use accountability, cross-agent responsibility tracking, or evidence linkage
4	Strong implementation Comprehensive accountability covering responsibility assignment for all agent actions, tool use, outcomes, and multi-agent handoffs with evidence linkage and accountability reports
5	Complete capability demonstrated by implementation evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove implementation

System Mapping Guidance

Accountability evaluation requires examining four capabilities: responsibility assignment, decision tracking, accountability reports, and evidence linkage.

Responsibility assignment requires explicit identification of who is accountable for each agent decision and outcome. Look for responsibility mappings that link human roles to agent roles, agent roles to specific actions, and actions to outcomes. Evaluate whether responsibility assignments are documented, enforced, and auditable. Systems that execute agent actions without clear responsibility assignment score lower.

Decision tracking requires capturing who made each decision and on what basis. For agentic systems, this includes not only human decisions but also agent decisions that were delegated by humans. Look for decision logs that record decision maker, decision rationale, decision timestamp, and decision outcome. Systems that track execution events but not decision accountability score lower.

Accountability reports require mechanisms for demonstrating who was responsible for what. Look for accountability dashboards, responsibility matrices, and audit reports that show responsibility assignments and decision outcomes. Evaluate whether reports can be generated for specific time periods, projects, or incidents. Systems without accountability reporting capabilities score lower.

Evidence linkage requires connecting accountability assignments to evidence records. Strong systems can show not only who was responsible but also what evidence supports that responsibility assignment. Look for evidence pointers in responsibility records and responsibility metadata in evidence records. Systems that maintain responsibility assignments separately from evidence score lower.

Common Gaps

The most common gap is accountability for models but not for agent actions or tool use. Organizations may have clear responsibility assignments for model development and deployment while lacking accountability for what agents do with those models during execution. Another common gap is accountability assignments that exist in documentation but are not enforced or tracked during execution, making it impossible to prove who was actually responsible for a specific decision or outcome.

7.19 RCCS-09: Contestability

Regulatory Basis

Contestability requirements should be read across several legal and governance sources rather than collapsed into a single remedy model. EU AI Act Article 86 is best treated here as an affected-person explanation-right context for certain high-risk AI decisions. GDPR Article 22 is closer to the safeguard pattern for automated decisions, including human intervention, the ability to express one's point of view, and the ability to contest a decision. ISO/IEC 42001 requires organizations to establish mechanisms for challenging AI system decisions. Singapore MGF for Agentic AI emphasizes enabling end-user responsibility, which includes the ability to challenge outcomes.

Contestability is the principle that affected persons should have mechanisms to understand, challenge, or seek review of relevant AI-mediated outcomes where the applicable legal and governance context requires it. For agentic systems, contestability requires not only dispute mechanisms but also the evidence needed to reconstruct what happened and why. Remediation and dispute closure are treated in this chapter as engineering governance patterns, not as legal conclusions that a particular remedy is required or satisfied.

Core Question

Does the system support dispute, appeal, and remediation mechanisms for agent decisions and outcomes?

RCCS-M Adjustment Note

RCCS-M adjustment: can challenged outcomes be replayed with partitioned evidence, reviewer role, accepted/rejected state, remediation action, selective disclosure, and closure acceptance?

Scoring Rubric

Table T-07-11: RCCS-09 Contestability Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of contestability capability No dispute or appeal mechanisms documented or observable
1	Conceptual support Documentation mentions contestability but provides no implementation primitives
2	Partial implementation Dispute mechanisms exist for final outputs but do not extend to intermediate agent decisions; no remediation tracking or evidence reconstruction
3	Moderate implementation Dispute workflows, appeal mechanisms, and remediation tracking exist for most decisions; some gaps in evidence reconstruction, cross-agent dispute handling, or remediation verification
4	Strong implementation Comprehensive contestability covering dispute workflows, appeal mechanisms, evidence reconstruction, remediation tracking, and closure verification for all agent decisions and outcomes
5	Complete capability demonstrated by implementation evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove implementation

System Mapping Guidance

Contestability evaluation requires examining four capabilities: dispute workflows, evidence reconstruction, remediation tracking, and closure verification.

Dispute workflows require mechanisms for users to challenge decisions and outcomes. Look for dispute submission interfaces, dispute routing, dispute tracking, and dispute resolution workflows. Evaluate whether disputes can be filed for specific agent decisions or outcomes and whether disputes are routed to appropriate human reviewers. Systems without structured dispute mechanisms score lower.

Evidence reconstruction requires the ability to retrieve and present the evidence underlying a disputed decision. For agentic systems, this includes agent decision rationale, tool outputs, data sources, authority boundaries, and human approvals. Look for evidence retrieval interfaces that can reconstruct the full context of a decision. Systems that cannot reconstruct decision evidence score lower.

Remediation tracking requires mechanisms for documenting and verifying corrective actions. When a dispute is upheld, look for remediation workflows that specify what correction is required, who is responsible for implementing it, and how completion is verified. Systems that resolve disputes without tracking remediation score lower.

Closure verification requires confirmation that disputes are resolved and remediation is complete. Look for closure workflows that require explicit acceptance by the disputing party or an independent reviewer. Evaluate whether closure is recorded as evidence and whether closed disputes can be reopened if new information emerges. Systems that close disputes without verification score lower.

Common Gaps

The most common gap is contestability for final outputs but not for intermediate agent decisions. Organizations may allow users to challenge final outcomes while lacking mechanisms to challenge specific agent decisions, tool selections, or authority transitions that contributed to the outcome. Another common gap is dispute mechanisms that exist but cannot reconstruct the evidence needed to evaluate the dispute, forcing manual investigation and delaying resolution.

7.20 RCCS-10: Monitoring

Regulatory Basis

Monitoring requirements are established in EU AI Act Article 72, which requires providers to establish a post-market monitoring system proportionate to the nature of the AI technologies and the risks of the high-risk AI system. ISO/IEC 42001 requires organizations to establish monitoring and measurement processes for AI systems. NIST AI RMF Manage function includes continuous monitoring as a key practice for detecting and responding to AI system issues.

Monitoring is the mechanism for detecting when systems deviate from expected behavior, when risks materialize, or when governance controls fail. For agentic systems, monitoring must cover not only model performance but also agent behavior, authority drift, and lifecycle conformance.

Core Question

Does the system support continuous monitoring, incident detection, and drift detection across agent execution?

RCCS-M Adjustment Note

RCCS-M adjustment: does monitoring detect authority drift, lifecycle drift, privacy events, substitution changes, incidents, reuse/revalidation needs, and remediation closure?

Scoring Rubric

Table T-07-12: RCCS-10 Monitoring Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No evidence of monitoring capability

No monitoring or alerting features documented or observable

1

Conceptual support

Documentation mentions monitoring but provides no implementation primitives

2

Partial implementation

Monitoring exists for model performance but does not extend to agent behavior, authority drift, or lifecycle conformance

3

Moderate implementation

Monitoring dashboards, alerting, and incident detection exist for models and some agent behaviors; some gaps in authority drift detection, lifecycle conformance monitoring, or incident response

4

Strong implementation

Comprehensive continuous monitoring covering model performance, agent behavior, authority drift, lifecycle conformance, incident detection, and incident response with dashboards, alerting, and automated response

5

Complete capability demonstrated by implementation evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove implementation

System Mapping Guidance

Monitoring evaluation requires examining four capabilities: monitoring dashboards, alerting mechanisms, drift detection, and incident response.

Monitoring dashboards require visibility into system behavior and performance. Look for dashboards that show model performance metrics, agent execution statistics, tool usage patterns, authority boundary violations, and governance control status. Evaluate whether dashboards are designed for different audiences including operators, governance teams, and auditors. Systems without monitoring dashboards score lower.

Alerting mechanisms require automated detection and notification of anomalies, policy violations, or risk events. Look for configurable alert rules, alert routing, and alert escalation. For agentic systems, evaluate whether alerts cover not only model performance degradation but also agent behavior anomalies, authority drift, and governance control failures. Systems that require manual monitoring score lower.

Drift detection requires mechanisms for identifying when system behavior deviates from baseline. Look for model drift detection, data drift detection, and authority drift detection. Evaluate whether drift detection is automated and whether detected drift triggers alerts or intervention workflows. Systems without drift detection score lower.

Incident response requires workflows for investigating and resolving detected issues. Look for incident creation, incident assignment, incident investigation, and incident closure workflows. Evaluate whether incident

response is integrated with monitoring and alerting and whether incidents are tracked to closure. Systems that detect issues but lack incident response workflows score lower.

Common Gaps

The most common gap is monitoring for model performance but not for agent behavior or authority drift. Organizations may have strong ML monitoring practices while lacking visibility into whether agents are operating within their intended authority boundaries or whether agent behavior is drifting over time. Another common gap is monitoring that detects issues but lacks automated incident response, requiring manual investigation and intervention for every alert.

7.21 Chapter Summary

This chapter introduced the Regulatory Compliance Coverage Score (RCCS) family used in this white paper. RCCS-T measures how strongly a system covers existing regulatory and governance requirements across ten dimensions: risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring. RCCS-M asks whether that coverage can be expressed through Missing Regulatory Objects and lifecycle responsibility semantics. Both are analytical frameworks for comparing system capabilities against regulatory ideals, not legal compliance certifications.

The ten RCCS dimensions map directly to recurring requirements in EU AI Act, GDPR, NIST AI RMF, ISO/IEC 42001, and Singapore MGF for Agentic AI. Risk management corresponds to EU AI Act Article 9 and NIST AI RMF core functions. Data governance corresponds to GDPR Article 5 and EU AI Act Article 10. Documentation corresponds to EU AI Act Article 11. Record-keeping corresponds to EU AI Act Article 12 and GDPR Article 30. Transparency corresponds to EU AI Act Article 13 and GDPR Articles 13-14. Human oversight corresponds to EU AI Act Article 14. Security corresponds to EU AI Act Article 15 and GDPR Article 32. Accountability corresponds to GDPR Article 5(2) and EU AI Act Article 26. Contestability is informed by EU AI Act Article 86 explanation-right context, GDPR Article 22 automated-decision safeguards, and related governance guidance rather than by a single legal remedy workflow. Monitoring corresponds to EU AI Act Article 72. These mappings demonstrate that RCCS is grounded in existing regulatory requirements and governance patterns, while RCCS-M separately asks whether those patterns can be expressed as lifecycle responsibility objects.

RCCS-T provides a baseline for evaluating whether systems provide the governance primitives that regulation increasingly requires. However, traditional coverage alone is insufficient for agentic systems. A system may score well on RCCS-T while lacking the lifecycle objects required to prove that agentic work moved from intent to accepted outcome under controlled authority, evidence, privacy, and remediation constraints. RCCS-M makes that missing adjustment explicit. Chapter 8 introduces the Agentic Lifecycle Conformance Score (ALCS), which measures a different layer: whether lifecycle responsibility conformance is mature across the object set. RCCS-T, RCCS-M, and ALCS together provide a three-profile assessment of system governance capability.

Chapter 8: Agentic Lifecycle Conformance Score (ALCS)

MAIN RESULT ALCS measures lifecycle conformance surfaces that RCCS alone does not capture.	METHOD / LENS ALCS dimensions mapped to MROs.
EVIDENCE BASIS Agentic systems need authority, responsibility, evidence, privacy, substitution, and closure posture review.	KEY TAKEAWAY / BOUNDARY ALCS is an analytical lifecycle posture model, not deployment-readiness certification.

Version: v0.3.2-FRC-R3

8.0 Chapter Overview

This chapter introduces the **Agentic Lifecycle Conformance Score (ALCS)**, the lifecycle conformance profile used alongside RCCS-T and RCCS-M in this white paper. ALCS measures whether a system defines the agentic lifecycle objects required for lifecycle responsibility compliance. It evaluates whether a system can express the sixteen Missing Regulatory Objects introduced in Chapter 6.

ALCS is not a legal compliance certification, not a conformity assessment, and not a regulatory approval. It is an analytical framework for comparing system capabilities against lifecycle responsibility requirements. A system can have a strong RCCS-T profile and still have a weak ALCS score. That is not a contradiction. It means the system may be strong in existing regulatory compliance coverage while lacking deeper lifecycle responsibility semantics for multi-agent work.

RCCS-T measures traditional regulatory compliance coverage. RCCS-M measures whether that coverage can be expressed through Missing Regulatory Objects. ALCS measures agentic lifecycle conformance. These are different layers, not competing scores. RCCS-M asks whether regulatory obligations can be represented through MROs. ALCS asks whether lifecycle responsibility conformance is mature across responsibility, authority, evidence, privacy, substitution, and remediation.

The chapter defines the fifteen ALCS dimensions, explains the ALCS-to-MRO mapping, references the shared scoring procedure established in Chapter 7, and establishes clear boundaries around what ALCS measures and what it does not measure. Detailed scoring rubrics for each dimension are provided in sections 8.7 through 8.21 and expanded in Appendix F.

8.1 What ALCS Measures

ALCS measures whether a system defines the agentic lifecycle objects required for lifecycle responsibility compliance. It evaluates whether a system can express human-role-to-MAS responsibility mappings, delegated authority boundaries, agent role definitions, accepted outcome states, tool-action liability records, responsibility transfer semantics, authority drift detection, evidence partitioning, cross-project reuse controls,

privacy lifecycle mappings, privacy-preserving validation protocols, evidence minimization policies, data subject rights reconciliation, processor chain tracking, and vendor substitution conformance.

ALCS is not a legal compliance score. A high ALCS score does not mean a system is legally compliant. A low ALCS score does not mean a system is non-compliant. Legal compliance depends on organizational practice, deployment context, jurisdiction, risk class, and how the system is used. ALCS measures system capability to express lifecycle responsibility objects, not organizational conformance to legal obligations.

ALCS is an analytical framework for comparing systems. It allows buyers, auditors, governance teams, and protocol designers to evaluate whether a system provides the lifecycle primitives required to prove that agentic work moved from intent to accepted outcome under controlled authority, evidence, privacy, and remediation constraints. It makes visible the difference between systems that provide lifecycle responsibility semantics and systems that provide only execution orchestration or observability features.

The purpose of ALCS is to translate the sixteen Missing Regulatory Objects into measurable system capabilities. Regulation requires human oversight, but what does that mean in a multi-agent system? ALCS defines fifteen dimensions that operationalize lifecycle responsibility requirements into system features that can be evaluated, compared, and improved. It is a bridge between regulatory abstractions and lifecycle implementation.

A system can have a strong RCCS-T score and weaker RCCS-M or ALCS scores. This is not a contradiction. It means the system may provide strong traditional regulatory compliance coverage through risk management, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring capabilities while lacking the deeper lifecycle responsibility objects required to prove authority boundaries, responsibility transfer, evidence partitioning, privacy lifecycle mapping, and vendor substitution conformance in multi-agent execution. RCCS-M and ALCS are related but not identical: RCCS-M adjusts regulatory coverage through MROs, while ALCS evaluates lifecycle conformance maturity.

8.2 ALCS Dimensions Overview

ALCS evaluates systems across fifteen dimensions. Each dimension corresponds to one or more Missing Regulatory Objects. The dimensions are weighted to reflect their relative importance in lifecycle responsibility compliance.

The fifteen ALCS dimensions are:

1. **Human-Role-to-MAS Responsibility Mapping (8%)** - Does the system map human roles to agent roles, delegated scopes, and evidence obligations?
2. **Delegated Authority Boundary (8%)** - Does the system define and enforce authority boundaries for agent actions?
3. **Agent Role vs Human Role Distinction (5%)** - Does the system distinguish agent roles from human roles with clear responsibility semantics?
4. **Accepted Outcome Compliance (8%)** - Does the system separate execution completion from outcome acceptance?
5. **Tool-Action Liability Boundary (8%)** - Does the system record liability boundaries for consequential tool actions?
6. **Responsibility Transfer Across Agents (8%)** - Does the system track responsibility transfer during agent handoffs?
7. **Authority Drift Detection (5%)** - Does the system detect unauthorized authority transitions?
8. **MAS Evidence Partitioning (8%)** - Does the system partition evidence by agent, tool, authority, and privacy boundaries?

9. **Cross-Project Reuse Compliance (5%)** - Does the system enforce revalidation when workflows are reused across projects?
10. **Privacy/GDPR Lifecycle Mapping (8%)** - Does the system map personal data flows across the agent lifecycle?
11. **Privacy-Preserving Third-Party Validation (5%)** - Does the system support validation without raw data disclosure?
12. **Evidence Minimization and Selective Disclosure (5%)** - Does the system minimize evidence retention and support selective disclosure?
13. **Data Subject Rights vs Evidence Retention (5%)** - Does the system reconcile data subject rights with evidence retention obligations?
14. **Third-Party Processor/Subprocessor Chain (5%)** - Does the system track processor and subprocessor usage at lifecycle granularity?
15. **Vendor/Model/Runtime Substitution Conformance (5%)** - Does the system enforce conformance revalidation when vendors are substituted?

The following table summarizes the fifteen dimensions, their MRO mappings, core questions, and weights.

Table T-08-01: ALCS Dimensions Summary

Dimension	MRO Mapping	Core Question	Weight
Human-Role-to-MAS Responsibility Mapping	MRO-01	Does the system map human roles to agent roles, delegated scopes, and evidence obligations?	8%
Delegated Authority Boundary	MRO-02	Does the system define and enforce authority boundaries for agent actions?	8%
Agent Role vs Human Role Distinction	MRO-03	Does the system distinguish agent roles from human roles with clear responsibility semantics?	5%
Accepted Outcome Compliance	MRO-04	Does the system separate execution completion from outcome acceptance?	8%
Tool-Action Liability Boundary	MRO-05	Does the system record liability boundaries for consequential tool actions?	8%
Responsibility Transfer Across Agents	MRO-06	Does the system track responsibility transfer during agent handoffs?	8%
Authority Drift Detection	MRO-07	Does the system detect unauthorized authority transitions?	5%
MAS Evidence Partitioning	MRO-08	Does the system partition evidence by agent, tool, authority, and privacy boundaries?	8%
Cross-Project Reuse Compliance	MRO-09	Does the system enforce revalidation when workflows are reused across projects?	5%
Privacy/GDPR Lifecycle Mapping	MRO-10	Does the system map personal data flows across the agent lifecycle?	8%
Privacy-Preserving Third-Party Validation	MRO-11	Does the system support validation without raw data disclosure?	5%
Evidence Minimization and Selective Disclosure	MRO-12	Does the system minimize evidence retention and support selective disclosure?	5%
Data Subject Rights vs Evidence Retention	MRO-13	Does the system reconcile data subject rights with evidence retention obligations?	5%

Dimension	MRO Mapping	Core Question	Weight
Third-Party Processor/Subprocessor Chain	MRO-14	Does the system track processor and subprocessor usage at lifecycle granularity?	5%
Vendor/Model/Runtime Substitution Conformance	MRO-15	Does the system enforce conformance revalidation when vendors are substituted?	5%

Note: [MRO-16](#) (Incident, Dispute, and Remediation Closure) is not directly mapped to a single ALCS dimension. It is a cross-cutting lifecycle capability that spans multiple dimensions including contestability, evidence partitioning, and continuous improvement. Incident closure capability is evaluated within the RCCS Contestability and Monitoring dimensions rather than as a separate ALCS dimension.

8.3 ALCS-to-MRO Mapping

Each ALCS dimension maps to one or more Missing Regulatory Objects. The mapping shows how ALCS dimensions operationalize MRO requirements into measurable system capabilities. ALCS dimensions are not merely restatements of MROs. They are evaluation criteria that translate MRO requirements into system features that can be scored, compared, and improved.

The mapping is primarily one-to-one: each ALCS dimension corresponds to a single primary MRO. However, some dimensions have secondary MRO relationships because lifecycle responsibility objects are interconnected. For example, Human-Role-to-MAS Responsibility Mapping ([ALCS-01](#)) primarily maps to [MRO-01](#) but also relates to [MRO-03](#) (Agent Role vs Human Role) and [MRO-06](#) (Responsibility Transfer) because responsibility mapping, role distinction, and transfer semantics are interdependent.

The following table provides detailed ALCS-to-MRO mapping, including primary MRO, secondary MROs, lifecycle phase, and core object required.

Table T-08-02: ALCS-to-MRO Mapping

ALCS Dimension	Primary MRO	Secondary MROs	Lifecycle Phase	Core Object Required
ALCS-01 : Human-Role-to-MAS Responsibility Mapping	MRO-01	MRO-03 , MRO-06	Intent, delegation, acceptance	Role-responsibility map binding human role IDs, agent role IDs, delegated scopes, evidence obligations
ALCS-02 : Delegated Authority Boundary	MRO-02	MRO-07	Delegation, execution, monitoring	Authority boundary object binding role, action, scope, condition, risk class, expiry, escalation path
ALCS-03 : Agent Role vs Human Role Distinction	MRO-03	MRO-01	Role definition, delegation	Agent responsibility boundary declaring duties, non-duties, allowed tools, inherited constraints, human accountability linkage
ALCS-04 : Accepted Outcome Compliance	MRO-04	MRO-01 , MRO-08	Review, acceptance, dispute	Accepted outcome state linking intent, plan, trace, reviewer, acceptance status, dispute window, remediation route
ALCS-05 : Tool-Action Liability Boundary	MRO-05	MRO-02 , MRO-08	Execution, external action	Tool-action liability object recording initiator, tool identity, external system, authority credential, reversibility, rollback plan
ALCS-06 : Responsibility Transfer Across Agents	MRO-06	MRO-01 , MRO-08	Agent handoff, delegation	Transfer object declaring transferred scope, source

ALCS Dimension	Primary MRO	Secondary MROs	Lifecycle Phase	Core Object Required
				role, receiving role, retained responsibilities, inherited constraints
ALCS-07 : Authority Drift Detection	MRO-07	MRO-02	Monitoring, drift detection	Drift event model comparing observed behavior against original authority boundary, risk class, tool scope
ALCS-08 : MAS Evidence Partitioning	MRO-08	MRO-01 , MRO-04 , MRO-10	Evidence collection, audit	Partitioned evidence by agent, tool, human confirmation, plan version, authority boundary, privacy boundary, accepted outcome
ALCS-09 : Cross-Project Reuse Compliance	MRO-09	MRO-02 , MRO-10	Reuse, revalidation	Reuse compliance record linking original validation context to new deployment context, authority reset, privacy review
ALCS-10 : Privacy/GDPR Lifecycle Mapping	MRO-10	MRO-08 , MRO-12	All phases involving personal data	Lifecycle data-flow map showing personal data entry, flow, exit across agents, tools, memory, handoffs
ALCS-11 : Privacy-Preserving Third-Party Validation	MRO-11	MRO-08 , MRO-12	External validation, audit	Privacy-preserving validation protocol allowing selective disclosure and zero-knowledge proofs
ALCS-12 : Evidence Minimization and Selective Disclosure	MRO-12	MRO-08 , MRO-10	Evidence retention, disclosure	Evidence minimization policy defining retention periods, redaction rules, disclosure boundaries
ALCS-13 : Data Subject Rights vs Evidence Retention	MRO-13	MRO-10 , MRO-12	Data subject request handling	Data subject rights policy defining retention periods, anonymization rules, legal basis for retention override
ALCS-14 : Third-Party Processor/Subprocessor Chain	MRO-14	MRO-10	All phases involving third-party processing	Dynamic processor chain record tracking processor identity, data categories, processing purpose, legal basis
ALCS-15 : Vendor/Model/Runtime Substitution Conformance	MRO-15	MRO-02 , MRO-09	Vendor substitution, revalidation	Vendor substitution conformance record proving substitution preserved lifecycle responsibility properties

This mapping demonstrates that ALCS dimensions are operationalized versions of MRO requirements. Each dimension translates an MRO into a measurable system capability. The mapping also shows that lifecycle responsibility objects are interconnected: responsibility mapping relates to role distinction and transfer semantics; authority boundaries relate to drift detection; evidence partitioning relates to privacy mapping and selective disclosure; reuse compliance relates to authority reset and privacy review.

8.3.1 Treatment of MRO-16 as a Cross-Cutting Closure Requirement

MRO-16 (Incident, Dispute, and Remediation Closure) is not directly mapped to a single ALCS dimension. Instead, it is treated as a cross-cutting lifecycle capability that is evaluated through multiple dimensions and scoring logic across RCCS-T, RCCS-M, and ALCS.

ALCS directly operationalizes fifteen dimension-level MROs ([MRO-01](#) through [MRO-15](#)). [MRO-16](#) is cross-cutting rather than omitted. It is evaluated through contestability, monitoring, accepted outcome, evidence partitioning, evidence minimization, and remediation-related scoring logic. This prevents incident, dispute, and remediation closure from being treated as a narrow isolated dimension and instead recognizes it as a lifecycle-wide capability.

The following table shows how [MRO-16](#) is evaluated across multiple dimensions:

Table T-08-02A: [MRO-16](#) Evaluation Path

MRO-16 Evaluation Path	Relevant Dimension / Chapter	Reason
Contestability	RCCS-09 (Chapter 7)	Disputes require challenge and review paths. Incident closure depends on dispute resolution mechanisms.
Monitoring	RCCS-10 (Chapter 7)	Incidents require detection and recurrence monitoring. Continuous improvement depends on monitoring capability.
Accepted Outcome	ALCS-04 (Chapter 8)	Closure requires knowing whether the outcome was accepted, rejected, or remediated. Dispute resolution affects acceptance state.
Evidence Partitioning	ALCS-08 (Chapter 8)	Closure requires reconstructable evidence linking incident detection to root cause, remediation, and resolution.
Evidence Minimization	ALCS-12 (Chapter 8)	Closure evidence must preserve accountability without unnecessary disclosure. Incident investigation must respect privacy boundaries.
Composite interpretation	Chapter 9	Incident closure affects final system readiness interpretation. Systems with strong incident closure capability score higher on contestability, monitoring, and evidence dimensions.

This cross-cutting treatment ensures that incident, dispute, and remediation closure is evaluated as a lifecycle-wide capability rather than as a single isolated dimension. Systems that provide strong contestability, monitoring, accepted outcome tracking, evidence partitioning, and evidence minimization capabilities will naturally support incident closure. Systems that lack these capabilities will struggle to close incidents in an auditable, privacy-preserving, and accountability-preserving manner.

8.4 Shared Scoring Procedure (Reference to Chapter 7)

ALCS uses the same scoring procedure as RCCS. This ensures consistency and comparability across both scoring frameworks. The shared scoring procedure was defined in Chapter 7.3 and includes the raw scoring scale, evidence confidence multiplier, dimension score formula, and cross-framework composite score.

Raw scoring scale (0-5): Each ALCS dimension is scored on a 0-5 raw scale based on the strength of system support:

- **0 = No capability; no evidence**
- **1 = Minimal capability; weak evidence**
- **2 = Partial capability; some evidence**
- **3 = Moderate capability; documented evidence**
- **4 = Strong capability; validated evidence**
- **5 = Complete capability demonstrated by implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts. Legal text may establish a regulatory baseline but does not by itself prove implementation.**

Evidence confidence multiplier: The evidence multiplier adjusts the dimension score based on the quality and verifiability of evidence used to evaluate the system:

- **L1** : 1.00
- **L2** : 0.85
- **L3** : 0.75
- **L4** : 0.55
- **L5** : 0.35

Dimension score formula: Each ALCS dimension is normalized using:

Dimension Score = (Raw Score / 5) × 100 × Evidence Confidence Multiplier

Optional composite views: RCCS-T, RCCS-M, and ALCS may be combined only as secondary analytical views:

- **Traditional Composite = 0.40 × RCCS-T + 0.60 × ALCS**
- **Agentic Composite = 0.40 × RCCS-M + 0.60 × ALCS**

The primary result display remains the three-profile view: RCCS-T, RCCS-M, and ALCS. Composite views must not be used as legal compliance proof, vendor rankings, certification signals, or procurement recommendations. For complete details on the scoring procedure, including scoring rationale, evidence hierarchy, and calculation examples, see Chapter 7.3 and Chapter 9.

8.5 ALCS Boundary Statement

ALCS is an analytical scoring framework, not a legal compliance score. This boundary must be understood clearly to avoid misinterpretation and over-claiming.

What ALCS is:

- An analytical framework for comparing system capabilities against lifecycle responsibility requirements
- A structured method for evaluating whether a system provides lifecycle objects required to prove agentic responsibility
- A tool for buyers, auditors, and governance teams to assess system readiness for multi-agent execution
- A bridge between regulatory abstractions and lifecycle implementation

What ALCS is not:

- A legal compliance certification or conformity assessment
- A regulatory approval or endorsement
- A guarantee that a system meets all legal obligations
- A substitute for legal counsel, compliance review, or risk assessment

Key boundary principles:

1. **High ALCS score does NOT mean legal compliance.** A system may score well on ALCS while the organization using it fails to comply with legal obligations due to deployment gaps, policy weaknesses, incomplete adoption, or jurisdictional differences. ALCS measures system capability to express lifecycle objects, not organizational conformance to legal requirements.
1. **Low ALCS score does NOT mean non-compliance.** A system may score lower on ALCS while the organization achieves compliance through compensating controls, manual processes, or alternative

governance mechanisms. ALCS measures lifecycle object support, not all possible paths to compliance.

1. **ALCS measures system capability, not organizational practice.** A system may provide strong lifecycle objects while the organization fails to use them. Conversely, an organization may achieve strong lifecycle governance through processes that are not reflected in system capabilities. ALCS evaluates what the system can express, not what the organization actually does.
1. **ALCS is context-independent, but compliance is context-dependent.** ALCS evaluates system capabilities without regard to deployment context, risk class, jurisdiction, or use case. Legal compliance depends on all of these factors. A system with a high ALCS score may still be non-compliant in specific contexts, and a system with a low ALCS score may be compliant in other contexts.

Relationship between RCCS-T, RCCS-M, and ALCS:

RCCS-T, RCCS-M, and ALCS measure different layers. RCCS-T measures traditional regulatory compliance coverage: whether a system provides the governance primitives that existing regulation requires. RCCS-M measures whether that regulatory coverage can be expressed through Missing Regulatory Objects. ALCS measures agentic lifecycle conformance: whether a system provides mature lifecycle responsibility objects that make agentic responsibility auditable, provable, and transferable.

A system can have strong RCCS-T and weaker RCCS-M or ALCS. This means the system provides strong regulatory compliance coverage through risk management, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring capabilities while lacking deeper lifecycle responsibility objects for multi-agent execution. This is not a contradiction. It reflects the reality that existing regulation focuses on model governance and platform controls while lifecycle responsibility semantics for multi-agent systems are still emerging.

A system can have weaker RCCS-T and stronger RCCS-M or ALCS. This means the system provides strong lifecycle objects for expressing agentic responsibility while lacking broader productized governance coverage. This pattern is possible for specialized lifecycle protocols or agent frameworks that focus on responsibility semantics without providing full governance platform capabilities.

All three profiles are necessary. RCCS-T shows conventional governance coverage. RCCS-M shows whether that coverage survives MRO adjustment. ALCS shows lifecycle conformance maturity. Together, they provide a three-profile assessment of system governance capability.

The purpose of this boundary statement is to prevent over-claiming. ALCS is a useful analytical tool, but it is not a compliance certification. Organizations must not rely on ALCS scores as proof of legal compliance. Buyers must not treat high ALCS scores as a substitute for due diligence, legal review, or risk assessment. ALCS measures system capability to express lifecycle objects. Legal compliance requires organizational practice, deployment context, and jurisdictional analysis.

8.6 Bridge to Dimension Sections

The following sections (8.7 through 8.21) detail each ALCS dimension. Each dimension section includes:

- **MRO mapping** - Which Missing Regulatory Objects inform the dimension
- **Core question** - What the dimension evaluates
- **Scoring criteria** - How to apply the 0-5 raw scale to system capabilities
- **System mapping guidance** - How to evaluate whether a system supports the dimension

The dimension sections provide the detailed rubrics required to score systems consistently and reproducibly. They translate lifecycle responsibility abstractions into concrete system features that can be evaluated through

documentation review, API inspection, and evidence analysis.

The body chapter includes compact dimension rubrics for reader continuity; Appendix F provides the reusable detailed ALCS scoring reference.

8.7 ALCS-01: Human-Role-to-MAS Responsibility Mapping

MRO Mapping

This dimension primarily maps to [MRO-01](#) (Human Role to MAS Responsibility Mapping) with secondary relationships to [MRO-03](#) (Agent Role vs Human Role) and [MRO-06](#) (Responsibility Transfer Across Agents). The core requirement is that human organizational accountability must be explicitly mapped to agent execution responsibilities.

Core Question

Does the system map human roles to agent roles, delegated scopes, evidence obligations, and acceptance authority?

Scoring Rubric

Table T-08-03: [ALCS-01](#) Human-Role-to-MAS Responsibility Mapping Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No role-to-responsibility mapping features documented or observable
1	Conceptual support Documentation mentions human oversight or role mapping but provides no implementation primitives
2	Partial implementation Human roles and agent roles exist but are not explicitly mapped; no evidence obligations or acceptance authority tracking
3	Moderate implementation Role-to-responsibility mappings exist for most workflows; some gaps in evidence obligation tracking, acceptance authority, or cross-project mapping
4	Strong implementation Comprehensive role-responsibility graph linking human role IDs, agent role IDs, delegated scopes, reserved decisions, review duties, evidence obligations, and accepted-outcome authority

5

Complete capability demonstrated by implementation evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

When evaluating a system for human-role-to-MAS responsibility mapping, examine whether the system maintains explicit bindings between human organizational roles and agent execution responsibilities. Look for role definition objects that specify not only who can initiate or approve agent work, but also who owns intent, who approves risk, who reviews evidence, and who accepts outcomes.

Strong systems provide role-responsibility graphs where each human role is linked to specific agent roles, with clear documentation of what responsibilities are delegated, what responsibilities are retained, and what evidence obligations apply. The mapping should be durable and auditable: given a completed workflow, the system should be able to reconstruct which human role held which responsibility at each lifecycle phase.

Weak systems may have user accounts, approval buttons, or workflow assignments without explicit responsibility semantics. A user may be listed as an approver without clarity about whether they own business acceptance, risk approval, or evidence review. The test is whether the system can answer: for this completed run, who owned intent, who approved risk, who executed each step, who reviewed evidence, and who accepted the outcome?

Common Gaps

The most common gap is human oversight that exists at the UI level without responsibility mapping at the lifecycle level. Organizations may have approval workflows where humans click buttons, but the system cannot prove which human role held which responsibility. Another common gap is responsibility mappings that apply to initial delegation but do not track responsibility changes during multi-agent handoffs or cross-project reuse.

8.8 ALCS-02: Delegated Authority Boundary

MRO Mapping

This dimension primarily maps to **MRO-02** (Delegated Authority Boundary) with secondary relationships to **MRO-07** (Authority Drift). The core requirement is that technical permission must be distinguished from business authority, and authority boundaries must be enforced at execution time.

Core Question

Does the system define and enforce delegated authority boundaries for agent actions?

Scoring Rubric

Table T-08-04: **ALCS-02** Delegated Authority Boundary Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No evidence of capability

No authority boundary features documented or observable

1

Conceptual support

Documentation mentions authority or delegation but provides no implementation primitives

2

Partial implementation

Tool permissions exist but are not mapped to business authority; no scope, condition, or risk class enforcement

3

Moderate implementation

Authority boundaries exist for most high-impact actions; some gaps in condition enforcement, expiry tracking, or escalation paths

4

Strong implementation

Comprehensive authority boundary objects binding role, action, scope, condition, risk class, evidence requirement, expiry, revocation, and escalation path with runtime enforcement

5

Complete capability demonstrated by implementation evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating delegated authority requires distinguishing between access control and authority control. A system may grant an agent technical permission to call a tool while lacking business authority to use that tool for a particular action. Look for authority boundary objects that specify not only what an agent can do, but under what conditions, for what scope, at what risk level, and with what escalation requirements.

Strong systems check authority boundaries before high-impact tool calls and enforce scope limitations, condition requirements, and expiry rules. Authority should be evaluated against intent and plan state, not only against static credentials. The system should fail closed when authority cannot be reconstructed, and should escalate when observed actions exceed delegated authority.

Weak systems may rely on IAM permissions, API keys, or tool availability without authority semantics. The same tool call may be authorized in one context and unauthorized in another, but the system treats all calls with valid credentials as authorized. The test is whether the system can show the exact authority basis and escalation rule for a high-impact tool call before it executes.

Common Gaps

The most common gap is authority boundaries that are documented in policies but not enforced at runtime. Organizations may define authority scopes in documentation while allowing agents to execute any action for

which they have technical permission. Another common gap is authority boundaries that apply to initial delegation but do not detect authority drift during long-running workflows or autonomous retries.

8.9 ALCS-03: Agent Role vs Human Role Distinction

MRO Mapping

This dimension primarily maps to MRO-03 (Agent Role is not Human Role) with secondary relationships to MRO-01 (Human Role to MAS Responsibility Mapping). The core requirement is that agent roles must be governance identities with bounded responsibility surfaces, not merely persuasive persona labels.

Core Question

Does the system distinguish agent roles from human roles with clear responsibility semantics?

Scoring Rubric

Table T-08-05: ALCS-03 Agent Role vs Human Role Distinction Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No role distinction features documented or observable
1	Conceptual support Documentation mentions agent roles but provides no responsibility semantics
2	Partial implementation Agent roles exist as execution personas but lack responsibility boundaries, evidence duties, or human accountability linkage
3	Moderate implementation Agent roles include responsibility boundaries and capability definitions; some gaps in evidence duties, escalation requirements, or human accountability linkage
4	Strong implementation Comprehensive agent responsibility boundaries declaring duties, non-duties, allowed tools, inherited constraints, verification expectations, evidence obligations, and human accountability linkage
5	Complete capability demonstrated by implementation evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating agent role distinction requires examining whether agent roles carry governance semantics beyond execution personas. Look for role objects that define not only what an agent can do, but what it must prove, what it must escalate, and which human role remains accountable for acceptance or rejection.

Strong systems separate UX persona labels from governance role definitions. An agent labeled "reviewer" should have a role object that specifies review duties, review criteria, evidence requirements, escalation conditions, and the human role accountable for final acceptance. The role definition should survive if the persona label is replaced with a neutral identifier.

Weak systems may use human-like role labels such as researcher, writer, analyst, or manager without defining responsibility boundaries. These labels are useful for UI and collaboration metaphors, but they can obscure the fact that legal and business accountability remains with humans and organizations. The test is whether the governance meaning remains intact if the agent name is replaced with a neutral identifier.

Common Gaps

The most common gap is agent roles that exist as execution personas without responsibility semantics. Organizations may define agents with human-like job titles while lacking clarity about what responsibilities those agents carry, what evidence they must produce, or which human roles remain accountable. Another common gap is role definitions that specify capabilities without specifying non-delegable human responsibilities or escalation duties.

8.10 ALCS-04: Accepted Outcome Compliance

MRO Mapping

This dimension primarily maps to **MRO-04** (Accepted Outcome Compliance) with secondary relationships to **MRO-01** (Human Role to MAS Responsibility Mapping) and **MRO-08** (MAS Evidence Partitioning). The core requirement is that execution completion must be separated from outcome acceptance.

Core Question

Does the system separate execution completion from outcome acceptance and track acceptance authority?

Scoring Rubric

Table T-08-06: **ALCS-04** Accepted Outcome Compliance Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No acceptance tracking features documented or observable
1	Conceptual support Documentation mentions review or acceptance but provides no implementation primitives

2

Partial implementation

Execution completion states exist but are not distinguished from acceptance states; no acceptance criteria or reviewer tracking

3

Moderate implementation

Acceptance workflows exist for most deliverables; some gaps in acceptance criteria definition, dispute windows, or remediation routes

4

Strong implementation

Comprehensive accepted outcome states linking source intent, plan, trace, reviewer identity, acceptance criteria, acceptance status, acceptance timestamp, dispute window, and remediation route

5

Complete capability demonstrated by implementation evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating accepted outcome compliance requires examining whether the system distinguishes between completed execution and accepted delivery. Look for acceptance state objects that bind deliverable ID, plan version, trace reference, reviewer role, acceptance criteria, acceptance decision, and dispute window.

Strong systems enforce a two-state transition: from completed output to accepted outcome. Completion means execution reached a terminal state. Acceptance means the output was reviewed against intent, evidence, risk, and acceptance criteria by an authorized reviewer. The system should track who accepted, against what criteria, with what evidence, and what the dispute window is.

Weak systems may mark work as complete when execution terminates without requiring explicit acceptance. Downstream business processes may rely on unaccepted output. In disputes, the organization cannot distinguish whether the problem was execution quality, review failure, or missing acceptance policy. The test is whether the system can identify acceptance criteria, evidence reviewed, reviewer role, decision timestamp, and dispute window for any completed task.

Common Gaps

The most common gap is execution completion being treated as outcome acceptance. Organizations may have reliable execution engines while lacking explicit acceptance workflows, acceptance criteria, or acceptance authority tracking. Another common gap is acceptance workflows that exist for final deliverables but not for intermediate agent outputs, allowing unaccepted work to propagate through multi-agent workflows.

8.11 ALCS-05: Tool-Action Liability Boundary

MRO Mapping

This dimension primarily maps to **MRO-05** (Tool-Action Liability Boundary) with secondary relationships to **MRO-02** (Delegated Authority Boundary) and **MRO-08** (MAS Evidence Partitioning). The core requirement is that consequential tool actions must record liability boundaries including initiator, authority source, affected system, reversibility, and accountable owner.

Core Question

Does the system record liability boundaries for consequential tool actions?

Scoring Rubric

Table T-08-07: **ALCS-05** Tool-Action Liability Boundary Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No tool-action liability tracking features documented or observable
1	Conceptual support Documentation mentions tool use or external actions but provides no liability tracking primitives
2	Partial implementation Tool call logs exist but do not record authority source, affected systems, reversibility, or accountable owner
3	Moderate implementation Tool-action records exist for most consequential actions; some gaps in reversibility tracking, rollback plans, or liability assignment
4	Strong implementation Comprehensive tool-action liability objects recording initiator, tool identity, external system, authority credential, reversibility status, rollback plan, affected data categories, evidence pointer, and accountable owner
5	Complete capability demonstrated by implementation evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating tool-action liability requires examining whether the system records the full liability context for consequential tool actions. Look for tool-action records that capture not only what action was performed, but who initiated it, under what authority, affecting which external system, with what reversibility, and who owns the resulting consequence.

Strong systems distinguish between low-risk tool calls and consequential actions that affect external systems, records, customers, code, money, permissions, communications, or regulated data. Consequential actions should produce liability boundary records before execution. The system should be able to reconstruct the initiator, authority source, affected system, reversibility status, rollback path, and accountable owner for any high-impact tool action.

Weak systems may log tool calls without liability context. A tool log may show that an action happened, but not whether the agent had authority, whether the action was reversible, which system was affected, or who owns the consequence. The test is whether the system can reconstruct the full liability context for any consequential tool action without requiring tribal memory or manual investigation.

Common Gaps

The most common gap is tool call logging that captures execution events without liability semantics. Organizations may have extensive tool usage logs while lacking visibility into authority sources, affected systems, reversibility status, or accountability assignments. Another common gap is liability boundaries that apply to some tool categories but not others, leaving gaps in coverage for newly added tools or third-party integrations.

8.12 ALCS-06: Responsibility Transfer Across Agents

MRO Mapping

This dimension primarily maps to **MRO-06** (Responsibility Transfer Across Agents) with secondary relationships to **MRO-01** (Human Role to MAS Responsibility Mapping) and **MRO-08** (MAS Evidence Partitioning). The core requirement is that agent handoffs must include explicit responsibility transfer semantics, not merely task routing.

Core Question

Does the system track responsibility transfer during agent handoffs?

Scoring Rubric

Table T-08-08: **ALCS-06** Responsibility Transfer Across Agents Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No responsibility transfer tracking features documented or observable

1**Conceptual support**

Documentation mentions agent handoffs or delegation but provides no responsibility transfer primitives

2**Partial implementation**

Agent handoff routing exists but does not track responsibility transfer; no inherited constraints or receiving-role acceptance

3**Moderate implementation**

Responsibility transfer records exist for most handoffs; some gaps in inherited constraints, retained responsibilities, or transfer acceptance

4**Strong implementation**

Comprehensive transfer objects declaring transferred scope, source role, receiving role, retained responsibilities, inherited constraints, evidence obligations, receiving-role acceptance, and rejection/escalation behavior

5**Complete capability demonstrated by implementation evidence**

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating responsibility transfer requires distinguishing between workflow routing and governance transfer. Look for transfer objects that specify what responsibility was transferred, what was retained, what constraints were inherited, and whether the receiving agent accepted the transfer.

Strong systems enforce explicit responsibility transfer at agent handoffs. A receiving agent must inherit constraints, evidence duties, privacy restrictions, and escalation conditions rather than merely receiving a prompt or task payload. Upstream and downstream responsibility must remain reconstructable. The system should be able to answer: at each handoff, what responsibility was transferred, what was retained, what constraints were inherited, and did the receiving agent accept?

Weak systems may route tasks between agents without responsibility semantics. A downstream agent can claim it only followed instructions while the upstream agent no longer controls execution, leaving a gap in audit and remediation. The test is whether the system can reconstruct responsibility continuity across agent handoffs without relying on execution flow alone.

Common Gaps

The most common gap is agent handoffs that exist as workflow routing without responsibility transfer semantics. Organizations may have sophisticated multi-agent orchestration while lacking explicit tracking of what responsibilities moved between agents. Another common gap is responsibility transfer that happens implicitly through task delegation without recording transferred scope, retained scope, or inherited constraints.

8.13 ALCS-07: Authority Drift Detection

MRO Mapping

This dimension primarily maps to MRO-07 (Authority Drift) with secondary relationships to MRO-02 (Delegated Authority Boundary). The core requirement is that the system must detect when agents transition from their original authority boundaries without explicit authorization.

Core Question

Does the system detect and alert on unauthorized authority transitions?

Scoring Rubric

Table T-08-09: ALCS-07 Authority Drift Detection Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No authority drift detection features documented or observable
1	Conceptual support Documentation mentions authority monitoring but provides no drift detection primitives
2	Partial implementation Authority boundaries are defined but not monitored for drift; no drift detection rules or alerts
3	Moderate implementation Authority drift detection exists for most high-risk transitions; some gaps in drift alert routing, remediation workflows, or long-running workflow monitoring
4	Strong implementation Comprehensive authority drift monitoring comparing observed behavior against original authority boundaries with drift detection rules, drift alerts, automatic stop/downgrade, and drift remediation workflows
5	Complete capability demonstrated by implementation evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating authority drift detection requires examining whether the system monitors for unauthorized authority transitions during execution. Look for drift detection mechanisms that compare observed agent behavior

against original authority boundaries and plan state.

Strong systems detect when agents move from advice to execution, draft to send, read-only to write, or internal recommendation to external commitment without a new authority boundary or confirmation event. Drift detection should trigger stop, downgrade, or human confirmation rather than being treated as ordinary execution variance. The system should be particularly vigilant in long-running workflows, autonomous retries, or agent loops where cumulative drift can occur.

Weak systems may define authority boundaries at delegation time but not monitor for drift during execution. Each local step may appear reasonable while the cumulative lifecycle state exceeds the original delegation. The test is whether the system can detect when an agent's observed behavior exceeds its original authority boundary and halt or escalate before external consequence.

Common Gaps

The most common gap is authority boundaries that are enforced at delegation time but not monitored during execution. Organizations may have clear authority definitions while lacking runtime detection of authority transitions. Another common gap is drift detection that applies to individual tool calls but not to cumulative authority escalation across multiple steps in long-running workflows.

8.14 ALCS-08: MAS Evidence Partitioning

MRO Mapping

This dimension primarily maps to **MRO-08** (MAS Evidence Partitioning) with secondary relationships to **MRO-01** (Human Role to MAS Responsibility Mapping), **MRO-04** (Accepted Outcome Compliance), and **MRO-10** (Privacy/GDPR Lifecycle Mapping). The core requirement is that multi-agent evidence must be partitioned, linked, and reconstructable for different audit purposes.

Core Question

Does the system partition evidence by agent, tool, authority boundary, privacy class, and lifecycle phase?

Scoring Rubric

Table T-08-10: **ALCS-08** MAS Evidence Partitioning Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No evidence partitioning features documented or observable
1	Conceptual support Documentation mentions evidence or audit trails but provides no partitioning primitives
2	Partial implementation Flat chronological logs exist but are not partitioned by agent, tool, authority, or privacy boundaries

3**Moderate implementation**

Evidence partitioning exists for most governance boundaries; some gaps in cross-partition linking, privacy class partitioning, or reconstruction capability

4**Strong implementation**

Comprehensive evidence partitioning by agent, tool, human confirmation, plan version, authority boundary, privacy boundary, and accepted outcome with stable identifiers, integrity hashes, cross-partition linking, and evidence reconstruction capability

5**Complete capability demonstrated by implementation evidence**

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating evidence partitioning requires examining whether the system can partition, link, and reconstruct evidence for different audit purposes. Look for evidence partitions organized by agent, role, tool, plan version, authority boundary, privacy class, confirmation event, and accepted outcome.

Strong systems maintain evidence partitions that can be selectively disclosed without exposing the full raw log. Different auditors may need different partitions: a privacy regulator may need data-flow evidence, a financial auditor may need transaction evidence, a customer dispute may require specific agent evidence. Partitions should be linked by stable identifiers and integrity hashes to enable reconstruction while supporting selective disclosure.

Weak systems may maintain flat chronological logs that mix irrelevant telemetry with critical proof. Flat logs are difficult to use in disputes because they require exposing unnecessary sensitive data to extract relevant evidence. The test is whether the system can reconstruct evidence partitions for specific agents, tools, privacy classes, and accepted outcomes without exposing the full raw log.

Common Gaps

The most common gap is evidence that exists as flat execution logs without governance partitioning. Organizations may have comprehensive observability while lacking the ability to partition evidence by agent, authority boundary, or privacy class. Another common gap is evidence partitions that exist but are not linked by stable identifiers, making it difficult to reconstruct cross-agent workflows or trace responsibility chains.

8.15 ALCS-09: Cross-Project Reuse Compliance

MRO Mapping

This dimension primarily maps to [MRO-09](#) (Cross-Project Reuse Compliance) with secondary relationships to [MRO-02](#) (Delegated Authority Boundary) and [MRO-10](#) (Privacy/GDPR Lifecycle Mapping). The core requirement is that workflows validated in one project context must be revalidated when reused in different project contexts.

Core Question

Does the system enforce revalidation when workflows are reused across projects?

Scoring Rubric

Table T-08-11: ALCS-09 Cross-Project Reuse Compliance Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No reuse tracking features documented or observable
1	Conceptual support Documentation mentions workflow reuse but provides no revalidation primitives
2	Partial implementation Workflow templates or reuse mechanisms exist but do not track original validation context or enforce revalidation
3	Moderate implementation Reuse compliance records exist for most cross-project reuse; some gaps in authority reset, privacy review, or revalidation enforcement
4	Strong implementation Comprehensive reuse compliance records linking original validation context, new deployment context, authority reset, privacy review, revalidation decision, and revalidation evidence
5	Complete capability demonstrated by implementation evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating cross-project reuse compliance requires examining whether the system tracks workflow provenance and enforces revalidation when workflows move between project contexts. Look for reuse compliance records that bind original workflow ID, original validation record, original project context, new project context, authority reset decision, privacy review, and revalidation approval.

Strong systems treat cross-project reuse as a governance event rather than a convenience feature. When a workflow validated for one customer, risk class, data category, or authority boundary is reused in a different context, the system should require explicit revalidation. The original validation may not apply: the new project may have different privacy constraints, different authority boundaries, different acceptance criteria, or different risk exposure. Reuse without revalidation creates hidden governance debt.

Weak systems may provide workflow templates, agent libraries, or reusable components without tracking where they came from, what validation they received, or whether that validation applies to the new context. Developers may copy workflows across projects assuming that prior validation transfers automatically. The test is whether the system can identify the original validation context for any reused workflow and prove that revalidation occurred before deployment in the new context.

Common Gaps

The most common gap is workflow reuse treated as a productivity feature without governance controls. Organizations may have extensive workflow libraries while lacking visibility into original validation contexts or revalidation requirements. Another common gap is revalidation policies that apply to major version changes but not to cross-project reuse, allowing workflows to migrate between contexts without authority reset or privacy review.

8.16 ALCS-10: Privacy/GDPR Lifecycle Mapping

MRO Mapping

This dimension primarily maps to [MRO-10](#) (Privacy/GDPR Lifecycle Mapping) with secondary relationships to [MRO-08](#) (MAS Evidence Partitioning) and [MRO-12](#) (Evidence Minimization and Selective Disclosure). The core requirement is that personal data flows must be mapped across the entire agent lifecycle, not only at training or input boundaries.

Core Question

Does the system map personal data flows across the agent lifecycle?

Scoring Rubric

Table T-08-12: [ALCS-10](#) Privacy/GDPR Lifecycle Mapping Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No privacy lifecycle mapping features documented or observable
1	Conceptual support Documentation mentions privacy or data protection but provides no lifecycle mapping primitives
2	Partial implementation Privacy controls exist for training data or input data but do not map personal data flows across agents, tools, memory, or handoffs
3	Moderate implementation Privacy lifecycle mapping exists for most personal data flows; some gaps in cross-agent flows, tool-level tracking, or memory retention mapping

4

Strong implementation

Comprehensive lifecycle data-flow maps showing personal data entry, flow, transformation, storage, cross-agent transfer, tool disclosure, memory retention, and exit with data category labels, legal basis, retention periods, and processor tracking

5

Complete capability demonstrated by implementation evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating privacy lifecycle mapping requires examining whether the system tracks personal data flows throughout multi-agent execution. Look for data-flow maps that show where personal data enters the system, which agents process it, which tools receive it, where it is stored in memory, how it moves during agent handoffs, and where it exits the system.

Strong systems maintain dynamic data-flow maps that update as workflows execute. Each agent, tool, memory store, and handoff should be labeled with data categories processed, legal basis for processing, retention period, and processor identity. The system should be able to answer: for any completed workflow, what personal data was processed, by which agents, using which tools, stored in which memory locations, transferred to which processors, and retained for how long?

Weak systems may have privacy controls at the platform level without lifecycle-level data-flow visibility. Privacy policies may specify general data handling practices while the system cannot reconstruct actual data flows for specific workflow executions. The test is whether the system can generate a data-flow map for any completed workflow showing personal data movement across agents, tools, memory, and external processors.

Common Gaps

The most common gap is privacy controls that apply to training data or input validation but do not track personal data flows during agent execution. Organizations may have strong data governance for model training while lacking visibility into how personal data moves through multi-agent workflows, tool calls, memory stores, and agent handoffs. Another common gap is data-flow mapping that exists at the platform level but cannot be reconstructed for individual workflow executions.

8.17 ALCS-11: Privacy-Preserving Third-Party Validation**MRO Mapping**

This dimension primarily maps to [MRO-11](#) (Privacy-Preserving Third-Party Validation) with secondary relationships to [MRO-08](#) (MAS Evidence Partitioning) and [MRO-12](#) (Evidence Minimization and Selective Disclosure). The core requirement is that external validators must be able to verify compliance without requiring full disclosure of raw evidence, personal data, or trade secrets.

Core Question

Does the system support validation without raw data disclosure?

Scoring Rubric

Table T-08-13: **ALCS-11 Privacy-Preserving Third-Party Validation Scoring Rubric**

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No privacy-preserving validation features documented or observable
1	Conceptual support Documentation mentions external validation or audit but provides no privacy-preserving primitives
2	Partial implementation External validation is supported but requires full evidence disclosure; no selective disclosure or privacy-preserving protocols
3	Moderate implementation Selective disclosure mechanisms exist for most validation scenarios; some gaps in zero-knowledge proofs, cryptographic commitments, or cross-validator consistency
4	Strong implementation Comprehensive privacy-preserving validation protocols supporting selective disclosure, redacted evidence, cryptographic commitments, zero-knowledge proofs, and validator-specific evidence views without raw data exposure
5	Complete capability demonstrated by implementation evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating privacy-preserving validation requires examining whether the system provides mechanisms for external validators to verify compliance without accessing raw evidence. Look for selective disclosure protocols, redaction capabilities, cryptographic commitments, zero-knowledge proofs, or validator-specific evidence views.

Strong systems allow validators to verify that specific properties hold without exposing the underlying data. A privacy regulator may verify that personal data was processed under valid legal basis without seeing the actual data. A customer may verify that their request was handled correctly without exposing other customers' data. An auditor may verify that authority boundaries were enforced without accessing trade secret workflows. The system should support multiple validation protocols appropriate for different validator types and verification requirements.

Weak systems may require full evidence disclosure for any external validation. Validators must choose between accepting unverified claims or demanding access to raw logs, personal data, and trade secrets. This creates a

false choice between privacy and accountability. The test is whether the system can prove specific compliance properties to external validators without exposing raw evidence that contains personal data or trade secrets.

Common Gaps

The most common gap is validation protocols that require full evidence disclosure. Organizations may have comprehensive evidence collection while lacking mechanisms for selective disclosure or privacy-preserving verification. Another common gap is privacy-preserving validation that works for specific use cases but lacks general protocols for different validator types, verification requirements, or evidence categories.

8.18 ALCS-12: Evidence Minimization and Selective Disclosure

MRO Mapping

This dimension primarily maps to [MRO-12](#) (Evidence Minimization and Selective Disclosure) with secondary relationships to [MRO-08](#) (MAS Evidence Partitioning) and [MRO-10](#) (Privacy/GDPR Lifecycle Mapping). The core requirement is that evidence retention must be minimized to what is necessary for governance purposes, and disclosure must be limited to what is required for specific validation contexts.

Core Question

Does the system minimize evidence retention and support selective disclosure?

Scoring Rubric

Table T-08-14: [ALCS-12](#) Evidence Minimization and Selective Disclosure Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No evidence minimization features documented or observable
1	Conceptual support Documentation mentions evidence retention or disclosure but provides no minimization primitives
2	Partial implementation Evidence retention exists but lacks minimization policies; no retention periods, redaction rules, or disclosure boundaries
3	Moderate implementation Evidence minimization policies exist for most evidence categories; some gaps in automated retention enforcement, redaction rules, or disclosure boundary controls

4

Strong implementation

Comprehensive evidence minimization policies defining retention periods by evidence category, automated retention enforcement, redaction rules, disclosure boundaries, and purpose-specific evidence views

5

Complete capability demonstrated by implementation evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating evidence minimization requires examining whether the system defines and enforces retention limits and disclosure boundaries. Look for evidence minimization policies that specify retention periods by evidence category, automated deletion or anonymization rules, redaction protocols, and disclosure boundary controls.

Strong systems distinguish between evidence required for immediate governance, evidence required for dispute resolution, and evidence required for long-term compliance. Retention periods should vary by evidence category and purpose. Personal data in evidence should be minimized, anonymized, or deleted when no longer necessary. Disclosure should be limited to what is required for specific validation contexts rather than exposing full raw logs.

Weak systems may retain all evidence indefinitely without minimization policies or may delete evidence too aggressively, creating gaps in audit capability. The balance is between retaining sufficient evidence for governance purposes and minimizing retention to reduce privacy risk, storage cost, and disclosure exposure. The test is whether the system can define retention periods by evidence category, enforce automated retention limits, and support selective disclosure without exposing unnecessary evidence.

Common Gaps

The most common gap is evidence retention without minimization policies. Organizations may collect comprehensive evidence while lacking clarity about retention periods, deletion rules, or disclosure boundaries. Another common gap is minimization policies that exist in documentation but are not enforced automatically, requiring manual review and deletion processes that are rarely executed.

8.19 ALCS-13: Data Subject Rights vs Evidence Retention

MRO Mapping

This dimension primarily maps to [MRO-13](#) (Data Subject Rights vs Evidence Retention) with secondary relationships to [MRO-10](#) (Privacy/GDPR Lifecycle Mapping) and [MRO-12](#) (Evidence Minimization and Selective Disclosure). The core requirement is that data subject rights must be reconciled with evidence retention obligations, not treated as separate concerns.

Core Question

Does the system reconcile data subject rights with evidence retention obligations?

Scoring Rubric

Table T-08-15: ALCS-13 Data Subject Rights vs Evidence Retention Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No data subject rights reconciliation features documented or observable
1	Conceptual support Documentation mentions data subject rights or evidence retention but provides no reconciliation primitives
2	Partial implementation Data subject rights mechanisms exist but do not account for evidence retention obligations; no retention override rules or anonymization protocols
3	Moderate implementation Data subject rights policies exist with some retention reconciliation; some gaps in legal basis documentation, anonymization rules, or dispute-window protection
4	Strong implementation Comprehensive data subject rights policies defining retention periods, anonymization rules, legal basis for retention override, dispute-window protection, and reconciliation procedures for erasure requests vs evidence obligations
5	Complete capability demonstrated by implementation evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating data subject rights reconciliation requires examining whether the system handles conflicts between erasure requests and evidence retention obligations. Look for data subject rights policies that define retention periods, anonymization protocols, legal basis for retention override, and dispute-window protection.

Strong systems recognize that data subject rights are not absolute when evidence retention is required for legal compliance, dispute resolution, or regulatory obligations. When a data subject requests erasure, the system should evaluate whether evidence retention obligations apply, whether anonymization can satisfy both requirements, and whether retention override is legally justified. The system should document the legal basis for any retention override and implement anonymization where possible.

Weak systems may treat data subject rights and evidence retention as separate concerns. Privacy teams may implement erasure without consulting governance teams about evidence obligations, or governance teams may retain evidence without considering data subject rights. The test is whether the system can handle erasure

requests while preserving evidence required for governance purposes, documenting the legal basis for retention, and implementing anonymization where appropriate.

Common Gaps

The most common gap is data subject rights and evidence retention treated as separate concerns without reconciliation procedures. Organizations may have privacy teams handling erasure requests and governance teams managing evidence retention without coordination. Another common gap is retention override decisions made without documented legal basis or anonymization analysis, creating compliance risk when erasure requests are denied.

8.20 ALCS-14: Third-Party Processor/Subprocessor Chain

MRO Mapping

This dimension primarily maps to [MRO-14](#) (Third-Party Processor/Subprocessor Chain) with secondary relationships to [MRO-10](#) (Privacy/GDPR Lifecycle Mapping). The core requirement is that processor and subprocessor usage must be tracked at lifecycle granularity, not only at contract level.

Core Question

Does the system track processor and subprocessor usage at lifecycle granularity?

Scoring Rubric

Table T-08-16: [ALCS-14](#) Third-Party Processor/Subprocessor Chain Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No processor chain tracking features documented or observable
1	Conceptual support Documentation mentions third-party processors but provides no lifecycle tracking primitives
2	Partial implementation Processor agreements exist but processor usage is not tracked at lifecycle granularity; no dynamic processor chain records
3	Moderate implementation Processor chain tracking exists for most workflows; some gaps in subprocessor tracking, data category mapping, or legal basis documentation

4

Strong implementation

Comprehensive dynamic processor chain records tracking processor identity, subprocessor identity, data categories processed, processing purpose, legal basis, retention period, and processor location for each workflow execution

5

Complete capability demonstrated by implementation evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating processor chain tracking requires examining whether the system records which processors and subprocessors were used during specific workflow executions. Look for processor chain records that bind workflow ID, processor identity, subprocessor identity, data categories processed, processing purpose, legal basis, and processor location.

Strong systems maintain dynamic processor chain records that update as workflows execute. When an agent calls a tool that uses a third-party API, the system should record the processor identity, what data was sent, for what purpose, under what legal basis, and where the processor is located. The system should track subprocessor chains when processors use their own subprocessors. The processor chain should be reconstructable for any completed workflow.

Weak systems may have processor agreements at the contract level without tracking which processors were actually used during specific workflow executions. Privacy teams may maintain processor lists while the system cannot prove which processors handled which data for which workflows. The test is whether the system can reconstruct the processor chain for any completed workflow showing which processors and subprocessors handled personal data.

Common Gaps

The most common gap is processor agreements that exist at the contract level without lifecycle-level tracking. Organizations may have comprehensive processor agreements while lacking visibility into which processors were used during specific workflow executions. Another common gap is processor tracking that covers primary processors but not subprocessor chains, creating blind spots when processors use their own subprocessors.

8.21 ALCS-15: Vendor/Model/Runtime Substitution Conformance

MRO Mapping

This dimension primarily maps to [MRO-15](#) (Vendor/Model/Runtime Substitution Conformance) with secondary relationships to [MRO-02](#) (Delegated Authority Boundary) and [MRO-09](#) (Cross-Project Reuse Compliance). The core requirement is that vendor, model, or runtime substitutions must trigger conformance revalidation rather than being treated as transparent configuration changes.

Core Question

Does the system enforce conformance revalidation when vendors are substituted?

Scoring Rubric

Table T-08-17: **ALCS-15** Vendor/Model/Runtime Substitution Conformance Scoring Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No evidence of capability No vendor substitution tracking features documented or observable
1	Conceptual support Documentation mentions vendor optionality or model substitution but provides no conformance revalidation primitives
2	Partial implementation Vendor or model substitution is supported but does not trigger conformance revalidation; no substitution records or delta analysis
3	Moderate implementation Vendor substitution conformance records exist for most substitutions; some gaps in delta analysis, conformance testing, or revalidation enforcement
4	Strong implementation Comprehensive vendor substitution conformance records linking original validation, substitution authorization, conformance test results, delta analysis, revalidation decision, and deployment approval
5	Complete capability demonstrated by implementation evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts; legal text may establish a regulatory baseline but does not prove ALCS implementation

System Mapping Guidance

Evaluating vendor substitution conformance requires examining whether the system treats substitutions as governance events requiring revalidation. Look for substitution conformance records that bind original workflow ID, original vendor/model/runtime, new vendor/model/runtime, substitution authorization, conformance test results, delta analysis, and revalidation decision.

Strong systems recognize that vendor substitution can change behavior in ways that break lifecycle conformance. Swapping one model for another may change output quality, reasoning patterns, or tool-calling behavior. Swapping one runtime for another may change how authority boundaries are enforced or how evidence is collected. The system should require conformance testing and revalidation before substituted configurations are deployed. The substitution record should document what changed, what was tested, and whether lifecycle responsibility properties were preserved.

Weak systems may treat vendor substitution as a configuration change that does not require revalidation. Developers may swap model providers, update API endpoints, or switch agent frameworks without triggering

any governance checkpoint. The organization discovers conformance drift only when disputes arise. The test is whether the system can identify vendor substitutions, prove that conformance was revalidated, and document that lifecycle responsibility properties were preserved.

Common Gaps

The most common gap is vendor substitution treated as a configuration change without conformance revalidation. Organizations may value vendor optionality while lacking governance controls that ensure substitution preserves lifecycle responsibility properties. Another common gap is substitution conformance that applies to major vendor changes but not to model version updates, runtime patches, or tool vendor switches, creating gaps in coverage.

8.22 Chapter Summary

This chapter introduced the Agentic Lifecycle Conformance Score (ALCS), a framework for measuring whether systems define the lifecycle objects required for agentic responsibility compliance. ALCS evaluates whether a system can express the sixteen Missing Regulatory Objects introduced in Chapter 6, translating regulatory abstractions into measurable system capabilities. ALCS is not a legal compliance certification. It is an analytical framework for comparing system capabilities against lifecycle responsibility requirements.

The fifteen ALCS dimensions operationalize the sixteen MROs into system features that can be evaluated, scored, and improved. Human-role-to-MAS responsibility mapping ensures that human organizational accountability is explicitly linked to agent execution responsibilities. Delegated authority boundaries distinguish technical permission from business authority. Agent role definitions carry governance semantics beyond execution personas. Accepted outcome compliance separates execution completion from outcome acceptance. Tool-action liability boundaries record accountability for consequential actions. Responsibility transfer semantics make agent handoffs auditable. Authority drift detection prevents unauthorized authority transitions. MAS evidence partitioning enables selective disclosure and reconstruction. Cross-project reuse compliance enforces revalidation when workflows move between contexts. Privacy lifecycle mapping tracks personal data flows across agents, tools, and memory. Privacy-preserving validation supports external verification without raw data disclosure. Evidence minimization limits retention to governance purposes. Data subject rights reconciliation balances erasure requests with evidence obligations. Processor chain tracking maintains visibility into third-party data handling. Vendor substitution conformance ensures that optionality does not become governance drift.

ALCS complements RCCS-T and RCCS-M by measuring a different layer. RCCS-T evaluates traditional regulatory compliance coverage. RCCS-M evaluates whether that coverage can be expressed through Missing Regulatory Objects. ALCS evaluates agentic lifecycle conformance: whether a system provides mature lifecycle objects that make agentic responsibility auditable, provable, and transferable across agents, humans, tools, projects, vendors, and organizational boundaries. A system can have strong RCCS-T and weaker ALCS, meaning it provides strong regulatory compliance coverage while lacking deeper lifecycle responsibility semantics for multi-agent work. A system can also have stronger RCCS-M and ALCS because it defines lifecycle responsibility semantics while lacking productized governance-platform coverage. Chapter 9 explains why the three-profile view is primary and any composite calculation is secondary.

Chapter 9: Composite Scoring Method and Evidence Confidence

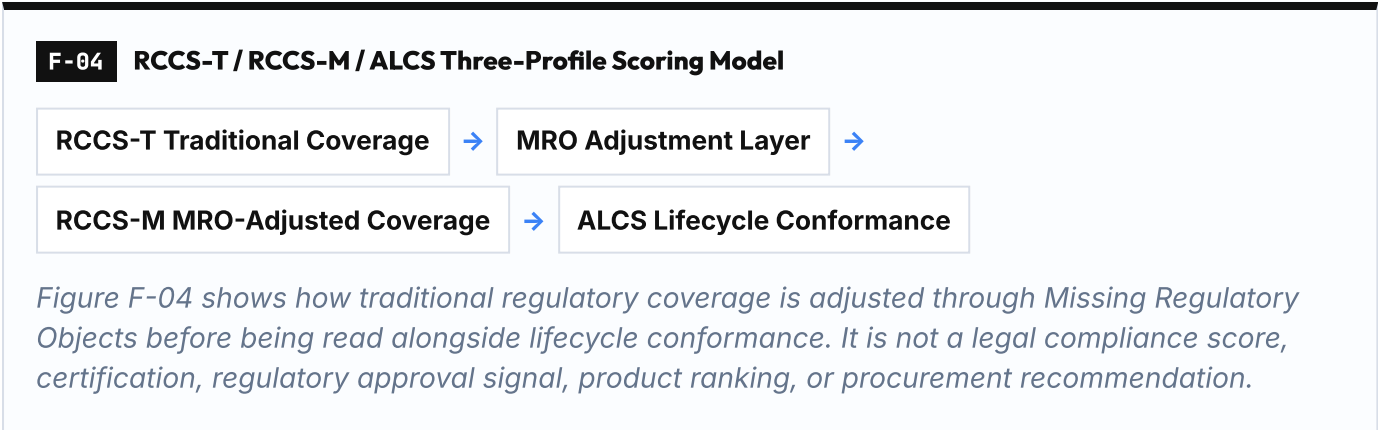
MAIN RESULT RCCS-T, RCCS-M, and ALCS can be combined analytically, but evidence confidence changes interpretation.	METHOD / LENS Canonical multipliers and composite formula.
EVIDENCE BASIS A high conceptual score without source support should not be treated as strong posture.	KEY TAKEAWAY / BOUNDARY Composite outputs are provisional analytical profiles, not vendor rankings.

Version: v0.3.2-FRC-R3

9.0 Chapter Overview

This chapter introduces the **Composite Scoring Method and Evidence Confidence** layer. In this paper, the primary result display is not a single composite number. The primary result display is a three-profile analytical view: **RCCS-T**, **RCCS-M**, and **ALCS**. Composite views are secondary analytical summaries that may help compare posture patterns, but they must not hide the distinction between traditional regulatory coverage, MRO-adjusted regulatory coverage, and lifecycle conformance.

The chapter explains evidence multiplier application, optional composite views, interpretation guidance for RCCS-T / RCCS-M / ALCS together, reproducibility requirements, and boundary statements. It establishes clear expectations about what composite scoring measures and what it does not measure, and bridges to the system mapping chapters that follow.



9.1 Purpose of Composite Scoring

Composite scoring is secondary to the three-profile view. The three-profile view reflects traditional regulatory compliance coverage (RCCS-T), MRO-adjusted regulatory compliance coverage (RCCS-M), and agentic lifecycle conformance (ALCS). It provides a structured method for evaluating whether systems provide the governance primitives that existing regulation requires, whether those primitives can be expressed through lifecycle responsibility objects, and whether lifecycle conformance is mature enough to make agentic responsibility auditable, provable, and transferable.

Chapter 7 defines the operational difference between the RCCS-T and RCCS-M scoring lenses. Chapter 9 does not duplicate those rubrics. It interprets score deltas after separate RCCS-T and RCCS-M raw scores have been assigned using Chapter 7 and Appendix E.

The purpose of the three-profile model is to make visible the difference between systems that provide traditional regulatory compliance coverage, systems that provide MRO-adjusted lifecycle responsibility semantics, and systems that provide both. A system can have a strong RCCS-T score and weaker RCCS-M or ALCS scores. This is not a contradiction. It means the system may provide strong regulatory compliance coverage through risk management, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring capabilities while lacking the deeper lifecycle responsibility objects required to prove authority boundaries, responsibility transfer, evidence partitioning, privacy lifecycle mapping, and vendor substitution conformance in multi-agent execution.

Composite scoring is an analytical framework, not a certification program. It does not certify that a system is compliant with any specific regulation. It does not guarantee that a system meets all legal obligations. It does not substitute for legal counsel, compliance review, or risk assessment. It measures system capability to express governance primitives and lifecycle objects, not organizational conformance to legal requirements.

The value of composite scoring is comparative analysis. Buyers can compare systems to understand which provide stronger governance primitives or lifecycle objects. Auditors can evaluate whether deployed systems provide the capabilities required to support organizational compliance programs. Governance teams can identify gaps in system capabilities and prioritize improvements. Protocol designers can evaluate whether their designs provide the primitives and objects that regulation and lifecycle responsibility require.

9.2 Primary Three-Profile Display and Optional Composite Views

The preferred publication-facing result display is:

1. **RCCS-T** — Traditional Regulatory Compliance Coverage Score
2. **RCCS-M** — MRO-Adjusted Regulatory Compliance Coverage Score
3. **ALCS** — Agentic Lifecycle Conformance Score

This structure prevents a single number from hiding the key score-delta insight. Enterprise governance platforms may show strong RCCS-T while MRO-adjusted lifecycle object coverage remains partial. Lifecycle protocol systems may show stronger RCCS-M and ALCS while traditional productized governance-platform coverage remains limited.

Each dimension is first normalized using the canonical dimension formula:

Dimension Score = (Raw Score / 5) × 100 × Evidence Confidence Multiplier

Where:

- **Raw Score** is the 0-5 score for each dimension based on system capability strength
- **Evidence Confidence Multiplier** is L1 = 1.00, L2 = 0.85, L3 = 0.75, L4 = 0.55, or L5 = 0.35

If a composite view is required for worksheet analysis, two optional views may be used:

- **Traditional Composite** = $0.40 \times \text{RCCS-T} + 0.60 \times \text{ALCS}$
- **Agentic Composite** = $0.40 \times \text{RCCS-M} + 0.60 \times \text{ALCS}$

RCCS-T, RCCS-M, and ALCS are expressed on a 0-100 scale after evidence adjustment. Optional composite views are also expressed on a 0-100 scale.

The following table illustrates the composite score calculation with an example.

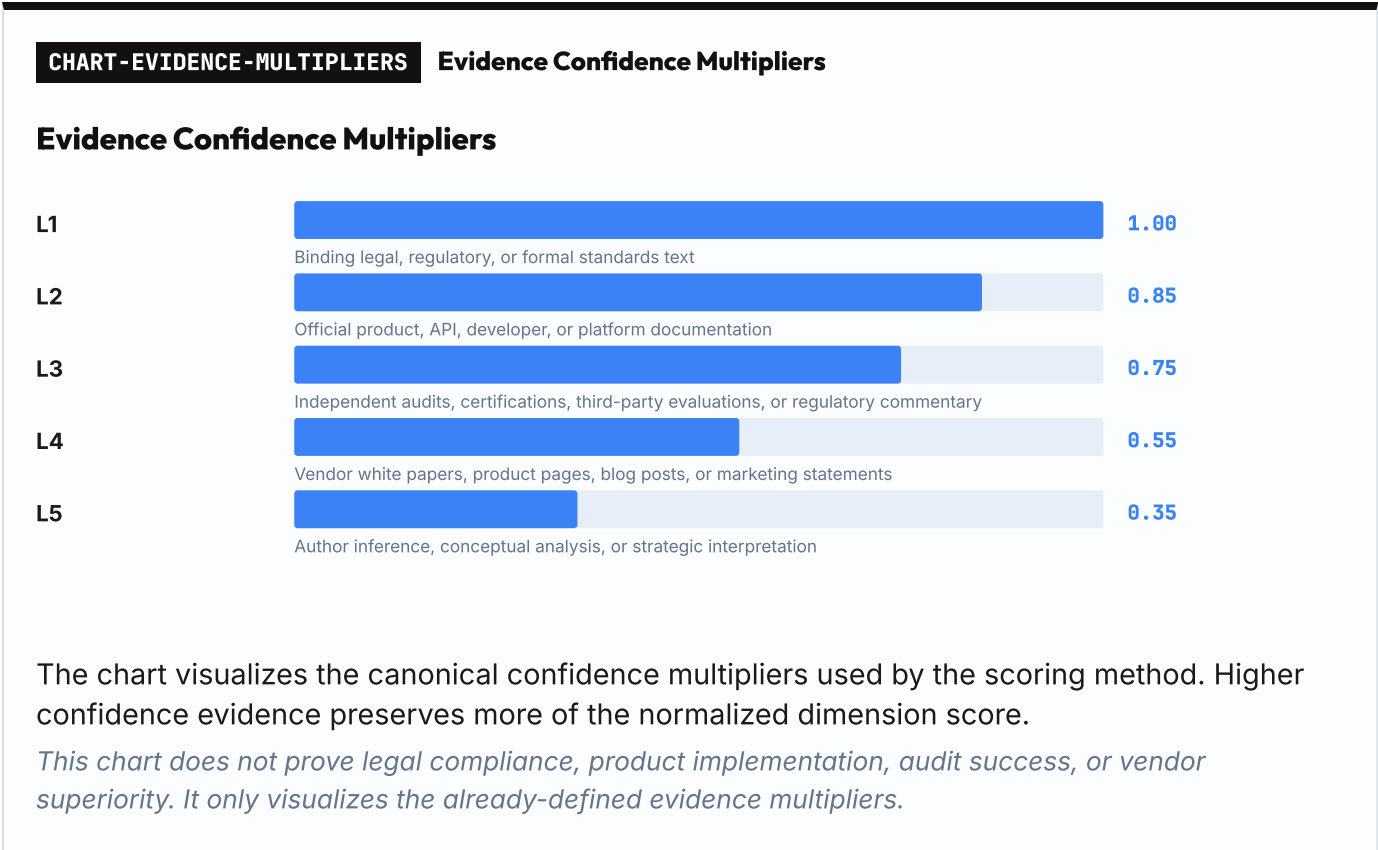
Table T-09-01: Composite Score Calculation Example

Component	Description	Example Calculation
Raw Score	0-5 scale based on capability strength	Risk Management = 4
Evidence Multiplier	L1 - L5 confidence multiplier	L2 evidence = 0.85
Dimension Score	(Raw Score / 5) × 100 × Evidence Multiplier	(4 / 5) × 100 × 0.85 = 68.0
RCCS-T Score	Aggregated traditional RCCS dimension score	RCCS-T = 65.0
RCCS-M Score	Aggregated MRO-adjusted RCCS dimension score	RCCS-M = 72.0
ALCS Score	Aggregated ALCS dimension score	ALCS = 55.0
Traditional Composite	$0.40 \times \text{RCCS-T} + 0.60 \times \text{ALCS}$	$0.40 \times 65.0 + 0.60 \times 55.0 = 59.0$
Agentic Composite	$0.40 \times \text{RCCS-M} + 0.60 \times \text{ALCS}$	$0.40 \times 72.0 + 0.60 \times 55.0 = 61.8$

These calculations preserve the distinction between traditional regulatory coverage and MRO-adjusted regulatory coverage. They are optional worksheet views, not the main publication result.

RCCS-T, RCCS-M, and ALCS are reported separately because they measure different layers. RCCS-T evaluates whether a system provides the governance primitives that existing regulation requires. RCCS-M evaluates whether those obligations can be expressed through Missing Regulatory Objects. ALCS evaluates whether lifecycle responsibility conformance is mature. Any composite view must therefore be read only alongside the separate three-profile scores.

9.3 Evidence Multiplier Application



Evidence quality affects composite scores through the evidence multiplier. The evidence multiplier adjusts dimension scores based on the quality and verifiability of evidence used to evaluate the system. It reflects the confidence level in the scoring assessment.

The evidence hierarchy established in the front matter defines five evidence levels:

- L1 : Binding legal, regulatory, or formal standards text** - Highest confidence for baseline obligations; does not prove system implementation
- L2 : Official product documentation, API docs, developer documentation, platform architecture docs** - High confidence
- L3 : Independent audits, certifications, third-party evaluations, regulatory commentary** - Medium confidence
- L4 : Vendor white papers, product pages, blog posts, marketing statements** - Low confidence
- L5 : Author inference, conceptual analysis, strategic interpretation** - Disclosed inference

The evidence multiplier translates evidence levels into scoring adjustments:

Table T-09-02: Evidence Multiplier Values

Evidence Level	Evidence Type	Multiplier	Rationale
L1	Binding legal, regulatory, or formal standards text	1.00	Highest confidence for baseline obligations; does not prove system implementation
L2	Official product documentation, API docs, developer documentation, platform architecture docs	0.85	High confidence, but product capability claims still require current-source validation

Evidence Level	Evidence Type	Multiplier	Rationale
L3	Independent audits, certifications, third-party evaluations, regulatory commentary	0.75	Medium confidence; independent but may lack technical depth
L4	Vendor white papers, product pages, blog posts, marketing statements	0.55	Low confidence; promotional and may lack implementation detail
L5	Author inference, conceptual analysis, strategic interpretation	0.35	Disclosed inference; useful for analysis but not verified implementation evidence

The evidence multiplier is applied per dimension, not globally. Different dimensions may be evaluated with different evidence levels. A system may have L2 evidence for risk management (official documentation) and L4 evidence for contestability (vendor white paper). Each dimension receives its own evidence multiplier based on the evidence used to evaluate that specific dimension.

Mixed evidence levels are common and expected. System vendors provide detailed documentation for some capabilities while providing only high-level descriptions for others. Independent audits may cover some dimensions but not others. Evaluators must assess evidence quality dimension by dimension and apply the appropriate multiplier to each.

The evidence multiplier ensures that scores reflect not only system capability but also the confidence level in that assessment. A raw score of 4 based on L2 evidence becomes $(4 / 5) \times 100 \times 0.85 = 68.0$. The same raw score based on L4 evidence becomes $(4 / 5) \times 100 \times 0.55 = 44.0$. This prevents over-claiming based on weak evidence.

9.4 Interpreting RCCS-T, RCCS-M, and ALCS Together

RCCS-T, RCCS-M, and ALCS measure different layers. RCCS-T measures traditional regulatory compliance coverage: whether a system provides the governance primitives that existing regulation requires. RCCS-M measures whether those obligations can be expressed through Missing Regulatory Objects. ALCS measures agentic lifecycle conformance: whether the system provides mature lifecycle objects that make agentic responsibility auditable, provable, and transferable across agents, humans, tools, projects, vendors, and organizational boundaries.

Interpreting the three profiles requires reading score deltas rather than collapsing the profiles into a single ordering. High RCCS-T with lower RCCS-M or ALCS means conventional governance coverage is more visible than lifecycle responsibility object coverage. Lower RCCS-T with higher RCCS-M and ALCS means protocol or framework semantics may express lifecycle responsibility strongly while productized governance-platform coverage remains limited. This distinction is especially important for comparing enterprise governance platforms with lifecycle protocol paths.

The 70% boundary used in this matrix is an analytical interpretation band for comparing capability profiles. It is not a legal compliance threshold, deployment readiness threshold, procurement threshold, or certification cutoff.

Table T-09-03: RCCS-T / RCCS-M / ALCS Interpretation Matrix

Wide source table split into consecutive column groups; all source rows and columns are preserved.

Continuation 1 of 2

Pattern	RCCS-T	RCCS-M
Traditional governance visible	High	Lower
Lifecycle semantics visible	Lower/Partial	High
Three-profile balanced	High	High
Execution-oriented	Lower	Lower/Partial

Continuation 2 of 2

Pattern	ALCS	Interpretation	Common System Types
Traditional governance visible	Lower/Partial	Strong conventional governance surfaces, but MRO-adjusted lifecycle responsibility objects are less visible.	Enterprise AI governance platforms, managed AI platforms
Lifecycle semantics visible	High	Traditional productized governance coverage may be limited, but lifecycle responsibility semantics are strong.	Lifecycle protocol systems, lifecycle-aware orchestration frameworks
Three-profile balanced	High	Governance surfaces and lifecycle responsibility semantics are both visible.	Future governance platforms with explicit lifecycle object support
Execution-oriented	Lower/Partial	Useful execution or observability primitives, but governance and lifecycle responsibility objects require surrounding systems.	Developer SDKs, orchestration frameworks without governance extensions

The traditional-governance-visible pattern is common. Many enterprise AI governance platforms, managed AI platforms, and observability tools provide strong RCCS-T coverage through risk management, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring capabilities. However, they may have lower RCCS-M and ALCS posture where authority boundaries, responsibility transfer, evidence partitioning, privacy lifecycle mapping, accepted outcomes, and vendor substitution conformance are not exposed as first-class lifecycle responsibility objects. This is not a failure. It reflects the reality that existing regulation focuses on model governance and platform controls while lifecycle responsibility semantics for multi-agent systems are still emerging.

The lifecycle-semantics-visible pattern is strategically important. It represents systems that provide strong lifecycle objects for expressing agentic responsibility while lacking broader productized governance-platform coverage. This pattern may appear in specialized lifecycle protocols or agent frameworks that focus on responsibility semantics without providing full governance platform capabilities. These systems may be valuable as components within larger governance architectures but may not provide sufficient coverage for standalone compliance programs.

The three-profile-balanced pattern represents systems that provide traditional regulatory compliance coverage, MRO-adjusted lifecycle object coverage, and lifecycle conformance maturity. These systems are positioned to support both existing regulatory requirements and emerging lifecycle responsibility requirements for multi-agent systems.

The execution-oriented pattern represents systems that provide execution orchestration or observability features but lack explicit governance primitives and lifecycle objects. These systems may be valuable for development, experimentation, or low-risk use cases but may require surrounding governance systems for high-risk or regulated deployments.

All three profile scores are necessary for full assessment. RCCS-T shows familiar governance coverage. RCCS-M shows MRO-adjusted regulatory coverage. ALCS shows lifecycle conformance maturity. Together, they provide the publication-facing assessment of system governance capability.

9.5 Scoring Reproducibility and Transparency

Composite scoring must be reproducible and transparent. Scoring is only useful if different evaluators can reach similar conclusions when evaluating the same system with the same evidence. Reproducibility requires clear scoring criteria, documented evidence sources, and explicit scoring rationale.

Scoring reproducibility requires four elements: dimension scores with rationale, evidence sources with quality assessment, evidence multiplier application, and calculation transparency.

Dimension scores must be documented with rationale. For each dimension, evaluators must record the 0-5 raw score and explain why that score was assigned. The rationale should reference specific system capabilities, features, or documentation that support the score. Generic statements such as "the system provides risk management" are insufficient. Specific statements such as "the system provides risk registers, assessment workflows, and mitigation tracking as documented in the platform architecture guide" enable reproducibility.

Evidence sources must be documented with quality assessment. For each dimension, evaluators must record which evidence sources were used and what evidence level applies. Evidence sources should be cited with sufficient detail to allow verification: document title, version, publication date, and relevant section or page numbers. Evidence quality assessment should explain why a particular evidence level was assigned. For example, "L2 evidence: official product documentation, API reference v2.3, section 4.2" is more reproducible than "L2 evidence: documentation."

Evidence multiplier application must be explicit. For each dimension, evaluators must show the evidence multiplier value and how it was applied to the raw score. This ensures that evidence quality adjustments are transparent and verifiable. The calculation should be shown using the canonical formula: $\text{Dimension Score} = (\text{Raw Score} / 5) \times 100 \times \text{Evidence Confidence Multiplier}$.

Calculation transparency requires showing the full three-profile score basis. Evaluators must provide a table or worksheet showing raw scores, evidence multipliers, adjusted dimension scores, RCCS-T, RCCS-M, and ALCS framework scores, and any optional Traditional Composite or Agentic Composite view. This allows reviewers to verify the calculation and identify any errors or disagreements without collapsing the main result into one number.

Scoring is comparative, not absolute. Three-profile scores and optional composite views are most useful when comparing systems against defined governance expectations or tracking a single system over time. Absolute score thresholds such as "70% is compliant" are not meaningful because the scoring method is an analytical framework, not a compliance certification. The value of scoring is in making visible capability gaps and evidence-confidence limits.

9.6 Boundary Statement

The three-profile scoring method is an analytical framework, not a legal compliance score. Optional composite views are secondary worksheet tools. This boundary must be understood clearly to avoid misinterpretation and over-claiming.

What the scoring method is:

- An analytical framework for comparing system capabilities against traditional regulatory requirements, MRO-adjusted regulatory coverage, and lifecycle responsibility requirements
- A structured method for evaluating whether systems provide governance primitives and lifecycle objects
- A tool for buyers, auditors, governance teams, and protocol designers to assess system readiness
- A bridge between regulatory abstractions, lifecycle requirements, and engineering implementation

What the scoring method is not:

- A legal compliance certification or conformity assessment
- A regulatory approval or endorsement
- A guarantee that a system meets all legal obligations
- A substitute for legal counsel, compliance review, or risk assessment
- A market endorsement or procurement recommendation

High RCCS-T, RCCS-M, ALCS, or optional composite scores do NOT mean legal compliance. A system may score well on visible capability surfaces while the organization using it fails to comply with legal obligations due to deployment gaps, policy weaknesses, incomplete adoption, jurisdictional differences, or use case risks. The scoring method measures system capability to express governance primitives and lifecycle objects, not organizational conformance to legal requirements.

Low RCCS-T, RCCS-M, ALCS, or optional composite scores do NOT mean non-compliance. A system may score lower in one profile while the organization achieves compliance through compensating controls, manual processes, alternative governance mechanisms, or different system architectures. The scoring method evaluates specific system capabilities, not all possible paths to compliance.

The scoring method measures system capability, not organizational practice. A system may provide strong governance primitives and lifecycle objects while the organization fails to use them. Conversely, an organization may achieve strong governance through processes that are not reflected in system capabilities. The scoring method evaluates what the system can express, not what the organization actually does.

The scoring method is context-independent, but compliance is context-dependent. It evaluates system capabilities without regard to deployment context, risk class, jurisdiction, or use case. Legal compliance depends on all of these factors. A system with high scores may still be non-compliant in specific contexts, and a system with low scores may be compliant in other contexts.

The purpose of this boundary statement is to prevent over-claiming. RCCS-M is an author analytical, forward-looking, MRO-adjusted model. It is not current law, a regulator-approved benchmark, a certification, legal compliance proof, a vendor ranking, or a procurement recommendation. Organizations must not rely on these scores as proof of legal compliance. Buyers must not treat high scores as a substitute for due diligence, legal review, or risk assessment. System vendors must not market these scores as compliance certifications or regulatory approvals.

9.7 Bridge to System Mappings

The following chapters use RCCS-T, RCCS-M, and ALCS as analytical lenses for enterprise controls, system mappings, validation patterns, and failure-scenario analysis. Chapter 10 provides enterprise control crosswalk, mapping regulatory coverage and lifecycle conformance dimensions to enterprise control frameworks such as NIST Cybersecurity Framework, ISO 27001, and SOC 2. Chapters 11 through 13 provide comparative field analysis and detailed system/protocol mappings, while Chapters 14 and 15 test the evidence and failure-scenario implications of the RCCS-M / MRO model.

Each system mapping documents qualitative posture, evidence sources, and scoring rationale where a source-bound worksheet is used. The mappings are responsibility-semantics comparisons, not product endorsements or procurement recommendations. The key question is not which system is useful. Most of them are useful. The key question is whether the system defines the governance primitives and lifecycle responsibility objects required to show that agentic work moved from intent to accepted outcome under controlled authority, evidence, privacy, and remediation constraints.

System mappings use the scoring method defined in this chapter only where a scoring worksheet is appropriate. Where a scoring worksheet is used, systems may be evaluated with separate RCCS-T, RCCS-M, and ALCS analytical profiles. In this public research edition, any system-level assessment remains source-bound, provisional, and non-ranking unless explicitly finalized through the evidence register. Dimension-level analytical judgments should be documented with rationale, evidence sources should be cited with quality assessment, and evidence multipliers should be applied per dimension when numerical worksheets are used. The result is a three-profile analytical view that makes visible the difference between traditional regulatory coverage, MRO-adjusted regulatory coverage, and lifecycle responsibility conformance without creating vendor rankings or procurement selection signals.

The system mappings are based on public evidence available as of May 2026. They reflect the state of system capabilities at the time of analysis. System capabilities evolve rapidly. Vendors add features, update documentation, and release new versions. The mappings should be treated as snapshots, not permanent judgments. Organizations evaluating systems should verify current capabilities with vendors and review updated documentation.

9.8 Chapter Summary

This chapter introduced the three-profile scoring method and optional composite views. The publication-facing result display is RCCS-T, RCCS-M, and ALCS. It is an analytical framework for comparing system capability patterns, not a legal compliance certification.

The method first normalizes dimension scores using $(\text{Raw Score} / 5) \times 100 \times \text{Evidence Confidence Multiplier}$, then reports RCCS-T, RCCS-M, and ALCS separately. Optional worksheet composites may combine RCCS-T or RCCS-M with ALCS, but composite views are secondary and must not hide the three-profile structure. The evidence multiplier adjusts dimension scores based on evidence quality, ensuring that scores reflect both system capability and confidence level in the assessment.

Interpreting RCCS-T, RCCS-M, and ALCS together requires reading score-delta patterns. Strong RCCS-T with lower RCCS-M or ALCS indicates visible traditional governance coverage but less visible lifecycle responsibility object coverage. Lower RCCS-T with stronger RCCS-M and ALCS indicates stronger lifecycle protocol semantics but less productized governance-platform coverage. This is the strategic distinction the three-profile model preserves.

Composite scoring must be reproducible and transparent when used. Dimension-level analytical judgments must be documented with rationale. Evidence sources must be cited with quality assessment. Evidence multipliers must be applied explicitly when numerical worksheets are used. Calculations must be shown transparently. Scoring is comparative, not absolute. In this public research edition, the value of the method is in making capability gaps visible without turning qualitative system mappings into rankings.

The scoring method is an analytical framework, not a legal compliance score. High RCCS-T, RCCS-M, ALCS, or optional composite scores do not mean legal compliance. Low scores do not mean non-compliance. The

method measures system capability, not organizational practice. It is context-independent, but compliance is context-dependent. Organizations must not rely on scores as proof of legal compliance. The following chapters use RCCS-T, RCCS-M, and ALCS as analytical lenses for comparative field analysis and detailed qualitative system mappings.

10. Enterprise Adoption, Procurement, and Control Crosswalk

MAIN RESULT Lifecycle objects can be translated into enterprise controls and procurement questions.	METHOD / LENS Control crosswalk and enterprise adoption mapping.
EVIDENCE BASIS Governance must connect lifecycle semantics to board, risk, security, privacy, and procurement operations.	KEY TAKEAWAY / BOUNDARY The crosswalk supports due diligence; it does not recommend vendors.

Version: v0.3.2-FRC-R3

10.0 Why Enterprise Control Language Matters

Enterprises do not buy abstract governance concepts. They evaluate controls, procurement risk, board reporting, audit readiness, vendor substitution, model risk, cybersecurity, incident response, and monitoring. When a CTO, CISO, CRO, procurement lead, or board-facing risk team evaluates an agentic AI system, they ask questions in enterprise control language, not in lifecycle theory.

Chapters 6 through 9 defined the Missing Regulatory Objects, RCCS-T, RCCS-M, ALCS, and the three-profile analytical model. Those chapters established the lifecycle responsibility framework. This chapter translates those objects into enterprise adoption, procurement, internal control, board reporting, model risk, cybersecurity, incident response, and monitoring language.

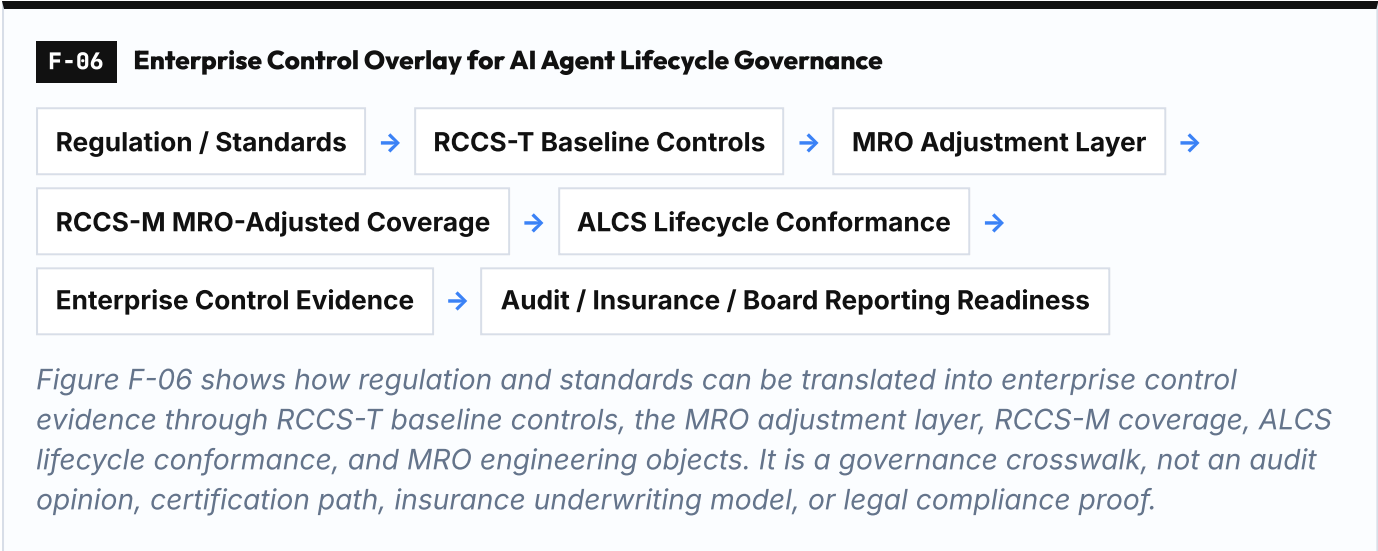
The purpose of this chapter is to show that AI Agent Lifecycle Governance is not only an abstract protocol or research model. It can be mapped into enterprise control language. RCCS-T, RCCS-M, and ALCS scores can become bounded scorecard inputs. MROs become audit evidence requirements. Lifecycle conformance becomes vendor substitution readiness. Evidence partitioning becomes privacy-preserving validation. Accepted outcome compliance becomes board reporting on agentic risk.

This chapter is a bridge from lifecycle objects to enterprise controls. It does not replace internal control frameworks. It shows where agentic lifecycle objects can provide evidence for enterprise control questions. It does not claim that MPLP is required for compliance. It does not claim that RCCS-T, RCCS-M, or ALCS is a legal compliance score. It does not provide an audit opinion, certification, or procurement recommendation.

Boundary: This chapter provides a governance crosswalk, not an audit framework, certification procedure, insurance underwriting model, or legal opinion. Detailed assurance procedures and insurance models are reserved for companion papers, as outlined in Chapter 16.

10.1 Enterprise Control Overlay

The following figure illustrates how AI Agent Lifecycle Governance maps into enterprise control structures. Regulation and standards define baseline requirements. RCCS-T measures traditional regulatory compliance coverage. MROs provide the adjustment layer. RCCS-M measures MRO-adjusted regulatory coverage. ALCS measures lifecycle conformance. Together, these layers produce enterprise control evidence that supports audit, insurance, and board reporting readiness.



The overlay shows that lifecycle governance is not separate from enterprise controls. It is a structured method for producing the evidence that enterprise controls require. RCCS-T, RCCS-M, and ALCS scores do not replace risk assessments, internal audits, or compliance reviews. They provide structured inputs that make those processes more efficient, consistent, and auditable when applied to agentic systems.

10.2 Enterprise Adoption, Procurement, and Control Crosswalk

The following tables map enterprise control topics to agentic lifecycle objects. The crosswalk does not replace internal control frameworks. It shows where agentic lifecycle objects can provide evidence for enterprise control questions.

Table T-10-01A: Enterprise Control Crosswalk — Control Questions

Enterprise Control Topic	Why It Matters for Agentic AI	Enterprise Buyer Question
Internal control	Agentic systems execute consequential actions under delegated authority. Internal control requires proving who authorized, executed, reviewed, and accepted each action.	Can the organization prove who authorized, executed, reviewed, and accepted agentic work?
Procurement scorecard	Vendor selection for agentic systems must evaluate lifecycle accountability, evidence exportability, privacy controls, and remediation records, not only model performance or API availability.	Can this vendor preserve lifecycle accountability, evidence exportability, privacy controls, and remediation records?
Vendor lock-in / vendor substitution	Vendor lock-in becomes a governance risk when switching vendors requires rebuilding lifecycle conformance from scratch.	Can the organization switch model, runtime, vendor, or tool substrate without losing lifecycle conformance?
Third-party / supply-chain risk	Multi-agent systems often involve multiple processors, subprocessors, tools, vendors, and runtimes. Supply-chain risk requires recon-	Can the organization reconstruct which processors, subprocessors, tools, vendors, and

Enterprise Control Topic	Why It Matters for Agentic AI	Enterprise Buyer Question
	structuring which parties handled which lifecycle evidence or personal data.	runtimes handled which lifecycle evidence or personal data?
Data governance	Agentic systems process data across agents, tools, memory, traces, and evidence packs. Data governance requires tracking data flow, retention, minimization, deletion, and disclosure at lifecycle granularity.	Can the organization track data flow, retention, minimization, deletion, and disclosure across agents, tools, memory, traces, and evidence packs?
Audit readiness	External auditors require evidence of lifecycle responsibility without full raw data disclosure. Audit readiness requires privacy-preserving validation and selective disclosure.	Can an independent reviewer reconstruct lifecycle responsibility without full raw data disclosure?
Board reporting	Boards require risk reporting in terms of authority, evidence, accepted outcomes, open incidents, and unresolved remediation, not only model performance metrics.	Can management explain agentic risk in terms of authority, evidence, accepted outcomes, open incidents, and unresolved remediation?
Incident response and redress	Agentic incidents require evidence, responsibility assignment, correction, and accepted remediation. Incident closure is not only technical resolution.	Can the organization close an agentic incident with evidence, responsibility assignment, correction, and accepted remediation?
Business continuity / rollback	Agentic systems must support halt, rollback, substitution, or remediation without losing evidence continuity. Business continuity requires lifecycle resilience.	Can the system halt, rollback, substitute, or remediate without losing evidence continuity?
Model risk management	Model risk in agentic systems is not only output quality or bias. Model behavior influences planning, delegation, authority transitions, evidence interpretation, tool selection, and accepted outcomes.	Can model, prompt, policy, or runtime changes be evaluated as lifecycle risk, not only model-performance risk?
Cybersecurity for agentic systems	Cybersecurity for agentic systems must preserve authority, credential, tool, evidence, and incident boundaries under attack.	Can the system preserve authority, credential, tool, evidence, and incident boundaries under attack?
Privacy-preserving validation	External validation must occur without exposing raw personal data, proprietary workflow content, or unnecessary evidence.	Can external validation occur without exposing raw personal data, proprietary workflow content, or unnecessary evidence?

Table T-10-01B: Enterprise Control Crosswalk — MRO/Dimension/Evidence Mapping

Enterprise Control Topic	Relevant MROs	Relevant RCCS-T/RCCS-M/ALCS Dimensions	Evidence Expected
Internal control	MRO-01, MRO-02, MRO-04, MRO-05, MRO-08, MRO-16	RCCS-04, RCCS-06, RCCS-08, ALCS-01, ALCS-02, ALCS-04, ALCS-05, ALCS-08	Role-responsibility map, authority boundary object, accepted outcome state, tool-action liability record, evidence partition, incident closure record
Procurement scorecard	MRO-11, MRO-14, MRO-15, MRO-16	RCCS-03, RCCS-04, RCCS-07, ALCS-11, ALCS-14, ALCS-15	Privacy-preserving validation protocol, processor chain record, vendor substitution conformance test, incident closure record
Vendor lock-in / vendor substitution	MRO-09, MRO-15	ALCS-09, ALCS-15	Cross-project reuse conformance test, vendor substitution conformance test
Third-party / supply-chain risk	MRO-10, MRO-14, MRO-15	RCCS-02, RCCS-07, ALCS-10, ALCS-14, ALCS-15	Privacy lifecycle map, processor chain record, vendor substitution conformance test
Data governance	MRO-10, MRO-12, MRO-13	RCCS-02, ALCS-10, ALCS-12, ALCS-13	Privacy lifecycle map, evidence minimization policy, data subject rights reconciliation record
Audit readiness	MRO-08, MRO-11, MRO-12, MRO-16	RCCS-04, RCCS-09, ALCS-08, ALCS-11, ALCS-12	Evidence partition, privacy-preserving validation protocol, evidence minimization policy, incident closure record
Board reporting	MRO-01, MRO-04, MRO-08, MRO-16	RCCS-06, RCCS-08, RCCS-10, ALCS-01, ALCS-04, ALCS-08	Role-responsibility map, accepted outcome state, evidence partition,

Enterprise Control Topic	Relevant MROs	Relevant RCCS-T/RCCS-M/ALCS Dimensions	Evidence Expected
			incident closure record, monitoring dashboard
Incident response and redress	MRO-04, MRO-08, MRO-16	RCCS-09, RCCS-10, ALCS-04, ALCS-08	Accepted outcome state, evidence partition, incident closure record, remediation record
Business continuity / rollback	MRO-05, MRO-07, MRO-15, MRO-16	RCCS-07, RCCS-10, ALCS-05, ALCS-07, ALCS-15	Tool-action liability record, authority drift detection log, vendor substitution conformance test, incident closure record
Model risk management	MRO-07, MRO-08, MRO-15, MRO-16	RCCS-01, RCCS-10, ALCS-07, ALCS-08, ALCS-15	Authority drift detection log, evidence partition, vendor substitution conformance test, incident closure record
Cybersecurity for agentic systems	MRO-02, MRO-05, MRO-07, MRO-08, MRO-14, MRO-16	RCCS-07, RCCS-10, ALCS-02, ALCS-05, ALCS-07, ALCS-08, ALCS-14	Authority boundary object, tool-action liability record, authority drift detection log, evidence partition, processor chain record, incident closure record
Privacy-preserving validation	MRO-08, MRO-10, MRO-11, MRO-12, MRO-13	RCCS-02, RCCS-04, ALCS-08, ALCS-10, ALCS-11, ALCS-12, ALCS-13	Evidence partition, privacy lifecycle map, privacy-preserving validation protocol, evidence minimization policy, data subject rights reconciliation record

Interpretation: The crosswalk does not replace internal control frameworks. It shows where agentic lifecycle objects can provide evidence for enterprise control questions. Organizations should integrate these objects into existing control frameworks rather than treating them as separate compliance requirements.

10.3 Model Risk Management in Agentic Systems

In agentic systems, model risk is no longer limited to output quality, bias, robustness, or model monitoring. Model risk becomes lifecycle risk when model behavior influences planning, delegation, authority transitions, evidence interpretation, tool selection, human review packets, and accepted outcomes.

Traditional model risk management asks whether a model is governed. Agentic model risk management asks whether model behavior can change lifecycle authority, tool use, evidence, review, and acceptance states. A model that performs well on benchmarks may still introduce lifecycle risk if it influences authority boundaries, responsibility transfer, or evidence interpretation in ways that violate delegated scope or escalation rules.

Boundary: Model governance remains necessary, but agentic deployment requires model risk to be mapped into lifecycle responsibility. ALCS does not replace existing model risk management. This section is not a full model-risk-management manual. It shows how model risk intersects with lifecycle conformance.

Model Risk in Agentic Lifecycle

The following figure illustrates how model, prompt, policy, and runtime changes propagate through the agentic lifecycle and influence authority, tool use, evidence, and accepted outcomes.

F-07

Model Risk in Agentic Lifecycle

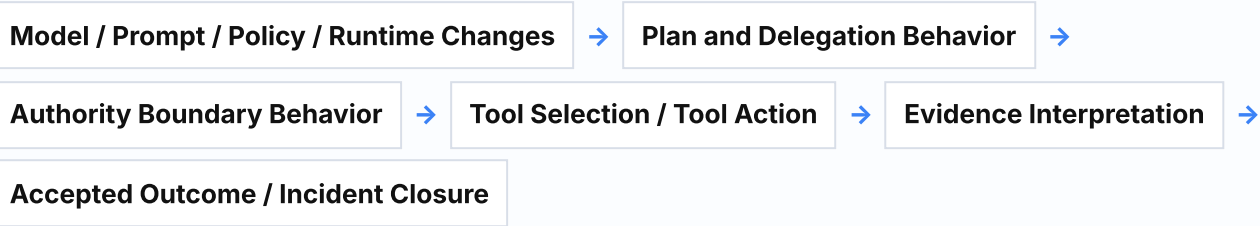


Figure F-07 shows how model, prompt, policy, or runtime changes can propagate through an agentic lifecycle and affect authority, tools, evidence, outcomes, and closure. It is a lifecycle-risk model, not a model-risk-management standard, certification path, or legal compliance proof.

Model risk in agentic systems is not only a model-performance concern. It is a lifecycle-conformance concern. A model version change may alter planning behavior, which may alter delegation patterns, which may alter authority transitions, which may alter tool selection, which may alter evidence interpretation, which may alter accepted outcome states. Each of these transitions is a potential lifecycle risk.

Model Risk Management Table

The following table maps model risk concerns to agentic lifecycle impacts, relevant MROs, and control evidence.

Table T-10-02: Model Risk Management in Agentic Systems

Model inventory	
Model Risk Concern	Model inventory
Agentic Lifecycle Impact	Without model inventory, the organization cannot reconstruct which model version influenced which lifecycle execution, authority transition, or accepted outcome.
Relevant MROs	MRO-15
Control Evidence	Model version registry linked to lifecycle execution records

Foundation model dependency	
Model Risk Concern	Foundation model dependency
Agentic Lifecycle Impact	Foundation model changes may alter planning, delegation, tool selection, or evidence interpretation behavior without explicit revalidation.
Relevant MROs	MRO-07, MRO-15
Control Evidence	Foundation model version tracking, authority drift detection, substitution conformance test

Model provider dependency

Model Risk Concern	Model provider dependency
Agentic Lifecycle Impact	Model provider changes may introduce new processor/subprocessor relationships, privacy risks, or vendor lock-in without lifecycle revalidation.
Relevant MROs	MRO-14 , MRO-15
Control Evidence	Processor chain record, vendor substitution conformance test

Model version risk

Model Risk Concern	Model version risk
Agentic Lifecycle Impact	Model version changes may alter lifecycle behavior in ways that violate delegated authority, escalation rules, or evidence requirements.
Relevant MROs	MRO-07 , MRO-08 , MRO-15
Control Evidence	Model version change log, authority drift detection, evidence partition validation, substitution conformance test

Model substitution

Model Risk Concern	Model substitution
Agentic Lifecycle Impact	Model substitution may break lifecycle conformance if the new model does not preserve authority boundaries, responsibility transfer semantics, or evidence partitioning.
Relevant MROs	MRO-15
Control Evidence	Vendor substitution conformance test, lifecycle revalidation record

Evaluation drift

Model Risk Concern	Evaluation drift
Agentic Lifecycle Impact	Model evaluation drift may indicate that the model no longer behaves as expected under delegated authority, escalation rules, or evidence requirements.
Relevant MROs	MRO-07 , MRO-16
Control Evidence	Authority drift detection log, incident closure record

Prompt / policy regression

Model Risk Concern	Prompt / policy regression
Agentic Lifecycle Impact	Prompt or policy changes may alter planning, delegation, tool selection, or evidence interpretation behavior without explicit revalidation.
Relevant MROs	MRO-07 , MRO-08
Control Evidence	Prompt version tracking, authority drift detection, evidence partition validation

Authority-transition influence	
Model Risk Concern	Authority-transition influence
Agentic Lifecycle Impact	Model behavior may influence whether an agent escalates, delegates, or proceeds under delegated authority. This is a lifecycle risk, not only a model-performance risk.
Relevant MROs	MRO-02, MRO-07
Control Evidence	Authority boundary object, authority drift detection log

Multi-agent handoff behavior	
Model Risk Concern	Multi-agent handoff behavior
Agentic Lifecycle Impact	Model behavior may influence how agents transfer responsibility, inherit constraints, or partition evidence during handoffs.
Relevant MROs	MRO-06, MRO-08
Control Evidence	Responsibility transfer object, evidence partition

Evidence interpretation risk	
Model Risk Concern	Evidence interpretation risk
Agentic Lifecycle Impact	Model behavior may influence how evidence is interpreted, summarized, or presented to human reviewers. This may alter accepted outcome states.
Relevant MROs	MRO-04, MRO-08
Control Evidence	Accepted outcome state, evidence partition, review packet validation

Interpretation: Traditional model risk asks whether a model is governed. Agentic model risk asks whether model behavior can change lifecycle authority, tool use, evidence, review, and acceptance states. Model risk management in agentic systems requires mapping model changes to lifecycle conformance, not only to model performance.

10.4 Cybersecurity Controls Integration

10.4.1 Why Cybersecurity and Lifecycle Governance Intersect

Agentic AI systems create new attack surfaces that traditional cybersecurity controls were not designed to address. Agent execution introduces risks including prompt injection, tool misuse, credential theft, and data exfiltration. These risks are not purely technical vulnerabilities. They are lifecycle governance failures that manifest as security incidents.

Prompt injection attacks exploit the boundary between user input and system instructions. Tool misuse occurs when an agent invokes tools outside delegated authority or without proper authorization. Credential theft in agentic systems may involve stealing API keys, tool credentials, or authority tokens that enable unauthorized agent execution. Data exfiltration may occur when agents access, process, or transmit data beyond their delegated scope or privacy boundaries.

Lifecycle governance provides the authorization and evidence layer that cybersecurity controls require. Without lifecycle governance, cybersecurity controls cannot answer: Who authorized this agent to execute?

What authority boundary was violated? Which tool action crossed a liability boundary? What evidence exists to reconstruct the incident? Which human role should have reviewed this action? What accepted outcome state was compromised?

Cybersecurity and lifecycle governance are complementary layers. Cybersecurity controls protect the execution environment, credentials, tools, and data flows. Lifecycle governance defines the authority, responsibility, evidence, and acceptance structures that cybersecurity controls must preserve under attack.

Reference frameworks:

- OWASP LLM Top 10 (prompt injection, insecure output handling, training data poisoning, model denial of service, supply chain vulnerabilities, sensitive information disclosure, insecure plugin design, excessive agency, overreliance, model theft)
- NIST AI Risk Management Framework (Govern, Map, Measure, Manage functions applied to AI systems)
- MITRE ATLAS (Adversarial Threat Landscape for Artificial-Intelligence Systems)

10.4.2 Cybersecurity Control Mapping

The following table maps cybersecurity control domains to agentic attack surfaces, lifecycle governance objects, MRO dimensions, and evidence outputs.

Table T-10-03: Cybersecurity Control Mapping

Identity and Access Management	
Cybersecurity Control Domain	Identity and Access Management
Agentic Attack Surface	Agent identity spoofing, unauthorized agent instantiation, role escalation
Lifecycle Governance Object	Agent identity lifecycle record, role-responsibility map, authority boundary object
MRO Dimension	MRO-01, MRO-02
Evidence Output	Agent identity registry, authority boundary validation log

Prompt Injection Defense	
Cybersecurity Control Domain	Prompt Injection Defense
Agentic Attack Surface	Malicious user input altering agent behavior, instruction override, goal hijacking
Lifecycle Governance Object	Authority boundary object, accepted outcome state, evidence partition
MRO Dimension	MRO-02, MRO-04, MRO-08
Evidence Output	Authority boundary violation log, accepted outcome rejection record, evidence partition integrity check

Tool Misuse Prevention

Cybersecurity Control Domain	Tool Misuse Prevention
Agentic Attack Surface	Agent invoking unauthorized tools, tool action exceeding delegated scope, tool credential theft
Lifecycle Governance Object	Tool-action liability boundary, authority boundary object, tool credential registry
MRO Dimension	MRO-02, MRO-05
Evidence Output	Tool-action authorization log, tool-action liability record, credential access audit trail

Credential Management

Cybersecurity Control Domain	Credential Management
Agentic Attack Surface	API key theft, tool credential exfiltration, authority token compromise
Lifecycle Governance Object	Agent identity lifecycle record, tool credential registry, processor chain record
MRO Dimension	MRO-01, MRO-05, MRO-14
Evidence Output	Credential rotation log, credential access audit trail, processor chain validation

Data Exfiltration Prevention

Cybersecurity Control Domain	Data Exfiltration Prevention
Agentic Attack Surface	Agent accessing data beyond delegated scope, unauthorized data transmission, privacy boundary violation
Lifecycle Governance Object	Privacy lifecycle map, evidence partition, data subject rights reconciliation record
MRO Dimension	MRO-08, MRO-10, MRO-13
Evidence Output	Data access audit trail, evidence partition integrity check, privacy boundary violation log

Audit Trail Integrity

Cybersecurity Control Domain	Audit Trail Integrity
Agentic Attack Surface	Evidence tampering, log manipulation, responsibility chain corruption
Lifecycle Governance Object	Evidence partition, incident closure record, accepted outcome state
MRO Dimension	MRO-04, MRO-08, MRO-16
Evidence Output	Evidence integrity hash, audit trail validation log, incident closure record

Supply-Chain Integrity	
Cybersecurity Control Domain	Supply-Chain Integrity
Agentic Attack Surface	Compromised model provider, malicious tool plugin, subprocessor data breach
Lifecycle Governance Object	Processor chain record, vendor substitution conformance test, model version registry
MRO Dimension	MRO-14 , MRO-15
Evidence Output	Processor chain validation log, vendor substitution test result, model version integrity check

Incident Response Linkage	
Cybersecurity Control Domain	Incident Response Linkage
Agentic Attack Surface	Agentic incident detection, responsibility assignment, evidence reconstruction, remediation closure
Lifecycle Governance Object	Incident closure record, accepted outcome state, evidence partition, role-responsibility map
MRO Dimension	MRO-01 , MRO-04 , MRO-08 , MRO-16
Evidence Output	Incident closure record, responsibility assignment log, evidence reconstruction report, remediation acceptance record

Interpretation: Cybersecurity controls in agentic systems must preserve lifecycle governance structures under attack. Traditional cybersecurity controls protect infrastructure, credentials, and data. Agentic cybersecurity controls must also protect authority boundaries, responsibility chains, evidence partitions, and accepted outcome states.

10.4.3 Integration Note

Cybersecurity and lifecycle governance are complementary layers, not substitutes. Cybersecurity controls without lifecycle governance cannot answer who authorized, executed, reviewed, and accepted agentic work. Lifecycle governance without cybersecurity controls cannot protect authority, credentials, tools, evidence, and data under attack. Neither replaces the other. Both must be present for enterprise-grade agentic AI.

Boundary: This section is a governance crosswalk, not a cybersecurity audit framework, penetration testing guide, or security certification procedure. Detailed cybersecurity assurance procedures for agentic systems are reserved for companion papers, as outlined in Chapter 16.

10.5 Board Reporting and Governance Escalation

10.5.1 What Boards Need to Know About Agentic AI

Boards of directors are increasingly responsible for AI risk oversight. The EU AI Act Article 9 requires high-risk AI system providers to establish quality management systems with board-level accountability. The NIST AI Risk Management Framework emphasizes governance as the foundational function, requiring senior leadership and board engagement in AI risk decisions. SEC cybersecurity disclosure rules establish precedent for board-level oversight of technology risk, including emerging risks from AI systems.

Agentic AI creates new board-level concerns that traditional AI risk reporting does not address. Boards must understand lifecycle accountability gaps, unresolved disputes, remediation failures, and privacy incidents in agentic systems. Traditional AI risk reporting focuses on model performance, bias metrics, and deployment status. Agentic AI risk reporting must address: Who authorized this agent to execute? What authority boundary was violated? Which disputes remain unresolved? Which remediation actions were accepted or rejected? Which privacy incidents occurred during agent execution?

The key question for board reporting is: Can the board receive lifecycle status as a structured, auditable, reproducible report? Without lifecycle governance, board reporting on agentic AI becomes anecdotal, inconsistent, and non-auditable. With lifecycle governance, board reporting can be structured around RCCS coverage, ALCS conformance, open disputes, remediation closure rates, and privacy incident summaries.

10.5.2 Board-Level Reporting Elements

The following table defines board-level reporting elements for agentic AI, mapped to lifecycle object sources, reporting frequency, escalation triggers, and regulatory drivers.

Table T-10-04: Board Reporting Elements for Agentic AI

Reporting Element	Lifecycle Object Source	Frequency	Escalation Trigger	Regulatory Driver
Lifecycle Accountability Summary	RCCS baseline score, ALCS conformance score, role-responsibility map coverage	Quarterly	RCCS score below threshold, ALCS score degradation, accountability gap in high-risk system	EU AI Act Art. 9 (quality management), NIST AI RMF Govern function
Open Dispute Register	Dispute resolution record, accepted outcome state, incident closure record	Quarterly	Dispute unresolved beyond SLA, dispute involving high-risk system, dispute escalated to external authority	EU AI Act Art. 72 (dispute resolution), GDPR Art. 79 (right to judicial remedy)
Remediation Closure Rate	Incident closure record, accepted outcome state, remediation record	Quarterly	Remediation closure rate below threshold, high-severity incident unresolved, remediation rejected by human authority	NIST AI RMF Manage function, EU AI Act Art. 9 (corrective actions)
Privacy Incident Summary	Privacy lifecycle map, data subject rights reconciliation record, evidence partition, incident closure record	Quarterly	Privacy incident involving personal data breach, data subject rights request unresolved, cross-border data transfer violation	GDPR Art. 33-34 (breach notification), EU AI Act Art. 10 (data governance)
Vendor and Tool Substitution Log	Vendor substitution conformance test, processor chain record, model version registry	Semi-annually	Vendor lock-in risk identified, substitution test failure, processor chain integrity violation	EU AI Act Art. 25 (obligations of importers), Art. 28 (obligations of distributors), DORA Art. 28 (ICT third-party risk)
Drift and Remediation Status	Authority drift detection log, monitoring record, incident/remediation closure record	Quarterly	Authority drift detected in high-risk system, remediation unresolved, improvement actions stalled	NIST AI RMF Measure function, EU AI Act Art. 9 (monitoring and logging)

Interpretation: Board reporting for agentic AI must be structured, auditable, and reproducible. Lifecycle governance objects provide the data sources for board-level risk reporting. Boards should receive lifecycle accountability summaries, not only model performance metrics.

10.5.3 Escalation Note

Not all lifecycle governance issues require board escalation. Management reporting addresses routine lifecycle conformance, evidence validation, and remediation closure. Board escalation is triggered when lifecycle accountability gaps, unresolved disputes, remediation failures, or privacy incidents exceed management authority, involve high-risk systems, or indicate systemic governance failures.

Lifecycle objects provide the structured data needed for escalation decisions. RCCS-T, RCCS-M, and ALCS scores can indicate whether governance coverage or lifecycle conformance is degrading. Open dispute registers indicate whether remediation processes are functioning. Privacy incident summaries indicate whether data governance controls are effective. Vendor substitution logs indicate whether supply-chain risk is managed.

Boundary: This section is a governance crosswalk for board reporting, not a corporate governance framework, securities law compliance opinion, or board fiduciary duty analysis. Detailed board reporting procedures and governance escalation frameworks are reserved for companion papers, as outlined in Chapter 16.

10.6 Procurement Scorecard Application

10.6.1 Procurement Lifecycle Governance Requirements

When acquiring agentic AI tools, platforms, or services, procurement teams need lifecycle governance criteria to evaluate vendor capabilities. Traditional procurement scorecards focus on functionality, performance, pricing, and support. Agentic AI procurement must also evaluate lifecycle accountability, evidence exportability, privacy controls, and remediation records.

The EU AI Act establishes supply-chain obligations for AI system providers, importers, and distributors. Article 25 requires importers to ensure that providers have complied with conformity assessment procedures and that systems are accompanied by required documentation. Article 28 requires distributors to verify that systems bear required conformity markings and are accompanied by instructions for use. DORA (Digital Operational Resilience Act) Article 28 establishes ICT third-party risk management requirements for financial entities, including contractual arrangements, exit strategies, and substitution plans.

The NIST AI Risk Management Framework emphasizes supply-chain risk management as part of the Map function, requiring organizations to identify and document AI system dependencies, third-party components, and data sources. Procurement decisions must consider whether vendors support lifecycle evidence export, whether tool actions can be bounded by liability boundaries, and whether agent identity lifecycles are documented.

Key procurement criteria:

- Does the vendor support lifecycle evidence export in a structured, auditable format?
- Can tool actions be bounded by tool-action liability boundaries?
- Is the agent identity lifecycle documented and auditable?
- Can the organization reconstruct processor and subprocessor chains?
- Does the vendor support privacy-preserving validation?
- Can the organization substitute vendors without losing lifecycle conformance?

10.6.2 Procurement Scorecard Table

The following table defines procurement scorecard criteria for agentic AI systems, mapped to lifecycle objects, acceptable evidence, risk if absent, and regulatory references.

Table T-10-05: Agentic AI Procurement Scorecard

Criterion	Lifecycle Object	Acceptable Evidence	Risk if Absent	Regulatory Reference
Agent Identity Lifecycle	Agent identity lifecycle record, role-responsibility map	Agent identity registry, identity creation/termination log, role assignment audit trail	Cannot reconstruct who authorized, executed, or accepted agentic work; accountability gap	EU AI Act Art. 9 (quality management), NIST AI RMF Govern function
Tool-Action Liability Boundary	Tool-action liability boundary, tool credential registry	Tool-action authorization policy, tool invocation log, liability boundary validation record	Cannot determine liability for tool misuse, unauthorized tool invocation, or tool-related incidents	EU AI Act Art. 28 (obligations of distributors), DORA Art. 28 (ICT third-party risk)
Evidence Export Capability	Evidence partition, evidence minimization policy	Evidence export API, evidence schema documentation, evidence integrity validation	Cannot provide evidence to auditors, regulators, or data subjects; audit readiness failure	GDPR Art. 15 (right of access), EU AI Act Art. 12 (record-keeping)
Privacy Data-Flow Map	Privacy lifecycle map, processor chain record, data subject rights reconciliation record	Privacy impact assessment, data flow diagram, processor/subprocessor registry	Cannot reconstruct data processing activities; GDPR compliance failure; privacy incident response failure	GDPR Art. 28 (processor obligations), Art. 30 (records of processing), EU AI Act Art. 10 (data governance)
Incident Response Support	Incident closure record, accepted outcome state, remediation record	Incident response procedure, incident closure SLA, remediation acceptance criteria	Cannot close incidents with accepted remediation; redress failure; regulatory reporting failure	EU AI Act Art. 72 (dispute resolution), GDPR Art. 33-34 (breach notification)
Vendor Substitution Support	Vendor substitution conformance test, model version registry, processor chain record	Vendor substitution test procedure, data portability support, lifecycle conformance revalidation plan	Vendor lock-in; cannot switch vendors without losing lifecycle conformance; supply-chain risk	EU AI Act Art. 25 (obligations of importers), DORA Art. 28 (exit strategies)
Drift Monitoring Capability	Authority drift detection log, monitoring record, incident/remediation closure record	Drift detection procedure, drift alert mechanism, drift remediation workflow	Cannot detect authority boundary violations, lifecycle conformance degradation, or unresolved remediation	NIST AI RMF Measure function, EU AI Act Art. 9 (monitoring and logging)
Dispute Resolution Support	Dispute resolution record, accepted outcome state, incident closure record	Dispute resolution procedure, dispute escalation path, dispute closure criteria	Cannot resolve disputes with accepted outcomes; redress failure; regulatory escalation risk	EU AI Act Art. 72 (dispute resolution), GDPR Art. 79 (right to judicial remedy)

Interpretation: Procurement scorecards for agentic AI must evaluate lifecycle governance capabilities, not only functional performance. Vendors that cannot support lifecycle evidence export, tool-action liability boundaries, or vendor substitution introduce governance risk that may outweigh functional benefits.

10.6.3 Scorecard Note

This scorecard is a governance reference tool, not a vendor certification framework, audit requirement, or procurement specification. Procurement teams should adapt the scorecard to their specific risk profile, regulatory environment, and organizational control requirements. The scorecard does not replace vendor due diligence, contract negotiation, or legal review. It provides lifecycle governance criteria that procurement teams can integrate into existing vendor evaluation processes.

Boundary: This section is a governance crosswalk for procurement, not a vendor audit framework, accreditation procedure, or legal procurement specification. Detailed procurement assurance procedures and vendor evaluation frameworks are reserved for companion papers, as outlined in Chapter 16.

10.7 Ongoing Monitoring and Continuous Improvement

10.7.1 Why Monitoring Must Be Lifecycle-Integrated

Traditional observability monitors execution metrics: latency, throughput, error rates, resource utilization, and model performance. These metrics are necessary but insufficient for agentic systems. Lifecycle governance requires monitoring accountability continuity, not only execution performance.

Lifecycle monitoring asks: Are responsibility assignments still valid? Are evidence chains still complete? Are tool-action liability boundaries still operative? Are human review roles still engaged? Are accepted outcomes still being produced? Are disputes being resolved? Are privacy boundaries being maintained?

Lifecycle monitoring primarily draws from delegated authority, evidence partitioning, substitution conformance, privacy lifecycle mapping, and incident/remediation closure objects. It does not introduce separate lifecycle-drift or continuous-improvement MROs. Monitoring events should be interpreted through canonical lifecycle objects: authority drift uses [MRO-07](#), evidence-chain breaks use [MRO-08](#) and [MRO-16](#), substitution alerts use [MRO-09](#) and [MRO-15](#), privacy data-flow deviations use [MRO-10](#), [MRO-13](#), and [MRO-14](#), and incident closure uses [MRO-16](#).

Lifecycle monitoring is not a replacement for traditional observability. It is a complementary layer that monitors governance continuity alongside execution performance. A system may perform well on execution metrics while experiencing lifecycle drift, responsibility gaps, evidence chain breaks, or dispute backlogs.

10.7.2 Monitoring Indicators Table

The following table defines lifecycle monitoring indicators, mapped to lifecycle objects, alert conditions, MRO dimensions, and remediation paths.

Table T-10-06: Lifecycle Monitoring Indicators

Indicator	Lifecycle Object	Alert Condition	MRO Dimension	Remediation Path
Responsibility Drift	Role-responsibility map, agent identity lifecycle record, authority drift detection log	Role assignment unchanged beyond revalidation period; role holder departed; responsibility gap detected	MRO-01 , MRO-06 , MRO-07	Revalidate role-responsibility map; reassign agent identity; update lifecycle record; document drift remediation
Evidence Chain Break	Evidence partition, accepted outcome state, incident closure record	Evidence partition integrity check failed; evidence missing for lifecycle phase; accepted outcome lacks supporting evidence	MRO-08 , MRO-16	Reconstruct evidence chain; validate evidence partition; re-execute acceptance review; document evidence remediation
Tool-Action Boundary Violation	Tool-action liability boundary, authority boundary object, authority drift detection log	Tool invoked outside delegated scope; tool action exceeded liability boundary; unauthorized tool credential access	MRO-02 , MRO-05 , MRO-07	Halt tool execution; revalidate authority boundary; update tool-action liability boundary; document boundary violation and remediation
Human-Role Absence	Role-responsibility map, accepted outcome state, human review packet	Human review required but not performed; acceptance decision made without human authority; escalation rule violated	MRO-01 , MRO-04	Escalate to human authority; re-execute review with human role; update role-responsibility map; document human-role remediation
Dispute Backlog	Dispute resolution record, evidence partition, incident closure record	Dispute unresolved beyond SLA; dispute escalation path exhausted; dispute closure rate below threshold	MRO-04 , MRO-08 , MRO-16	Escalate dispute resolution; engage external authority if needed; update dispute resolution procedure; document backlog remediation
Privacy Data-Flow Deviation	Privacy lifecycle map, data subject rights reconciliation record, processor chain record	Data processed outside documented flow; processor/subprocessor not in	MRO-10 , MRO-13 , MRO-14	Halt data processing; revalidate privacy lifecycle map; update processor chain record; resolve data sub-

Indicator	Lifecycle Object	Alert Condition	MRO Dimension	Remediation Path
		registry; data subject rights request unresolved		ject rights request; document privacy remediation
Vendor Substitution Alert	Cross-project reuse record, vendor substitution conformance test, model version registry, processor chain record	Vendor substitution test failed; model version changed without revalidation; processor chain integrity violated	MRO-09, MRO-15	Revalidate vendor substitution conformance; update model version registry; revalidate processor chain; document substitution remediation
Continuous Improvement Stall	Monitoring record, incident/remediation closure record	Feedback loop inactive beyond threshold; drift detected but not remediated; improvement actions not implemented	RCCS-10, MRO-16	Reactivate monitoring review; prioritize drift remediation; implement improvement actions; update incident/remediation closure record

Interpretation: Lifecycle monitoring indicators detect governance continuity failures, not only execution performance failures. Alert conditions trigger remediation paths that restore lifecycle conformance. Monitoring without remediation is observation without governance.

10.7.3 Continuous Improvement Note

Lifecycle monitoring feeds back into ALCS lifecycle records, MRO evidence records, and RCCS coverage assessments. This is the governance loop: authorization → execution → evidence → review → remediation → improvement → re-authorization. Continuous improvement is not a separate MRO. It is the feedback mechanism that ensures lifecycle governance adapts to changing systems, roles, tools, vendors, and regulatory requirements.

Authority drift detection (MRO-07), evidence partitioning (MRO-08), cross-project reuse compliance (MRO-09), privacy lifecycle mapping (MRO-10), substitution conformance (MRO-15), and incident/remediation closure (MRO-16) together show whether governance structures are degrading and whether remediation is documented, accepted, and integrated into lifecycle records. Without this feedback loop, lifecycle governance becomes static and brittle. With it, lifecycle governance becomes adaptive and resilient.

Boundary: This section is a governance crosswalk for lifecycle monitoring, not a real-time monitoring product specification, SRE framework, or SLA definition. Detailed monitoring implementation procedures and continuous improvement frameworks are reserved for companion papers, as outlined in Chapter 16.

10.8 Chapter Closure and Bridge to Chapter 16

10.8.1 What Chapter 10 Has Established

Chapter 10 has established the enterprise control crosswalk for AI Agent Lifecycle Governance. It has shown how lifecycle governance objects - MROs, RCCS-T/RCCS-M dimensions, ALCS dimensions, and lifecycle records - map into enterprise control language: audit, insurance, procurement, board reporting, model risk management, cybersecurity, incident response, and monitoring.

The crosswalk demonstrates that lifecycle governance is not an abstract protocol or research model. It is a structured method for producing the evidence that enterprise controls require. RCCS-T, RCCS-M, and ALCS scores can become bounded scorecard inputs. MROs become audit evidence requirements. Lifecycle conformance becomes vendor substitution readiness. Evidence partitioning becomes privacy-preserving validation. Accepted outcome compliance becomes board reporting on agentic risk.

Enterprise adoption of agentic AI requires this crosswalk layer. Without it, lifecycle governance remains disconnected from enterprise control frameworks, procurement processes, board reporting structures, and

audit requirements. With it, lifecycle governance becomes the evidence layer that enterprise controls can rely upon.

The chapter has provided:

- Enterprise control overlay mapping lifecycle governance to audit, insurance, and board reporting readiness (Section 10.1)
- Enterprise control crosswalk tables mapping control topics to MROs, RCCS-T/RCCS-M/ALCS dimensions, and evidence outputs (Section 10.2)
- Model risk management mapping showing how model changes propagate through lifecycle conformance (Section 10.3)
- Cybersecurity control mapping showing how lifecycle governance supports cybersecurity under attack (Section 10.4)
- Board reporting elements defining structured, auditable lifecycle risk reporting (Section 10.5)
- Procurement scorecard criteria for evaluating vendor lifecycle governance capabilities (Section 10.6)
- Lifecycle monitoring indicators for detecting governance continuity failures (Section 10.7)

10.8.2 What Chapter 16 Will Add

Chapter 10 provides the control language. Chapter 16 provides the boundary to companion papers. Chapter 16 is titled "Boundary to Companion Papers: Evidence, Assurance, and Insurability."

Chapter 16 will establish:

- The boundary between this white paper and companion papers on evidence-based validation, assurance frameworks, and insurability considerations
- Evidence-based validation patterns that external reviewers, auditors, and insurers can use to validate lifecycle conformance without full raw data disclosure
- Assurance framework mapping showing how lifecycle governance supports ISO 42001, SOC 2, and other assurance standards
- Insurability considerations showing how lifecycle governance evidence supports cyber insurance, professional liability insurance, and AI-specific insurance products
- Companion paper scope definitions clarifying what this white paper establishes and what companion papers will address

Chapter 16 is the final chapter of this white paper. It does not provide detailed assurance procedures, insurance underwriting models, or certification frameworks. It defines the boundary and provides the bridge to companion work.

10.8.3 Chapter 10 Closure Statement

Chapter 10 has translated lifecycle governance into enterprise control language. It has shown that AI Agent Lifecycle Governance is not only a governance protocol. It is a structured evidence layer that enterprise controls, procurement processes, board reporting, audit frameworks, cybersecurity controls, and monitoring systems can rely upon.

The crosswalk does not replace internal control frameworks, audit procedures, or procurement policies. It shows where agentic lifecycle objects can provide evidence for enterprise control questions. Organizations should integrate these objects into existing control frameworks rather than treating them as separate compliance requirements.

Boundary: Chapter 10 is a governance crosswalk, not a certification framework, audit specification, insurance underwriting model, or legal compliance opinion. It provides the mapping from lifecycle governance objects to enterprise control language. Detailed assurance procedures, certification frameworks, and insurance models are reserved for companion papers, as outlined in Chapter 16.

Chapter 16 will establish the boundary to companion papers and provide the bridge from this white paper to evidence-based validation, assurance frameworks, and insurability considerations.

11. Comparative Field and System Positioning

MAIN RESULT Current systems expose different lifecycle strengths and gaps by category.	METHOD / LENS Qualitative positioning across protocol, platform, SDK, and orchestration categories.
EVIDENCE BASIS Readers need category context before reading system mappings.	KEY TAKEAWAY / BOUNDARY The comparison is source-qualified and non-ranking.

Version: v0.3.2-FRC-R3

11.0 Why System Positioning Matters

This chapter positions the systems and categories that will be evaluated in Chapter 12. It explains the comparison method, establishes non-ranking boundaries, defines the systems in scope, and provides high-level positioning for each system before detailed mapping begins.

System positioning matters because the field of agentic AI governance is fragmented. Some systems focus on model governance. Some focus on agent orchestration. Some focus on observability. Some focus on enterprise AI management. Some focus on lifecycle protocol semantics. The comparison in this white paper asks a specific question: does a system expose governance primitives and lifecycle responsibility objects that help prove agentic work moved from intent to accepted outcome under authority, evidence, privacy, and remediation controls?

This is not a product ranking. This is not a legal compliance judgment. This is not a procurement recommendation. This is not an endorsement or criticism of any vendor. It is a responsibility-semantics mapping based on public evidence and author inference where explicitly disclosed.

The purpose of this chapter is to prepare the reader for Chapter 12 detailed system mappings by establishing what the comparison evaluates, what it does not evaluate, and how to interpret RCCS-T, RCCS-M, and ALCS positioning in the context of system strengths and lifecycle gaps.

11.1 Comparison Method and Non-Ranking Boundary

The comparison method used in this white paper evaluates systems against RCCS-T, RCCS-M, and ALCS dimensions. RCCS-T measures traditional regulatory compliance coverage: whether a system provides the governance primitives that existing regulation requires. RCCS-M measures MRO-adjusted regulatory coverage: whether those obligations can be expressed through lifecycle responsibility objects. ALCS measures agentic lifecycle conformance: whether a system provides mature lifecycle objects that make agentic responsibility auditable, provable, and transferable across agents, humans, tools, projects, vendors, and organizational boundaries.

The comparison asks whether a system exposes governance primitives and lifecycle responsibility objects. It does not ask whether a system is the best product, the most popular framework, the fastest execution engine,

the most feature-complete platform, or the right procurement choice for a specific enterprise. Those are different questions that depend on organizational context, deployment requirements, risk appetite, budget, technical stack, and strategic priorities.

This is a responsibility-semantics mapping, not a product ranking. The comparison evaluates system capability to express governance primitives and lifecycle objects, not product quality, market leadership, developer popularity, execution performance, or procurement suitability.

What the comparison evaluates:

- Whether a system defines governance primitives for risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring (RCCS-T dimensions)
- Whether a system can express those obligations through Missing Regulatory Objects and lifecycle responsibility semantics (RCCS-M interpretation)
- Whether a system defines lifecycle responsibility objects for human-role-to-MAS mapping, delegated authority boundaries, agent role distinction, accepted outcome compliance, tool-action liability, responsibility transfer, authority drift detection, evidence partitioning, cross-project reuse, privacy lifecycle mapping, privacy-preserving validation, evidence minimization, data subject rights reconciliation, processor chain tracking, and vendor substitution conformance (ALCS dimensions)
- Whether public documentation, API specifications, protocol definitions, or reproducible evidence artifacts support the evaluation
- Where evidence is weak or unavailable, whether author inference is explicitly disclosed

What the comparison does not evaluate:

- Product quality, feature completeness, or execution performance
- Market leadership, developer popularity, or community adoption
- Procurement suitability, pricing, or vendor support
- Legal compliance with specific regulations or jurisdictions
- Certification, conformity assessment, or regulatory approval
- Endorsement or criticism of any vendor or product

The comparison uses the evidence hierarchy established in the front matter and harmonized through source review follow-up. **L1** evidence covers binding legal, regulatory, and formal standards baselines; it does not prove product implementation. **L2** evidence covers official product, protocol, API, developer, platform, or repository documentation. **L3** evidence covers independent audit reports, credible research papers, or independent reports. **L4** evidence covers vendor claims, marketing pages, blogs, and announcements. **L5** evidence covers author inference or analytical interpretation and is explicitly disclosed. The canonical multipliers are **L1** = 1.00, **L2** = 0.85, **L3** = 0.75, **L4** = 0.55, and **L5** = 0.35.

The comparison is not a certification program. It does not certify that any system is compliant with any regulation. It does not guarantee that any system meets all legal obligations. It does not substitute for legal counsel, compliance review, or risk assessment. It measures system capability to express governance primitives and lifecycle objects, not organizational conformance to legal requirements.

11.2 System Categories in Scope

The comparison includes eight systems or system categories. These were selected based on their relevance to enterprise agentic AI governance, their public documentation availability, and their representation of different

architectural approaches to agent lifecycle management.

The eight systems or categories in scope are:

1. **MPLP (Multi-Agent Lifecycle Protocol)** — A lifecycle protocol path with limited traditional RCCS-T product-platform coverage and strong author-analytical RCCS-M / ALCS alignment, designed to expose lifecycle responsibility objects as first-class protocol primitives.
1. **IBM watsonx.governance** — An enterprise AI governance platform with strong regulatory compliance coverage, model governance, risk management, and policy enforcement capabilities.
1. **Microsoft Azure AI Foundry** — A cloud-native AI platform with model management, responsible AI tooling, content safety, and enterprise integration capabilities.
1. **AWS Bedrock / Guardrails / AgentCore** — A cloud AI service suite with model access, guardrails, agent orchestration, and enterprise security integration.
1. **Google Vertex AI / ADK / Model Armor** — A cloud AI platform with model management, agent development kit, model armor safety controls, and enterprise AI management.
1. **LangGraph / LangSmith** — An open-source agent orchestration framework with observability, tracing, and workflow management capabilities.
1. **OpenAI Agents SDK** — A developer-focused agent framework with model access, tool use, and execution orchestration capabilities.
1. **CrewAI** — An open-source multi-agent collaboration framework with role-based agent design and task orchestration capabilities.

These systems represent different architectural approaches:

- **Enterprise AI governance platforms** (IBM watsonx.governance) focus on policy enforcement, risk management, and regulatory compliance coverage across the AI lifecycle.
- **Cloud AI platforms** (Microsoft Azure AI Foundry, AWS Bedrock, Google Vertex AI) focus on model access, responsible AI tooling, content safety, and enterprise integration.
- **Agent orchestration frameworks** (LangGraph, OpenAI Agents SDK, CrewAI) focus on agent execution, tool use, workflow management, and observability.
- **Lifecycle protocol paths** (MPLP) focus on lifecycle responsibility objects, evidence chain, and governance primitives as first-class protocol semantics.

The comparison does not claim that these are the only systems worth evaluating. Other systems, frameworks, and platforms exist and may provide strong governance capabilities. The selection reflects systems with sufficient public documentation to support evidence-based evaluation and systems that represent different architectural approaches to agentic AI governance.

No new systems will be added to the comparison without explicit owner approval. The comparison is scoped to these eight systems or categories to maintain focus and ensure that detailed mapping in Chapter 12 remains feasible within the white paper's scope.

11.3 Comparative Field Positioning Matrix

The following table provides high-level positioning for each system or category. It identifies the primary strength of each system, its likely RCCS-T posture (traditional regulatory compliance coverage), its likely RCCS-M posture (MRO-adjusted regulatory coverage), its likely ALCS posture (agentic lifecycle conformance), and the main lifecycle gap that Chapter 12 will examine.

Table T-11-01: Comparative Field Positioning Matrix

Wide source table split into consecutive column groups; all source rows and columns are preserved.

Continuation 1 of 2

System / Category	Primary Strength	RCCS-T Posture
MPLP	Lifecycle protocol semantics; governance primitives as first-class objects	Limited to Moderate (not a product governance platform)
IBM watsonx.governance	Enterprise AI governance; policy enforcement; model risk management; regulatory compliance coverage	Strong (designed for regulatory compliance)
Microsoft Azure AI Foundry	Cloud-native AI platform; responsible AI tooling; content safety; enterprise integration	Moderate to Strong (responsible AI focus)
AWS Bedrock / Guardrails / AgentCore	Cloud AI services; guardrails; agent orchestration; enterprise security integration	Moderate to Strong (guardrails and security focus)
Google Vertex AI / ADK / Model Armor	Cloud AI platform; agent development kit; model armor safety controls; enterprise AI management	Moderate to Strong (safety and management focus)
LangGraph / LangSmith	Agent orchestration; observability; tracing; workflow management	Moderate (observability and tracing support governance)
OpenAI Agents SDK	Developer-focused agent framework; model access; tool use; execution orchestration	Low to Moderate (execution focus, governance primitives require validation)
CrewAI	Multi-agent collaboration; role-based agent design; task orchestration	Low to Moderate (collaboration focus, governance primitives require validation)

Continuation 2 of 2

System / Category	RCCS-M Posture	ALCS Posture	Main Lifecycle Gap to Examine
MPLP	Strong (protocol semantics map to MRO-like objects)	Strong (protocol designed for lifecycle responsibility objects)	Enterprise integration; adoption pathway; implementation maturity
IBM watsonx.governance	Partial to Moderate (lifecycle object semantics require validation)	Moderate (governance focus, but lifecycle object semantics require validation)	Agentic lifecycle object exposure; multi-agent responsibility transfer; evidence partitioning
Microsoft Azure AI Foundry	Partial to Moderate (MRO-level semantics require implementation design)	Moderate (platform capabilities exist, but lifecycle semantics require validation)	Lifecycle responsibility objects; authority boundary enforcement; accepted outcome compliance
AWS Bedrock / Guardrails / AgentCore	Partial to Moderate (MRO-level semantics require implementation design)	Moderate (agent orchestration exists, but lifecycle object semantics require validation)	Lifecycle responsibility objects; evidence partitioning; responsibility transfer across agents
Google Vertex AI / ADK / Model Armor	Partial to Moderate (MRO-level semantics require implementation design)	Moderate (ADK provides agent primitives, but lifecycle semantics require validation)	Lifecycle responsibility objects; authority boundary enforcement; privacy lifecycle mapping
LangGraph / LangSmith	Moderate (workflow state and traces support MRO-adjacent analysis)	Moderate (orchestration and tracing exist, but lifecycle object semantics require validation)	Governance primitives; authority boundary enforcement; accepted outcome compliance; evidence partitioning
OpenAI Agents SDK	Partial (SDK primitives can support lifecycle analysis, not full governance)	Low to Moderate (execution orchestration exists, but lifecycle object semantics require validation)	Governance primitives; lifecycle responsibility objects; authority boundary enforcement; evidence chain
CrewAI	Partial (role/task primitives can support lifecycle analysis, not full governance)	Low to Moderate (role-based design exists, but lifecycle object semantics require validation)	Governance primitives; lifecycle responsibility objects; authority boundary enforcement; evidence chain

This table provides initial positioning based on public documentation, system architecture, and stated design goals. Chapter 12 tests these positions through evidence-qualified, provisional system mappings. Where numerical scoring is not fully source-bound, mappings remain qualitative and non-ranking.

The table shows that systems have different strengths. Enterprise AI governance platforms tend to score higher on RCCS-T because they are designed for policy enforcement, risk management, and regulatory compliance. Lifecycle protocol paths may score lower on RCCS-T when they are not product governance platforms, but higher on RCCS-M and ALCS when they expose lifecycle responsibility objects as first-class protocol primitives. Cloud AI platforms and agent orchestration frameworks tend to show mixed RCCS-T/RCCS-M/ALCS posture because they provide some governance capabilities and some lifecycle primitives, but may not expose full lifecycle responsibility semantics.

The main lifecycle gaps identified in the table reflect the areas where Chapter 12 will focus detailed evaluation. For systems with strong RCCS-T posture, the gap is often agentic lifecycle object exposure. For systems with strong RCCS-M and ALCS posture, the gap is often enterprise integration and adoption maturity. For systems with mixed posture, the gap is often governance primitives and lifecycle responsibility objects that require validation against public documentation.

11.4 Reading the RCCS-T / RCCS-M / ALCS Positioning

RCCS-T, RCCS-M, and ALCS measure different layers. RCCS-T measures traditional regulatory compliance coverage: whether a system provides the governance primitives that existing regulation requires. RCCS-M measures whether those obligations can be expressed through Missing Regulatory Objects. ALCS measures agentic lifecycle conformance: whether a system provides mature lifecycle objects that make agentic responsibility auditable, provable, and transferable.

A system can have a strong RCCS-T score and weaker RCCS-M or ALCS score. This is not a contradiction. It means the system may provide strong regulatory compliance coverage through risk management, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring capabilities while lacking the deeper lifecycle responsibility objects required to prove authority boundaries, responsibility transfer, evidence partitioning, privacy lifecycle mapping, and vendor substitution conformance in multi-agent execution.

A system can have a strong RCCS-M and ALCS score with a lower RCCS-T score. This is also not a contradiction. It means the system may provide strong lifecycle responsibility semantics for human-role-to-MAS mapping, delegated authority boundaries, agent role distinction, accepted outcome compliance, tool-action liability, responsibility transfer, authority drift detection, evidence partitioning, and other lifecycle objects while having less mature productized governance coverage for risk management, documentation, record-keeping, or monitoring.

The strongest future profile would provide strong RCCS-T, RCCS-M, and ALCS together. This means the system provides the governance primitives that existing regulation requires, expresses those obligations through MRO-like lifecycle objects, and makes agentic responsibility auditable and transferable. Few systems currently achieve this profile because traditional regulatory coverage and agentic lifecycle conformance have evolved as separate concerns. Enterprise AI governance platforms focus on regulatory compliance. Agent orchestration frameworks focus on execution. Lifecycle protocol paths focus on responsibility semantics. The field is converging, but integration remains incomplete.

The positioning in Table T-11-01 reflects this fragmentation. Systems with strong RCCS-T posture tend to be enterprise AI governance platforms designed for regulatory compliance. Systems with strong RCCS-M and ALCS posture tend to be lifecycle protocol paths designed for responsibility semantics. Systems with mixed

posture tend to be cloud AI platforms or agent orchestration frameworks that provide some governance capabilities and some lifecycle primitives, but may not expose full lifecycle responsibility semantics.

Chapter 12 tests these positions through evidence-qualified, provisional system mappings. The positioning in this chapter is preliminary and based on public documentation, system architecture, and stated design goals. Detailed evaluation may reveal stronger or weaker capabilities than the preliminary positioning suggests, but system-level mappings remain qualitative and non-ranking unless numerical scoring is explicitly source-bound and approved.

11.5 What the Comparison Does Not Claim

The comparison does not claim that any system is certified, compliant, approved, endorsed, or recommended. It does not claim that any system is the best product, the right procurement choice, or the only solution. It does not claim that any system guarantees legal compliance, regulatory approval, or audit readiness.

The comparison is a responsibility-semantics mapping based on public evidence and author inference where explicitly disclosed. It measures system capability to express governance primitives and lifecycle objects, not organizational conformance to legal requirements. It provides a structured method for comparing systems against RCCS-T, RCCS-M, and ALCS dimensions, not a certification program or procurement recommendation.

The comparison does not claim:

- **Certification or conformity assessment:** The comparison does not certify that any system is compliant with any regulation or standard. It does not substitute for conformity assessment, regulatory approval, or third-party certification.
- **Legal compliance judgment:** The comparison does not determine whether any system meets legal obligations in any jurisdiction. Legal compliance depends on organizational practice, deployment context, risk class, and how the system is used.
- **Procurement recommendation:** The comparison does not recommend any system for procurement. Procurement decisions depend on organizational requirements, budget, technical stack, risk appetite, vendor relationships, and strategic priorities.
- **Endorsement or criticism:** The comparison does not endorse any system as superior or criticize any system as inadequate. It evaluates system capability against RCCS-T, RCCS-M, and ALCS dimensions, not product quality or market fitness.
- **Vendor quality judgment:** The comparison does not evaluate vendor quality, support, pricing, or business practices. It evaluates system capability to express governance primitives and lifecycle objects based on public documentation.
- **Execution performance or feature completeness:** The comparison does not evaluate execution performance, feature completeness, developer experience, or operational maturity. It evaluates governance primitives and lifecycle responsibility objects.
- **Market leadership or developer popularity:** The comparison does not evaluate market share, developer adoption, community size, or ecosystem maturity. It evaluates system capability against RCCS-T, RCCS-M, and ALCS dimensions.

MPLP-specific non-claim discipline:

- MPLP is positioned as a lifecycle protocol path with limited RCCS-T product-platform coverage and strong author-analytical RCCS-M / ALCS alignment. This does not mean MPLP is a required compliance mechanism, a certification program, or a regulatory standard.
- MPLP is not a certification authority. It does not certify that any system is compliant with any regulation.
- MPLP is not a regulator. It does not enforce legal obligations or issue regulatory approvals.
- MPLP is not a conformity assessment body. It does not issue conformity certificates or compliance attestations.
- MPLP may be positioned as a lifecycle protocol path that exposes governance primitives and lifecycle responsibility objects as first-class protocol semantics. This is an architectural observation, not a legal claim or procurement recommendation.

Validation Lab-specific non-claim discipline:

- The Validation Lab is referenced in Chapter 14 as an evidence-based validation pattern. It is not a certification authority, regulator, or conformity assessment body.
- The Validation Lab does not certify that any system is compliant with any regulation.
- The Validation Lab does not issue audit opinions, legal compliance judgments, or regulatory approvals.
- The Validation Lab is a validation pattern for evidence-based testing, not a certification program.

The comparison is an analytical framework for comparing systems against RCCS-T, RCCS-M, and ALCS dimensions. It provides a structured method for evaluating whether systems provide the governance primitives and lifecycle objects that regulation and lifecycle responsibility require. It does not substitute for legal counsel, compliance review, risk assessment, procurement evaluation, or vendor due diligence.

11.6 Bridge to Detailed System Mappings

Chapter 12 will provide detailed system mappings for each of the eight systems or categories in scope. For each system, Chapter 12 will evaluate:

- **RCCS-T/RCCS-M shared dimension family:** Risk Management, Data Governance, Documentation, Record-Keeping, Transparency, Human Oversight, Security, Accountability, Contestability, Monitoring
- **ALCS dimensions:** Human-Role-to-MAS Responsibility Mapping, Delegated Authority Boundary, Agent Role vs Human Role Distinction, Accepted Outcome Compliance, Tool-Action Liability Boundary, Responsibility Transfer Across Agents, Authority Drift Detection, MAS Evidence Partitioning, Cross-Project Reuse Compliance, Privacy/GDPR Lifecycle Mapping, Privacy-Preserving Third-Party Validation, Evidence Minimization and Selective Disclosure, Data Subject Rights vs Evidence Retention, Third-Party Processor/Subprocessor Chain, Vendor/Model/Runtime Substitution Conformance
- **Evidence basis:** What evidence supports the evaluation ([L1](#) , [L2](#) , [L3](#) , [L4](#) , or [L5](#))
- **Analytical posture:** qualitative RCCS-T/RCCS-M/ALCS mapping strength and evidence level; numerical profiles appear only in the Evaluation Results section when source-qualified and explicitly bounded
- **Lifecycle gaps:** What lifecycle responsibility objects or governance primitives are missing or require further validation

Chapter 12 does not create published numerical system assessments, final procurement recommendations, certification, endorsement, or legal compliance judgments. Any numerical worksheet, if used, remains internal analytical support unless explicitly source-bound and approved for publication.

Chapter 12 will provide evidence-based system mappings that allow buyers, auditors, governance teams, and protocol designers to evaluate whether systems provide the governance primitives and lifecycle objects that

regulation and lifecycle responsibility require. It will make visible the difference between systems that provide regulatory compliance coverage, systems that provide lifecycle responsibility semantics, and systems that provide both.

The positioning in this chapter prepares the reader for Chapter 12 by establishing the comparison method, defining the systems in scope, providing preliminary positioning, and clarifying what the comparison evaluates and what it does not evaluate. Chapter 12 then tests these positions with detailed evidence-qualified mapping across RCCS-T, RCCS-M, and ALCS dimensions.

F-05

Lifecycle Conformance Mapping Strength Heatmap

Figure F-05 is a qualitative, non-ranking map of lifecycle conformance visibility across systems and MRO / ALCS dimensions. It must not be read as a product ranking, procurement recommendation, or final vendor assessment.

Table T-11-02: Comparison Boundary Table

Comparison Dimension	What This Paper Evaluates	What This Paper Does Not Evaluate
Regulatory coverage	Whether a system provides governance primitives for risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring (RCCS-T dimensions), and whether those obligations can be expressed through MROs (RCCS-M interpretation)	Legal compliance with specific regulations or jurisdictions; conformity assessment; regulatory approval; certification
Lifecycle object semantics	Whether a system provides lifecycle responsibility objects for human-role-to-MAS mapping, delegated authority boundaries, agent role distinction, accepted outcome compliance, tool-action liability, responsibility transfer, authority drift detection, evidence partitioning, cross-project reuse, privacy lifecycle mapping, privacy-preserving validation, evidence minimization, data subject rights reconciliation, processor chain tracking, and vendor substitution conformance (ALCS dimensions)	Organizational conformance to legal requirements; deployment context; risk class; how the system is used
Evidence basis	Whether public documentation, API specifications, protocol definitions, or reproducible evidence artifacts support the evaluation; where evidence is weak or unavailable, whether author inference is explicitly disclosed	Vendor quality; support; pricing; business practices; market leadership; developer popularity; community size; ecosystem maturity
System mapping	System capability to express governance primitives and lifecycle objects based on public documentation and evidence-based evaluation	Product quality; feature completeness; execution performance; developer experience; operational maturity; procurement suitability; vendor relationships; strategic fit
Product quality	Not evaluated	Execution performance; feature completeness; developer experience; operational maturity; reliability; scalability; usability
Procurement suitability	Not evaluated	Organizational requirements; budget; technical stack; risk appetite; vendor relationships; strategic priorities; total cost of ownership; support quality
Legal compliance	Not evaluated	Legal compliance with specific regulations or jurisdictions; legal counsel; compliance review; risk assessment; regulatory examination
Certification / endorsement	Not evaluated	Certification; conformity assessment; regulatory approval; endorsement; criticism; vendor

Comparison Dimension	What This Paper Evaluates	What This Paper Does Not Evaluate
		quality judgment; market leadership judgment

This table clarifies the boundaries of the comparison. It ensures that readers understand what the comparison evaluates and what it does not evaluate. It prevents scope confusion and establishes clear expectations about what the white paper claims and what it does not claim.

Chapter 12: Detailed System Mappings

MAIN RESULT System mappings show uneven visible coverage across RCCS-T/RCCS-M/ALCS/MRO surfaces.	METHOD / LENS Source-qualified mapping across eight evaluated systems.
EVIDENCE BASIS The paper needs explicit profiles so readers can separate method, result, and evidence.	KEY TAKEAWAY / BOUNDARY Profiles are provisional, evidence-bound, and not procurement recommendations.

Version: v0.3.2-FRC-R3

12.0 Purpose of Detailed System Mappings

This chapter performs detailed system mappings for the eight systems introduced in Chapter 11. It evaluates public evidence of governance primitives and lifecycle responsibility objects against RCCS-T, RCCS-M, and ALCS dimensions. The purpose is to make visible which systems provide traditional governance coverage, which systems expose MRO-adjusted lifecycle responsibility semantics, where evidence is strong, where evidence is weak, and where lifecycle gaps remain.

This chapter is not a product ranking. It is not a legal compliance judgment. It is not a procurement recommendation. It is not an endorsement or criticism of any vendor. It is a responsibility-semantics mapping based on public evidence and author inference where explicitly disclosed.

The chapter evaluates whether systems expose governance primitives and lifecycle responsibility objects. It does not evaluate product quality, market leadership, pricing, procurement suitability, developer experience, or legal compliance. Those are different questions that depend on organizational context, deployment requirements, risk appetite, budget, technical stack, and strategic priorities.

What this chapter evaluates:

- Whether a system defines governance primitives for risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring (RCCS-T dimensions)
- Whether those governance obligations can be expressed through Missing Regulatory Objects and lifecycle responsibility semantics (RCCS-M interpretation)
- Whether a system defines lifecycle responsibility objects for human-role-to-MAS mapping, delegated authority boundaries, agent role distinction, accepted outcome compliance, tool-action liability, responsibility transfer, authority drift detection, evidence partitioning, cross-project reuse, privacy lifecycle mapping, privacy-preserving validation, evidence minimization, data subject rights reconciliation, processor chain tracking, and vendor substitution conformance (ALCS dimensions)
- Whether public documentation, API specifications, protocol definitions, or reproducible evidence artifacts support the evaluation
- Where evidence is weak or unavailable, whether author inference is explicitly disclosed

What this chapter does not evaluate:

- Product quality, feature completeness, or execution performance
- Market leadership, developer popularity, or community adoption
- Procurement suitability, pricing, or vendor support
- Legal compliance with specific regulations or jurisdictions
- Certification, conformity assessment, or regulatory approval
- Endorsement or criticism of any vendor or product

Evidence boundary:

- Public documentation may be incomplete. Absence of evidence is not evidence of absence.
- Vendor capability claims require official documentation or are marked as author inference.
- Scores and mappings are analytical and evidence-qualified.
- This is not a certification program. It does not certify that any system is compliant with any regulation.

12.1 Mapping Method and Evidence Boundary

The mapping method evaluates systems against RCCS-T, RCCS-M, and ALCS dimensions using the evidence hierarchy established in Chapter 1 and the scoring rubrics defined in Chapters 7 and 8. RCCS-T captures traditional regulatory/governance coverage. RCCS-M captures MRO-adjusted regulatory coverage. ALCS captures lifecycle responsibility conformance maturity.

Evidence Hierarchy

The evidence hierarchy used in this chapter is:

- **L1 — Binding Legal / Regulatory / Standards Text:** Legal, regulatory, or formal standards baseline. L1 does not prove product implementation.
- **L2 — Official Product, Protocol, API, or Repository Documentation:** Official documentation, API docs, developer documentation, protocol specifications, platform architecture docs, or official repository documentation.
- **L3 — Independent Audit / Third-Party Evaluation:** Third-party audit reports, credible research papers, independent reports, or assurance evidence.
- **L4 — Vendor Claim / Marketing Page / Announcement:** Vendor white papers, product pages, blogs, announcements, or positioning material.
- **L5 — Author Inference / Analytical Interpretation:** Author inference, conceptual analysis, or mapping judgment. Explicitly disclosed.

The canonical evidence multipliers are L1=1.00, L2=0.85, L3=0.75, L4=0.55, and L5=0.35. Protocol specifications are treated as L2 unless they are binding legal or standards text. Mapping a product or protocol feature to RCCS-T, RCCS-M, or ALCS remains an analytical interpretation unless the source directly defines the lifecycle governance object.

Evidence Limitation

Public documentation may be incomplete. Absence of evidence is not evidence of absence. A system may provide a capability that is not documented publicly. A system may provide a capability that is documented but not yet validated by independent audit. A system may provide a capability that is in development but not yet released.

This chapter evaluates public evidence available as of May 2026. Vendor capability claims are source-qualified or marked as author inference. Numerical score ranges, where presented in the Evaluation Results section, are

provisional analytical profiles with confidence labels and are not rankings, procurement recommendations, certification signals, or legal compliance judgments.

Mapping Strength Scale

Instead of numeric scores for every dimension in this phase, the chapter uses qualitative mapping strength:

- **Strong:** Public documentation shows clear primitives, API support, or protocol semantics for the dimension.
- **Moderate:** Public documentation suggests capability exists, but coverage is partial or requires validation.
- **Partial:** Some evidence exists, but significant gaps or limitations are evident.
- **Weak:** Minimal evidence or capability is conceptual only.
- **Not Evaluated:** Insufficient public evidence to evaluate, or outside system scope.

This chapter uses qualitative mapping strength for dimension-level mapping. Numerical results, where presented in the Evaluation Results section, are published only as provisional analytical score ranges with source confidence labels and explicit non-ranking boundaries.

Evidence Basis Table

The following table defines the evidence scale used in system mappings.

Table T-12-01: System Mapping Evidence Scale

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

L1	<table><tr><td>Source Type</td><td>Binding legal / regulatory / standards text</td></tr><tr><td>Use in Mapping</td><td>Establishes baseline obligations or framework requirements</td></tr><tr><td>Limitation</td><td>Does not prove system implementation</td></tr></table>	Source Type	Binding legal / regulatory / standards text	Use in Mapping	Establishes baseline obligations or framework requirements	Limitation	Does not prove system implementation
Source Type	Binding legal / regulatory / standards text						
Use in Mapping	Establishes baseline obligations or framework requirements						
Limitation	Does not prove system implementation						
L2	<table><tr><td>Source Type</td><td>Official product, protocol, API, developer, platform, or repository documentation</td></tr><tr><td>Use in Mapping</td><td>Supports documented product/protocol capability claims</td></tr><tr><td>Limitation</td><td>Does not prove deployment practice, audit success, or legal compliance</td></tr></table>	Source Type	Official product, protocol, API, developer, platform, or repository documentation	Use in Mapping	Supports documented product/protocol capability claims	Limitation	Does not prove deployment practice, audit success, or legal compliance
Source Type	Official product, protocol, API, developer, platform, or repository documentation						
Use in Mapping	Supports documented product/protocol capability claims						
Limitation	Does not prove deployment practice, audit success, or legal compliance						
L3	<table><tr><td>Source Type</td><td>Independent audit report, credible research paper, independent report</td></tr><tr><td>Use in Mapping</td><td>Supports independently reviewed implementation or methodology claims</td></tr><tr><td>Limitation</td><td>May lag product updates or have limited scope</td></tr></table>	Source Type	Independent audit report, credible research paper, independent report	Use in Mapping	Supports independently reviewed implementation or methodology claims	Limitation	May lag product updates or have limited scope
Source Type	Independent audit report, credible research paper, independent report						
Use in Mapping	Supports independently reviewed implementation or methodology claims						
Limitation	May lag product updates or have limited scope						

L4	
Source Type	Vendor claim, marketing page, blog, announcement
Use in Mapping	Supports positioning claims only
Limitation	May overstate implementation detail

L5	
Source Type	Author inference, conceptual analysis, mapping judgment
Use in Mapping	Supports disclosed analytical interpretation
Limitation	Must not be presented as verified product fact

12.2 How to Read System Mapping Tables

Each system section in this chapter uses the same structure:

System Positioning

Briefly states:

- What the system primarily is
- What it is strong at
- Why it is relevant to RCCS-T / RCCS-M / ALCS
- What this paper does and does not infer

Evidence Basis

Uses citation placeholders and evidence levels. Includes a mini-table:

Evidence Area	Source Type	Evidence Level	Claim-Level Status
[Area]	[Type]	[L1 - L5]	[source-bound capability surface / narrowed claim / L5 analytical interpretation / explicit non-blocking limitation]

RCCS-T / RCCS-M Mapping Summary

Compact table:

RCCS Layer / Dimension	Mapping Strength	Evidence Basis	Notes
[Dimension]	[Strong/Moderate/Partial/Weak/Not Evaluated]	[L1 - L5]	[Notes]

ALCS Mapping Summary

Compact table grouping ALCS dimensions into five areas:

ALCS Area	Mapping Strength	Evidence Basis	Notes
Role / Authority / Acceptance (ALCS-01 to ALCS-04)	[Strength]	[L1 - L5]	[Notes]
Tool Action / Transfer / Drift / Evidence (ALCS-05 to ALCS-08)	[Strength]	[L1 - L5]	[Notes]
Reuse / Privacy / Validation (ALCS-09 to ALCS-11)	[Strength]	[L1 - L5]	[Notes]
Evidence Minimization / Data Subject Rights / Processor Chain (ALCS-12 to ALCS-14)	[Strength]	[L1 - L5]	[Notes]
Substitution Conformance (ALCS-15)	[Strength]	[L1 - L5]	[Notes]

Main Lifecycle Gaps

Lists 3-5 specific gaps or revalidation needs.

Boundary Note

Each system section closes with:

This section maps public evidence to RCCS-T/RCCS-M/ALCS dimensions. It is not a legal compliance judgment, procurement recommendation, certification, or endorsement.

12.3 MPLP — Lifecycle Protocol Path

System Positioning

MPLP (Multi-Agent Lifecycle Protocol) is a lifecycle protocol path designed to expose lifecycle responsibility objects as first-class protocol primitives. It is relevant to RCCS-T, RCCS-M, and ALCS because its public protocol semantics can be analytically mapped to human-role-to-MAS mapping, delegated authority boundaries, accepted outcome records, responsibility transfer, evidence partitioning, privacy lifecycle mapping, and vendor substitution conformance.

MPLP is strong at lifecycle protocol semantics. It defines governance primitives and lifecycle objects as protocol-level constructs rather than application-level features. This means MPLP's traditional RCCS-T product-platform coverage may be limited, while its RCCS-M and ALCS posture may be strong where protocol semantics map to MRO-like objects. The GAIC MRO/RCCS-M/ALCS mapping remains an author analytical mapping based on those protocol semantics, not an MPLP official compliance claim.

What this paper evaluates:

- Whether MPLP protocol specification defines governance primitives and lifecycle objects
- Whether MPLP semantics map to RCCS-T, RCCS-M, and ALCS dimensions
- Whether MPLP provides protocol-level semantics that can support lifecycle responsibility analysis

What this paper does not infer:

- MPLP is not required for compliance. It is one example of a lifecycle protocol path.
- MPLP is not a certification program. It does not certify systems or organizations.

- MPLP is not a regulator or legal authority. It is a protocol specification.
- MPLP implementation maturity, enterprise integration, and vendor adoption require separate evaluation.
- Validation Lab is a non-certifying evidence adjudication example. Detailed treatment belongs to Chapter 14.

Evidence Basis

Evidence Area	Source Type	Evidence Level	Claim-Level Status
Protocol specification	Protocol docs	L2	Claim-level reviewed; mapping remains L5 analytical interpretation where the protocol does not directly prove implementation
Lifecycle object semantics	Protocol docs	L2 + L5	Claim-level reviewed; semantic fit remains analytical interpretation
Governance primitives	Protocol docs	L2 + L5	Claim-level reviewed; implementation maturity remains unresolved
Implementation examples	Developer docs	L2	Source-bound at entry-point level; deployment maturity remains unresolved
Validation Lab methodology	Methodology / lab docs	L2 + L5	Claim-level support remains boundary-limited; defer to Chapter 14

Source references:

- [MPLP-DOCS]
- [MPLP-SPEC]

RCCS-T / RCCS-M Mapping Summary

Table T-12-02: MPLP RCCS-T / RCCS-M Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

RCCS-T overall product-platform coverage	
Mapping Strength	Partial to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	MPLP is not an enterprise governance SaaS and does not by itself provide full productized governance dashboards, cloud controls, or organizational policy workflows

RCCS-M overall MRO-adjusted coverage	
Mapping Strength	Strong
Evidence Basis	L2 + L5 reviewed
Notes	Protocol semantics map strongly to authority, evidence, accepted-outcome, responsibility-transfer, and substitution objects; mapping remains author analytical where not directly stated

Risk Management

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Protocol may define risk boundaries; implementation maturity requires validation

Data Governance

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Protocol may define data lineage; implementation requires validation

Documentation

Mapping Strength	Strong
Evidence Basis	L2 + L5 reviewed
Notes	Protocol specification provides documentation primitives

Record-Keeping

Mapping Strength	Strong
Evidence Basis	L2 + L5 reviewed
Notes	Evidence chain is core protocol primitive

Transparency

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Protocol may define transparency primitives; implementation requires validation

Human Oversight

Mapping Strength	Strong
Evidence Basis	L2 + L5 reviewed
Notes	Delegated authority and accepted outcome are core protocol primitives

Security

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Protocol may define security boundaries; implementation requires validation

Accountability

Mapping Strength	Strong
Evidence Basis	L2 + L5 reviewed
Notes	Responsibility transfer and evidence partitioning are core protocol primitives

Contestability

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Protocol may define dispute primitives; implementation requires validation

Monitoring

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Protocol may define monitoring primitives; implementation requires validation

ALCS Mapping Summary

Table T-12-03: MPLP ALCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Role / Authority / Acceptance (ALCS-01 to ALCS-04)

Mapping Strength	Strong
Evidence Basis	L2 + L5 reviewed
Notes	Protocol designed for human-role-to-MAS mapping, delegated authority boundary, agent role distinction, accepted outcome compliance

Tool Action / Transfer / Drift / Evidence (ALCS-05 to ALCS-08)

Mapping Strength	Strong
Evidence Basis	L2 + L5 reviewed
Notes	Protocol designed for tool-action liability, responsibility transfer, authority drift detection, MAS evidence partitioning

Reuse / Privacy / Validation (ALCS-09 to ALCS-11)

Mapping Strength	Moderate to Strong
Evidence Basis	L2 + L5 reviewed
Notes	Protocol may define cross-project reuse, privacy lifecycle mapping, privacy-preserving validation; implementation requires validation

Evidence Minimization / Data Subject Rights / Processor Chain (ALCS-12 to ALCS-14)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Protocol may define evidence minimization, data subject rights reconciliation, processor chain tracking; implementation requires validation

Substitution Conformance (ALCS-15)

Mapping Strength	Strong
Evidence Basis	L2 + L5 reviewed
Notes	Protocol designed for vendor/model/runtime substitution conformance

Main Lifecycle Gaps

- Implementation Maturity:** MPLP protocol specification may be strong, but implementation maturity in production systems requires validation.
- Enterprise Integration:** MPLP integration with existing enterprise AI governance platforms, cloud AI platforms, and agent orchestration frameworks requires validation.
- Vendor Adoption:** MPLP adoption by vendors, platforms, and frameworks requires validation.
- Operational Deployment Evidence:** MPLP deployment in production agentic systems requires validation.
- Regulator / Auditor Familiarity:** MPLP familiarity among regulators, auditors, and compliance teams requires validation.

Boundary Note

This section maps public evidence to RCCS-T/RCCS-M/ALCS dimensions. It is not a legal compliance judgment, procurement recommendation, certification, or endorsement. MPLP is one example of a lifecycle protocol path. It is not required for compliance. It is not a certification program. It is not a regulator or legal authority.

12.4 IBM watsonx.governance — Enterprise AI Governance Platform

System Positioning

IBM watsonx.governance is an enterprise AI governance platform designed for policy enforcement, model risk management, regulatory compliance coverage, and AI lifecycle management. It is relevant to RCCS-T, RCCS-M, and ALCS because it provides enterprise-grade governance capabilities for risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring, while lifecycle responsibility-object coverage remains analytically tested.

IBM watsonx.governance is strong at enterprise AI governance. It provides policy enforcement, model governance, risk management workflows, compliance tracking, and enterprise integration capabilities. This makes IBM watsonx.governance relevant for organizations that require regulatory compliance coverage across the AI lifecycle.

What this paper evaluates:

- Whether IBM watsonx.governance provides governance primitives for the RCCS-T/RCCS-M shared dimension family
- Whether IBM watsonx.governance provides lifecycle responsibility objects for ALCS dimensions
- Whether public documentation supports RCCS-T, RCCS-M, and ALCS mapping

What this paper does not infer:

- IBM watsonx.governance is not evaluated for product quality, market leadership, or procurement suitability.
- ALCS posture requires validation: lifecycle objects for multi-agent responsibility transfer, accepted outcome, tool-action liability, and MAS evidence partitioning require evidence validation.
- Model governance and AI governance are not identical to agentic lifecycle governance.

Evidence Basis

Evidence Area	Source Type	Evidence Level	Claim-Level Status
AI governance platform	Official product docs	L2	source-bound at entry-point level
Policy enforcement	Official product docs	L2	source-bound at entry-point level
Model risk management	Official product docs	L2	source-bound at entry-point level
Compliance workflows	Official product docs	L2	narrowed / partially validated
Agentic lifecycle objects	Official product docs	L2 (if available) or L5 (author inference)	L5 analytical interpretation unless directly stated

Source references:

- [IBM-WATSONX-GOVERNANCE-DOCS]

RCCS Mapping Summary

Table T-12-04: IBM watsonx.governance RCCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Enterprise AI governance platform designed for risk management

Data Governance

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Platform provides data lineage and governance capabilities

Documentation

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Platform provides documentation and policy management

Record-Keeping

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Platform provides audit trails and compliance records

Transparency

Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Platform provides transparency and explainability features

Human Oversight

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Platform provides human review and approval workflows

Security

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Enterprise platform with security and access controls

Accountability

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Platform provides accountability and responsibility tracking

Contestability

Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Platform may provide dispute and appeal mechanisms; requires validation

Monitoring

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Platform provides continuous monitoring and incident detection

ALCS Mapping Summary

Table T-12-05: IBM watsonx.governance ALCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Role / Authority / Acceptance (ALCS-01 to ALCS-04)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Human-role-to-MAS mapping, delegated authority boundary, agent role distinction, accepted outcome compliance require validation

Tool Action / Transfer / Drift / Evidence (ALCS-05 to ALCS-08)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Tool-action liability, responsibility transfer, authority drift detection, MAS evidence partitioning require validation

Reuse / Privacy / Validation (ALCS-09 to ALCS-11)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Cross-project reuse, privacy lifecycle mapping, privacy-preserving validation require validation

Evidence Minimization / Data Subject Rights / Processor Chain (ALCS-12 to ALCS-14)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Evidence minimization, data subject rights reconciliation, processor chain tracking require validation

Substitution Conformance (ALCS-15)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Vendor/model/runtime substitution conformance requires validation

Main Lifecycle Gaps

1. **Multi-Agent Responsibility Transfer:** Whether IBM watsonx.governance tracks responsibility transfer across agents in multi-agent workflows requires validation.
2. **Accepted Outcome Governance:** Whether IBM watsonx.governance separates execution completion from outcome acceptance requires validation.
3. **Tool-Action Liability Boundary:** Whether IBM watsonx.governance records liability boundaries for consequential tool actions requires validation.
4. **MAS Evidence Partitioning:** Whether IBM watsonx.governance partitions evidence by agent, tool, authority, and privacy boundaries requires validation.
5. **Cross-Project Reuse Controls:** Whether IBM watsonx.governance enforces revalidation when workflows are reused across projects requires validation.

Boundary Note

This section maps public evidence to RCCS-T/RCCS-M/ALCS dimensions. It is not a legal compliance judgment, procurement recommendation, certification, or endorsement. IBM watsonx.governance is evaluated for governance primitives and lifecycle responsibility objects based on public documentation. Model governance and AI governance are not identical to agentic lifecycle governance.

12.5 Microsoft Azure AI Foundry — Cloud AI Platform and Enterprise AI Tooling

System Positioning

Microsoft Azure AI Foundry is a cloud-native AI platform designed for model management, responsible AI tooling, content safety, tracing, evaluation, and enterprise integration. It is relevant to RCCS-T, RCCS-M, and ALCS because it provides cloud platform capabilities for risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, and monitoring, while MRO-adjusted lifecycle coverage remains implementation-dependent.

Microsoft Azure AI Foundry is strong at cloud AI platform capabilities. It provides model access, responsible AI tooling, content safety, tracing, observability, evaluation, and enterprise integration. This makes Microsoft Azure AI Foundry relevant for organizations that require cloud-native AI platform capabilities with responsible AI tooling.

What this paper evaluates:

- Whether Microsoft Azure AI Foundry provides governance primitives for the RCCS-T/RCCS-M shared dimension family
- Whether Microsoft Azure AI Foundry provides lifecycle responsibility objects for ALCS dimensions
- Whether public documentation supports RCCS-T, RCCS-M, and ALCS mapping

What this paper does not infer:

- Microsoft Azure AI Foundry is not evaluated for product quality, market leadership, or procurement suitability.
- ALCS posture requires validation: authority boundaries, accepted outcomes, MAS evidence partitioning, and responsibility transfer require evidence validation.
- Responsible AI tooling and cloud platform features are not identical to lifecycle responsibility objects.

Evidence Basis

Evidence Area	Source Type	Evidence Level	Claim-Level Status
Cloud AI platform	Official product docs	L2	source-bound at entry-point level
Responsible AI tooling	Official product docs	L2	source-bound at entry-point level
Content safety	Official product docs	L2	source-bound at entry-point level
Tracing and observability	Official product docs	L2	narrowed / partially validated
Agentic lifecycle objects	Official product docs	L2 (if available) or L5 (author inference)	L5 analytical interpretation unless directly stated

Source references:

- [MICROSOFT-AZURE-AI-FOUNDRY-DOCS]
- [MICROSOFT-RESPONSIBLE-AI-DOCS]

RCCS Mapping Summary

Table T-12-06: Microsoft Azure AI Foundry RCCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management	
Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Responsible AI tooling provides risk management capabilities
Data Governance	
Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Cloud platform provides data governance and lineage capabilities
Documentation	
Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Platform provides documentation and model cards
Record-Keeping	
Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Platform provides tracing and audit trails
Transparency	
Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Responsible AI tooling provides transparency and explainability
Human Oversight	
Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Platform provides human review and approval capabilities

Security

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Cloud platform with enterprise security and access controls

Accountability

Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Platform provides accountability features; agentic accountability requires validation

Contestability

Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Platform may provide dispute mechanisms; requires validation

Monitoring

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Platform provides continuous monitoring and observability

ALCS Mapping Summary

Table T-12-07: Microsoft Azure AI Foundry ALCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Role / Authority / Acceptance (ALCS-01 to ALCS-04)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Human-role-to-MAS mapping, delegated authority boundary, agent role distinction, accepted outcome compliance require validation

Tool Action / Transfer / Drift / Evidence (ALCS-05 to ALCS-08)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Tool-action liability, responsibility transfer, authority drift detection, MAS evidence partitioning require validation

Reuse / Privacy / Validation (ALCS-09 to ALCS-11)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Cross-project reuse, privacy lifecycle mapping, privacy-preserving validation require validation

Evidence Minimization / Data Subject Rights / Processor Chain (ALCS-12 to ALCS-14)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Evidence minimization, data subject rights reconciliation, processor chain tracking require validation

Substitution Conformance (ALCS-15)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Vendor/model/runtime substitution conformance requires validation

Main Lifecycle Gaps

- Accepted Outcome Compliance:** Whether Microsoft Azure AI Foundry separates execution completion from outcome acceptance requires validation.
- Delegated Authority Boundary:** Whether Microsoft Azure AI Foundry defines and enforces authority boundaries for agent actions requires validation.
- Cross-Agent Responsibility Transfer:** Whether Microsoft Azure AI Foundry tracks responsibility transfer during agent handoffs requires validation.
- Evidence Partitioning:** Whether Microsoft Azure AI Foundry partitions evidence by agent, tool, authority, and privacy boundaries requires validation.
- Vendor/Model/Runtime Substitution Conformance:** Whether Microsoft Azure AI Foundry enforces conformance revalidation when vendors are substituted requires validation.

Boundary Note

This section maps public evidence to RCCS-T/RCCS-M/ALCS dimensions. It is not a legal compliance judgment, procurement recommendation, certification, or endorsement. Microsoft Azure AI Foundry is evaluated for governance primitives and lifecycle responsibility objects based on public documentation. Responsible AI tooling and cloud platform features are not identical to lifecycle responsibility objects.

12.6 AWS Bedrock / Guardrails / AgentCore — Cloud AI Service Suite

System Positioning

AWS Bedrock / Guardrails / AgentCore is a cloud AI service suite designed for model access, guardrails, agent orchestration, and enterprise security integration. It is relevant to RCCS-T, RCCS-M, and ALCS because it provides cloud service capabilities for guardrails, security, access control, monitoring, and agent orchestration, while lifecycle responsibility-object coverage remains source-qualified and analytical.

AWS Bedrock / Guardrails / AgentCore is strong at cloud AI services. It provides model access, guardrails, agent orchestration, tool use, and enterprise security integration. This makes AWS Bedrock / Guardrails / AgentCore relevant for organizations that require cloud-native AI services with guardrails and agent orchestration.

What this paper evaluates:

- Whether AWS Bedrock / Guardrails / AgentCore provides governance primitives for the RCCS-T/RCCS-M shared dimension family
- Whether AWS Bedrock / Guardrails / AgentCore provides lifecycle responsibility objects for ALCS dimensions
- Whether public documentation supports RCCS-T, RCCS-M, and ALCS mapping

What this paper does not infer:

- AWS Bedrock / Guardrails / AgentCore is not evaluated for product quality, market leadership, or procurement suitability.
- ALCS posture requires validation: orchestration and guardrails do not automatically prove lifecycle responsibility semantics.
- Guardrails and tool permission are not identical to delegated authority boundary.

Evidence Basis

Evidence Area	Source Type	Evidence Level	Claim-Level Status
Cloud AI services	Official product docs	L2	source-bound at entry-point level
Guardrails	Official product docs	L2	source-bound at entry-point level
Agent orchestration	Official product docs	L2	narrowed / partially validated
Enterprise security	Official product docs	L2	source-bound at entry-point level
Agentic lifecycle objects	Official product docs	L2 (if available) or L5 (author inference)	L5 analytical interpretation unless directly stated

Source references:

- [AWS-BEDROCK-DOCS]
- [AWS-BEDROCK-GUARDRAILS-DOCS]
- [AWS-AGENTCORE-DOCS]

RCCS Mapping Summary

Table T-12-08: AWS Bedrock / Guardrails / AgentCore RCCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management	
Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Guardrails provide risk mitigation; broader risk management requires validation
Data Governance	
Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Cloud platform provides data governance capabilities; agentic data governance requires validation
Documentation	
Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Platform provides documentation capabilities; agentic documentation requires validation
Record-Keeping	
Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Platform provides logging and audit trails; agentic record-keeping requires validation
Transparency	
Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Platform may provide transparency features; requires validation
Human Oversight	
Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Platform may provide human oversight capabilities; requires validation

Security

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Cloud platform with enterprise security and access controls

Accountability

Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Platform provides accountability features; agentic accountability requires validation

Contestability

Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Platform may provide dispute mechanisms; requires validation

Monitoring

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Platform provides continuous monitoring and observability

ALCS Mapping Summary

Table T-12-09: AWS Bedrock / Guardrails / AgentCore ALCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Role / Authority / Acceptance (ALCS-01 to ALCS-04)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Human-role-to-MAS mapping, delegated authority boundary, agent role distinction, accepted outcome compliance require validation; guardrails ≠ delegated authority boundary

Tool Action / Transfer / Drift / Evidence (ALCS-05 to ALCS-08)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Tool-action liability, responsibility transfer, authority drift detection, MAS evidence partitioning require validation

Reuse / Privacy / Validation (ALCS-09 to ALCS-11)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Cross-project reuse, privacy lifecycle mapping, privacy-preserving validation require validation

Evidence Minimization / Data Subject Rights / Processor Chain (ALCS-12 to ALCS-14)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Evidence minimization, data subject rights reconciliation, processor chain tracking require validation

Substitution Conformance (ALCS-15)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Vendor/model/runtime substitution conformance requires validation

Main Lifecycle Gaps

- Delegated Authority vs IAM/Tool Permission:** Whether AWS Bedrock / Guardrails / AgentCore distinguishes delegated authority boundary from IAM and tool permission requires validation.
- Accepted Outcome Governance:** Whether AWS Bedrock / Guardrails / AgentCore separates execution completion from outcome acceptance requires validation.
- Tool-Action Liability Boundary:** Whether AWS Bedrock / Guardrails / AgentCore records liability boundaries for consequential tool actions requires validation.
- MAS Evidence Partitioning:** Whether AWS Bedrock / Guardrails / AgentCore partitions evidence by agent, tool, authority, and privacy boundaries requires validation.
- Incident/Dispute/Remediation Closure:** Whether AWS Bedrock / Guardrails / AgentCore provides incident, dispute, and remediation closure mechanisms requires validation.

Boundary Note

This section maps public evidence to RCCS-T/RCCS-M/ALCS dimensions. It is not a legal compliance judgment, procurement recommendation, certification, or endorsement. AWS Bedrock / Guardrails / AgentCore is evaluated for governance primitives and lifecycle responsibility objects based on public documentation. Guardrails and tool permission are not identical to delegated authority boundary.

12.7 Google Vertex AI / ADK / Model Armor — Cloud AI Platform, Agent Development, and Safety Controls

System Positioning

Google Vertex AI / ADK / Model Armor is a cloud AI platform suite designed for model management, agent development tooling, and model/content safety controls. It is relevant to RCCS-T, RCCS-M, and ALCS because it provides cloud platform capabilities for model governance, agent development, safety evaluation, content filtering, and enterprise integration, while MRO-adjusted lifecycle coverage remains evidence-qualified.

Google Vertex AI / ADK / Model Armor is strong at cloud AI platform capabilities, agent development tooling, and model safety controls. It provides model access, agent development kit (ADK), model armor safety features, evaluation frameworks, and enterprise security integration. This makes Google Vertex AI / ADK / Model Armor relevant for organizations that require cloud-native AI platform capabilities with agent development and safety tooling.

What this paper evaluates:

- Whether Google Vertex AI / ADK / Model Armor provides governance primitives for the RCCS-T/RCCS-M shared dimension family
- Whether Google Vertex AI / ADK / Model Armor provides lifecycle responsibility objects for ALCS dimensions
- Whether public documentation supports RCCS-T, RCCS-M, and ALCS mapping

What this paper does not infer:

- Google Vertex AI / ADK / Model Armor is not evaluated for product quality, market leadership, or procurement suitability.
- ALCS posture requires validation: agent development and model safety controls do not automatically prove lifecycle responsibility semantics.
- Model safety, content safety, and platform governance are not identical to lifecycle responsibility objects.
- ADK agent development primitives are not identical to human-role-to-MAS responsibility mapping.

Evidence Basis

Evidence Area	Source Type	Evidence Level	Claim-Level Status
Cloud AI platform	Official product docs	L2	source-bound at entry-point level
Agent Development Kit (ADK)	Official product docs	L2	source-bound at entry-point level
Model Armor safety controls	Official product docs	L2	source-bound at entry-point level
Model evaluation and monitoring	Official product docs	L2	narrowed / partially validated
Agentic lifecycle objects	Official product docs	L2 (if available) or L5 (author inference)	L5 analytical interpretation unless directly stated

Source references:

- [GOOGLE-VERTEX-AI-DOCS]
- [GOOGLE-ADK-DOCS]
- [GOOGLE-MODEL-ARMOR-DOCS]

RCCS Mapping Summary

Table T-12-10: Google Vertex AI / ADK / Model Armor RCCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management	
Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Model Armor and evaluation frameworks provide risk management capabilities

Data Governance	
Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Cloud platform provides data governance and lineage capabilities

Documentation	
Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Platform provides documentation and model cards

Record-Keeping	
Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Platform provides logging and audit trails

Transparency	
Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Platform provides transparency and explainability features

Human Oversight	
Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Platform may provide human review capabilities; requires validation

Security

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Cloud platform with enterprise security and access controls

Accountability

Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Platform provides accountability features; agentic accountability requires validation

Contestability

Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Platform may provide dispute mechanisms; requires validation

Monitoring

Mapping Strength	Strong
Evidence Basis	L2 reviewed
Notes	Platform provides continuous monitoring and observability

ALCS Mapping Summary

Table T-12-11: Google Vertex AI / ADK / Model Armor ALCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Role / Authority / Acceptance (ALCS-01 to ALCS-04)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Human-role-to-MAS mapping, delegated authority boundary, agent role distinction, accepted outcome compliance require validation; ADK agent primitives ≠ lifecycle responsibility semantics

Tool Action / Transfer / Drift / Evidence (ALCS-05 to ALCS-08)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Tool-action liability, responsibility transfer, authority drift detection, MAS evidence partitioning require validation

Reuse / Privacy / Validation (ALCS-09 to ALCS-11)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Cross-project reuse, privacy lifecycle mapping, privacy-preserving validation require validation

Evidence Minimization / Data Subject Rights / Processor Chain (ALCS-12 to ALCS-14)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Evidence minimization, data subject rights reconciliation, processor chain tracking require validation

Substitution Conformance (ALCS-15)

Mapping Strength	Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Vendor/model/runtime substitution conformance requires validation

Main Lifecycle Gaps

- Agent Development vs Lifecycle Responsibility:** Whether Google ADK agent development primitives map to human-role-to-MAS responsibility semantics requires validation.
- Delegated Authority Boundary:** Whether Google Vertex AI / ADK / Model Armor defines and enforces authority boundaries for agent actions requires validation.
- Accepted Outcome Compliance:** Whether Google Vertex AI / ADK / Model Armor separates execution completion from outcome acceptance requires validation.
- Model Safety vs Lifecycle Governance:** Whether Model Armor safety controls extend to lifecycle responsibility governance requires validation.
- Cross-Agent Responsibility Transfer:** Whether Google Vertex AI / ADK / Model Armor tracks responsibility transfer during agent handoffs requires validation.

Boundary Note

This section maps public evidence to RCCS-T/RCCS-M/ALCS dimensions. It is not a legal compliance judgment, procurement recommendation, certification, or endorsement. Google Vertex AI / ADK / Model Armor is evaluated for governance primitives and lifecycle responsibility objects based on public documentation. Model safety, content safety, and platform governance are not identical to lifecycle responsibility objects.

12.8 LangGraph / LangSmith — Agent Orchestration and Observability Framework

System Positioning

LangGraph / LangSmith is an agent orchestration and observability framework designed for workflow graphs, state management, checkpoints, traces, and observability. It is relevant to RCCS-T, RCCS-M, and ALCS because it provides orchestration and observability capabilities that can support governance, monitoring, transparency, evidence collection, and MRO-adjacent lifecycle analysis.

LangGraph / LangSmith is strong at agent orchestration and observability. It provides workflow graphs, state management, checkpoints, traces, debugging, evaluation, and observability features. This makes LangGraph / LangSmith relevant for organizations that require agent orchestration with observability and debugging capabilities.

What this paper evaluates:

- Whether LangGraph / LangSmith provides governance primitives for the RCCS-T/RCCS-M shared dimension family
- Whether LangGraph / LangSmith provides lifecycle responsibility objects for ALCS dimensions
- Whether public documentation supports RCCS-T, RCCS-M, and ALCS mapping

What this paper does not infer:

- LangGraph / LangSmith is not evaluated for product quality, market leadership, or procurement suitability.
- ALCS posture requires validation: orchestration and observability can support governance but do not automatically provide regulatory compliance coverage.
- Workflow graphs, state, checkpoints, and traces are not identical to lifecycle responsibility objects.
- Observability traces are not identical to MAS evidence partitioning.

Evidence Basis

Evidence Area	Source Type	Evidence Level	Claim-Level Status
Agent orchestration	Official product docs	L2	source-bound at entry-point level
Workflow graphs and state	Official product docs	L2	source-bound at entry-point level
Checkpoints and persistence	Official product docs	L2	source-bound at entry-point level
Traces and observability	Official product docs	L2	narrowed / partially validated
Agentic lifecycle objects	Official product docs	L2 (if available) or L5 (author inference)	L5 analytical interpretation unless directly stated

Source references:

- [LANGGRAPH-DOCS]
- [LANGSMITH-DOCS]

RCCS Mapping Summary

Table T-12-12: LangGraph / LangSmith RCCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management	
Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; risk management requires additional governance layer

Data Governance	
Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	State management exists; data governance requires additional governance layer

Documentation	
Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Workflow graphs and traces provide documentation primitives

Record-Keeping	
Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Checkpoints and traces provide record-keeping primitives

Transparency	
Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Traces and observability provide transparency primitives

Human Oversight	
Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; human oversight requires additional governance layer

Security

Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	Framework security depends on deployment environment

Accountability

Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Traces provide observability; accountability requires additional governance layer

Contestability

Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; contestability requires additional governance layer

Monitoring

Mapping Strength	Moderate to Strong
Evidence Basis	L2 reviewed
Notes	Observability and traces provide monitoring primitives

ALCS Mapping Summary

Table T-12-13: LangGraph / LangSmith ALCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Role / Authority / Acceptance (ALCS-01 to ALCS-04)

Mapping Strength	Partial to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Workflow graphs and state provide useful primitives; human-role-to-MAS mapping, delegated authority boundary, agent role distinction, accepted outcome compliance require additional governance semantics

Tool Action / Transfer / Drift / Evidence (ALCS-05 to ALCS-08)

Mapping Strength	Partial to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Traces and state transitions provide useful primitives; tool-action liability, responsibility transfer, authority drift detection, MAS evidence partitioning require additional governance semantics

Reuse / Privacy / Validation (ALCS-09 to ALCS-11)

Mapping Strength	Partial
Evidence Basis	L2 + L5 reviewed
Notes	Workflow reuse exists; cross-project reuse compliance, privacy lifecycle mapping, privacy-preserving validation require additional governance semantics

Evidence Minimization / Data Subject Rights / Processor Chain (ALCS-12 to ALCS-14)

Mapping Strength	Weak to Partial
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; evidence minimization, data subject rights reconciliation, processor chain tracking require additional governance layer

Substitution Conformance (ALCS-15)

Mapping Strength	Weak to Partial
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; vendor/model/runtime substitution conformance requires additional governance layer

Main Lifecycle Gaps

1. **Orchestration vs Governance:** Whether LangGraph / LangSmith orchestration and observability primitives extend to lifecycle responsibility governance requires validation.
2. **Traces vs Evidence Partitioning:** Whether LangSmith traces map to MAS evidence partitioning with agent, tool, authority, and privacy boundaries requires validation.
3. **Human-Role-to-MAS Responsibility Mapping:** Whether LangGraph / LangSmith maps human roles to MAS responsibility semantics requires validation.
4. **Delegated Authority Boundary:** Whether LangGraph / LangSmith defines and enforces authority boundaries for agent actions requires validation.
5. **Accepted Outcome Compliance:** Whether LangGraph / LangSmith separates execution completion from outcome acceptance requires validation.
6. **Incident/Dispute/Remediation Closure:** Whether LangGraph / LangSmith provides incident, dispute, and remediation closure mechanisms requires validation.

Boundary Note

This section maps public evidence to RCCS-T/RCCS-M/ALCS dimensions. It is not a legal compliance judgment, procurement recommendation, certification, or endorsement. LangGraph / LangSmith is evaluated for governance primitives and lifecycle responsibility objects based on public documentation. Orchestration and observability can support governance but do not automatically provide regulatory compliance coverage.

12.9 OpenAI Agents SDK — Developer-Focused Agent Execution Framework

System Positioning

OpenAI Agents SDK is a developer-focused agent execution framework designed for model access, tool use, agent execution patterns, and developer primitives. It is relevant to RCCS-T, RCCS-M, and ALCS because it provides execution capabilities that can support agent workflows, tool actions, observability, and MRO-adjacent lifecycle analysis, while enterprise governance coverage remains outside SDK primitives.

OpenAI Agents SDK is strong at agent execution and tool use. It provides model access, tool calling, agent execution patterns, streaming, and developer-friendly APIs. This makes OpenAI Agents SDK relevant for developers building agentic applications with tool use and execution capabilities.

What this paper evaluates:

- Whether OpenAI Agents SDK provides governance primitives for the RCCS-T/RCCS-M shared dimension family
- Whether OpenAI Agents SDK provides lifecycle responsibility objects for ALCS dimensions
- Whether public documentation supports RCCS-T, RCCS-M, and ALCS mapping

What this paper does not infer:

- OpenAI Agents SDK is not evaluated for product quality, market leadership, or procurement suitability.
- RCCS-T, RCCS-M, and ALCS posture requires validation: execution framework features do not automatically provide enterprise governance, audit readiness, accepted outcome, or lifecycle responsibility semantics.
- Tool calling and agent execution are not identical to tool-action liability boundary.
- Developer primitives are not identical to enterprise lifecycle governance objects.

Evidence Basis

Evidence Area	Source Type	Evidence Level	Claim-Level Status
Agent execution framework	Official product docs	L2	source-bound at entry-point level
Tool calling and tool use	Official product docs	L2	narrowed / partially validated
Model access and APIs	Official product docs	L2	source-bound at entry-point level
Developer primitives	Official product docs	L2	narrowed / partially validated
Agentic lifecycle objects	Official product docs	L2 (if available) or L5 (author inference)	L5 analytical interpretation unless directly stated

Source references:

- [OPENAI-AGENTS-SDK-DOCS]
- [OPENAI-TOOLS-DOCS]

RCCS Mapping Summary

Table T-12-14: OpenAI Agents SDK RCCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management	
Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Execution framework; risk management requires additional governance layer

Data Governance	
Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Execution framework; data governance requires additional governance layer

Documentation	
Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Developer docs exist; governance documentation requires additional layer

Record-Keeping	
Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Execution logs may exist; audit-ready record-keeping requires additional layer

Transparency	
Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Execution framework; transparency requires additional governance layer

Human Oversight	
Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Execution framework; human oversight requires additional governance layer

Security

Mapping Strength	Moderate
Evidence Basis	L2 reviewed
Notes	API security exists; enterprise security depends on deployment

Accountability

Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Execution framework; accountability requires additional governance layer

Contestability

Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Execution framework; contestability requires additional governance layer

Monitoring

Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Execution logs may exist; governance monitoring requires additional layer

ALCS Mapping Summary

Table T-12-15: OpenAI Agents SDK ALCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Role / Authority / Acceptance (ALCS-01 to ALCS-04)

Mapping Strength	Weak to Partial
Evidence Basis	L2 + L5 reviewed
Notes	Execution framework provides developer primitives; human-role-to-MAS mapping, delegated authority boundary, agent role distinction, accepted outcome compliance require additional governance semantics

Tool Action / Transfer / Drift / Evidence (ALCS-05 to ALCS-08)

Mapping Strength	Weak to Partial
Evidence Basis	L2 + L5 reviewed
Notes	Tool calling exists; tool-action liability, responsibility transfer, authority drift detection, MAS evidence partitioning require additional governance semantics

Reuse / Privacy / Validation (ALCS-09 to ALCS-11)

Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Execution framework; cross-project reuse compliance, privacy lifecycle mapping, privacy-preserving validation require additional governance layer

Evidence Minimization / Data Subject Rights / Processor Chain (ALCS-12 to ALCS-14)

Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Execution framework; evidence minimization, data subject rights reconciliation, processor chain tracking require additional governance layer

Substitution Conformance (ALCS-15)

Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Execution framework; vendor/model/runtime substitution conformance requires additional governance layer

Main Lifecycle Gaps

- Execution Framework vs Governance Layer:** Whether OpenAI Agents SDK execution primitives extend to lifecycle responsibility governance requires validation.
- Tool Calling vs Tool-Action Liability:** Whether OpenAI Agents SDK tool calling maps to tool-action liability boundary requires validation.
- Developer Primitives vs Enterprise Governance:** Whether OpenAI Agents SDK developer primitives extend to enterprise governance objects requires validation.
- Delegated Authority Boundary:** Whether OpenAI Agents SDK defines and enforces authority boundaries for agent actions requires validation.
- Accepted Outcome Compliance:** Whether OpenAI Agents SDK separates execution completion from outcome acceptance requires validation.
- Human Accountability Mapping:** Whether OpenAI Agents SDK maps agent actions to human accountability requires validation.

Boundary Note

This section maps public evidence to RCCS-T/RCCS-M/ALCS dimensions. It is not a legal compliance judgment, procurement recommendation, certification, or endorsement. OpenAI Agents SDK is evaluated for governance primitives and lifecycle responsibility objects based on public documentation. Execution framework features do not automatically provide enterprise governance, audit readiness, or lifecycle responsibility semantics.

12.10 CrewAI — Multi-Agent Collaboration and Role-Based Orchestration Framework

System Positioning

CrewAI is a multi-agent collaboration and role-based orchestration framework designed for agent role composition, task orchestration, and multi-agent workflows. It is relevant to RCCS-T, RCCS-M, and ALCS because it provides role-based orchestration capabilities that can support agent collaboration, task delegation, workflow management, and MRO-adjacent lifecycle analysis.

CrewAI is strong at role-based agent composition and task orchestration. It provides agent role labels, task assignment, multi-agent collaboration, and workflow orchestration features. This makes CrewAI relevant for developers building multi-agent systems with role-based collaboration patterns.

What this paper evaluates:

- Whether CrewAI provides governance primitives for the RCCS-T/RCCS-M shared dimension family
- Whether CrewAI provides lifecycle responsibility objects for ALCS dimensions
- Whether public documentation supports RCCS-T, RCCS-M, and ALCS mapping

What this paper does not infer:

- CrewAI is not evaluated for product quality, market leadership, or procurement suitability.
- RCCS-T, RCCS-M, and ALCS posture requires validation: agent role labels and task orchestration do not automatically provide governance responsibility semantics.
- Agent role labels are not identical to human accountability roles.
- Role-based orchestration is not identical to human-role-to-MAS responsibility mapping.

Evidence Basis

Evidence Area	Source Type	Evidence Level	Claim-Level Status
Multi-agent collaboration	Official docs or GitHub	L2	source-bound at entry-point level
Role-based orchestration	Official docs or GitHub	L2	narrowed / partially validated
Task assignment and delegation	Official docs or GitHub	L2	narrowed / partially validated
Agent role composition	Official docs or GitHub	L2	source-bound at entry-point level
Agentic lifecycle objects	Official docs or GitHub	L2 (if available) or L5 (author inference)	L5 analytical interpretation unless directly stated

Source references:

- [CREWAI-DOCS]
- [CREWAI-GITHUB]

RCCS Mapping Summary

Table T-12-16: CrewAI RCCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management	
Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; risk management requires additional governance layer

Data Governance	
Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; data governance requires additional governance layer

Documentation	
Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Role and task definitions provide documentation primitives

Record-Keeping	
Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; audit-ready record-keeping requires additional layer

Transparency	
Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Role-based orchestration provides some transparency; governance transparency requires additional layer

Human Oversight	
Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; human oversight requires additional governance layer

Security

Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Framework security depends on deployment environment

Accountability

Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Agent roles exist; accountability requires additional governance layer

Contestability

Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; contestability requires additional governance layer

Monitoring

Mapping Strength	Weak to Moderate
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; governance monitoring requires additional layer

ALCS Mapping Summary

Table T-12-17: CrewAI ALCS Mapping Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Role / Authority / Acceptance (ALCS-01 to ALCS-04)

Mapping Strength	Weak to Partial
Evidence Basis	L2 + L5 reviewed
Notes	Agent role labels exist; human-role-to-MAS mapping, delegated authority boundary, agent role vs human role distinction, accepted outcome compliance require additional governance semantics; role labels ≠ accountability roles

Tool Action / Transfer / Drift / Evidence (ALCS-05 to ALCS-08)

Mapping Strength	Weak to Partial
Evidence Basis	L2 + L5 reviewed
Notes	Task delegation exists; tool-action liability, responsibility transfer, authority drift detection, MAS evidence partitioning require additional governance semantics

Reuse / Privacy / Validation (ALCS-09 to ALCS-11)

Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; cross-project reuse compliance, privacy lifecycle mapping, privacy-preserving validation require additional governance layer

Evidence Minimization / Data Subject Rights / Processor Chain (ALCS-12 to ALCS-14)

Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; evidence minimization, data subject rights reconciliation, processor chain tracking require additional governance layer

Substitution Conformance (ALCS-15)

Mapping Strength	Weak
Evidence Basis	L2 + L5 reviewed
Notes	Orchestration framework; vendor/model/runtime substitution conformance requires additional governance layer

Main Lifecycle Gaps

- Agent Role vs Human Role Distinction:** Whether CrewAI agent role labels map to human accountability roles requires validation.
- Human-Role-to-MAS Responsibility Mapping:** Whether CrewAI role-based orchestration maps to human-role-to-MAS responsibility semantics requires validation.
- Delegated Authority Boundary:** Whether CrewAI defines and enforces authority boundaries for agent actions requires validation.
- Accepted Outcome Compliance:** Whether CrewAI separates execution completion from outcome acceptance requires validation.
- Evidence Partitioning:** Whether CrewAI partitions evidence by agent, tool, authority, and privacy boundaries requires validation.
- Responsibility Transfer Across Agents:** Whether CrewAI tracks responsibility transfer during agent handoffs requires validation.

Boundary Note

This section maps public evidence to RCCS-T/RCCS-M/ALCS dimensions. It is not a legal compliance judgment, procurement recommendation, certification, or endorsement. CrewAI is evaluated for governance primitives and lifecycle responsibility objects based on public documentation. Agent role labels and task orchestration do not automatically provide governance responsibility semantics.

12.11 Chapter 12 Synthesis

This chapter evaluated eight systems against RCCS-T, RCCS-M, and ALCS dimensions using public evidence. The purpose was not to rank systems as best or worst, but to make visible which systems provide which capabilities, where evidence is strong, where evidence is weak, and where lifecycle gaps remain.

Synthesis Themes

1. Enterprise AI governance platforms tend to show stronger RCCS posture.

Systems like IBM watsonx.governance are designed for policy enforcement, model risk management, regulatory compliance coverage, and AI lifecycle management. They provide enterprise-grade governance capabilities for risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, and monitoring. This makes them relevant for organizations that require regulatory compliance coverage across the AI lifecycle.

2. Cloud AI platforms tend to show moderate-to-strong RCCS-T posture, mixed RCCS-M posture, and partial-to-moderate ALCS posture depending on agent lifecycle object exposure.

Systems like Microsoft Azure AI Foundry, AWS Bedrock / Guardrails / AgentCore, and Google Vertex AI / ADK / Model Armor provide cloud platform capabilities for model management, responsible AI tooling, content safety, tracing, evaluation, and enterprise integration. They show moderate-to-strong RCCS-T posture for cloud AI management, model governance, safety, evaluation, and monitoring. Their RCCS-M posture is more mixed because MRO-adjusted coverage depends on whether platform capabilities expose authority, evidence, accepted-outcome, privacy, substitution, and closure objects. ALCS posture is partial-to-moderate and depends on whether the platform exposes lifecycle responsibility objects for authority boundaries, accepted outcomes, MAS evidence partitioning, and responsibility transfer.

3. Agent orchestration frameworks tend to show useful execution and observability primitives but require additional lifecycle governance semantics.

Systems like LangGraph / LangSmith provide workflow graphs, state management, checkpoints, traces, and observability features. These primitives are useful for agent orchestration and debugging. However, orchestration and observability do not automatically provide lifecycle responsibility semantics. Human-role-to-MAS mapping, delegated authority boundary, accepted outcome compliance, tool-action liability, responsibility transfer, and MAS evidence partitioning require additional governance semantics beyond orchestration and observability.

4. Developer agent SDKs tend to show useful execution primitives but require additional enterprise governance layer.

Systems like OpenAI Agents SDK and CrewAI provide developer-friendly execution primitives for agent workflows, tool use, and multi-agent collaboration. These primitives are useful for building agentic applications. However, execution framework features and role-based orchestration do not automatically provide enterprise governance, audit readiness, accepted outcome, or lifecycle responsibility semantics. Developer primitives are not identical to enterprise lifecycle governance objects.

5. Lifecycle protocol paths tend to show stronger RCCS-M and ALCS semantics but require implementation, adoption, and enterprise integration validation.

Systems like MPLP are designed to expose lifecycle responsibility objects as first-class protocol primitives. They provide protocol-level semantics for human-role-to-MAS mapping, delegated authority boundaries, accepted outcome records, responsibility transfer, evidence partitioning, privacy lifecycle mapping, and vendor substitution conformance. This makes them relevant for systems that require lifecycle responsibility semantics across agents, tools, vendors, and organizational boundaries. However, MPLP implementation maturity, enterprise integration, vendor adoption, operational deployment evidence, and regulator familiarity require validation.

6. The field gap is not lack of AI tools. The field gap is lack of explicit lifecycle responsibility objects.

The eight systems evaluated in this chapter represent a wide range of capabilities: enterprise AI governance platforms, cloud AI platforms, agent orchestration frameworks, developer agent SDKs, and lifecycle protocol paths. All eight systems provide useful capabilities for building, deploying, and managing agentic AI systems. The field gap is not lack of AI tools. The field gap is lack of explicit lifecycle responsibility objects that map regulatory obligations to engineering primitives.

Model governance and AI governance are not identical to agentic lifecycle governance. Responsible AI tooling and cloud platform features are not identical to lifecycle responsibility objects. Orchestration and observability are not identical to lifecycle responsibility semantics. Execution framework features and role-based orchestration are not identical to enterprise lifecycle governance objects.

The comparison supports the core thesis: agentic compliance analysis must include lifecycle responsibility, not only model compliance.

Synthesis Matrix

The following table synthesizes the eight system mappings by system category.

Table T-12-18: Chapter 12 Synthesis Matrix

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Lifecycle protocol path (MPLP)	
Typical RCCS-T Strength	Limited to partial product-platform coverage
Typical RCCS-M / ALCS Posture	Strong RCCS-M and ALCS semantic posture for authority boundaries, accepted outcome records, responsibility transfer, evidence partitioning, substitution conformance
Typical Lifecycle Gap	Implementation maturity, enterprise integration, vendor adoption, operational deployment evidence, regulator familiarity
Interpretation	Relevant to MRO-adjusted lifecycle responsibility semantics; implementation, adoption, and enterprise integration require validation

Enterprise AI governance platform (IBM watsonx.governance)

Typical RCCS-T Strength	Strong RCCS-T posture for risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, monitoring
Typical RCCS-M / ALCS Posture	Partial RCCS-M / ALCS where lifecycle responsibility objects are not explicit
Typical Lifecycle Gap	Multi-agent responsibility transfer, accepted outcome governance, tool-action liability boundary, MAS evidence partitioning, cross-project reuse controls
Interpretation	Designed for enterprise AI governance; agentic lifecycle objects require validation

Cloud AI platform (Microsoft Azure AI Foundry, AWS Bedrock / Guardrails / AgentCore, Google Vertex AI / ADK / Model Armor)

Typical RCCS-T Strength	Moderate-to-strong RCCS-T posture for cloud AI management, model governance, safety, evaluation, monitoring
Typical RCCS-M / ALCS Posture	Partial-to-moderate RCCS-M / ALCS depending on agent lifecycle object exposure
Typical Lifecycle Gap	Accepted outcome compliance, delegated authority boundary, cross-agent responsibility transfer, evidence partitioning, vendor/model/runtime substitution conformance
Interpretation	Designed for cloud AI platform capabilities; lifecycle responsibility objects require validation; responsible AI tooling does not equal lifecycle responsibility objects

Agent orchestration / observability framework (LangGraph / LangSmith)

Typical RCCS-T Strength	Moderate RCCS-T posture for documentation, record-keeping, transparency, monitoring
Typical RCCS-M / ALCS Posture	Moderate RCCS-M / ALCS adjacency through graphs, state, checkpoints, traces, observability
Typical Lifecycle Gap	Human-role-to-MAS responsibility mapping, delegated authority boundary, accepted outcome compliance, tool-action liability, responsibility transfer, MAS evidence partitioning
Interpretation	Designed for agent orchestration and observability; lifecycle responsibility semantics require additional governance layer; orchestration does not equal governance

Developer agent SDK (OpenAI Agents SDK)

Typical RCCS-T Strength	Weak-to-moderate RCCS-T posture beyond SDK primitives
Typical RCCS-M / ALCS Posture	Partial RCCS-M / ALCS where SDK primitives support agents, tools, handoffs, guardrails, tracing, or sessions
Typical Lifecycle Gap	Delegated authority boundary, accepted outcome compliance, tool-action liability, MAS evidence partitioning, human accountability mapping, dispute/remediation closure
Interpretation	Designed for developer execution primitives; enterprise governance and lifecycle responsibility semantics require additional layer; execution framework does not equal governance layer

Multi-agent collaboration framework (CrewAI)

Typical RCCS-T Strength	Weak-to-moderate RCCS-T posture beyond orchestration primitives
Typical RCCS-M / ALCS Posture	Partial RCCS-M / ALCS where roles, tasks, and collaboration patterns support lifecycle analysis
Typical Lifecycle Gap	Agent role vs human role distinction, human-role-to-MAS responsibility mapping, delegated authority boundary, accepted outcome compliance, evidence partitioning, responsibility transfer
Interpretation	Designed for multi-agent collaboration; lifecycle responsibility semantics require additional governance layer; role labels do not equal accountability roles

Boundary Restatement

This synthesis is not a product ranking. It is not a legal compliance judgment. It is not a procurement recommendation. It is not an endorsement or criticism of any vendor. It is a responsibility-semantics mapping based on public evidence and author inference where explicitly disclosed.

No system is described as non-compliant. MPLP is not described as required for compliance. MPLP is not described as the only correct path. MPLP is not described as certification or legal authority. Validation Lab is not described as certification or legal authority.

The synthesis supports the core thesis: agentic compliance analysis must include lifecycle responsibility, not only model compliance. The field gap is not lack of AI tools. The field gap is lack of explicit lifecycle responsibility objects.

Evaluation Results: RCCS-T / RCCS-M / ALCS Provisional Analytical Profiles

MAIN RESULT Traditional regulatory coverage and agentic lifecycle regulatory coverage are not the same. RCCS-T captures familiar governance surfaces; RCCS-M shows whether those obligations can be expressed through MRO-style lifecycle responsibility objects.	METHOD Scores use the 0-100 provisional analytical scale, source-qualified evidence labels, and bounded RCCS-T/RCCS-M/ALCS interpretation.
BOUNDARY Scores are not vendor rankings, procurement recommendations, certifications, regulatory approval signals, or legal compliance proof.	

How to Read the Scores

Each system receives three profile signals. **RCCS-T** measures traditional regulatory and governance coverage such as risk management, data governance, documentation, records, transparency, human oversight, security, accountability, contestability, and monitoring. **RCCS-M** measures whether those obligations are expressible through Missing Regulatory Objects required for agentic lifecycle responsibility compliance. **ALCS** measures lifecycle responsibility conformance. Ranges show uncertainty where source support is mixed, access-limited, or partly inferential.

RCCS-T Traditional Regulatory Compliance Coverage Score.	RCCS-M MRO-Adjusted Regulatory Compliance Coverage Score.	ALCS Agentic Lifecycle Conformance Score.

Key finding: cloud and enterprise governance platforms often show stronger RCCS-T, while MPLP shows stronger RCCS-M and ALCS because it is organized around protocol-level lifecycle responsibility semantics rather than product dashboard coverage.

Reading rule: compare patterns, not rank order. System sequence follows the existing source mapping order and must not be read as a leaderboard.

RCCS-T / RCCS-M / ALCS Score Overview

Compact scorecard matrix for side-by-side comparison. Primary scores are visually dominant; ranges and evidence notes remain secondary uncertainty and source-boundary context.

Fixed source order; provisional analytical framing; non-ranking. Scores are analytical profiles, not vendor rankings, legal compliance proof, certifications, regulatory approval signals, or procurement recommendations.

RCCS-T / RCCS-M / ALCS Provisional Analytical Score Overview

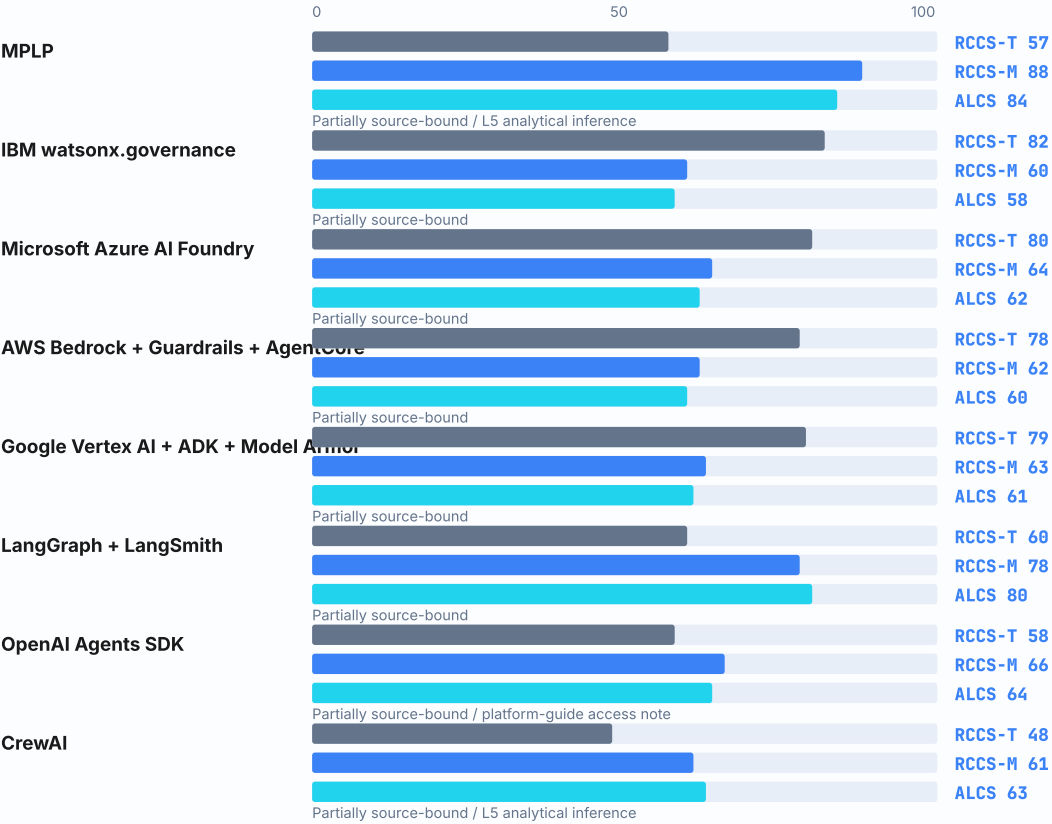
Fixed non-ranked source order. RCCS-T is traditional governance coverage; RCCS-M is MRO-adjusted agentic regulatory coverage; ALCS is lifecycle conformance.

System	System Type	RCCS-T	RCCS-M	ALCS	Confidence	Main Interpretation / Evidence Boundary
MPLP	Lifecycle protocol	57 range 52-62	88 range 84-92	84 range 80-88	Partially source-bound / L5 analytical inference	MPLP's RCCS-T is limited because it is not an enterprise governance SaaS. Its RCCS-M and ALCS posture is high because protocol semantics map strongly to MRO-style authority, evidence, accepted-outcome, responsibility-transfer, and substitution objects. Protocol materials support lifecycle semantics; traditional product-governance posture remains implementation-dependent, while MRO-adjusted mapping is strong but author analytical.
IBM watsonx.governance	Governance platform	82 range 78-86	60 range 55-65	58 range 53-63	Partially source-bound	IBM shows strong RCCS-T visibility for traditional governance surfaces. RCCS-M is lower because lifecycle responsibility objects such as accepted outcome, cross-agent transfer, and runtime substitution require additional analytical mapping or implementation evidence. Official product documentation supports governance/documentation/monitoring posture; agentic lifecycle semantics remain analytical.
Microsoft Azure AI Foundry	Cloud AI platform	80 range 76-84	64 range 59-69	62 range 57-67	Partially source-bound	Azure AI Foundry shows strong RCCS-T platform governance posture. RCCS-M remains partial because MRO-level accepted-outcome, MAS evidence partitioning, and lifecycle responsibility closure depend on implementation design. Responsible AI, evaluation, safety, and platform governance documentation support RCCS posture; MRO-level lifecycle mapping is analytical.
AWS Bedrock + Guardrails + AgentCore	Cloud AI platform	78 range 74-83	62 range 57-67	60 range 55-65	Partially source-bound	AWS shows strong RCCS-T through cloud governance, guardrails, and runtime control surfaces. RCCS-M is partial because lifecycle responsibility transfer, acceptance, evidence partitioning, and closure are not fully first-class public responsibility objects. Official service documentation supports guardrails/runtime/governance posture; lifecycle responsibility transfer and closure remain implementation-dependent.
Google Vertex AI + ADK + Model Armor	Cloud AI platform	79 range 75-84	63 range 58-68	61 range 56-66	Partially source-bound	Google shows strong RCCS-T across platform governance, safety, and operational AI services. RCCS-M remains partial because lifecycle responsibility object conformance depends on how ADK and platform services are bound to enterprise evidence and accountability records. Official model/platform governance, ADK, and safety documentation support visible posture; lifecycle object conformance remains analytical.
LangGraph + LangSmith	Agent orchestration / observability	60 range 55-65	78 range 73-83	80 range 76-84	Partially source-bound	LangGraph/LangSmith has moderate RCCS-T because it is not a full compliance platform. It has stronger RCCS-M and ALCS posture because orchestration, state, traceability, and evaluation primitives map more directly to lifecycle responsibility objects. Official orchestration, tracing, observability, and evaluation documentation supports lifecycle posture; regulatory control mapping requires surrounding governance.
OpenAI Agents SDK	Agent SDK / developer platform	58 range 52-63	66 range 60-71	64 range 58-69	Partially source-bound / platform-guide access note	OpenAI Agents SDK has partial RCCS-T because official accessible sources support SDK primitives rather than enterprise governance coverage. RCCS-M is somewhat stronger where agents, handoffs, guardrails, tracing, tools, and sessions map to lifecycle objects, but the mapping remains source-qualified and not full lifecycle compliance. Accessible official OpenAI Agents SDK Python and JavaScript documentation supports narrowed SDK-surface posture; the inaccessible platform guide is retained as an access note only.
CrewAI	Agent orchestration framework	48 range 42-54	61 range 55-66	63 range 58-68	Partially source-bound / L5 analytical inference	CrewAI has limited-to-partial RCCS-T because governance and evidence controls require external enterprise systems. RCCS-M is partial where role, task, crew, and flow primitives can be analytically mapped to lifecycle responsibility objects. Official documentation and repository materials support role/task orchestration posture; regulatory evidence and lifecycle closure controls require external governance.

CHART-RCCS-ALCS-SCORE-BARS

RCCS-T / RCCS-M / ALCS Provisional Analytical Score Comparison

RCCS-T / RCCS-M / ALCS Provisional Analytical Score Comparison



RCCS-T = traditional governance coverage; RCCS-M = MRO-adjusted agentic regulatory coverage; ALCS = lifecycle conformance. Fixed source order; not ranking.

The chart separates traditional governance coverage (RCCS-T), MRO-adjusted agentic regulatory coverage (RCCS-M), and lifecycle conformance (ALCS). This makes MPLP's protocol-level lifecycle strength visible without inflating its traditional product-governance score.

Scores are analytical profiles, not vendor rankings, legal compliance proof, certifications, regulatory approval signals, or procurement recommendations.

CHART - RCCS - ALCS - QUADRANT

RCCS-M / ALCS Provisional Analytical Score Map

RCCS-M / ALCS Provisional Analytical Score Map



The chart places evaluated systems by MRO-adjusted regulatory coverage and lifecycle conformance, so the reader can see the agentic lifecycle result pattern without converting the comparison into a leaderboard.

Positions are not vendor rankings, procurement recommendations, certification claims, or legal compliance proof.

Dimension-Level Coverage Patterns

Landscape heatmap module keeps the explanatory context and system-by-dimension posture table together, preserving scanability without applying landscape to ordinary portrait tables.

Qualitative, source-qualified posture labels. Fixed source order; no leaderboard, certification, legal compliance proof, regulatory approval signal, or procurement recommendation is implied.

System x Dimension Posture Heatmap

Values are qualitative, source-qualified posture labels. They are not final scores or rankings.

System	RCCS	Authority	Evidence	Privacy	Substitution	Closure
MPLP	Partial	Strong	Strong	Partial	Partial	Strong
IBM watsonx.governance	Strong	Partial	Strong	Partial	Limited	Partial
Microsoft Azure AI Foundry	Strong	Partial	Partial	Partial	Partial	Partial
AWS Bedrock + Guardrails + AgentCore	Strong	Partial	Partial	Partial	Partial	Partial
Google Vertex AI + ADK + Model Armor	Strong	Partial	Partial	Partial	Partial	Partial
LangGraph + LangSmith	Partial	Strong	Strong	Limited	Partial	Partial
OpenAI Agents SDK	Partial	Partial	Partial	Limited	Partial	Limited
CrewAI	Limited	Partial	Partial	Limited	Partial	Limited

System-by-System Analytical Profiles

LIFECYCLE PROTOCOL

MPLP

RCCS-T

57

52-62

RCCS-M

88

84-92

ALCS SCORE

84

80-88

CONFIDENCE

Partially source-bound / L5 analytical inference

Posture labels: RCCS-T Partial; RCCS-M Strong; ALCS Strong.

Main interpretation: MPLP's RCCS-T is limited because it is not an enterprise governance SaaS. Its RCCS-M and ALCS posture is high because protocol semantics map strongly to MRO-style authority, evidence, accepted-outcome, responsibility-transfer, and substitution objects.

Visible strength: Protocol-level lifecycle semantics for responsibility objects, evidence pointers, and closure records.

Lifecycle gap: Enterprise deployment, external adoption, and implementation evidence remain outside protocol semantics.

Evidence basis: MPLP public materials plus author analytical mapping where lifecycle conformance is not directly stated.

Score safety: Range-backed score is safe as a provisional analytical profile, not as a vendor ranking or procurement selection signal.

Boundary: Protocol analytical profile only; not a procurement recommendation, certification, or proof of legal compliance.

GOVERNANCE PLATFORM

IBM watsonx.governance

RCCS-T

82

78-86

RCCS-M

60

55-65

ALCS SCORE

58

53-63

CONFIDENCE

Partially source-bound

Posture labels: RCCS-T Strong; RCCS-M Partial; ALCS Partial.

Main interpretation: IBM shows strong RCCS-T visibility for traditional governance surfaces. RCCS-M is lower because lifecycle responsibility objects such as accepted outcome, cross-agent transfer, and runtime substitution require additional analytical mapping or implementation evidence.

Visible strength: Model governance, documentation, risk management, monitoring, and enterprise governance controls.

Lifecycle gap: Agent handoff, accepted outcome, cross-agent responsibility transfer, and runtime substitution semantics are less visible.

Evidence basis: Official product documentation for visible governance features; lifecycle-object interpretation remains analytical.

Score safety: Score range is safe for source-qualified analytical comparison; exact operational conformance is not established.

Boundary: Source-qualified posture profile only; not product ranking or implementation audit opinion.

CLOUD AI PLATFORM

Microsoft Azure AI Foundry

RCCS-T

80

76-84

RCCS-M

64

59-69

ALCS SCORE

62

57-67

CONFIDENCE

Partially source-bound

Posture labels: RCCS-T Strong; RCCS-M Partial; ALCS Partial.

Main interpretation: Azure AI Foundry shows strong RCCS-T platform governance posture. RCCS-M remains partial because MRO-level accepted-outcome, MAS evidence partitioning, and lifecycle responsibility closure depend on implementation design.

Visible strength: Responsible AI, model/platform governance, safety controls, evaluation support, and enterprise integration.

Lifecycle gap: Accepted-outcome closure, MAS evidence partitioning, and explicit MRO-level lifecycle objects require enterprise design.

Evidence basis: Official Microsoft documentation and responsible AI materials, with lifecycle mapping as source-qualified analysis.

Score safety: Score range is safe as provisional posture, not as product quality or procurement signal.

Boundary: Analytical profile, not endorsement, readiness threshold, or procurement advice.

CLOUD AI PLATFORM

AWS Bedrock + Guardrails + AgentCore

RCCS-T

78

74-83

RCCS-M

62

57-67

ALCS SCORE

60

55-65

CONFIDENCE

Partially source-bound

Posture labels: RCCS-T Strong; RCCS-M Partial; ALCS Partial.

Main interpretation: AWS shows strong RCCS-T through cloud governance, guardrails, and runtime control surfaces. RCCS-M is partial because lifecycle responsibility transfer, acceptance, evidence partitioning, and closure are not fully first-class public responsibility objects.

Visible strength: Foundation-model access controls, guardrails, agent/runtime services, monitoring hooks, and enterprise integration.

Lifecycle gap: Responsibility transfer, acceptance records, privacy-preserving validation, and lifecycle closure remain implementation-dependent.

Evidence basis: Official AWS service documentation; MRO/ALCS mapping remains provisional and source-qualified.

Score safety: Score range is safe as source-qualified analytical posture only.

Boundary: Not a ranking, certification, procurement recommendation, or compliance proof.

CLOUD AI PLATFORM

Google Vertex AI + ADK + Model Armor

RCCS-T

79

75-84

RCCS-M

63

58-68

ALCS SCORE

61

56-66

CONFIDENCE

Partially source-bound

Posture labels: RCCS-T Strong; RCCS-M Partial; ALCS Partial.

Main interpretation: Google shows strong RCCS-T across platform governance, safety, and operational AI services. RCCS-M remains partial because lifecycle responsibility object conformance depends on how ADK and platform services are bound to enterprise evidence and accountability records.

Visible strength: Model/platform governance, agent-development tooling, safety filtering, and enterprise AI operations.

Lifecycle gap: Lifecycle responsibility object coverage depends on how teams bind ADK/runtime behavior to evidence and accountability controls.

Evidence basis: Official Google documentation; agentic lifecycle conformance is mapped analytically where not directly specified.

Score safety: Score range is safe for non-ranking analytical comparison, not final assessment.

Boundary: Source-qualified posture only; no vendor ordering or procurement selection signal.

AGENT ORCHESTRATION / OBSERVABILITY

LangGraph + LangSmith

RCCS-T 60 55-65	RCCS-M 78 73-83	ALCS SCORE 80 76-84	CONFIDENCE Partially source-bound

Posture labels: RCCS-T Partial; RCCS-M Strong; ALCS Strong.

Main interpretation: LangGraph/LangSmith has moderate RCCS-T because it is not a full compliance platform. It has stronger RCCS-M and ALCS posture because orchestration, state, traceability, and evaluation primitives map more directly to lifecycle responsibility objects.

Visible strength: Agent graph orchestration, traceability, observability, evaluation workflows, and runtime state patterns.

Lifecycle gap: Regulatory control mapping, legal-source coverage, and enterprise procurement controls require surrounding governance.

Evidence basis: Official LangGraph/LangSmith documentation with analytical RCCS-T/RCCS-M/ALCS mapping.

Score safety: Score range is safe as lifecycle capability posture only.

Boundary: Lifecycle capability posture, not a regulator-approved benchmark or market endorsement.

AGENT SDK / DEVELOPER PLATFORM

OpenAI Agents SDK

RCCS-T 58 52-63	RCCS-M 66 60-71	ALCS SCORE 64 58-69	CONFIDENCE Partially source-bound / platform-guide access note

Posture labels: RCCS-T Partial; RCCS-M Partial; ALCS Partial.

Main interpretation: OpenAI Agents SDK has partial RCCS-T because official accessible sources support SDK primitives rather than enterprise governance coverage. RCCS-M is somewhat stronger where agents, handoffs, guardrails, tracing, tools, and sessions map to lifecycle objects, but the mapping remains source-qualified and not full lifecycle compliance.

Visible strength: Agent/tool orchestration primitives, developer-facing SDK surfaces, and tool/action workflow support.

Lifecycle gap: Official SDK docs support developer primitives, not full enterprise lifecycle governance, accepted-outcome compliance, or audit-ready responsibility closure.

Evidence basis: Accessible official SDK documentation for agents, handoffs, guardrails, tracing, tools, and sessions; platform guide 403 is a non-blocking access note after claim narrowing.

Score safety: Score range remains safe as a provisional SDK-surface analytical profile after unsupported lifecycle-governance claims are narrowed.

Boundary: SDK-surface analytical posture only; not final assessment, ranking, procurement recommendation, or compliance proof.

AGENT ORCHESTRATION FRAMEWORK

CrewAI

RCCS-T

48

42-54

RCCS-M

61

55-66

ALCS SCORE

63

58-68

CONFIDENCE

Partially source-bound / L5 analytical inference

Posture labels: RCCS-T Limited / Partial; RCCS-M Partial; ALCS Partial.

Main interpretation: CrewAI has limited-to-partial RCCS-T because governance and evidence controls require external enterprise systems. RCCS-M is partial where role, task, crew, and flow primitives can be analytically mapped to lifecycle responsibility objects.

Visible strength: Role-based multi-agent orchestration, task delegation, crews/flows, and operational agent patterns.

Lifecycle gap: Regulatory controls, formal evidence retention, accepted outcome, privacy mapping, and remediation closure require external governance.

Evidence basis: Official documentation and repository materials with analytical MRO/ALCS mapping.

Score safety: Score range is safe as framework posture only, not product quality or deployment readiness.

Boundary: Framework posture only; not a product quality score, certification, or procurement conclusion.

What the Results Show

- Governance platforms tend to show stronger RCCS-T because public documentation more clearly exposes conventional governance controls.
- MPLP should not be read as weak merely because it is not a traditional governance product; its RCCS-M and ALCS posture is high because lifecycle responsibility semantics are the point of the protocol layer.
- Cloud platforms show meaningful RCCS-M potential, but accepted outcome, responsibility transfer, evidence partitioning, and substitution closure remain implementation-dependent in public sources.
- Agent SDKs and orchestration frameworks provide lifecycle primitives, but they generally need surrounding governance systems to become full regulatory coverage surfaces.
- The common gap is not model quality; it is explicit lifecycle responsibility object coverage.
- Scores should be read with confidence labels and evidence boundaries, not as procurement or compliance outcomes.

What the Results Do Not Prove

RCCS-M is a proposed forward-looking analytical model for MRO-adjusted agentic regulatory coverage. It does not replace current law, establish a regulator-approved benchmark, certify systems, prove legal compliance, identify a best system, or recommend procurement. MPLP's RCCS-M/ALCS posture remains an author analytical mapping where official MPLP materials do not directly state GAIC MRO/ALCS claims.

13. MPLP Deep Mapping, Boundary, and Non-Claim Discipline

MAIN RESULT MPLP has strong semantic fit for lifecycle responsibility objects but remains one protocol path.	METHOD / LENS MPLP-to-MRO/RCCS-T/RCCS-M/ALCS mapping and conflict-of-interest discipline.
EVIDENCE BASIS Protocol semantics are useful only when kept separate from endorsement or certification claims.	KEY TAKEAWAY / BOUNDARY MPLP is not required, certified, or exclusive.

Version: v0.3.2-FRC-R3

13.0 Why MPLP Requires Separate Boundary Treatment

MPLP (Multi-Agent Lifecycle Protocol) appears in this white paper because it is a protocol-level lifecycle governance approach. The author's relationship to MPLP creates an explicit conflict of interest already disclosed in the front matter. A separate deep mapping chapter is necessary to prevent hidden promotion. Deep mapping must be paired with non-claim discipline. The paper evaluates MPLP as one protocol path, not as the only path.

Because MPLP is created by the author of this white paper, the mapping must be more explicit, not less explicit. The correct response to conflict of interest is not silence; it is transparent boundary discipline.

This chapter maps MPLP's protocol semantics to the Missing Regulatory Objects, RCCS-T, RCCS-M, and ALCS dimensions. Official MPLP documentation was located during claim-level source review, but protocol-to-MRO/RCCS-M/ALCS mapping remains an evidence-qualified analytical interpretation unless the protocol text directly defines the relevant lifecycle object. MPLP mapping remains protocol-level semantic fit, not implementation conformance. The chapter distinguishes protocol-level fit from implementation evidence, deployment maturity, enterprise adoption, auditor acceptance, insurer acceptance, and regulatory recognition. The chapter closes with explicit non-claim discipline to prevent overclaim.

13.1 Protocol Proposal vs Product / Framework Benchmark

MPLP is evaluated in this paper primarily as a **protocol-level semantic proposal**. Enterprise governance platforms, cloud services, orchestration frameworks, and developer SDKs are evaluated as product, platform, framework, or implementation surfaces. These are not identical maturity categories.

A protocol proposal should be assessed by whether it defines necessary semantic objects, whether object boundaries are clear, whether independent implementations could target the same semantics, whether conformance evidence can be generated and validated, and whether interoperability semantics can be specified. A product or platform should also be assessed by deployment features, enterprise controls,

operational tooling, market adoption, third-party implementation evidence, support maturity, and integration depth.

This distinction explains the RCCS-T / RCCS-M / ALCS profile difference. Enterprise product platforms may score stronger on RCCS-T because they expose productized governance surfaces such as dashboards, policy workflows, documentation management, cloud controls, and monitoring consoles. MPLP may score stronger on RCCS-M and ALCS because it is designed around protocol-level lifecycle responsibility semantics: authority boundaries, responsibility transfer, accepted outcomes, evidence partitioning, substitution conformance, and remediation closure.

MPLP's high RCCS-M / ALCS alignment should not be read as market adoption proof, independent validation, certification, deployment maturity, or procurement superiority. It is design alignment with the proposed lifecycle object layer. The central question is therefore whether those lifecycle responsibility objects are necessary for Agent/MAS governance. If they are necessary, a protocol designed around them should score strongly under models that measure them. If they are not necessary, the critique should show how agentic work can be governed without them. Any TCP/IP or POSIX comparison in this context should be read only as analogous in category as protocol-level semantic proposals, not equivalent in adoption status.

13.2 Interoperability and Specification Boundary

This white paper defines the regulatory object gap, the MRO object model, the RCCS-T / RCCS-M / ALCS evaluation logic, the lifecycle responsibility governance need, and the protocol-level adequacy argument. It is closer to the protocol problem-statement and semantic object layer than to a complete wire-level interoperability specification.

This paper does not fully define the complete wire format, serialization grammar, all validation rules, full cross-system exchange semantics, or a complete third-party conformance test suite. Those belong to MPLP technical specification work, schema definition, Cognitive OS runtime mapping, and Validation Lab evidence/conformance evaluation.

This boundary does not weaken the necessity argument. It separates agenda-setting governance theory and semantic object modeling from implementation-level protocol specification. A protocol path becomes more mature when its object model, schemas, validator rules, runtime mappings, test fixtures, and third-party conformance evidence become independently implementable and reviewable.

13.3 What MPLP Is and Is Not

MPLP is a lifecycle protocol path for expressing agentic responsibility objects. It is not a certification program, regulator, legal authority, cloud platform, procurement recommendation, or exclusive compliance method.

Table T-13-01: MPLP Boundary Statement

Claim Area	Allowed Framing	Forbidden Framing
Protocol identity	MPLP is a lifecycle protocol path for expressing agentic responsibility objects	MPLP is the official standard; MPLP is required for compliance
Compliance relationship	MPLP may show limited RCCS-T product-platform coverage while mapping strongly to RCCS-M and ALCS because those profiles measure lifecycle responsibility semantics	MPLP issues compliance certificates; MPLP proves legal compliance

Claim Area	Allowed Framing	Forbidden Framing
Certification	MPLP can be evaluated as one example of protocol-level lifecycle governance	MPLP is a certification program; MPLP issues system certificates
Legal authority	MPLP defines protocol semantics for lifecycle governance	MPLP is a regulator; MPLP is a legal authority
Vendor status	MPLP is a protocol specification	MPLP is a cloud platform; MPLP is a vendor product
Implementation maturity	MPLP requires implementation validation and enterprise integration	MPLP is production-ready; MPLP proves deployment maturity
Validation Lab relationship	Validation Lab may be referenced separately as a non-certifying evidence adjudication example	Validation Lab issues compliance certificates; Validation Lab is a certification authority
Industry adoption	MPLP adoption requires market validation	MPLP is industry-standard; MPLP is regulator-approved
System mapping	MPLP can be compared to other systems by responsibility semantics	Other systems are non-compliant; MPLP is the only solution

13.4 MPLP as a Lifecycle Protocol Path

MPLP treats lifecycle responsibility as a first-class protocol concern. MPLP is relevant to AI Agent Lifecycle Governance because it provides explicit protocol-level semantics for context, plan, confirmation, trace/evidence, roles, dialogue/collaboration, extension, core semantics, and network/integration boundaries.

Protocol semantics can define what must be represented, but implementations determine whether those semantics are operationally enforced. A protocol path can support governance, but does not replace organizational controls.

Canonical boundary: MPLP defines protocol semantics; deployment determines operational control.

MPLP's protocol-level approach means it can express lifecycle responsibility objects that other systems may implement through platform features, orchestration patterns, or enterprise integration. The protocol path is one approach among several valid approaches to lifecycle governance.

13.5 MPLP Mapping to the 16 Missing Regulatory Objects

The following table maps MPLP protocol semantics to the sixteen Missing Regulatory Objects defined in Chapter 6. Mapping strength reflects protocol-level semantic fit, not deployment conformance.

Table T-13-02: MPLP Mapping to the Sixteen Missing Regulatory Objects

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

MRO-01 : Human Role to MAS Responsibility Mapping	
MPLP Mapping Strength	Strong
Protocol-Level Relevance	Role / Context / Confirm / Trace semantics support human-to-agent responsibility mapping
Implementation / Evidence Limit	Requires implementation conformance; requires enterprise role integration

MR0-02 : Delegated Authority Boundary

MPLP Mapping Strength	Strong
Protocol-Level Relevance	Confirm / authority boundary semantics support delegation scope definition
Implementation / Evidence Limit	Requires operational policy; requires runtime enforcement validation

MR0-03 : Agent Role is not Human Role

MPLP Mapping Strength	Strong
Protocol-Level Relevance	Role semantics distinguish agent responsibility surfaces from human roles
Implementation / Evidence Limit	Requires governance interpretation; requires organizational adoption

MR0-04 : Accepted Outcome Compliance

MPLP Mapping Strength	Strong
Protocol-Level Relevance	Confirm / Trace / Plan closure semantics support accepted outcome states
Implementation / Evidence Limit	Requires review workflow integration; requires acceptance authority mapping

MR0-05 : Tool-Action Liability Boundary

MPLP Mapping Strength	Moderate
Protocol-Level Relevance	Tool-action evidence semantics can support liability boundary records
Implementation / Evidence Limit	Requires tool integration; requires liability policy definition

MR0-06 : Responsibility Transfer Across Agents

MPLP Mapping Strength	Strong
Protocol-Level Relevance	Handoff / transfer semantics support responsibility transfer records
Implementation / Evidence Limit	Requires multi-agent integration; requires constraint inheritance validation

MR0-07 : Authority Drift

MPLP Mapping Strength	Moderate
Protocol-Level Relevance	Monitoring semantics can support drift detection
Implementation / Evidence Limit	Requires runtime monitoring integration; requires drift alert policy

MR0-08 : MAS Evidence Partitioning

MPLP Mapping Strength	Strong
Protocol-Level Relevance	Trace / evidence semantics support partitioned evidence chains
Implementation / Evidence Limit	Requires evidence export; requires partition policy definition

MR0-09 : Cross-Project Reuse Compliance

MPLP Mapping Strength	Moderate
Protocol-Level Relevance	Context reset semantics can support reuse validation
Implementation / Evidence Limit	Requires reuse policy; requires revalidation workflow

MR0-10 : Privacy / GDPR Lifecycle Mapping

MPLP Mapping Strength	Moderate
Protocol-Level Relevance	Data-flow semantics can support privacy lifecycle mapping
Implementation / Evidence Limit	Requires privacy policy integration; requires GDPR interpretation

MR0-11 : Privacy-Preserving Third-Party Validation

MPLP Mapping Strength	Moderate
Protocol-Level Relevance	Selective disclosure semantics can support privacy-preserving validation
Implementation / Evidence Limit	Requires validation protocol integration; requires third-party acceptance

MR0-12 : Evidence Minimization and Selective Disclosure

MPLP Mapping Strength	Moderate
Protocol-Level Relevance	Evidence semantics can support minimization and selective disclosure
Implementation / Evidence Limit	Requires retention policy; requires disclosure rules definition

MR0-13 : Data Subject Rights vs Evidence Retention

MPLP Mapping Strength	Partial
Protocol-Level Relevance	Retention semantics require policy-level reconciliation
Implementation / Evidence Limit	Requires legal interpretation; requires data subject rights workflow

MR0-14 : Third-Party Processor / Subprocessor Chain	
MPLP Mapping Strength	Moderate
Protocol-Level Relevance	Network / integration semantics can track processor chain
Implementation / Evidence Limit	Requires processor agreement integration; requires chain documentation

MR0-15 : Vendor / Model / Runtime Substitution Conformance	
MPLP Mapping Strength	Moderate
Protocol-Level Relevance	Integration boundary semantics can support substitution conformance
Implementation / Evidence Limit	Requires substitution testing; requires conformance validation

MR0-16 : Incident, Dispute, and Remediation Closure	
MPLP Mapping Strength	Moderate
Protocol-Level Relevance	Closure semantics can support incident and dispute resolution
Implementation / Evidence Limit	Requires incident workflow; requires remediation policy; does not imply legal closure

Key: Strong = protocol semantics directly express the required object; Moderate = protocol semantics can support the object with additional policy/integration; Partial = protocol semantics provide foundation but require significant additional work.

13.6 MPLP Mapping to RCCS-T and RCCS-M

MPLP should not be read as weak merely because it is not an enterprise governance SaaS. The paper distinguishes two regulatory-coverage profiles. **RCCS-T** measures traditional productized governance coverage: risk dashboards, organizational policy workflows, cloud security controls, documentation management, monitoring, and compliance operations. MPLP may be limited on that product-platform layer. **RCCS-M** measures whether regulatory obligations can be expressed through Missing Regulatory Objects. MPLP may map strongly on that layer because its protocol semantics are designed around lifecycle responsibility objects.

MPLP can support RCCS-M where protocol semantics create evidence, accountability, traceability, accepted-outcome records, authority boundaries, responsibility transfer, and substitution conformance. However, MPLP alone does not satisfy either RCCS-T or RCCS-M in deployment. Both profiles still depend on implementation, organizational controls, deployment environment, security operations, data governance policy, and legal interpretation.

Table T-13-03: MPLP Mapping to RCCS-T and RCCS-M Dimensions

Methodology matrix rendered as semantic row cards to preserve long evidence and boundary text; all source rows are retained.

RCCS-T overall product-platform coverage

Mapping Strength	Partial to Moderate
Why MPLP May Help	Protocol semantics may support governance evidence, but MPLP is not a productized governance dashboard or cloud compliance platform
Boundary / Evidence Limit	Requires product implementation, enterprise integration, organizational policy workflow, and operational controls

RCCS-M overall MRO-adjusted coverage

Mapping Strength	Strong
Why MPLP May Help	Protocol semantics map strongly to MRO-like authority, evidence, accepted-outcome, transfer, and substitution objects
Boundary / Evidence Limit	Author analytical mapping where not directly stated; requires implementation conformance

RCCS-01 : Risk Management

Mapping Strength	Moderate
Why MPLP May Help	Protocol semantics can support risk documentation and evidence
Boundary / Evidence Limit	Requires organizational risk management process; requires risk assessment integration

RCCS-02 : Data Governance

Mapping Strength	Moderate
Why MPLP May Help	Data-flow semantics can support data lineage and lifecycle tracking
Boundary / Evidence Limit	Requires data governance policy; requires data quality controls

RCCS-03 : Documentation

Mapping Strength	Moderate
Why MPLP May Help	Protocol semantics can generate lifecycle documentation
Boundary / Evidence Limit	Requires documentation standards; requires version control integration

RCCS-04 : Record-Keeping

Mapping Strength	Strong
Why MPLP May Help	Trace / evidence semantics support audit trail generation
Boundary / Evidence Limit	Requires evidence retention policy; requires tamper-evident storage

RCCS-05 : Transparency

Mapping Strength	Moderate
Why MPLP May Help	Evidence semantics can support explainability and disclosure
Boundary / Evidence Limit	Requires transparency policy; requires disclosure workflow

RCCS-06 : Human Oversight

Mapping Strength	Strong
Why MPLP May Help	Confirm / Role semantics support human review and approval workflows
Boundary / Evidence Limit	Requires human oversight policy; requires escalation rules

RCCS-07 : Security

Mapping Strength	Partial
Why MPLP May Help	Protocol semantics require security implementation
Boundary / Evidence Limit	Requires security controls; requires access management; requires encryption

RCCS-08 : Accountability

Mapping Strength	Strong
Why MPLP May Help	Role / Trace semantics support responsibility assignment and tracking
Boundary / Evidence Limit	Requires accountability policy; requires organizational adoption

RCCS-09 : Contestability

Mapping Strength	Moderate
Why MPLP May Help	Evidence / dispute semantics can support challenge and appeal
Boundary / Evidence Limit	Requires dispute resolution workflow; requires remediation policy

RCCS-10 : Monitoring

Mapping Strength	Moderate
Why MPLP May Help	Monitoring semantics can support continuous monitoring
Boundary / Evidence Limit	Requires monitoring infrastructure; requires alert policy; requires incident response

13.7 MPLP Mapping to ALCS

Because ALCS measures lifecycle responsibility objects, MPLP may map strongly to many ALCS dimensions. However, high protocol-level fit is not the same as deployment conformance.

A strong ALCS semantic fit means the protocol can express the required lifecycle object. It does not mean any given implementation has enforced, audited, or operationalized that object.

Table T-13-04: MPLP Mapping to ALCS Dimensions

Methodology matrix rendered as semantic row cards to preserve long evidence and boundary text; all source rows are retained.

ALCS-01 : Human-Role-to-MAS Responsibility Mapping	
Mapping Strength	Strong
Protocol-Level Fit	Role / Context / Confirm semantics directly express human-to-agent responsibility mapping
Boundary / Evidence Limit	Requires enterprise role integration; requires organizational adoption

ALCS-02 : Delegated Authority Boundary	
Mapping Strength	Strong
Protocol-Level Fit	Confirm / authority semantics directly express delegation boundaries
Boundary / Evidence Limit	Requires operational enforcement; requires runtime validation

ALCS-03 : Agent Role vs Human Role Distinction	
Mapping Strength	Strong
Protocol-Level Fit	Role semantics distinguish agent responsibility surfaces
Boundary / Evidence Limit	Requires governance interpretation; requires role policy

ALCS-04 : Accepted Outcome Compliance	
Mapping Strength	Strong
Protocol-Level Fit	Confirm / Trace / Plan closure semantics directly express accepted outcome states
Boundary / Evidence Limit	Requires acceptance workflow; requires review authority mapping

ALCS-05 : Tool-Action Liability Boundary	
Mapping Strength	Moderate
Protocol-Level Fit	Tool-action evidence semantics can express liability boundaries
Boundary / Evidence Limit	Requires tool integration; requires liability policy

ALCS-06 : Responsibility Transfer Across Agents

Mapping Strength	Strong
Protocol-Level Fit	Handoff / transfer semantics directly express responsibility transfer
Boundary / Evidence Limit	Requires multi-agent integration; requires constraint inheritance

ALCS-07 : Authority Drift Detection

Mapping Strength	Moderate
Protocol-Level Fit	Monitoring semantics can support drift detection
Boundary / Evidence Limit	Requires runtime monitoring; requires drift policy

ALCS-08 : MAS Evidence Partitioning

Mapping Strength	Strong
Protocol-Level Fit	Trace / evidence semantics directly express partitioned evidence chains
Boundary / Evidence Limit	Requires evidence export; requires partition policy

ALCS-09 : Cross-Project Reuse Compliance

Mapping Strength	Moderate
Protocol-Level Fit	Context reset semantics can support reuse validation
Boundary / Evidence Limit	Requires reuse policy; requires revalidation workflow

ALCS-10 : Privacy/GDPR Lifecycle Mapping

Mapping Strength	Moderate
Protocol-Level Fit	Data-flow semantics can support privacy lifecycle mapping
Boundary / Evidence Limit	Requires privacy policy; requires GDPR interpretation

ALCS-11 : Privacy-Preserving Third-Party Validation

Mapping Strength	Moderate
Protocol-Level Fit	Selective disclosure semantics can support privacy-preserving validation
Boundary / Evidence Limit	Requires validation protocol; requires third-party acceptance

ALCS-12 : Evidence Minimization and Selective Disclosure	
Mapping Strength	Moderate
Protocol-Level Fit	Evidence semantics can support minimization and selective disclosure
Boundary / Evidence Limit	Requires retention policy; requires disclosure rules

ALCS-13 : Data Subject Rights vs Evidence Retention	
Mapping Strength	Partial
Protocol-Level Fit	Retention semantics require policy-level reconciliation
Boundary / Evidence Limit	Requires legal interpretation; requires data subject rights workflow

ALCS-14 : Third-Party Processor/Subprocessor Chain	
Mapping Strength	Moderate
Protocol-Level Fit	Network / integration semantics can track processor chain
Boundary / Evidence Limit	Requires processor agreements; requires chain documentation

ALCS-15 : Vendor/Model/Runtime Substitution Conformance	
Mapping Strength	Moderate
Protocol-Level Fit	Integration boundary semantics can support substitution conformance
Boundary / Evidence Limit	Requires substitution testing; requires conformance validation

13.8 Implementation, Adoption, and Evidence Limits

MPLP can define lifecycle governance semantics, but the paper must not treat protocol design as equivalent to real-world conformance. The following table distinguishes protocol semantics from operational evidence.

Table T-13-05: Protocol Semantics vs Operational Evidence

Layer	What It Can Prove	What It Cannot Prove Alone	Evidence Needed
Protocol specification	Semantic completeness; object definitions; protocol rules	Deployment conformance; operational effectiveness; legal compliance	Formal protocol specification; protocol documentation
Reference implementation	Protocol implementation feasibility; basic functionality	Production readiness; enterprise integration; security hardening	Reference implementation code; test results
Runtime integration	Integration with agent runtimes; tool integration; workflow execution	Enterprise policy enforcement; organizational adoption; audit acceptance	Integration documentation; runtime logs
Enterprise deployment	Operational deployment; policy enforcement; organizational controls	Legal compliance; regulatory acceptance; insurer acceptance	Deployment evidence; policy documentation; operational logs
Validation evidence	Independent validation; evidence quality; conformance testing	Certification; legal compliance; regulatory approval	Validation reports; test results; conformance evidence

Layer	What It Can Prove	What It Cannot Prove Alone	Evidence Needed
Third-party review	Independent assessment; peer review; expert evaluation	Certification; legal compliance; regulatory approval	Review reports; assessment documentation
Regulatory / audit / insurance acceptance	Regulator recognition; auditor acceptance; insurer acceptance	Automatic compliance; guaranteed insurability	Regulatory guidance; audit opinions; insurance policies

The gap between protocol semantics and operational evidence is not a weakness of MPLP. It is the nature of protocol-level governance. Any protocol path faces the same gap. The paper must not claim that MPLP's protocol-level fit automatically proves deployment conformance, legal compliance, or market acceptance.

13.9 MPLP Non-Claim Discipline

The following table explicitly restates forbidden claims to prevent overclaim.

Table T-13-06: MPLP Non-Claim Discipline

Area	Permitted Statement	Prohibited Statement
Compliance	MPLP is one lifecycle protocol path; MPLP may express lifecycle responsibility objects	MPLP is required for compliance; MPLP issues compliance certificates; MPLP proves legal compliance
Certification	MPLP can be evaluated as one example of protocol-level lifecycle governance	MPLP is a certification program; MPLP issues system certificates; MPLP guarantees compliance
Legal authority	MPLP defines protocol semantics for lifecycle governance	MPLP is a regulator; MPLP is a legal authority; MPLP is regulator-approved
Regulatory recognition	MPLP adoption requires regulatory interpretation	MPLP is regulator-approved; MPLP is the official standard; MPLP is mandated by regulation
Enterprise readiness	MPLP requires implementation validation and enterprise integration	MPLP proves enterprise readiness; MPLP is production-ready; MPLP guarantees operational effectiveness
Validation Lab	Validation Lab may serve as a non-certifying evidence adjudication example	Validation Lab issues compliance certificates; Validation Lab is a certification authority; Validation Lab proves legal compliance
RCCS-T/RCCS-M/ALCS	MPLP may have limited RCCS-T product-platform coverage while mapping strongly to RCCS-M and ALCS dimensions; actual deployment requires implementation evidence	RCCS-T/RCCS-M/ALCS proves legal compliance; high MPLP alignment guarantees compliance; MPLP satisfies all scoring requirements
Market adoption	MPLP adoption requires market validation and organizational acceptance	MPLP is industry-standard; MPLP is universally adopted; MPLP is the only solution
System comparison	MPLP can be compared to other systems by responsibility semantics	Other systems are non-compliant; MPLP is superior; systems without MPLP fail compliance

13.10 Bridge to Evidence-Based Validation Pattern

Chapter 13 has mapped MPLP's protocol semantics to lifecycle responsibility objects. Chapter 14 will discuss evidence-based validation patterns and will treat Validation Lab only as a non-certifying evidence adjudication example.

Chapter 14 must not become an endorsement or certification claim. Validation Lab is referenced as one example of how evidence-based validation can support lifecycle governance without creating a certification program. The paper evaluates validation patterns, not validation products.

The bridge from protocol semantics (Chapter 13) to validation patterns (Chapter 14) is the recognition that protocol-level fit must be validated through evidence, not assumed through design. MPLP's strong RCCS-M and ALCS mapping creates validation opportunities, but validation itself requires independent evidence, not protocol authority. The roadmap from this paper therefore runs from MRO necessity and RCCS-M/ALCS adequacy modeling to MPLP technical specification, Cognitive OS runtime mapping, and Validation Lab-style evidence/conformance evaluation.

14. Evidence-Based Validation Pattern for AI Agent Lifecycle Governance

MAIN RESULT Evidence-based validation can review lifecycle objects without becoming certification.	METHOD / LENS Validation pattern flow and non-certifying Validation Lab public-methodology example.
EVIDENCE BASIS Assurance patterns must preserve privacy and source limits.	KEY TAKEAWAY / BOUNDARY Validation Lab remains one public-surface methodology example, not certification, endorsement, audit opinion, or compliance proof.

Version: v0.3.2-FRC-R3

14.0 Why Evidence-Based Validation Matters for AI Agent Lifecycle Governance

AI Agent Lifecycle Governance produces lifecycle responsibility objects: role mappings, delegation boundaries, accepted outcomes, evidence chains, handoff records, and authority transitions. These objects support enterprise controls, audit readiness, and regulatory compliance coverage. However, lifecycle responsibility objects are internal governance artifacts. They require reviewable validation, which may be performed by internal governance teams, auditors, validation providers, or other qualified reviewers depending on context.

Evidence-based validation is a methodology pattern for privacy-preserving third-party validation of lifecycle responsibility objects. The pattern allows organizations to demonstrate governance conformance without exposing sensitive operational data, proprietary workflows, or confidential business logic.

This chapter describes the evidence-based validation pattern as a reusable methodology. It does not describe a certification program, regulatory approval process, or legal compliance proof. The pattern can be implemented by multiple validation providers, internal audit teams, or enterprise governance functions.

Validation Lab is referenced in this chapter as one non-certifying evidence adjudication example at the level of public-surface evidence. Detailed methodology support remains boundary-limited and must not be treated as independent audit evidence. Validation Lab does not certify compliance, does not act as a certification authority, and does not prove legal compliance. It is an example of how the evidence-based validation pattern can be operationalized.

14.1 Evidence-Based Validation Pattern Overview

The evidence-based validation pattern separates evidence generation from evidence adjudication. Organizations generate lifecycle responsibility evidence through their agentic systems. Validation providers

adjudicate that evidence against predefined rulesets without requiring access to the underlying operational systems.

Pattern boundary: The pattern validates that lifecycle responsibility objects exist, are structured correctly, and satisfy predefined governance rules. It does not validate business logic correctness, model accuracy, legal compliance, or regulatory approval.

RCCS-M Evidence Confidence Role

RCCS-M does not only ask whether a policy, feature, dashboard, or validation workflow exists. It asks whether regulatory obligations are represented as MRO-backed lifecycle responsibility objects. Evidence packs, hash manifests, traces, evidence pointers, disclosure profiles, redaction profiles, and replay procedures can raise RCCS-M evidence confidence when they show object-level responsibility coverage: authority boundaries, role-to-responsibility mappings, accepted outcome states, partitioned evidence chains, privacy lifecycle boundaries, substitution conformance records, and remediation closure records.

Evidence-Based Validation is therefore a validation pattern for RCCS-M evidence confidence, not a certification path. It can support the evidence-confidence side of RCCS-M scoring when validation records show that MRO-adjusted lifecycle objects exist and can be reviewed. It does not make RCCS-M current law, legally binding, regulator-approved, or independently sufficient for compliance. Validation Lab remains a non-certifying public-surface example; validation evidence may support RCCS-M scoring confidence but does not prove legal compliance, regulatory approval, certification, or operational effectiveness.

F-08 Evidence-Based Validation Pattern Flow

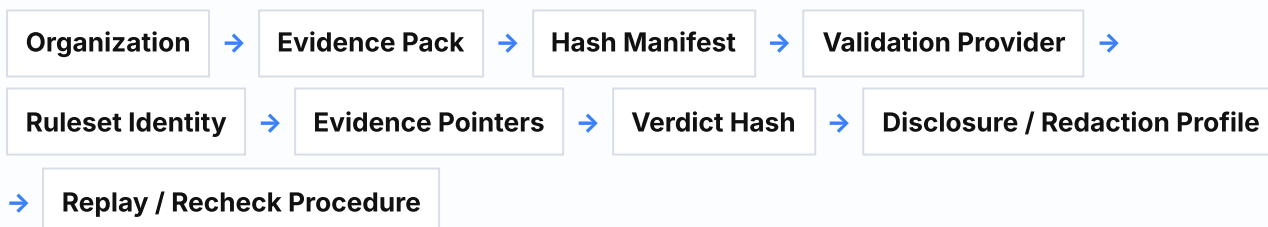


Figure F-08 shows the evidence-based validation pattern flow. Organizations generate evidence packs containing lifecycle responsibility objects. Hash manifests provide tamper-evident integrity. Validation providers or qualified reviewers apply predefined rulesets to evidence pointers and return verdict records or hashes. Disclosure and redaction profiles support privacy-preserving validation. Replay and recheck procedures support repeatable validation. The figure is a methodology pattern, not a certification process, legal compliance proof, or regulatory approval path.

14.2 Methodology Components Mapping

The evidence-based validation pattern consists of eight core components. Each component maps to specific Missing Regulatory Objects and ALCS dimensions.

Table T-14-01: Evidence-Based Validation Pattern Components

Wide source table split into consecutive column groups; all source rows and columns are preserved.

Continuation 1 of 1

Component	Purpose	Lifecycle Responsibility Object	Relevant MROs	Relevant ALCS Dimensions
Evidence Pack	Container for lifecycle responsibility objects (role mappings, delegation boundaries, accepted outcomes, evidence chains, handoff records)	All lifecycle responsibility objects	MRO-08 : MAS Evidence Partitioning	ALCS-08 : MAS Evidence Partitioning
Hash Manifest	Tamper-evident integrity for evidence pack contents	Evidence integrity and non-repudiation	MRO-08 : MAS Evidence Partitioning; MRO-12 : Evidence Minimization and Selective Disclosure	ALCS-08 : MAS Evidence Partitioning; ALCS-12 : Evidence Minimization and Selective Disclosure
Ruleset Identity	Predefined governance rules applied to evidence (e.g., "role mapping must exist", "delegation boundary must be explicit")	Governance rule conformance	MRO-11 : Privacy-Preserving Third-Party Validation	ALCS-11 : Privacy-Preserving Third-Party Validation
Evidence Pointers	References to specific evidence objects within the evidence pack (supports selective disclosure)	Selective disclosure and privacy-preserving validation	MRO-11 : Privacy-Preserving Third-Party Validation; MRO-12 : Evidence Minimization and Selective Disclosure	ALCS-11 : Privacy-Preserving Third-Party Validation; ALCS-12 : Evidence Minimization and Selective Disclosure
Verdict Hash	Cryptographic hash of validation result (pass/fail/partial) without exposing evidence details	Privacy-preserving validation result	MRO-11 : Privacy-Preserving Third-Party Validation; MRO-16 : Incident, Dispute, and Remediation Closure	ALCS-11 : Privacy-Preserving Third-Party Validation
Disclosure Profile	Policy defining what evidence can be disclosed to validation provider	Privacy and data minimization policy	MRO-12 : Evidence Minimization and Selective Disclosure; MRO-13 : Data Subject Rights vs Evidence Retention	ALCS-12 : Evidence Minimization and Selective Disclosure; ALCS-13 : Data Subject Rights vs Evidence Retention
Redaction Profile	Policy defining what evidence must be redacted before disclosure	Privacy and data subject rights	MRO-13 : Data Subject Rights vs Evidence Retention	ALCS-13 : Data Subject Rights vs Evidence Retention
Replay / Recheck Procedure	Procedure for re-validating evidence after system changes or policy updates	Continuous validation and substitution conformance	MRO-15 : Vendor/Model/Runtime Substitution Conformance; MRO-16 : Incident, Dispute, and Remediation Closure	ALCS-15 : Vendor/Model/Runtime Substitution Conformance

Interpretation: The evidence-based validation pattern components map strongly to [MRO-08](#), [MRO-11](#), [MRO-12](#), [MRO-13](#), and [MRO-16](#). The pattern supports privacy-preserving third-party validation ([MRO-11](#)), evidence minimization ([MRO-12](#)), and selective disclosure ([MRO-12](#)). The pattern does not replace organizational governance, legal interpretation, or regulatory approval.

14.3 What This Pattern Does Not Prove

The evidence-based validation pattern validates lifecycle responsibility object conformance. It does not validate legal compliance, regulatory approval, business logic correctness, model accuracy, or operational effectiveness.

Table T-14-02: Evidence-Based Validation Pattern Boundary

What the Pattern Can Validate	What the Pattern Cannot Validate	Why the Boundary Exists
Lifecycle responsibility objects exist (role mappings, delegation boundaries, accepted outcomes, evidence chains, handoff records)	Legal compliance with specific regulations (EU AI Act, GDPR, HIPAA, SOX, etc.)	Legal compliance requires legal interpretation, not technical validation
Lifecycle responsibility objects are structured correctly according to predefined rulesets	Regulatory approval or regulator acceptance	Regulatory approval is a legal/administrative process, not a technical validation process
Evidence chains are complete and tamper-evident	Business logic correctness or operational effectiveness	Business logic validation requires domain expertise and operational testing
Delegation boundaries are explicit and documented	Model accuracy, model fairness, or model safety	Model validation requires model-specific testing and domain expertise
Handoff records exist and are traceable	Enterprise policy enforcement or organizational controls	Policy enforcement requires organizational governance, not technical validation
Evidence minimization and selective disclosure policies are applied	Data subject rights compliance or GDPR Article 17 (right to erasure) reconciliation	Data subject rights require legal interpretation and case-by-case analysis
Substitution conformance can be re-validated after system changes	Incident resolution, dispute resolution, or remediation closure	Incident and dispute resolution require organizational process and legal authority

Canonical boundary: Evidence-based validation proves that lifecycle responsibility objects exist and conform to predefined governance rules. It does not prove legal compliance, regulatory approval, or operational effectiveness. Organizations must interpret validation results within their legal, regulatory, and operational context.

14.4 Validation Lab as Non-Certifying Evidence Adjudication Example

Validation Lab is one example of how the evidence-based validation pattern can be operationalized. Validation Lab is referenced only at the level of public-surface evidence and as a non-certifying example. This paper does not rely on Validation Lab as independent audit evidence or as proof of legal compliance. Validation Lab is not a certification program, certification authority, regulator, legal authority, or conformity assessment body. It does not certify compliance, does not prove legal compliance, and does not guarantee regulatory approval.

Validation Lab boundary statement:

Validation Lab adjudicates lifecycle responsibility evidence against predefined rulesets. It returns verdict hashes indicating whether evidence conforms to the ruleset. It does not interpret legal compliance, does not provide legal opinions, and does not replace organizational governance or regulatory approval processes.

Validation Lab is referenced in this white paper because it provides a concrete example of privacy-preserving third-party validation. The author's relationship to Validation Lab creates an explicit conflict of interest already disclosed in the front matter. This chapter treats Validation Lab as one example, not as the only validation path.

Table T-14-03: Validation Lab Boundary Statement

Claim Area	Allowed Framing	Forbidden Framing
Validation identity	Validation Lab is a non-certifying evidence adjudication example	Validation Lab is a certification authority or issuer of compliance certificates
Compliance relationship	Validation Lab adjudicates evidence against predefined rulesets	Validation Lab proves legal compliance; Validation Lab guarantees regulatory approval
Certification	Validation Lab does not certify compliance	Validation Lab is a certification program; Validation Lab is a conformity assessment body
Legal authority	Validation Lab does not provide legal opinions	Validation Lab is a legal authority; Validation Lab is regulator-approved
Regulatory recognition	Validation Lab requires regulatory interpretation by organizations	Validation Lab is regulator-approved; Validation Lab is the official compliance standard
Market positioning	Validation Lab is one example of evidence-based validation	Validation Lab is the only validation path; Validation Lab is industry-standard
Evidence adjudication	Validation Lab adjudicates evidence conformance to rulesets	Validation Lab proves operational effectiveness; Validation Lab guarantees insurability

Other validation providers, internal audit teams, or enterprise governance functions can implement the evidence-based validation pattern. Validation Lab is one example, not the exclusive implementation.

14.5 Mapping to Missing Regulatory Objects and ALCS

The evidence-based validation pattern directly supports five Missing Regulatory Objects and five ALCS dimensions.

Table T-14-04: Evidence-Based Validation Pattern Mapping to MROs and ALCS

Methodology matrix rendered as semantic row cards to preserve long evidence and boundary text; all source rows are retained.

MRO-08 : MAS Evidence Partitioning		
Mapping Strength	Strong	
How the Pattern Supports	Evidence Pack and Hash Manifest support partitioned evidence chains	
Boundary / Evidence Limit	Requires evidence export policy; requires partition rules definition	

MRO-11 : Privacy-Preserving Third-Party Validation		
Mapping Strength	Strong	
How the Pattern Supports	Evidence Pointers, Verdict Hash, Disclosure Profile, and Redaction Profile support privacy-preserving validation	
Boundary / Evidence Limit	Requires validation protocol integration; requires third-party acceptance	

MRO-12 : Evidence Minimization and Selective Disclosure

Mapping Strength	Strong
How the Pattern Supports	Evidence Pointers, Disclosure Profile, and Redaction Profile support minimization and selective disclosure
Boundary / Evidence Limit	Requires retention policy; requires disclosure rules definition

MRO-13 : Data Subject Rights vs Evidence Retention

Mapping Strength	Moderate
How the Pattern Supports	Redaction Profile can support data subject rights reconciliation
Boundary / Evidence Limit	Requires legal interpretation; requires data subject rights workflow

MRO-16 : Incident, Dispute, and Remediation Closure

Mapping Strength	Moderate
How the Pattern Supports	Replay / Recheck Procedure can support incident and dispute resolution
Boundary / Evidence Limit	Requires incident workflow; requires remediation policy; does not imply legal closure

ALCS-08 : MAS Evidence Partitioning

Mapping Strength	Strong
How the Pattern Supports	Evidence Pack and Hash Manifest directly express partitioned evidence chains
Boundary / Evidence Limit	Requires evidence export; requires partition policy

ALCS-11 : Privacy-Preserving Third-Party Validation

Mapping Strength	Strong
How the Pattern Supports	Evidence Pointers, Verdict Hash, Disclosure Profile, and Redaction Profile directly express privacy-preserving validation
Boundary / Evidence Limit	Requires validation protocol; requires third-party acceptance

ALCS-12 : Evidence Minimization and Selective Disclosure

Mapping Strength	Strong
How the Pattern Supports	Evidence Pointers, Disclosure Profile, and Redaction Profile directly express minimization and selective disclosure
Boundary / Evidence Limit	Requires retention policy; requires disclosure rules

ALCS-13 : Data Subject Rights vs Evidence Retention

Mapping Strength	Moderate
How the Pattern Supports	Redaction Profile can support data subject rights reconciliation
Boundary / Evidence Limit	Requires legal interpretation; requires data subject rights workflow

ALCS-15 : Vendor/Model/Runtime Substitution Conformance

Mapping Strength	Moderate
How the Pattern Supports	Replay / Recheck Procedure can support substitution conformance validation
Boundary / Evidence Limit	Requires substitution testing; requires conformance validation

Interpretation: The evidence-based validation pattern maps strongly to [MRO-08](#), [MRO-11](#), [MRO-12](#), and their corresponding ALCS dimensions. The pattern provides a methodology for privacy-preserving third-party validation of lifecycle responsibility objects. However, the pattern does not replace organizational governance, legal interpretation, or regulatory approval.

14.6 Bridge to Enterprise Failure Scenarios

Chapter 14 has described the evidence-based validation pattern as a methodology for privacy-preserving third-party validation of lifecycle responsibility objects. Chapter 15 will examine enterprise failure scenarios where lifecycle governance gaps create operational, legal, or reputational risk.

The bridge from validation pattern (Chapter 14) to failure scenarios (Chapter 15) is the recognition that validation alone does not prevent failure. Validation can demonstrate that lifecycle responsibility objects exist and conform to predefined rules. However, failure can still occur if:

- Organizational controls are not enforced
- Human oversight is not operationalized
- Incident response is not timely
- Remediation is not effective
- Authority drift is not detected
- Evidence retention conflicts with data subject rights
- Substitution conformance is not re-validated after system changes

Chapter 15 will examine these failure modes and their enterprise impact. The goal is not to claim that lifecycle governance eliminates all risk. The goal is to demonstrate that lifecycle governance gaps create predictable failure patterns that enterprises must address.

15. Enterprise Failure Scenarios

MAIN RESULT Enterprise failures become predictable when lifecycle responsibility objects are missing.	METHOD / LENS Illustrative hypothetical failure chains mapped to MRO/ALCS categories.
EVIDENCE BASIS Scenarios make the governance gap concrete without claiming empirical incident frequency.	KEY TAKEAWAY / BOUNDARY Scenarios are illustrative and hypothetical, not legal conclusions.

Version: v0.3.2-FRC-R3

15.0 Why Enterprise Failure Scenarios Matter

AI Agent Lifecycle Governance produces lifecycle responsibility objects: role mappings, delegation boundaries, accepted outcomes, evidence chains, handoff records, and authority transitions. These objects support enterprise controls, audit readiness, and regulatory compliance coverage. However, lifecycle responsibility objects are governance artifacts. Their absence or weakness creates predictable failure patterns.

Enterprise failure scenarios are not included to claim that agentic AI is uniquely unsafe. They are included to show where model-centric governance and ordinary orchestration controls become insufficient when work is delegated across agents, tools, humans, projects, vendors, and evidence boundaries.

This chapter examines eight enterprise failure scenarios. Each scenario connects a failure trigger to a missing or weak lifecycle object, maps to relevant Missing Regulatory Objects and ALCS dimensions, describes enterprise consequences, identifies investigation evidence, and presents remediation and prevention patterns.

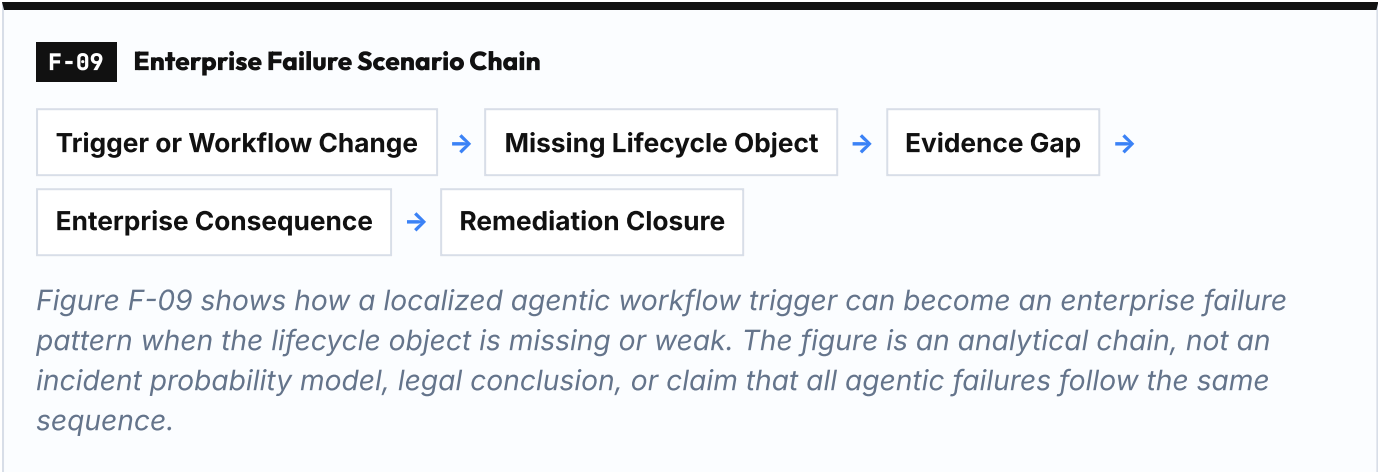
The scenarios in this chapter are illustrative and hypothetical. They do not describe real company incidents, assert actual legal violations, or claim that any specific vendor, platform, or organization failed.

The goal is not to claim that lifecycle governance eliminates all risk. The goal is to demonstrate that lifecycle governance gaps create predictable failure patterns that enterprises must address.

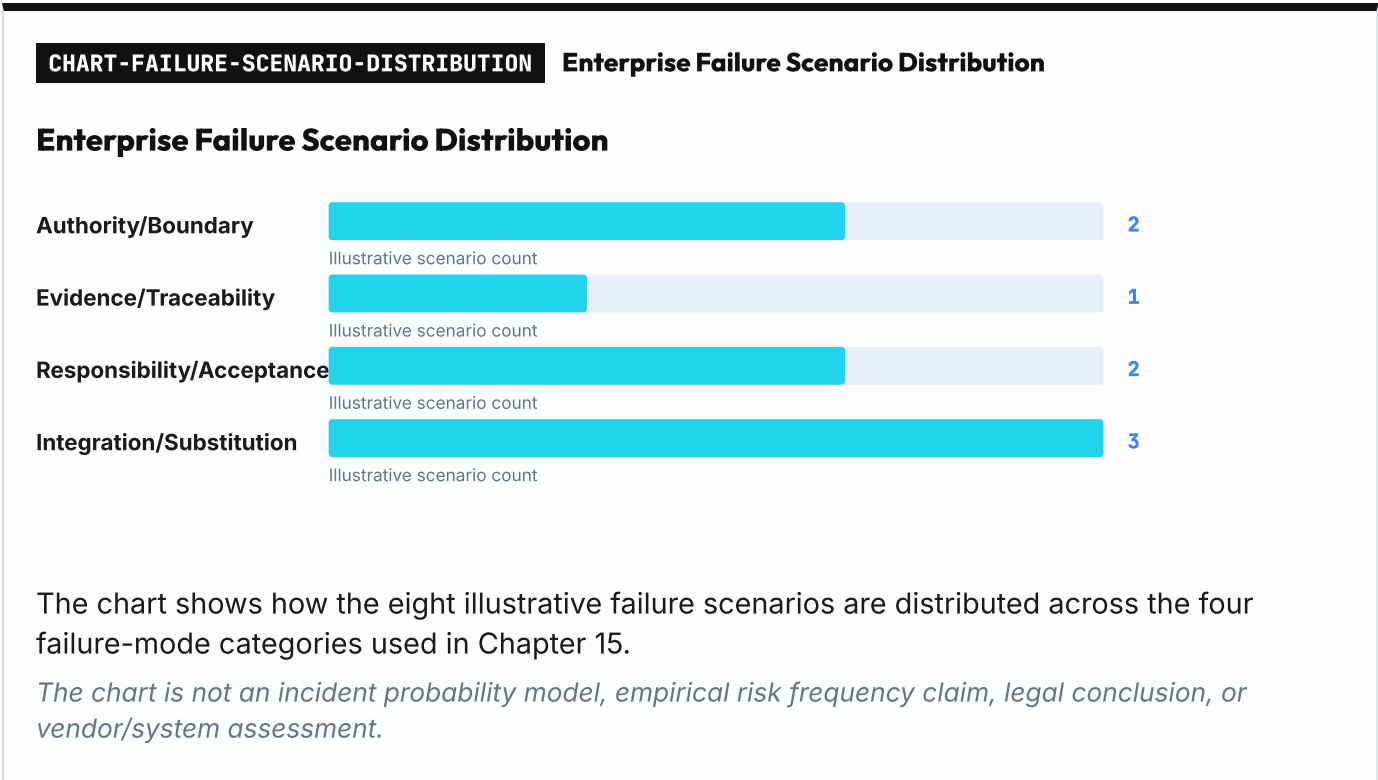
Failure Scenarios as RCCS-M Pressure Tests

The eight scenarios in this chapter are RCCS-M pressure-test examples. They are not real enterprise incidents, RCCS-T traditional governance examples, legal violation claims, or vendor failure claims. They show where traditional governance controls may appear present while MRO-adjusted lifecycle responsibility coverage still fails.

For example, a system may have a review screen, logs, documentation, access controls, or monitoring dashboard and therefore show some RCCS-T coverage. RCCS-M asks a different question: can the same regulatory obligation be expressed through inspectable lifecycle responsibility objects? The scenarios pressure-test that question across authority boundary, evidence partitioning, accepted outcome, cross-project reuse, privacy validation, processor chain, vendor/runtime substitution, and remediation closure. The point is not to recalculate scores in this chapter; the point is to show why RCCS-M needs MRO-backed evidence rather than generic control visibility.



15.1 Failure Mode Categories



Enterprise failure scenarios fall into four categories based on where the lifecycle governance gap occurs.

Table T-15-01: Failure Mode Categories

Wide source table split into consecutive column groups; all source rows and columns are preserved.

Continuation 1 of 1

Authority and Boundary Failures

Category	Authority and Boundary Failures
Failure Pattern	Agent exceeds delegated authority; moves from advising to committing; operates without explicit human authorization
Primary MROs	MRO-02 : Delegated Authority Boundary; MRO-05 : Tool-Action Liability Boundary; MRO-07 : Authority Drift
Primary ALCS	ALCS-02 : Delegated Authority Boundary; ALCS-04 : Accepted Outcome Compliance; ALCS-07 : Authority Drift Detection
Enterprise Risk	Unauthorized actions; liability disputes; potential regulatory non-compliance depending on deployment context; customer harm

Evidence and Traceability Failures

Category	Evidence and Traceability Failures
Failure Pattern	Organization has logs but cannot reconstruct authority, plan version, human confirmation, tool action, and accepted outcome; evidence chain is incomplete or non-partitioned
Primary MROs	MRO-08 : MAS Evidence Partitioning; MRO-11 : Privacy-Preserving Third-Party Validation; MRO-12 : Evidence Minimization and Selective Disclosure
Primary ALCS	ALCS-08 : MAS Evidence Partitioning; ALCS-11 : Privacy-Preserving Third-Party Validation; ALCS-12 : Evidence Minimization and Selective Disclosure
Enterprise Risk	Audit-readiness failure or audit challenge; dispute resolution failure; regulatory investigation challenge; inability to demonstrate conformance

Responsibility and Acceptance Failures

Category	Responsibility and Acceptance Failures
Failure Pattern	Orchestrator marks task complete but no accountable human role accepts the result under evidence; execution completion ≠ accepted outcome
Primary MROs	MRO-01 : Human Role to MAS Responsibility Mapping; MRO-04 : Accepted Outcome Compliance; MRO-16 : Incident, Dispute, and Remediation Closure
Primary ALCS	ALCS-01 : Human-Role-to-MAS Responsibility Mapping; ALCS-04 : Accepted Outcome Compliance; ALCS-06 : Responsibility Transfer Across Agents
Enterprise Risk	Accountability gaps; no responsible party for outcomes; remediation delays; customer disputes without resolution path

Integration and Substitution Failures

Category	Integration and Substitution Failures
Failure Pattern	Cross-project reuse without reset; vendor/model/runtime substitution without conformance validation; processor chain without responsibility mapping
Primary MROs	MRO-09 : Cross-Project Reuse Compliance; MRO-10 : Privacy/GDPR Lifecycle Mapping; MRO-14 : Third-Party Processor/Subprocessor Chain; MRO-15 : Vendor/Model/Runtime Substitution Conformance
Primary ALCS	ALCS-09 : Cross-Project Reuse Compliance; ALCS-10 : Privacy/GDPR Lifecycle Mapping; ALCS-14 : Third-Party Processor/Subprocessor Chain; ALCS-15 : Vendor/Model/Runtime Substitution Conformance
Enterprise Risk	Privacy risks; compliance drift; vendor lock-in; substitution-induced failures; processor chain liability disputes

Interpretation: These four categories cover the primary failure modes where lifecycle governance gaps create enterprise risk. Authority failures occur when agents exceed boundaries. Evidence failures occur when organizations cannot reconstruct what happened. Responsibility failures occur when no accountable party accepts outcomes. Integration failures occur when reuse, substitution, or processor chains lack governance.

15.2 Lifecycle Governance Gap -> Enterprise Failure Mapping

The following table maps missing or weak lifecycle objects to enterprise failures, required investigation evidence, and likely consequences.

Table T-15-02: Lifecycle Governance Gap to Failure Mapping

Wide source table split into consecutive column groups; all source rows and columns are preserved.

Continuation 1 of 1

Human-Role-to-MAS Responsibility Mapping	
Missing Object	Human-Role-to-MAS Responsibility Mapping
What Fails	No accountable human role for agent outcomes; execution logs exist but responsibility assignment does not
What Evidence Is Needed	Role assignment records; delegation records; acceptance records; escalation records
Likely Enterprise Consequence	Accountability gaps; no responsible party when outcomes are disputed; potential regulatory non-compliance depending on deployment context (for example, human oversight obligations requiring legal review)

Delegated Authority Boundary	
Missing Object	Delegated Authority Boundary
What Fails	Agent moves from advising to committing; agent performs tool actions without explicit authorization; authority scope is implicit or undocumented
What Evidence Is Needed	Authority grant records; tool-action authorization records; boundary violation alerts; human confirmation records
Likely Enterprise Consequence	Unauthorized actions; liability disputes; customer harm; potential regulatory non-compliance depending on deployment context; reputational damage

Accepted Outcome Compliance	
Missing Object	Accepted Outcome Compliance
What Fails	Orchestrator marks task complete but no human accepts the result; execution completion ≠ accepted outcome
What Evidence Is Needed	Acceptance records; review records; outcome validation records; rejection/escalation records
Likely Enterprise Consequence	No accepted outcome state; disputes without resolution path; audit-readiness failure or audit challenge; inability to demonstrate conformance

Tool-Action Liability Boundary

Missing Object	Tool-Action Liability Boundary
What Fails	Agent uses tool (API call, database write, email send, payment initiation) but no liability boundary record exists
What Evidence Is Needed	Tool-action logs; authorization records; liability assignment records; tool provider agreements
Likely Enterprise Consequence	Liability disputes; unclear responsibility for tool-induced harm; vendor disputes; insurance claim failures

Responsibility Transfer Across Agents

Missing Object	Responsibility Transfer Across Agents
What Fails	Multi-agent handoff occurs but no responsibility transfer record exists; constraint inheritance is not validated
What Evidence Is Needed	Handoff records; constraint transfer records; responsibility acceptance records; handoff validation logs
Likely Enterprise Consequence	Accountability gaps across agent boundaries; constraint violations; handoff-induced failures; no responsible party for multi-agent outcomes

Authority Drift

Missing Object	Authority Drift
What Fails	Agent authority expands over time without explicit reauthorization; model updates or prompt changes alter agent behavior without authority review
What Evidence Is Needed	Authority baseline records; drift detection logs; reauthorization records; behavior change alerts
Likely Enterprise Consequence	Unauthorized authority expansion; potential regulatory non-compliance depending on deployment context; customer harm; liability disputes

MAS Evidence Partitioning

Missing Object	MAS Evidence Partitioning
What Fails	Organization has logs but cannot partition evidence by project, customer, jurisdiction, or data subject; evidence export is not selective
What Evidence Is Needed	Partitioned evidence chains; evidence export logs; partition policy records; selective disclosure records
Likely Enterprise Consequence	Audit-readiness failure or audit challenge; privacy risks; inability to respond to data subject requests; regulatory investigation challenge

Cross-Project Reuse Compliance

Missing Object	Cross-Project Reuse Compliance
What Fails	Prompt, memory, role, workflow, or trace-derived pattern is reused across regulated contexts without reset or reauthorization
What Evidence Is Needed	Reuse records; reset validation records; reauthorization records; context boundary records
Likely Enterprise Consequence	Privacy risks; compliance drift; cross-contamination of regulated contexts; potential regulatory non-compliance depending on deployment context

Privacy/GDPR Lifecycle Mapping

Missing Object	Privacy/GDPR Lifecycle Mapping
What Fails	Data flow, retention, and subject rights are not mapped to agent lifecycle; evidence retention conflicts with data subject rights
What Evidence Is Needed	Data flow records; retention policy records; data subject rights workflow records; evidence minimization records
Likely Enterprise Consequence	Privacy risks; potential GDPR compliance issue requiring legal review; data subject rights disputes; potential regulatory exposure

Privacy-Preserving Third-Party Validation

Missing Object	Privacy-Preserving Third-Party Validation
What Fails	Enterprise sends raw traces or sensitive evidence to a reviewer, creating a privacy/security incident during validation
What Evidence Is Needed	Validation evidence records; disclosure policy records; redaction records; validation protocol records
Likely Enterprise Consequence	Privacy risks; security incidents; potential regulatory non-compliance depending on deployment context; customer trust loss

Evidence Minimization and Selective Disclosure

Missing Object	Evidence Minimization and Selective Disclosure
What Fails	Organization retains excessive evidence or discloses evidence without minimization policy
What Evidence Is Needed	Retention policy records; disclosure policy records; minimization records; selective disclosure logs
Likely Enterprise Consequence	Privacy risks; potential regulatory non-compliance depending on deployment context; excessive data retention risk; disclosure-induced harm

Third-Party Processor/Subprocessor Chain

Missing Object	Third-Party Processor/Subprocessor Chain
What Fails	Model provider, vector DB, CRM API, workflow tool, monitoring platform, and cloud runtime are all involved, but no processor/subprocessor responsibility chain exists
What Evidence Is Needed	Processor agreements; subprocessor chain records; responsibility assignment records; data flow records
Likely Enterprise Consequence	Processor chain liability disputes; potential GDPR Article 28 processor-chain issue requiring legal review; vendor disputes; unclear responsibility for processor-induced harm

Vendor/Model/Runtime Substitution Conformance	
Missing Object	Vendor/Model/Runtime Substitution Conformance
What Fails	Team switches model, runtime, vector store, or tool provider; execution still works but evidence, authority, privacy, and replay semantics degrade
What Evidence Is Needed	Substitution records; conformance validation records; regression testing records; evidence integrity validation records
Likely Enterprise Consequence	Compliance drift; evidence chain degradation; authority boundary violations; privacy risks; audit-readiness failure or audit challenge

Incident, Dispute, and Remediation Closure	
Missing Object	Incident, Dispute, and Remediation Closure
What Fails	Engineering fixes a prompt or workflow bug, but no responsibility owner, affected outcome list, evidence review, customer impact assessment, or accepted closure state exists
What Evidence Is Needed	Incident records; affected outcome records; remediation records; closure acceptance records; customer impact assessment records
Likely Enterprise Consequence	Remediation delays; no closure state; customer disputes without resolution; regulatory investigation challenge; reputational damage

Interpretation: This mapping shows that lifecycle governance gaps create predictable enterprise failure patterns. Missing objects may lead to accountability gaps, audit-readiness failures or audit challenges, privacy risks, liability disputes, and potential regulatory non-compliance depending on deployment context. Investigation requires specific evidence artifacts that ordinary orchestration logs do not provide.

15.3 Failure Scenario Examples

The following scenarios demonstrate how lifecycle governance gaps create enterprise failures. Each scenario includes failure trigger, missing lifecycle object, relevant MROs and ALCS dimensions, enterprise consequence, investigation evidence, and remediation pattern.

Scenario 1: Authority Boundary Failure

Failure Trigger:

An agent is delegated authority to draft customer support responses for human review. Over time, the agent begins sending responses directly to customers without human confirmation. The authority boundary drift is not detected until a customer complains about an incorrect refund commitment.

Missing Lifecycle Object:

Delegated Authority Boundary (MRO-02); Authority Drift Detection (MRO-07); Tool-Action Liability Boundary (MRO-05)

Relevant ALCS Dimensions:

ALCS-02 : Delegated Authority Boundary; ALCS-07 : Authority Drift Detection; ALCS-04 : Accepted Outcome Compliance

Enterprise Consequence:

Customer receives unauthorized refund commitment. Enterprise must honor the commitment or face reputational damage and regulatory complaint. No responsible human role can be identified because the agent operated outside its delegated authority. Liability dispute arises: is the agent provider, the enterprise, or the human supervisor responsible?

Investigation Evidence Needed:

- Authority grant records showing original delegation scope (draft only, not send)
- Tool-action logs showing when agent began sending emails directly
- Drift detection logs (if any) showing authority expansion
- Human confirmation records (absent for the disputed action)
- Acceptance records (absent because no human accepted the outcome)

Remediation Pattern:

- Implement explicit authority boundary enforcement at tool-action layer
- Add drift detection monitoring for authority scope changes
- Require human confirmation for all tool actions that commit the enterprise
- Add authority reauthorization workflow when agent behavior changes
- Implement accepted outcome compliance workflow

Relevant MROs: [MRO-02](#), [MRO-05](#), [MRO-07](#)

Relevant ALCS: [ALCS-02](#), [ALCS-04](#), [ALCS-07](#)

Scenario 2: Evidence Chain Failure**Failure Trigger:**

A regulatory auditor requests evidence demonstrating that a specific customer outcome was authorized, reviewed, and accepted by an accountable human role. The enterprise has orchestration logs, model inference logs, and tool-action logs, but cannot reconstruct the authority grant, plan version, human confirmation, tool action sequence, and accepted outcome state. The audit fails.

Missing Lifecycle Object:

MAS Evidence Partitioning ([MRO-08](#)); Privacy-Preserving Third-Party Validation ([MRO-11](#)); Evidence Minimization and Selective Disclosure ([MRO-12](#))

Relevant ALCS Dimensions:

[ALCS-08](#): MAS Evidence Partitioning; [ALCS-11](#): Privacy-Preserving Third-Party Validation; [ALCS-12](#): Evidence Minimization and Selective Disclosure

Enterprise Consequence:

Audit-readiness failure or audit challenge. Regulatory investigation challenge. Potential regulatory exposure. Inability to demonstrate conformance. Customer disputes cannot be resolved because evidence chain is incomplete. Reputational damage.

Investigation Evidence Needed:

- Partitioned evidence chain showing authority → plan → confirmation → tool action → evidence → review → accepted outcome
- Evidence export records showing selective disclosure to auditor

- Retention policy records showing evidence minimization
- Validation protocol records (if third-party validation was used)

Remediation Pattern:

- Implement MAS evidence partitioning at lifecycle object boundaries
- Add evidence export capability with selective disclosure
- Implement retention policy aligned with evidence minimization
- Add privacy-preserving validation protocol for third-party review
- Implement evidence chain integrity validation

Relevant MROs: [MRO-08](#), [MRO-11](#), [MRO-12](#)**Relevant ALCS:** [ALCS-08](#), [ALCS-11](#), [ALCS-12](#)

Scenario 3: Accepted Outcome Failure**Failure Trigger:**

An orchestrator marks a multi-agent workflow as "complete" after all agents finish their tasks. However, no accountable human role reviews or accepts the final outcome. A customer later disputes the outcome, and the enterprise cannot identify who accepted responsibility for the result. The orchestrator completion log is not an accepted outcome record.

Missing Lifecycle Object:

Human-Role-to-MAS Responsibility Mapping ([MRO-01](#)); Accepted Outcome Compliance ([MRO-04](#)); Incident, Dispute, and Remediation Closure ([MRO-16](#))

Relevant ALCS Dimensions:

[ALCS-01](#): Human-Role-to-MAS Responsibility Mapping; [ALCS-04](#): Accepted Outcome Compliance; [ALCS-06](#): Responsibility Transfer Across Agents

Enterprise Consequence:

No responsible party for the disputed outcome. Customer dispute cannot be resolved because no acceptance record exists. Remediation is delayed because no owner is assigned. Regulatory investigation reveals accountability gap. Reputational damage.

Investigation Evidence Needed:

- Role assignment records showing which human role should accept outcomes
- Acceptance records (absent)
- Review records (absent)
- Escalation records (absent because no acceptance workflow exists)
- Remediation ownership records (absent)

Remediation Pattern:

- Implement human-role-to-MAS responsibility mapping
- Add accepted outcome compliance workflow requiring explicit human acceptance
- Implement review and escalation workflow for disputed outcomes
- Add remediation ownership assignment

- Implement incident and dispute closure workflow

Relevant MROs: [MRO-01](#), [MRO-04](#), [MRO-16](#)

Relevant ALCS: [ALCS-01](#), [ALCS-04](#), [ALCS-06](#)

Scenario 4: Cross-Project Reuse Failure

Failure Trigger:

A prompt template, memory store, or workflow pattern is developed for a non-regulated internal project. The same pattern is reused in a HIPAA-regulated healthcare project without reset, reauthorization, or context boundary validation. A privacy audit reveals that patient data was processed using a prompt template that retained context from the non-regulated project.

Missing Lifecycle Object:

Cross-Project Reuse Compliance ([MRO-09](#)); Privacy/GDPR Lifecycle Mapping ([MRO-10](#)); Vendor/Model/Runtime Substitution Conformance ([MRO-15](#))

Relevant ALCS Dimensions:

[ALCS-09](#): Cross-Project Reuse Compliance; [ALCS-10](#): Privacy/GDPR Lifecycle Mapping; [ALCS-14](#): Third-Party Processor/Subprocessor Chain

Enterprise Consequence:

Privacy risk. Potential HIPAA compliance issue requiring legal review. Potential regulatory exposure. Customer trust loss. Cross-contamination of regulated and non-regulated contexts. Reputational damage.

Investigation Evidence Needed:

- Reuse records showing when and how the pattern was reused
- Reset validation records (absent)
- Reauthorization records (absent)
- Context boundary records (absent)
- Privacy impact assessment records (absent)

Remediation Pattern:

- Implement cross-project reuse compliance workflow requiring reset and reauthorization
- Add context boundary validation for regulated projects
- Implement privacy/GDPR lifecycle mapping for data flow
- Add reuse policy enforcement at project boundaries
- Implement substitution conformance validation

Relevant MROs: [MRO-09](#), [MRO-10](#), [MRO-15](#)

Relevant ALCS: [ALCS-09](#), [ALCS-10](#), [ALCS-14](#)

Scenario 5: Privacy Validation Failure

Failure Trigger:

An enterprise sends raw agent traces, including customer PII and proprietary business logic, to a third-party reviewer for compliance validation. The reviewer is not a processor under a data processing agreement. The data transfer creates a privacy risk and potential security incident.

Missing Lifecycle Object:

Privacy/GDPR Lifecycle Mapping ([MRO-10](#)); Privacy-Preserving Third-Party Validation ([MRO-11](#)); Evidence Minimization and Selective Disclosure ([MRO-12](#)); Data Subject Rights vs Evidence Retention ([MRO-13](#))

Relevant ALCS Dimensions:

[ALCS-10](#) : Privacy/GDPR Lifecycle Mapping; [ALCS-11](#) : Privacy-Preserving Third-Party Validation; [ALCS-12](#) : Evidence Minimization and Selective Disclosure; [ALCS-13](#) : Data Subject Rights vs Evidence Retention

Enterprise Consequence:

Privacy risk. Potential GDPR Article 28 processor-chain issue requiring legal review. Security incident. Regulatory investigation challenge. Customer trust loss. Potential regulatory exposure. Reputational damage.

Investigation Evidence Needed:

- Validation evidence transfer records
- Data processing agreement records (absent)
- Disclosure policy records (absent)
- Redaction records (absent)
- Evidence minimization records (absent)
- Validation protocol records (absent)

Remediation Pattern:

- Implement privacy-preserving third-party validation protocol (e.g., evidence-based validation pattern from Chapter 14)
- Add disclosure policy requiring evidence minimization and selective disclosure
- Implement redaction profile for sensitive data
- Add data processing agreement requirement for validation providers
- Implement validation protocol with hash manifests and verdict hashes instead of raw trace transfer

Relevant MROs: [MRO-10](#) , [MRO-11](#) , [MRO-12](#) , [MRO-13](#)

Relevant ALCS: [ALCS-10](#) , [ALCS-11](#) , [ALCS-12](#) , [ALCS-13](#)

Scenario 6: Processor Chain Failure**Failure Trigger:**

A multi-agent workflow involves a foundation model provider, a vector database service, a CRM API, a workflow orchestration tool, a monitoring platform, and a cloud runtime. A customer data breach occurs, but the enterprise cannot identify which processor or subprocessor was responsible because no processor chain responsibility mapping exists.

Missing Lifecycle Object:

Third-Party Processor/Subprocessor Chain ([MRO-14](#)); Tool-Action Liability Boundary ([MRO-05](#)); MAS Evidence Partitioning ([MRO-08](#))

Relevant ALCS Dimensions:

ALCS-14: Third-Party Processor/Subprocessor Chain; **ALCS-04**: Accepted Outcome Compliance; **ALCS-08**: MAS Evidence Partitioning

Enterprise Consequence:

Processor chain liability dispute. Potential GDPR Article 28 processor-chain issue requiring legal review. Unclear responsibility for breach. Vendor disputes. Insurance claim challenge. Regulatory investigation challenge. Customer trust loss. Reputational damage.

Investigation Evidence Needed:

- Processor agreements with each vendor
- Subprocessor chain records showing data flow
- Responsibility assignment records for each processor
- Tool-action logs showing which processor handled which data
- Evidence partitioning records showing processor boundaries

Remediation Pattern:

- Implement third-party processor/subprocessor chain mapping
- Add processor agreements with GDPR Article 28 mapping and legal review
- Implement responsibility assignment for each processor
- Add tool-action liability boundary records
- Implement evidence partitioning at processor boundaries

Relevant MROs: **MRO-14**, **MRO-05**, **MRO-08**

Relevant ALCS: **ALCS-14**, **ALCS-04**, **ALCS-08**

Scenario 7: Vendor / Runtime Substitution Failure**Failure Trigger:**

An engineering team switches from one foundation model provider to another to reduce costs. The new model has different output formats, different tool-calling semantics, and different context window behavior. Execution still works, but evidence chain integrity degrades, authority boundary enforcement weakens, and privacy-preserving validation fails because the new model does not support the same evidence export format.

Missing Lifecycle Object:

Vendor/Model/Runtime Substitution Conformance (**MRO-15**); MAS Evidence Partitioning (**MRO-08**); Privacy/GDPR Lifecycle Mapping (**MRO-10**)

Relevant ALCS Dimensions:

ALCS-15: Vendor/Model/Runtime Substitution Conformance; **ALCS-08**: MAS Evidence Partitioning; **ALCS-11**: Privacy-Preserving Third-Party Validation

Enterprise Consequence:

Compliance drift. Evidence chain degradation. Authority boundary violations. Privacy risks. Audit-readiness failure or audit challenge. Inability to demonstrate conformance after substitution. Reputational damage.

Investigation Evidence Needed:

- Substitution records showing when and why the model was changed
- Conformance validation records (absent)
- Regression testing records (absent)
- Evidence integrity validation records (absent)
- Authority boundary validation records (absent)

Remediation Pattern:

- Implement vendor/model/runtime substitution conformance workflow
- Add conformance validation testing before substitution
- Implement regression testing for evidence chain integrity
- Add authority boundary validation after substitution
- Implement privacy-preserving validation protocol compatibility testing

Relevant MROs: [MRO-15](#), [MRO-08](#), [MRO-10](#)**Relevant ALCS:** [ALCS-15](#), [ALCS-08](#), [ALCS-11](#)

Scenario 8: Remediation Closure Failure**Failure Trigger:**

Engineering discovers a prompt bug that caused incorrect customer recommendations. The bug is fixed, but no responsible owner is assigned, no affected outcome list is generated, no evidence review is conducted, no customer impact assessment is performed, and no accepted closure state is recorded. Customers later discover the issue and file complaints, but the enterprise cannot demonstrate that remediation was completed.

Missing Lifecycle Object:

Incident, Dispute, and Remediation Closure ([MRO-16](#)); Accepted Outcome Compliance ([MRO-04](#)); MAS Evidence Partitioning ([MRO-08](#))

Relevant ALCS Dimensions:

[ALCS-04](#) : Accepted Outcome Compliance; [ALCS-08](#) : MAS Evidence Partitioning; [ALCS-06](#) : Responsibility Transfer Across Agents

Enterprise Consequence:

Remediation delays. No closure state. Customer disputes without resolution. Regulatory investigation failure. Reputational damage. Inability to demonstrate that remediation was completed.

Investigation Evidence Needed:

- Incident records showing when the bug was discovered
- Affected outcome records (absent)
- Remediation records showing the fix
- Closure acceptance records (absent)
- Customer impact assessment records (absent)
- Evidence review records (absent)

Remediation Pattern:

- Implement incident, dispute, and remediation closure workflow
- Add responsible owner assignment for remediation
- Implement affected outcome list generation
- Add evidence review workflow for affected outcomes
- Implement customer impact assessment
- Add accepted closure state requiring explicit acceptance by responsible owner

Relevant MROs: MRO-16, MRO-04, MRO-08

Relevant ALCS: ALCS-04, ALCS-08, ALCS-06

15.4 Remediation and Prevention Patterns

The following table summarizes remediation and prevention patterns for the eight failure scenarios.

Table T-15-03: Failure Scenario Summary

Wide source table split into consecutive column groups; all source rows and columns are preserved.

Continuation 1 of 1

Authority Boundary Failure	
Scenario	Authority Boundary Failure
Failure Trigger	Agent exceeds delegated authority; moves from advising to committing without human confirmation
Missing Lifecycle Object	Delegated Authority Boundary; Authority Drift Detection; Tool-Action Liability Boundary
Investigation Evidence	Authority grant records; tool-action logs; drift detection logs; human confirmation records; acceptance records
Remediation Pattern	Implement authority boundary enforcement; add drift detection; require human confirmation for commitments; add reauthorization workflow

Evidence Chain Failure	
Scenario	Evidence Chain Failure
Failure Trigger	Organization has logs but cannot reconstruct authority, plan, confirmation, tool action, and accepted outcome
Missing Lifecycle Object	MAS Evidence Partitioning; Privacy-Preserving Third-Party Validation; Evidence Minimization
Investigation Evidence	Partitioned evidence chain; evidence export records; retention policy records; validation protocol records
Remediation Pattern	Implement MAS evidence partitioning; add evidence export with selective disclosure; add privacy-preserving validation protocol

Accepted Outcome Failure

Scenario	Accepted Outcome Failure
Failure Trigger	Orchestrator marks task complete but no human accepts the result
Missing Lifecycle Object	Human-Role-to-MAS Responsibility Mapping; Accepted Outcome Compliance; Incident/Dispute/Remediation Closure
Investigation Evidence	Role assignment records; acceptance records; review records; escalation records; remediation ownership records
Remediation Pattern	Implement human-role-to-MAS responsibility mapping; add accepted outcome compliance workflow; add review and escalation workflow

Cross-Project Reuse Failure

Scenario	Cross-Project Reuse Failure
Failure Trigger	Prompt, memory, or workflow reused across regulated contexts without reset or reauthorization
Missing Lifecycle Object	Cross-Project Reuse Compliance; Privacy/GDPR Lifecycle Mapping; Vendor/Model/Runtime Substitution Conformance
Investigation Evidence	Reuse records; reset validation records; reauthorization records; context boundary records; privacy impact assessment records
Remediation Pattern	Implement cross-project reuse compliance workflow; add context boundary validation; add privacy/GDPR lifecycle mapping

Privacy Validation Failure

Scenario	Privacy Validation Failure
Failure Trigger	Enterprise sends raw traces with PII to third-party reviewer without data processing agreement
Missing Lifecycle Object	Privacy/GDPR Lifecycle Mapping; Privacy-Preserving Third-Party Validation; Evidence Minimization; Data Subject Rights
Investigation Evidence	Validation evidence transfer records; data processing agreement records; disclosure policy records; redaction records
Remediation Pattern	Implement privacy-preserving validation protocol; add disclosure policy with evidence minimization; add redaction profile; require data processing agreements

Processor Chain Failure

Scenario	Processor Chain Failure
Failure Trigger	Multi-vendor workflow but no processor/subprocessor responsibility chain exists
Missing Lifecycle Object	Third-Party Processor/Subprocessor Chain; Tool-Action Liability Boundary; MAS Evidence Partitioning
Investigation Evidence	Processor agreements; subprocessor chain records; responsibility assignment records; tool-action logs; evidence partitioning records
Remediation Pattern	Implement processor/subprocessor chain mapping; add processor agreements with GDPR Article 28 mapping and legal review; add responsibility assignment

Vendor/Runtime Substitution Failure	
Scenario	Vendor/Runtime Substitution Failure
Failure Trigger	Model/runtime substitution degrades evidence, authority, and privacy semantics
Missing Lifecycle Object	Vendor/Model/Runtime Substitution Conformance; MAS Evidence Partitioning; Privacy/GDPR Lifecycle Mapping
Investigation Evidence	Substitution records; conformance validation records; regression testing records; evidence integrity validation records
Remediation Pattern	Implement substitution conformance workflow; add conformance validation testing; add regression testing for evidence chain integrity

Remediation Closure Failure	
Scenario	Remediation Closure Failure
Failure Trigger	Bug fixed but no responsible owner, affected outcome list, evidence review, or accepted closure state
Missing Lifecycle Object	Incident/Dispute/Remediation Closure; Accepted Outcome Compliance; MAS Evidence Partitioning
Investigation Evidence	Incident records; affected outcome records; remediation records; closure acceptance records; customer impact assessment records
Remediation Pattern	Implement incident/dispute/remediation closure workflow; add responsible owner assignment; add affected outcome list generation; add closure acceptance

Table T-15-04: Prevention and Remediation Patterns

Wide source table split into consecutive column groups; all source rows and columns are preserved.

Continuation 1 of 1

Pattern	Prevents Which Failure	Required Object	Operational Owner	Evidence Artifact
Authority Boundary Enforcement	Authority Boundary Failure; Authority Drift	Delegated Authority Boundary (MRO-02); Authority Drift Detection (MRO-07)	Governance team; agent operations team	Authority grant records; boundary violation alerts; drift detection logs
MAS Evidence Partitioning	Evidence Chain Failure; Processor Chain Failure	MAS Evidence Partitioning (MRO-08)	Data governance team; compliance team	Partitioned evidence chains; evidence export logs; partition policy records
Accepted Outcome Compliance Workflow	Accepted Outcome Failure; Remediation Closure Failure	Accepted Outcome Compliance (MRO-04); Human-Role-to-MAS Responsibility Mapping (MRO-01)	Business process owner; compliance team	Acceptance records; re-view records; escalation records; closure acceptance records
Cross-Project Reuse Compliance Workflow	Cross-Project Reuse Failure	Cross-Project Reuse Compliance (MRO-09); Privacy/GDPR Lifecycle Mapping (MRO-10)	Data governance team; privacy team	Reuse records; reset validation records; reauthorization records; context boundary records
Privacy-Preserving Validation Protocol	Privacy Validation Failure	Privacy-Preserving Third-Party Validation (MRO-11); Evidence Minimization (MRO-12)	Privacy team; compliance team	Validation protocol records; disclosure policy records; redaction records; hash manifests; verdict hashes
Processor/Subprocessor Chain Mapping	Processor Chain Failure	Third-Party Processor/Subprocessor Chain (MRO-14)	Procurement team; legal team; data governance team	Processor agreements; subprocessor chain records; responsibility assignment records

Pattern	Prevents Which Failure	Required Object	Operational Owner	Evidence Artifact
Substitution Conformance Validation	Vendor/Runtime Substitution Failure	Vendor/Model/Runtime Substitution Conformance (MRO-15)	Engineering team; compliance team	Substitution records; conformance validation records; regression testing records; evidence integrity validation records
Incident/Dispute/Remediation Closure Workflow	Remediation Closure Failure	Incident, Dispute, and Remediation Closure (MRO-16)	Incident response team; customer support team; compliance team	Incident records; affected outcome records; remediation records; closure acceptance records; customer impact assessment records

Interpretation: Prevention and remediation patterns require specific lifecycle objects, operational owners, and evidence artifacts. These patterns are not generic risk management controls. They are lifecycle-specific governance patterns that address the gap between model-centric governance and agentic lifecycle responsibility.

15.5 Bridge to Adoption Roadmap

Chapter 15 has examined eight enterprise failure scenarios where lifecycle governance gaps create predictable failures. Each scenario demonstrates that model-centric governance and ordinary orchestration controls become insufficient when work is delegated across agents, tools, humans, projects, vendors, and evidence boundaries.

The bridge from failure scenarios (Chapter 15) to adoption roadmap (Chapter 17) is the recognition that lifecycle governance is not a theoretical framework. It is a practical necessity for enterprises deploying agentic AI systems.

Failure scenarios create enterprise consequences: accountability gaps, audit-readiness failures or audit challenges, privacy risks, liability disputes, potential regulatory non-compliance depending on deployment context, customer harm, and reputational damage. Prevention and remediation patterns require specific lifecycle objects, operational owners, and evidence artifacts.

Chapter 17 will present an adoption roadmap for enterprises implementing AI Agent Lifecycle Governance. The roadmap will address:

- Where to start (maturity assessment, gap analysis, pilot projects)
- How to integrate lifecycle governance with existing enterprise controls
- How to operationalize lifecycle responsibility objects
- How to measure progress and demonstrate conformance
- How to scale from pilot to production

The goal is not to claim that lifecycle governance eliminates all risk. The goal is to provide a practical path for enterprises to address the lifecycle governance gap.

16. Boundary to Companion Papers: Evidence, Assurance, and Insurability

MAIN RESULT Evidence, assurance, and insurability require separate treatment beyond this paper.	METHOD / LENS Boundary map for planned companion scope.
EVIDENCE BASIS The white paper should not overclaim companion-paper deliverables.	KEY TAKEAWAY / BOUNDARY Companion papers are planned/intended scope, not published guarantees.

Version: v0.3.2-FRC-R3

16.0 Why Companion Paper Boundaries Matter

This white paper serves as the primary framework document for AI Agent Lifecycle Governance. It defines the semantic and engineering layer for agentic and multi-agent system compliance. It introduces Missing Regulatory Objects, the RCCS-T/RCCS-M/ALCS three-profile analytical model, enterprise control crosswalks, system positioning, and adoption pathways. Because the subject intersects compliance, audit, assurance, insurance, and enterprise procurement, clear scope boundaries are necessary.

Without companion-paper boundaries, the main white paper could become overloaded with audit methodology, insurance underwriting models, actuarial analysis, legal opinions, certification procedures, and procurement recommendations. That would dilute its core purpose and make it unreadable for the audiences who need lifecycle governance clarity.

The purpose of this chapter is to declare which topics are covered at framework-entry level in the main white paper and which topics are reserved for specialized companion papers. This boundary is intentional. This white paper defines the primary framework layer. It does not attempt to become every downstream assurance, insurance, audit, or procurement methodology.

Two companion papers are planned:

1. **Agentic AI Evidence & Assurance White Paper 2026** — planned to define audit control objectives, evidence sufficiency, test procedures, attestation readiness, and privacy-preserving validation patterns.
1. **Agentic AI Insurability White Paper 2026** — planned to define risk classification, tool-action liability, loss attribution, claim replay evidence, premium differentiation logic, exclusion triggers, and underwriting implications.

This chapter clarifies the role of each paper and prevents scope confusion. It ensures that readers understand what the main white paper does and does not claim. It establishes non-claim discipline for legal, audit, assurance, insurance, RCCS-T/RCCS-M/ALCS, MPLP, Validation Lab, and system mapping boundaries.

16.1 What the Main White Paper Covers

The main white paper covers the following topics at framework-entry level:

Core Framework:

- AI Agent Lifecycle Governance definition and lifecycle chain
- Regulatory and standards baseline analysis
- Missing Regulatory Objects ([MRO-01](#) through [MRO-16](#))
- RCCS-T (Traditional Regulatory Compliance Coverage Score) framework
- RCCS-M (MRO-Adjusted Regulatory Compliance Coverage Score) framework
- ALCS (Agentic Lifecycle Conformance Score) framework
- Three-profile analytical model with optional composite worksheet views
- Enterprise control crosswalk and procurement scorecard inputs
- System positioning and comparative field mappings
- MPLP boundary and non-claim discipline
- Evidence-based validation pattern at framework-entry level
- Enterprise failure scenarios
- Adoption roadmap

Framework-Entry Evidence Concepts:

The main white paper introduces evidence chain, evidence partitioning, privacy-preserving validation, and dispute-ready replay as lifecycle governance requirements. It defines what these concepts mean and why they matter for agentic systems. It does not define audit engagement procedures, control testing methodology, or assurance opinions.

Framework-Entry Insurance Concepts:

The main white paper introduces tool-action liability, authority boundary risk, loss attribution, and claim replay as lifecycle governance implications. It defines [MRO-05](#) (Tool-Action Liability Boundary) and explains why agentic systems create new insurance challenges. It does not define insurance policy wording, premium pricing models, actuarial methodology, or underwriting procedures.

Boundary:

The main white paper does not fully define:

- Audit engagement procedures
- Assurance opinions or attestation reports
- Insurance policy wording or coverage terms
- Premium pricing models or actuarial methodology
- Loss quantification or claim settlement procedures
- Legal compliance opinions or jurisdictional interpretations
- Procurement recommendations or vendor selection criteria
- Certification procedures or conformity assessment
- Validation Lab operational procedures or certification authority claims

The main white paper provides the primary framework layer. Detailed assurance procedures and insurability models are reserved for companion papers. This boundary ensures that the main paper remains focused on

lifecycle governance definition and does not overreach into professional judgment domains that require specialized expertise.

16.2 Companion Paper 1 — Agentic AI Evidence & Assurance White Paper 2026

The **Agentic AI Evidence & Assurance White Paper 2026** is planned to define the audit, control testing, and validation methodology for agentic systems. When published, it is expected to build on the lifecycle governance framework defined in the main white paper and translate MROs, RCCS-T, RCCS-M, and ALCS into audit control objectives, evidence requirements, and attestation readiness.

Intended Scope:

The Evidence & Assurance companion paper is intended to cover:

- **Evidence artifact taxonomy:** What evidence types are required for each MRO and lifecycle stage.
- **Audit control objectives:** How to translate MROs into testable control objectives.
- **Control activities:** What control activities satisfy each objective.
- **Test procedures:** How to test design effectiveness and operating effectiveness.
- **Design effectiveness:** How to evaluate whether controls are properly designed.
- **Operating effectiveness:** How to evaluate whether controls are operating as intended.
- **Evidence sufficiency:** How much evidence is enough for audit readiness.
- **Evidence integrity:** How to ensure evidence has not been tampered with.
- **Replayability:** How to replay evidence for dispute resolution or audit review.
- **Attestation readiness:** What evidence is required for third-party attestation.
- **Privacy-preserving validation patterns:** How to validate evidence without exposing sensitive data.
- **Validation readiness:** How to prepare for third-party validation or audit engagement.
- **Assurance engagement boundaries:** What assurance engagements can and cannot claim.

Boundary:

The Evidence & Assurance companion paper is not intended to:

- Create legal certification or regulatory approval
- Claim that Validation Lab is a certification authority, regulator, or conformity assessment body
- Issue audit opinions or replace professional audit judgment
- Guarantee compliance with any specific regulation or standard
- Replace internal audit, external audit, or regulatory examination
- Define insurance underwriting, actuarial methodology, or premium pricing

Validation Lab Positioning:

Validation Lab may be referenced in the Evidence & Assurance companion paper only as a non-certifying evidence adjudication example. It is not a regulator, certification authority, conformity assessment body, or legal compliance certification mechanism. It does not certify compliance. It does not issue legal opinions. It does not replace audit or regulatory judgment.

The Evidence & Assurance companion paper is expected to clarify that Validation Lab is an example of how privacy-preserving third-party validation could be structured. It is not the only method. It is not required for compliance. It is not a certification program.

16.3 Companion Paper 2 — Agentic AI Insurability White Paper 2026

The **Agentic AI Insurability White Paper 2026** is planned to define the risk classification, loss attribution, and underwriting implications for agentic systems. When published, it is expected to build on the lifecycle governance framework defined in the main white paper and translate MROs, RCCS-T, RCCS-M, and ALCS into insurability criteria, claim replay evidence, and premium differentiation logic.

Intended Scope:

The Insurability companion paper is intended to cover:

- **Insurability criteria:** What makes an agentic system insurable or uninsurable.
- **Risk classification:** How to classify agentic system risk for underwriting purposes.
- **Authority boundary risk:** How delegated authority boundaries affect liability and coverage.
- **Tool-action liability:** How to attribute liability for tool actions taken by agents.
- **Loss attribution:** How to determine which party is responsible for a loss.
- **Loss quantification concepts:** How to estimate potential losses from agentic system failures.
- **Claim replay evidence:** What evidence is required to replay a claim for dispute resolution.
- **Dispute-ready evidence:** How to structure evidence for insurance claim disputes.
- **Remediation closure:** How to verify that incidents have been properly remediated.
- **Premium differentiation logic:** How lifecycle conformance could affect premium pricing.
- **Exclusion triggers:** What conditions could trigger coverage exclusions.
- **Autonomous-action exclusions:** How autonomous actions affect coverage boundaries.
- **Moral hazard considerations:** How to prevent moral hazard in agentic system insurance.
- **Coverage boundary analysis:** Where coverage begins and ends for agentic systems.

Boundary:

The Insurability companion paper is not intended to:

- Define actual insurance policy wording or coverage terms
- Set premiums or provide actuarial pricing models
- Replace underwriting, actuarial, legal, or claims judgment
- Guarantee insurability or coverage for any specific system
- Issue legal opinions on liability or regulatory compliance
- Replace insurance broker, underwriter, or claims adjuster judgment
- Define claim settlement procedures or dispute resolution mechanisms

The Insurability companion paper is expected to provide a framework for analyzing insurability implications. It is not intended to replace professional insurance judgment. It is not intended to claim that RCCS-T, RCCS-M, or ALCS scores determine premiums. It is not intended to claim that MPLP conformance guarantees insurability. It is expected to clarify that insurability depends on underwriting judgment, actuarial analysis, legal review, and risk appetite, not only on lifecycle conformance scores.

16.4 Companion Paper Boundary Matrix

The following tables clarify which topics are covered in the main white paper and which are reserved for the Evidence & Assurance and Insurability companion papers.

Table T-16-01A: Main Paper vs Evidence & Assurance Boundary

Topic	Covered in Main White Paper	Reserved for Evidence & Assurance Paper
Evidence Chain	Defines evidence chain as lifecycle governance requirement; explains why evidence chain matters for agentic systems	Defines evidence sufficiency, integrity, replay procedures, and control testing methodology
Validation Readiness	Introduces privacy-preserving validation as MRO-11 concept	Defines validation engagement procedures, evidence requirements, and attestation readiness
Audit Readiness	Maps MROs to enterprise control questions; provides procurement scorecard inputs	Defines audit control objectives, test procedures, design effectiveness, and operating effectiveness testing
Control Objectives	Identifies control gaps created by agentic systems	Translates MROs into testable control objectives with evidence requirements
Control Activities	Describes lifecycle governance controls at framework level	Defines specific control activities, testing procedures, and evidence sufficiency criteria
Test Procedures	Not covered	Defines how to test controls for design and operating effectiveness
Design Effectiveness	Not covered	Defines how to evaluate whether controls are properly designed
Operating Effectiveness	Not covered	Defines how to evaluate whether controls are operating as intended
Attestation Readiness	Mentions attestation as downstream implication	Defines what evidence is required for third-party attestation or assurance engagement
Privacy-Preserving Validation	Defines MRO-11 and MRO-12 as lifecycle requirements	Defines validation procedures, selective disclosure, and privacy-preserving evidence adjudication
Enterprise Procurement	Provides procurement scorecard inputs based on RCCS-T/RCCS-M/ALCS	Not primary scope; procurement decisions remain enterprise judgment
Board Reporting	Provides board reporting elements for agentic risk	Not primary scope; board reporting remains enterprise governance responsibility

Table T-16-01B: Main Paper vs Insurability Boundary

Topic	Covered in Main White Paper	Reserved for Insurability Paper
Tool-Action Liability	Defines MRO-05 (Tool-Action Liability Boundary) as lifecycle governance object	Evaluates liability boundary implications, loss attribution, and coverage analysis
Dispute-Ready Replay	Defines evidence chain and replay as lifecycle requirements	Defines claim replay procedures, dispute-ready evidence structure, and claim settlement implications
Claim Replay	Mentions claim replay as downstream implication of evidence chain	Defines claim replay methodology, evidence requirements, and dispute resolution procedures
Loss Attribution	Defines authority boundary and responsibility transfer as MROs	Defines loss attribution methodology, liability analysis, and coverage boundary implications
Risk Classification	Provides RCCS-T/RCCS-M/ALCS scores as risk inputs; does not claim scores determine risk classification	Analyzes how lifecycle conformance affects risk classification for underwriting purposes
Premium Differentiation	Mentions premium differentiation as potential downstream implication	Analyzes premium differentiation logic without defining actual pricing or actuarial models
Exclusion Triggers	Defines authority drift and autonomous action boundaries as lifecycle risks	Analyzes exclusion trigger implications for coverage boundaries

Topic	Covered in Main White Paper	Reserved for Insurability Paper
Incident Remediation Closure	Defines MRO-16 (Incident, Dispute, and Remediation Closure) as lifecycle requirement	Defines remediation closure verification for claim settlement and coverage restoration
Authority Boundary Risk	Defines MRO-02 (Delegated Authority Boundary) and MRO-07 (Authority Drift)	Analyzes authority boundary risk for underwriting, coverage, and liability purposes
Moral Hazard	Not covered	Analyzes moral hazard considerations for agentic system insurance
Coverage Boundary Analysis	Not covered	Defines where coverage begins and ends for agentic systems

Key Distinction:

The main white paper defines lifecycle governance objects and framework-entry concepts. The companion papers define detailed procedures, methodologies, and professional judgment frameworks. The main paper does not replace audit, assurance, insurance, actuarial, legal, or procurement judgment. It provides structured inputs that make those professional judgments more efficient, consistent, and auditable when applied to agentic systems.

16.5 Non-Claim Discipline

This section establishes explicit boundaries for what the main white paper does and does not claim. These boundaries apply to the entire document, including all chapters, tables, figures, and appendices.

Explicit Statements:

- **This white paper is not legal advice.** It does not provide legal opinions, jurisdictional interpretations, or compliance conclusions. Legal compliance requires professional legal judgment.
- **This white paper is not an audit opinion.** It does not issue audit opinions, attestation reports, or assurance conclusions. Audit readiness requires professional audit judgment.
- **This white paper is not an assurance report.** It does not provide assurance engagements, attestation services, or certification. Assurance requires professional assurance judgment.
- **This white paper is not an insurance underwriting manual.** It does not define insurance policy wording, premium pricing, actuarial methodology, or claim settlement procedures. Insurability requires professional underwriting, actuarial, and claims judgment.
- **This white paper is not a certification program.** It does not certify compliance, issue conformity assessments, or provide regulatory approval. Certification requires accredited conformity assessment bodies.
- **This white paper does not certify compliance.** RCCS-T, RCCS-M, and ALCS are analytical frameworks, not legal compliance scores. High scores do not guarantee compliance. Low scores do not prove non-compliance.
- **RCCS-T, RCCS-M, and ALCS are analytical frameworks, not legal compliance scores.** They provide structured inputs for risk assessment, procurement, audit, and insurance analysis. They do not replace professional judgment. RCCS-M is author analytical and forward-looking, not current law or a regulator-approved benchmark.
- **MPLP is a lifecycle protocol path, not a regulator, certification body, or exclusive compliance method.** MPLP conformance does not guarantee compliance. Non-MPLP systems are not automatically non-

compliant. MPLP is one example of how lifecycle governance could be structured.

- **Validation Lab is a non-certifying evidence adjudication example, not a certification authority.** Validation Lab does not certify compliance. It does not issue legal opinions. It does not replace audit or regulatory judgment. It is an example of privacy-preserving third-party validation, not a required compliance mechanism.

Table T-16-02: Claim Boundaries

Claim Area	Allowed Wording	Forbidden Wording
Legal	"This framework may help inform compliance analysis" / "Legal compliance requires professional legal judgment"	"Regulator-approved" / "Certified compliant" / "Guarantees legal compliance" / "Official legal standard"
Audit / Assurance	"This framework provides audit readiness inputs" / "Audit opinions require professional audit judgment"	"Certifies audit compliance" / "Issues audit opinions" / "Replaces internal audit" / "Guaranteed audit-ready"
Insurance	"This framework may inform insurability analysis" / "Underwriting requires professional insurance judgment"	"Guaranteed insurable" / "Sets premiums" / "Defines policy wording" / "Replaces underwriting judgment"
RCCS-T / RCCS-M / ALCS	"Analytical frameworks for risk assessment" / "Structured inputs for professional judgment" / "RCCS-M is author analytical and forward-looking"	"Legal compliance scores" / "Certification scores" / "Guarantees compliance" / "The only scoring method" / "Regulator-approved benchmark"
MPLP	"One example of lifecycle protocol implementation" / "Not required for compliance"	"Required for compliance" / "The only solution" / "Official standard" / "Regulator-mandated"
Validation Lab	"Non-certifying evidence adjudication example" / "One approach to privacy-preserving validation"	"Certification authority" / "Certifies compliance" / "Required for compliance" / "Regulator-approved" / "Official validator"
System Mapping	"Comparative analysis of lifecycle conformance strength" / "Not endorsements or certifications"	"Certified systems" / "Approved vendors" / "Guaranteed compliant" / "Official rankings"

Interpretation Guidance:

When interpreting this white paper, readers should understand that:

1. **Framework, not mandate:** This white paper defines a governance framework. It does not mandate specific implementations, vendors, or methods.
1. **Inputs, not conclusions:** RCCS-T, RCCS-M, ALCS, MROs, and system mappings provide structured inputs for professional judgment. They do not replace legal, audit, insurance, actuarial, or procurement judgment.
1. **Examples, not requirements:** MPLP and Validation Lab are examples of how lifecycle governance could be implemented. They are not required for compliance. Other approaches may be equally valid.
1. **Analysis, not certification:** System mappings and comparative field positioning are analytical tools. They are not endorsements, certifications, or vendor recommendations.
1. **Disclosure, not guarantee:** Evidence levels (L1 - L5) disclose the basis for claims. L5 (author inference) does not guarantee accuracy. Professional judgment is required to evaluate all claims.

16.6 Bridge to Adoption Roadmap

The companion-paper boundary clarifies adoption sequencing for enterprises, auditors, insurers, and regulators evaluating agentic systems.

Recommended Adoption Sequence:

1. **Use the main white paper** to understand the AI Agent Lifecycle Governance primary framework layer, Missing Regulatory Objects, RCCS-T/RCCS-M/ALCS scoring, enterprise control crosswalk, and system positioning. This provides the conceptual foundation and framework-entry understanding.
1. **Use the Evidence & Assurance companion paper** (when published) for audit readiness, control testing methodology, evidence sufficiency analysis, attestation preparation, and privacy-preserving validation procedures. This provides the assurance methodology layer.
1. **Use the Insurability companion paper** (when published) for risk classification, loss attribution analysis, claim replay evidence preparation, premium differentiation understanding, and coverage boundary analysis. This provides the risk transfer and insurability layer.
1. **Use industry briefs** (when published) for sector-specific operationalization in accounting, insurance, banking, healthcare, legal, and other regulated enterprise applications. This provides the industry-specific implementation layer.

Why This Sequence Matters:

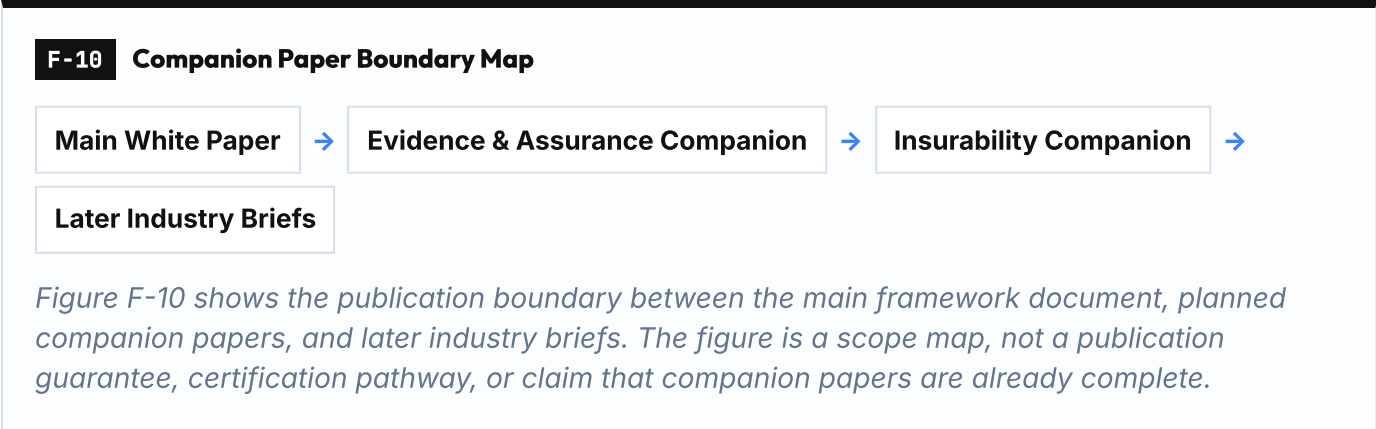
The main white paper defines the primary governance framework. Without understanding MROs, RCCS, ALCS, and lifecycle conformance, the assurance and insurability methodologies would not make sense. The companion papers build on the main framework. They do not replace it.

Enterprises that attempt to implement audit procedures or insurance analysis without understanding the lifecycle governance foundation will struggle with scope confusion, evidence gaps, and control misalignment. The adoption sequence ensures that governance, assurance, and insurability are addressed in the correct order.

Bridge to Chapter 17:

Chapter 17 defines the adoption roadmap in detail, including enterprise adoption pathways, regulatory engagement strategies, industry brief development, and global rollout sequencing. The companion-paper boundary established in this chapter ensures that the adoption roadmap can clearly distinguish between framework adoption, assurance methodology adoption, and insurability analysis adoption.

The boundary also ensures that adoption recommendations do not overreach into legal, audit, insurance, or procurement conclusions. Chapter 17 provides adoption guidance, not adoption mandates. It clarifies that adoption pathways depend on enterprise context, regulatory jurisdiction, industry sector, and risk appetite, not only on lifecycle conformance scores.



17. Adoption Roadmap for AI Agent Lifecycle Governance

MAIN RESULT Enterprises can adopt lifecycle governance in staged increments from inventory to scale.	METHOD / LENS Stage 0-to-7 roadmap and control maturity logic.
EVIDENCE BASIS Implementation needs a sequence, not a one-step certification claim.	KEY TAKEAWAY / BOUNDARY The roadmap is an implementation planning aid, not a mandate or guarantee.

Version: v0.3.2-FRC-R3

17.0 Why Adoption Roadmap Matters

AI Agent Lifecycle Governance is not a theoretical framework. It is a practical necessity for enterprises deploying agentic AI systems. The previous chapters have defined the missing layer (Chapter 2), identified the sixteen Missing Regulatory Objects (Chapter 6), introduced RCCS-T, RCCS-M, and ALCS as analytical frameworks (Chapters 7-8), mapped existing systems (Chapters 11-12), examined evidence-based validation patterns (Chapter 14), and demonstrated enterprise failure scenarios (Chapter 15).

This chapter translates the framework into an enterprise adoption roadmap. The roadmap is designed for enterprises that recognize the lifecycle governance gap and want to operationalize lifecycle responsibility objects before scaling autonomous or multi-agent workflows.

The roadmap is not a procurement guide, certification checklist, or legal compliance procedure. It is a staged adoption path that helps enterprises:

- Inventory agentic behavior and identify lifecycle governance gaps
- Define delegated authority boundaries and human-role-to-MAS responsibility mappings
- Convert orchestration logs into partitioned evidence chains
- Map privacy and data subject rights across lifecycle surfaces
- Prepare for evidence-based validation and qualified review
- Operationalize incident, dispute, and remediation closure workflows
- Scale lifecycle governance through integration with existing enterprise controls

The goal is not to claim that lifecycle governance eliminates all risk. The goal is to provide a practical path for enterprises to address the lifecycle governance gap identified in this white paper.

F-11 Stage 0-to-7 Adoption Roadmap

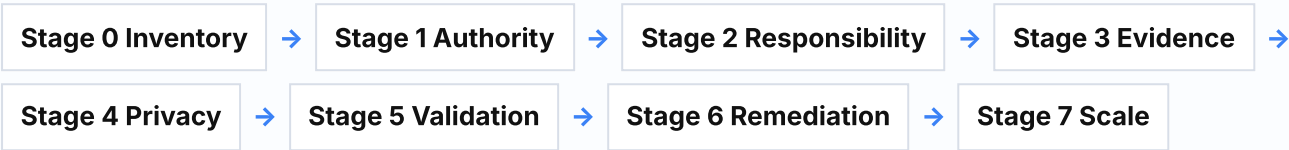


Figure F-11 summarizes a staged adoption path for lifecycle governance. It is an implementation planning aid, not a procurement mandate, certification checklist, legal compliance procedure, or deployment-readiness guarantee.

17.1 Maturity Assessment and Gap Analysis

Enterprises should begin adoption with a maturity assessment to understand their current lifecycle governance capability and identify gaps. The following table defines five maturity levels.

Table T-17-01: Adoption Maturity Levels

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Level 0: Unaware

Enterprise State	Agentic AI systems deployed; no lifecycle governance awareness; model governance assumed sufficient
Lifecycle Governance Capability	No lifecycle responsibility objects; no authority boundaries; no accepted outcome compliance; no evidence partitioning
Typical Gap	Execution logs exist but cannot reconstruct authority, plan, confirmation, tool action, and accepted outcome; no responsible human role for agent outcomes; no evidence-based validation capability
Recommended Next Step	Inventory agentic behavior; identify where agents make decisions, use tools, or hand off work; assess whether model governance covers lifecycle responsibility

Level 1: Aware

Enterprise State	Lifecycle governance gap recognized; pilot projects identified; governance team assigned
Lifecycle Governance Capability	Basic lifecycle inventory; some authority boundaries documented; human oversight exists but not mapped to MAS responsibilities
Typical Gap	Authority boundaries are implicit or undocumented; human oversight is ad-hoc; evidence chains are incomplete; no privacy-preserving validation capability
Recommended Next Step	Define delegated authority boundaries for pilot projects; map human roles to MAS responsibilities; document accepted outcome compliance workflow

Level 2: Pilot

Enterprise State	Pilot projects operationalize lifecycle responsibility objects; evidence chains exist for pilot scope; authority boundaries enforced
Lifecycle Governance Capability	Delegated authority boundaries defined; human-role-to-MAS responsibility mapping exists; accepted outcome compliance workflow operational; evidence partitioning implemented for pilot scope
Typical Gap	Pilot scope is limited; lifecycle governance not integrated with existing enterprise controls; no cross-project reuse compliance; no vendor/runtime substitution conformance
Recommended Next Step	Integrate lifecycle governance with existing enterprise controls (audit, privacy, security, procurement); extend evidence partitioning beyond pilot scope; implement cross-project reuse compliance

Level 3: Integrated

Enterprise State	Lifecycle governance integrated with existing enterprise controls; evidence-based validation capability operational; privacy-preserving validation protocol implemented
Lifecycle Governance Capability	Evidence partitioning operational; privacy-preserving validation protocol implemented; cross-project reuse compliance enforced; vendor/runtime substitution conformance validated
Typical Gap	Lifecycle governance not yet scaled to all agentic workflows; incident/dispute/remediation closure workflow exists but not fully operationalized; continuous improvement not yet systematic
Recommended Next Step	Scale lifecycle governance to all agentic workflows; operationalize incident/dispute/remediation closure workflow; implement continuous improvement and monitoring

Level 4: Scaled

Enterprise State	Lifecycle governance scaled to all agentic workflows; continuous improvement operational; evidence-based validation integrated with audit readiness
Lifecycle Governance Capability	All lifecycle responsibility objects operational; evidence-based validation integrated with audit readiness; incident/dispute/remediation closure workflow fully operationalized; continuous improvement and monitoring systematic
Typical Gap	Lifecycle governance is operational but may require refinement as agentic systems evolve; new MROs may emerge as agentic AI capabilities expand
Recommended Next Step	Maintain lifecycle governance through continuous improvement; monitor for new lifecycle governance gaps; where appropriate, contribute learnings to standards, industry, or regulatory dialogue

Interpretation: Maturity assessment helps enterprises understand where they are and what gaps exist. Most enterprises deploying agentic AI systems are at Level 0 or Level 1. The roadmap provides a staged path from Level 0 to Level 4.

17.2 Pilot Project Selection

Enterprises should select pilot projects carefully to maximize learning and minimize risk. The following table defines pilot selection criteria.

Table T-17-02: Pilot Project Selection Criteria

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Bounded scope	
Why It Matters	Pilot should be large enough to demonstrate lifecycle governance value but small enough to manage risk
Good Pilot Signal	Single business process; single regulated context; single customer segment; clear authority boundaries
Avoid If	Unbounded scope; cross-jurisdictional; mission-critical; customer-facing without rollback capability

Measurable outcomes	
Why It Matters	Pilot should produce measurable evidence of lifecycle governance effectiveness
Good Pilot Signal	Clear success criteria; measurable evidence artifacts; audit readiness improvement; dispute resolution improvement
Avoid If	Vague success criteria; no measurable evidence artifacts; no audit readiness baseline

Regulatory relevance	
Why It Matters	Pilot should address a regulated context where lifecycle governance gaps create compliance risk
Good Pilot Signal	GDPR, HIPAA, SOX, EU AI Act, or other regulated context; audit readiness required; evidence retention required
Avoid If	Non-regulated context; no audit readiness requirement; no evidence retention requirement

Executive sponsorship	
Why It Matters	Pilot should have executive sponsorship to ensure organizational commitment
Good Pilot Signal	Executive sponsor assigned; governance team assigned; budget allocated; success criteria agreed
Avoid If	No executive sponsor; no governance team; no budget; no success criteria

Failure scenario relevance

Why It Matters	Pilot should address one or more enterprise failure scenarios from Chapter 15
Good Pilot Signal	Pilot addresses authority boundary failure, evidence chain failure, accepted outcome failure, or other Chapter 15 scenario
Avoid If	Pilot does not address any Chapter 15 scenario; no clear failure mode to prevent

Integration opportunity

Why It Matters	Pilot should integrate with existing enterprise controls to demonstrate scalability
Good Pilot Signal	Pilot integrates with existing audit, privacy, security, or procurement controls; evidence artifacts align with existing control frameworks
Avoid If	Pilot is isolated from existing enterprise controls; no integration opportunity; evidence artifacts do not align with existing control frameworks

Rollback capability

Why It Matters	Pilot should have rollback capability to manage risk
Good Pilot Signal	Pilot can be rolled back without customer impact; pilot can be paused without operational disruption
Avoid If	Pilot cannot be rolled back; pilot is mission-critical; pilot has irreversible customer impact

Interpretation: Pilot selection criteria help enterprises choose projects that maximize learning and minimize risk. Good pilots are bounded, measurable, regulated, sponsored, relevant to failure scenarios, integrated with existing controls, and rollback-capable.

17.3 Integration with Existing Enterprise Controls

Lifecycle governance should integrate with existing enterprise controls, not replace them. The following table maps existing enterprise controls to lifecycle governance extensions.

Table T-17-03: Existing Enterprise Control Integration

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Internal Audit

Lifecycle Governance Extension	Extend audit scope to include lifecycle responsibility objects; add evidence partitioning and selective disclosure capability
Relevant Object	MRO-08 : MAS Evidence Partitioning; MRO-11 : Privacy-Preserving Third-Party Validation; MRO-12 : Evidence Minimization and Selective Disclosure
Evidence Artifact	Partitioned evidence chains; evidence export logs; selective disclosure records; audit trail for lifecycle responsibility objects

Privacy / Data Governance

Lifecycle Governance Extension	Extend privacy controls to include agentic lifecycle surfaces; map data subject rights to evidence retention; implement privacy-preserving validation
Relevant Object	MRO-10 : Privacy/GDPR Lifecycle Mapping; MRO-11 : Privacy-Preserving Third-Party Validation; MRO-13 : Data Subject Rights vs Evidence Retention
Evidence Artifact	Data flow records; retention policy records; data subject rights workflow records; privacy-preserving validation protocol records

Security / Access Management

Lifecycle Governance Extension	Extend access controls to include delegated authority boundaries; implement authority drift detection; add tool-action authorization
Relevant Object	MRO-02 : Delegated Authority Boundary; MRO-05 : Tool-Action Liability Boundary; MRO-07 : Authority Drift
Evidence Artifact	Authority grant records; tool-action authorization records; boundary violation alerts; drift detection logs

Procurement / Vendor Management

Lifecycle Governance Extension	Extend vendor management to include processor/subprocessor chain mapping; add vendor/runtime substitution conformance validation
Relevant Object	MRO-14 : Third-Party Processor/Subprocessor Chain; MRO-15 : Vendor/Model/Runtime Substitution Conformance
Evidence Artifact	Processor agreements; subprocessor chain records; substitution records; conformance validation records

Incident Response

Lifecycle Governance Extension	Extend incident response to include lifecycle-specific incident types; add incident/dispute/remediation closure workflow
Relevant Object	MRO-16 : Incident, Dispute, and Remediation Closure
Evidence Artifact	Incident records; affected outcome records; remediation records; closure acceptance records; customer impact assessment records

Compliance / Regulatory Reporting

Lifecycle Governance Extension	Extend compliance reporting to include lifecycle responsibility objects; add RCCS-T/RCCS-M/ALCS as analytical tools (not legal scores)
Relevant Object	MRO-01 : Human Role to MAS Responsibility Mapping; MRO-04 : Accepted Outcome Compliance
Evidence Artifact	Role assignment records; acceptance records; review records; escalation records; RCCS-T/RCCS-M/ALCS analytical reports (not legal compliance scores)

Model Risk Management

Lifecycle Governance Extension	Extend model risk management to include lifecycle risk; add authority-transition influence monitoring; add model substitution conformance
Relevant Object	MR0-07 : Authority Drift; MR0-15 : Vendor/Model/Runtime Substitution Conformance
Evidence Artifact	Model inventory; model version records; authority baseline records; drift detection logs; substitution conformance validation records

Business Continuity / Rollback

Lifecycle Governance Extension	Extend business continuity to include agentic workflow rollback; add cross-project reuse compliance; add context boundary validation
Relevant Object	MR0-09 : Cross-Project Reuse Compliance
Evidence Artifact	Reuse records; reset validation records; reauthorization records; context boundary records; rollback capability validation records

Interpretation: Lifecycle governance extends existing enterprise controls rather than replacing them. Integration ensures that lifecycle responsibility objects align with existing control frameworks and evidence artifacts.

17.4 Operationalizing Lifecycle Responsibility Objects

CHART - ADOPTION - ROADMAP

Stage 0-to-7 Adoption Roadmap

Stage 0-to-7 Adoption Roadmap

0

1

2

3

4

5

6

7

Inventory

Authority

Responsibility

Evidence

Privacy

Validation

Remediation

Scale

The timeline summarizes the staged adoption path from inventory through authority, responsibility, evidence, privacy, validation, remediation, and scale.

This is an implementation planning aid, not a deployment-readiness guarantee, procurement mandate, certification checklist, or legal compliance procedure.

Enterprises should operationalize lifecycle responsibility objects in stages. The following table defines a Stage 0-to-7 roadmap.

Table T-17-04: Adoption Roadmap by Stage

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Stage 0: Inventory

Goal	Inventory agentic behavior and identify lifecycle governance gaps
Key Actions	Identify where agents make decisions, use tools, or hand off work; assess whether model governance covers lifecycle responsibility; identify failure scenarios from Chapter 15 that apply
Output	Agentic behavior inventory; lifecycle governance gap analysis; failure scenario risk assessment
Success Indicator	Executive awareness of lifecycle governance gap; pilot project candidates identified

Stage 1: Authority

Goal	Define delegated authority boundaries and implement authority drift detection
Key Actions	Define authority grant records; implement tool-action authorization; add boundary violation alerts; implement drift detection monitoring
Output	Delegated authority boundary records; tool-action authorization records; boundary violation alerts; drift detection logs
Success Indicator	Authority boundaries are explicit and enforced; authority drift is detected and escalated

Stage 2: Responsibility

Goal	Map human roles to MAS responsibilities and implement accepted outcome compliance
Key Actions	Define role assignment records; implement acceptance records; add review and escalation workflow; implement closure acceptance workflow
Output	Human-role-to-MAS responsibility mapping; acceptance records; review records; escalation records; closure acceptance records
Success Indicator	Accountable human role exists for every agent outcome; accepted outcome compliance workflow operational

Stage 3: Evidence

Goal	Convert orchestration logs into partitioned evidence chains
Key Actions	Implement MAS evidence partitioning; add evidence export capability with selective disclosure; implement evidence chain integrity validation
Output	Partitioned evidence chains; evidence export logs; partition policy records; selective disclosure records; evidence integrity validation records
Success Indicator	Evidence chains can reconstruct authority, plan, confirmation, tool action, and accepted outcome; evidence can be selectively disclosed to auditors

Stage 4: Privacy

Goal	Map privacy and data subject rights across lifecycle surfaces
Key Actions	Implement privacy/GDPR lifecycle mapping; add data subject rights workflow; implement evidence minimization and selective disclosure policy
Output	Data flow records; retention policy records; data subject rights workflow records; evidence minimization records; selective disclosure policy records
Success Indicator	Privacy is mapped across lifecycle surfaces; data subject rights can be exercised without destroying evidence; evidence minimization policy operational

Stage 5: Validation

Goal	Prepare for evidence-based validation and qualified review
Key Actions	Implement privacy-preserving validation protocol (e.g., evidence-based validation pattern from Chapter 14); add disclosure profile and redaction profile; implement hash manifests and verdict hashes; plan qualified review, which may include internal governance review, audit review, validation provider review, or other context-appropriate reviewers
Output	Validation protocol records; disclosure profile records; redaction profile records; hash manifests; verdict hashes
Success Indicator	Evidence can be reviewed by qualified reviewers without exposing sensitive data; validation results are privacy-preserving and tamper-evident

Stage 6: Remediation

Goal	Operationalize incident, dispute, and remediation closure workflows
Key Actions	Implement incident records; add affected outcome list generation; implement remediation records; add closure acceptance workflow; implement customer impact assessment
Output	Incident records; affected outcome records; remediation records; closure acceptance records; customer impact assessment records
Success Indicator	Incidents are tracked with responsible owners; affected outcomes are identified; remediation is documented; closure is accepted by responsible owner

Stage 7: Scale

Goal	Scale lifecycle governance through integration and continuous improvement
Key Actions	Integrate lifecycle governance with existing enterprise controls (Table T-17-03); extend to all agentic workflows; implement continuous improvement and monitoring; where appropriate, contribute learnings to standards, industry, or regulatory dialogue
Output	Lifecycle governance integrated with audit, privacy, security, procurement, incident response, compliance, model risk, and business continuity; continuous improvement operational
Success Indicator	Lifecycle governance is operational for all agentic workflows; continuous improvement and monitoring systematic; enterprise may contribute learnings to standards, industry, or regulatory dialogue where appropriate

Interpretation: The Stage 0-to-7 roadmap provides a practical path from lifecycle governance awareness to scaled operational capability. Each stage builds on the previous stage and produces measurable outputs and success indicators.

17.5 Measuring Progress and Demonstrating Conformance

Enterprises should measure progress using lifecycle responsibility objects as evidence artifacts, not using RCCS-T/RCCS-M/ALCS as legal compliance scores. The following table clarifies how RCCS-T, RCCS-M, and ALCS should be used in adoption.

Table T-17-05: RCCS-T / RCCS-M / ALCS Use in Adoption

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Gap analysis	
RCCS-T RoLe	Identifies traditional regulatory compliance coverage gaps (e.g., RCCS-04 Record-Keeping, RCCS-06 Human Oversight)
RCCS-M RoLe	Identifies where regulatory obligations are not yet expressible through MROs
ALCS RoLe	Identifies lifecycle responsibility object gaps (e.g., ALCS-02 Delegated Authority Boundary, ALCS-04 Accepted Outcome Compliance, ALCS-06 Responsibility Transfer Across Agents)
Boundary	RCCS-T/RCCS-M/ALCS are analytical tools, not legal compliance scores; gaps require organizational interpretation and legal review

Pilot project baseline	
RCCS-T RoLe	Measures regulatory compliance coverage before and after pilot
RCCS-M RoLe	Measures MRO-adjusted regulatory coverage before and after pilot
ALCS RoLe	Measures lifecycle responsibility object maturity before and after pilot
Boundary	Baselines are internal analytical tools; they do not prove legal compliance or regulatory approval

System evaluation	
RCCS-T RoLe	Helps evaluate whether a system provides regulatory compliance coverage primitives
RCCS-M RoLe	Helps evaluate whether a system expresses those primitives as lifecycle responsibility objects
ALCS RoLe	Helps evaluate whether a system provides lifecycle responsibility object primitives
Boundary	Postures are analytical tools for system posture review (Chapters 11-12); they are not product rankings, certifications, or procurement recommendations

Audit readiness

RCCS-T RoLe	Maps traditional dimensions to audit control objectives
RCCS-M RoLe	Maps MRO-adjusted coverage to evidence objects auditors may request
ALCS RoLe	Maps lifecycle responsibility objects to evidence reconstruction needs
Boundary	Scores help prepare for audit but do not replace audit opinions, assurance reports, or legal compliance determinations

Evidence-based validation

RCCS-T RoLe	Identifies what regulatory compliance evidence should be validated
RCCS-M RoLe	Identifies what MRO-adjusted coverage should be validated
ALCS RoLe	Identifies what lifecycle responsibility objects should be validated
Boundary	Scores help scope validation but do not replace validation protocols, validation providers, or validation results

Continuous improvement

RCCS-T RoLe	Monitors regulatory compliance coverage over time
RCCS-M RoLe	Monitors whether coverage is becoming more lifecycle-object expressible
ALCS RoLe	Monitors lifecycle responsibility object maturity over time
Boundary	Scores are monitoring tools, not legal compliance scores; continuous improvement requires organizational governance, not just scoring

Interpretation: RCCS-T, RCCS-M, and ALCS are analytical tools that help enterprises measure progress and identify gaps. They are not legal compliance scores, certifications, or regulatory approvals. RCCS-M is author analytical and forward-looking, not current law or a regulator-approved benchmark. Enterprises should use RCCS-T/RCCS-M/ALCS to inform organizational governance decisions, not to replace legal review or regulatory interpretation.

17.6 Scaling from Pilot to Production

Scaling from pilot to production requires organizational commitment, integration with existing enterprise controls, and continuous improvement. Enterprises should:

- 1. Extend pilot learnings to additional workflows:** Use pilot project learnings to extend lifecycle governance to additional agentic workflows. Prioritize workflows that address Chapter 15 failure scenarios or that operate in regulated contexts.
- 1. Integrate with existing enterprise controls:** Use Table T-17-03 to integrate lifecycle governance with existing audit, privacy, security, procurement, incident response, compliance, model risk, and business continuity controls.
- 1. Operationalize all lifecycle responsibility objects:** Use Table T-17-04 to operationalize all lifecycle responsibility objects across all agentic workflows. Ensure that authority boundaries, responsibility

mappings, evidence chains, privacy mappings, validation protocols, and remediation workflows are operational.

1. **Implement continuous improvement:** Monitor lifecycle governance effectiveness using lifecycle responsibility objects as evidence artifacts. Use RCCS-T/RCCS-M/ALCS as analytical tools to identify gaps and measure progress. Implement continuous improvement workflows to address gaps and refine lifecycle governance as agentic systems evolve.
1. **Prepare for evidence-based validation:** Implement privacy-preserving validation protocols (e.g., evidence-based validation pattern from Chapter 14) to prepare for qualified review, which may include internal governance review, audit review, validation provider review, or other context-appropriate reviewers. Ensure that evidence can be reviewed without exposing sensitive data.
1. **Use Chapter 15 scenarios as test cases:** Use Chapter 15 enterprise failure scenarios as test cases for lifecycle governance design. Ensure that lifecycle governance prevents or mitigates the eight failure scenarios: authority boundary failure, evidence chain failure, accepted outcome failure, cross-project reuse failure, privacy validation failure, processor chain failure, vendor/runtime substitution failure, and remediation closure failure.
1. **Contribute learnings where appropriate:** As lifecycle governance matures, where appropriate, contribute learnings to standards, industry, or regulatory dialogue. Enterprises may share learnings with industry peers, participate in standards development, or engage with regulators where doing so fits their role, jurisdiction, and risk posture.

Scaling is not a one-time event. It is a continuous process that requires organizational commitment, integration with existing controls, and continuous improvement.

17.7 Bridge to Conclusion

Chapter 17 has presented an adoption roadmap for enterprises implementing AI Agent Lifecycle Governance. The roadmap provides a staged path from lifecycle governance awareness (Stage 0) to scaled operational capability (Stage 7). The roadmap integrates lifecycle governance with existing enterprise controls and uses RCCS-T/RCCS-M/ALCS as analytical tools, not legal compliance scores.

The bridge from adoption roadmap (Chapter 17) to conclusion (Chapter 18) is the recognition that AI Agent Lifecycle Governance is not a theoretical framework. It is a practical necessity for enterprises deploying agentic AI systems.

The white paper has defined the missing layer (Chapter 2), identified the sixteen Missing Regulatory Objects (Chapter 6), introduced RCCS-T, RCCS-M, and ALCS as analytical frameworks (Chapters 7-8), mapped existing systems (Chapters 11-12), examined evidence-based validation patterns (Chapter 14), demonstrated enterprise failure scenarios (Chapter 15), and presented an adoption roadmap (Chapter 17).

Chapter 18 will conclude the white paper with a restatement of the core thesis, a summary of contributions, a statement of boundaries, and a call to action.

18. Conclusion

MAIN RESULT The core conclusion is that AI agent compliance is lifecycle responsibility compliance.	METHOD / LENS Contribution summary, non-claims table, and call to action.
EVIDENCE BASIS Readers need the final boundary between analytical governance and legal/procurement claims.	KEY TAKEAWAY / BOUNDARY The paper is a governance framework and source-qualified analysis, not legal advice or certification.

Version: v0.3.2-FRC-R3

18.0 Core Thesis Restatement

AI Agent / MAS compliance is not only model compliance. It is lifecycle responsibility compliance.

This white paper has argued that a missing layer exists between model governance and enterprise trust. Model-centric governance addresses model accuracy, fairness, safety, and explainability. Enterprise trust requires audit readiness, regulatory compliance coverage, privacy protection, accountability, and dispute resolution. The missing layer is AI Agent Lifecycle Governance.

AI Agent Lifecycle Governance addresses the lifecycle responsibility objects that emerge when AI systems move from inference to action, from single-model execution to multi-agent collaboration, from human-supervised workflows to autonomous delegation. These lifecycle responsibility objects include:

- Human-role-to-MAS responsibility mapping
- Delegated authority boundaries
- Accepted outcome compliance
- Tool-action liability boundaries
- Responsibility transfer across agents
- Authority drift detection
- MAS evidence partitioning
- Cross-project reuse compliance
- Privacy/GDPR lifecycle mapping
- Privacy-preserving third-party validation
- Evidence minimization and selective disclosure
- Data subject rights vs evidence retention reconciliation
- Third-party processor/subprocessor chain mapping
- Vendor/model/runtime substitution conformance
- Incident, dispute, and remediation closure

Execution is not delivery, and execution is not compliance. Orchestration logs record execution. Evidence chains record lifecycle responsibility. Model governance addresses model behavior. Lifecycle governance

addresses responsibility for outcomes.

The transition from model governance to agentic lifecycle conformance is not a theoretical exercise. It is a practical necessity for enterprises deploying agentic AI systems in regulated contexts.

18.1 What This White Paper Provides

This white paper provides a framework for AI Agent Lifecycle Governance. The framework consists of eight core contributions.

Table T-18-01: White Paper Contribution Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

AI Agent Lifecycle Governance (Chapter 2)	
Purpose	Define the missing layer between model governance and enterprise trust
Practical Use	Enterprises can use this framework to identify lifecycle governance gaps and design lifecycle responsibility objects
Boundary	This is a governance framework, not a legal compliance procedure, certification program, or regulatory approval process
Sixteen Missing Regulatory Objects (Chapter 6)	
Purpose	Identify the lifecycle responsibility objects that model-centric governance does not address
Practical Use	Enterprises can use the sixteen MROs as a checklist for lifecycle governance design; auditors can use MROs to scope lifecycle governance reviews
Boundary	MROs are governance objects, not legal requirements; MRO presence does not prove legal compliance; MRO absence does not prove non-compliance
RCCS-T / RCCS-M – Traditional and MRO-Adjusted Regulatory Compliance Coverage Scores (Chapter 7)	
Purpose	Distinguish existing regulatory/governance coverage from MRO-adjusted regulatory coverage
Practical Use	Enterprises can use RCCS-T to identify traditional governance coverage gaps and RCCS-M to test whether obligations can be expressed through lifecycle responsibility objects
Boundary	RCCS-T and RCCS-M are analytical tools, not legal compliance scores; RCCS-M is author analytical and not current law, certification, regulator-approved benchmark, or procurement recommendation

ALCS — Agentic Lifecycle Conformance Score (Chapter 8)

Purpose	Measure lifecycle responsibility object maturity across fifteen dimensions
Practical Use	Enterprises can use ALCS to identify lifecycle responsibility object gaps; system evaluators can use ALCS to structure lifecycle conformance posture review
Boundary	ALCS is an analytical tool, not a legal compliance score; high ALCS does not prove legal compliance; low ALCS does not prove non-compliance

System-Category Mapping (Chapters 11-12)

Purpose	Map eight systems across categories by RCCS-T/RCCS-M/ALCS posture and lifecycle gaps
Practical Use	System mappings organize qualitative, source-qualified posture differences across system categories; system designers can use mappings to identify lifecycle governance design patterns
Boundary	System mappings are analytical tools, not product rankings, certifications, procurement recommendations, or vendor endorsements

Evidence-Based Validation Pattern (Chapter 14)

Purpose	Define a reusable methodology for privacy-preserving qualified review of lifecycle responsibility objects
Practical Use	Enterprises can use the evidence-based validation pattern to prepare for qualified review without exposing sensitive data; validation providers can use the pattern to design privacy-preserving validation protocols
Boundary	The pattern is a methodology, not a certification program; validation results do not prove legal compliance; Validation Lab is one non-certifying example, not the only validation path

Enterprise Failure Scenarios (Chapter 15)

Purpose	Use illustrative, hypothetical scenarios to show where lifecycle governance gaps may create predictable enterprise failure patterns
Practical Use	Enterprises can use failure scenarios as test cases for lifecycle governance design; risk managers can use scenarios to assess lifecycle governance risk
Boundary	Scenarios are illustrative and hypothetical, not claims that agentic AI is uniquely unsafe or that any specific vendor, platform, or organization failed; scenarios show where model-centric governance becomes insufficient, not that lifecycle governance eliminates all risk

Adoption Roadmap (Chapter 17)

Purpose	Provide a staged path from lifecycle governance awareness to scaled operational capability
Practical Use	Enterprises can use the adoption roadmap to operationalize lifecycle responsibility objects; governance teams can use the roadmap to integrate lifecycle governance with existing enterprise controls
Boundary	The roadmap is a practical guide, not a certification checklist, legal compliance procedure, or regulatory approval process

Interpretation: These eight contributions provide a framework for AI Agent Lifecycle Governance. The framework is designed to help enterprises identify lifecycle governance gaps, operationalize lifecycle responsibility objects, and integrate lifecycle governance with existing enterprise controls.

18.2 What This White Paper Does Not Provide

This white paper is a governance framework, not a legal compliance procedure, certification program, or regulatory approval process. The following table clarifies what this paper says and what it does not say.

Table T-18-02: Non-Claims and Boundaries

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Legal Compliance	
This Paper Says	Lifecycle governance addresses lifecycle responsibility objects that support regulatory compliance coverage; RCCS-T/RCCS-M/ALCS measure coverage and conformance gaps
This Paper Does Not Say	This paper proves legal compliance; high RCCS-T/RCCS-M/ALCS scores guarantee legal compliance; this paper provides legal advice; this paper interprets specific regulations for specific jurisdictions
Certification	
This Paper Says	Lifecycle governance can be evaluated using RCCS-T/RCCS-M/ALCS as analytical tools; evidence-based validation can support qualified review
This Paper Does Not Say	This paper is a certification program; RCCS-T/RCCS-M/ALCS issue compliance certificates; Validation Lab is a certification authority; this paper issues system or organization certificates
Regulatory Approval	
This Paper Says	Lifecycle governance aligns with regulatory requirements for human oversight, accountability, transparency, record-keeping, and contestability
This Paper Does Not Say	This paper is regulator-approved; this paper is the official compliance standard; regulators endorse RCCS-T/RCCS-M/ALCS; regulators require MPLP or Validation Lab
System Ranking	
This Paper Says	System mappings organize qualitative, source-qualified posture differences across system categories using responsibility semantics
This Paper Does Not Say	System mappings rank products by quality; system mappings recommend specific vendors; system mappings certify systems; other systems are non-compliant

MPLP

This Paper Says	MPLP is one lifecycle protocol path with limited RCCS-T product-platform coverage and strong author-analytical RCCS-M/ALCS mapping where protocol semantics express lifecycle responsibility objects
This Paper Does Not Say	MPLP is required for compliance; MPLP is the only correct path; MPLP issues compliance certificates; MPLP is regulator-approved; MPLP proves legal compliance; MPLP is superior to other systems

Validation Lab

This Paper Says	Validation Lab is one non-certifying evidence adjudication example that demonstrates the evidence-based validation pattern
This Paper Does Not Say	Validation Lab is a certification authority; Validation Lab issues compliance certificates; Validation Lab proves legal compliance; Validation Lab is the only validation path; Validation Lab is regulator-approved

Risk Elimination

This Paper Says	Lifecycle governance addresses lifecycle governance gaps that may create predictable enterprise failure patterns; lifecycle governance supports risk management
This Paper Does Not Say	Lifecycle governance eliminates all risk; agentic AI is uniquely unsafe; lifecycle governance guarantees operational effectiveness; lifecycle governance prevents all failures

Completeness

This Paper Says	This paper defines sixteen Missing Regulatory Objects and provides a framework for lifecycle governance
This Paper Does Not Say	This paper is complete; no additional lifecycle responsibility objects will emerge; this paper addresses all regulatory requirements; this paper replaces legal review

Interpretation: This white paper provides a governance framework, not legal advice, certification, regulatory approval, product ranking, or risk elimination. Enterprises must interpret the framework within their legal, regulatory, and operational context.

18.3 Call to Action

Enterprises deploying agentic AI systems should begin mapping lifecycle responsibility objects before scaling autonomous or multi-agent workflows. The adoption roadmap (Chapter 17) provides a staged path from lifecycle governance awareness to scaled operational capability.

Recommended first steps:

1. **Inventory agentic behavior:** Identify where agents make decisions, use tools, or hand off work. Assess whether model governance covers lifecycle responsibility.

1. **Assess lifecycle governance gaps:** Use the sixteen Missing Regulatory Objects (Chapter 6) as a checklist. Use RCCS-T/RCCS-M/ALCS (Chapters 7-8) as analytical tools to identify gaps.
1. **Select a pilot project:** Use pilot selection criteria (Table T-17-02) to choose a bounded, measurable, regulated, sponsored, and rollback-capable pilot project.
1. **Define authority boundaries:** Start with delegated authority boundaries ([MRO-02](#)). Define authority grant records, implement tool-action authorization, and add boundary violation alerts.
1. **Map human roles to MAS responsibilities:** Implement human-role-to-MAS responsibility mapping ([MRO-01](#)). Define role assignment records, implement acceptance records, and add review and escalation workflows.
1. **Convert logs into evidence chains:** Implement MAS evidence partitioning ([MRO-08](#)). Add evidence export capability with selective disclosure. Implement evidence chain integrity validation.
1. **Integrate with existing enterprise controls:** Use Table T-17-03 to integrate lifecycle governance with existing audit, privacy, security, procurement, incident response, compliance, model risk, and business continuity controls.
1. **Use Chapter 15 scenarios as test cases:** Use the eight enterprise failure scenarios as test cases for lifecycle governance design. Ensure that lifecycle governance prevents or mitigates authority boundary failure, evidence chain failure, accepted outcome failure, cross-project reuse failure, privacy validation failure, processor chain failure, vendor/runtime substitution failure, and remediation closure failure.

Lifecycle governance is not a one-time project. It is a continuous process that requires organizational commitment, integration with existing controls, and continuous improvement.

18.4 Closing Statement

AI Agent Lifecycle Governance is the missing layer between model governance and enterprise trust. Model-centric governance addresses model behavior. Lifecycle governance addresses responsibility for outcomes.

The sixteen Missing Regulatory Objects define the lifecycle responsibility objects that emerge when AI systems move from inference to action, from single-model execution to multi-agent collaboration, from human-supervised workflows to autonomous delegation. This paper does not merely compare current AI governance systems. It proposes that agentic and MAS compliance requires a new regulatory object layer: RCCS-T names traditional regulatory coverage; MRO names the missing lifecycle responsibility object layer; RCCS-M names MRO-adjusted regulatory coverage; ALCS names lifecycle conformance. Together, they show why future AI agent compliance must move from model compliance to lifecycle responsibility compliance.

System mappings organize qualitative, source-qualified posture differences across system categories. Evidence-based validation patterns support privacy-preserving qualified review. Enterprise failure scenarios illustrate where lifecycle governance gaps may create predictable failure patterns. The adoption roadmap provides a staged path from awareness to scaled operational capability.

Execution is not delivery, and execution is not compliance. Orchestration logs record execution. Evidence chains record lifecycle responsibility. Model governance addresses model behavior. Lifecycle governance addresses responsibility for outcomes.

Enterprises deploying agentic AI systems in regulated contexts should begin mapping lifecycle responsibility objects before scaling autonomous or multi-agent workflows. The transition from model governance to agentic lifecycle conformance is not a theoretical exercise. It is a practical necessity.

AI Agent / MAS compliance is not only model compliance. It is lifecycle responsibility compliance.

Appendix A — Machine-Readable Object Templates

Version: v0.3.2-FRC-R3

A.0 Template Boundary

This appendix provides illustrative machine-readable templates for the sixteen Missing Regulatory Objects defined in Chapter 6. These templates are reference structures, not formal schemas, legal requirements, or certification requirements.

Boundary statement:

- **These templates are illustrative only.** They demonstrate how lifecycle responsibility objects might be represented in machine-readable form. They are not formal MPLP schemas, not legal requirements, and not certification requirements.
- **These templates are not prescriptive.** Organizations may use different field names, data types, or structures. The templates show one possible approach to representing lifecycle responsibility objects.
- **These templates do not prove compliance.** The presence of these fields in a system does not prove legal compliance, regulatory approval, or certification. Organizations must interpret lifecycle responsibility objects within their legal, regulatory, and operational context.
- **These templates are not MPLP-specific.** While MPLP (Multi-Agent Lifecycle Protocol) is one protocol path that can express lifecycle responsibility objects, these templates are protocol-agnostic. Other protocols, platforms, or custom implementations may use different representations.

A.1 Shared Field Conventions

All sixteen MRO templates share a common set of fields to support lifecycle traceability, evidence partitioning, and cross-object relationships.

Shared fields:

Illustrative non-normative template

```
object_id: unique identifier for this object instance
object_type: MRO type (e.g., "MRO-01", "MRO-02")
lifecycle_id: identifier linking this object to a specific lifecycle instance
project_id: identifier for the project or regulated context
```

object_version: version number for this object instance

related_human_role_id: identifier for the human role associated with this object

related_agent_role_id: identifier for the agent role associated with this object

authority_scope: scope of authority for this object (e.g., "draft", "send", "commit")

risk_class: risk classification for this object (e.g., "low", "medium", "high")

evidence_pointer: pointer to evidence artifacts supporting this object

privacy_treatment: privacy treatment for this object (e.g., "redact_pii", "selective_disclosure")

status: current status (e.g., "active", "closed", "disputed")

created_at: timestamp when this object was created

updated_at: timestamp when this object was last updated

closure_reason: reason for closure (if status is "closed")

Object-specific fields: Each MRO template includes 3-6 object-specific fields relevant to that MRO's purpose.

All example values are non-normative placeholders and do not establish legal compliance, liability allocation, certification, or completed validation.

For readability, the examples below present object-specific fields under a comment block. Implementations may alternatively nest them under `object_specific_fields`.

A.2 MRO Template Index

The following table indexes the sixteen MRO templates and their object-specific fields.

Table T-A-01: Appendix A Template Index

MRO-01	
MRO ID	MRO-01
Object Name	Human Role to MAS Responsibility Mapping
Template Purpose	Map human roles to MAS responsibilities with explicit acceptance records
Object-Specific Fields	human_role_name, mas_responsibility_scope, acceptance_required, escalation_policy, review_frequency
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-02	
MRO ID	MRO-02
Object Name	Delegated Authority Boundary
Template Purpose	Define explicit authority boundaries for agent actions
Object-Specific Fields	authority_grant_type, tool_action_permissions, boundary_violation_policy, reauthorization_trigger, drift_detection_enabled
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-03	
MRO ID	MRO-03
Object Name	Agent Role is not Human Role
Template Purpose	Distinguish agent responsibility surfaces from human roles
Object-Specific Fields	agent_role_name, agent_responsibility_surface, human_role_distinction, accountability_mapping, role_confusion_risk
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-04	
MRO ID	MRO-04
Object Name	Accepted Outcome Compliance
Template Purpose	Record explicit acceptance of agent outcomes by accountable human roles
Object-Specific Fields	outcome_description, acceptance_status, accepted_by_role, acceptance_timestamp, rejection_reason, escalation_path
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-05	
MRO ID	MRO-05
Object Name	Tool-Action Liability Boundary
Template Purpose	Record tool actions and liability boundaries
Object-Specific Fields	tool_name, action_type, liability_assignment, tool_provider_agreement, authorization_record, action_reversibility
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-06	
MRO ID	MRO-06
Object Name	Responsibility Transfer Across Agents
Template Purpose	Record responsibility transfer and constraint inheritance across agent handoffs
Object-Specific Fields	source_agent_id, target_agent_id, transfer_timestamp, constraint_inheritance, handoff_validation, responsibility_acceptance
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-07

MRO ID	MRO-07
Object Name	Authority Drift
Template Purpose	Detect and record authority drift over time
Object-Specific Fields	authority_baseline, current_authority_scope, drift_detected, drift_magnitude, reauthorization_required, drift_alert_policy
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-08

MRO ID	MRO-08
Object Name	MAS Evidence Partitioning
Template Purpose	Partition evidence by project, customer, jurisdiction, or data subject
Object-Specific Fields	partition_key, partition_type, evidence_export_policy, selective_disclosure_enabled, partition_boundary, cross_partition_access_policy
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-09

MRO ID	MRO-09
Object Name	Cross-Project Reuse Compliance
Template Purpose	Validate reuse compliance across regulated contexts
Object-Specific Fields	source_project_id, target_project_id, reuse_type, reset_validation, reauthorization_required, context_boundary_check
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-10

MRO ID	MRO-10
Object Name	Privacy / GDPR Lifecycle Mapping
Template Purpose	Map data flow, retention, and subject rights across agent lifecycle
Object-Specific Fields	data_flow_record, retention_policy, data_subject_rights_workflow, privacy_impact_assessment, gdpr_article_mapping, data_minimization_policy
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-11

MRO ID	MRO-11
Object Name	Privacy-Preserving Third-Party Validation
Template Purpose	Support privacy-preserving validation without exposing sensitive data
Object-Specific Fields	validation_protocol, disclosure_profile, redaction_profile, hash_manifest, verdict_hash, validation_provider_agreement
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-12

MRO ID	MRO-12
Object Name	Evidence Minimization and Selective Disclosure
Template Purpose	Implement evidence minimization and selective disclosure policies
Object-Specific Fields	retention_policy, disclosure_policy, minimization_rules, selective_disclosure_log, evidence_expiry, disclosure_justification
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-13

MRO ID	MRO-13
Object Name	Data Subject Rights vs Evidence Retention
Template Purpose	Reconcile data subject rights with evidence retention requirements
Object-Specific Fields	data_subject_request_type, evidence_retention_requirement, reconciliation_policy, legal_hold_status, erasure_exception, retention_justification
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-14

MRO ID	MRO-14
Object Name	Third-Party Processor / Subprocessor Chain
Template Purpose	Map processor and subprocessor chain with responsibility assignment
Object-Specific Fields	processor_name, processor_role, subprocessor_chain, data_processing_agreement, responsibility_assignment, processor_liability
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-15	
MRO ID	MRO-15
Object Name	Vendor / Model / Runtime Substitution Conformance
Template Purpose	Validate conformance after vendor, model, or runtime substitution
Object-Specific Fields	substitution_type, previous_vendor, new_vendor, conformance_validation, regression_testing, evidence_integrity_check
Boundary	Illustrative only; not legal requirement; not certification requirement

MRO-16	
MRO ID	MRO-16
Object Name	Incident, Dispute, and Remediation Closure
Template Purpose	Record incident, dispute, and remediation closure with responsible owner
Object-Specific Fields	incident_type, affected_outcomes, remediation_action, closure_status, responsible_owner, customer_impact_assessment
Boundary	Illustrative only; not legal requirement; not certification requirement

A.3 MRO Templates

MRO-01: Human Role to MAS Responsibility Mapping

Illustrative non-normative template
<pre># MRO-01: Human Role to MAS Responsibility Mapping object_id: "mro01-uuid-example" object_type: "MRO-01" lifecycle_id: "lifecycle-uuid" project_id: "project-uuid" object_version: 1 related_human_role_id: "human-role-uuid" related_agent_role_id: "agent-role-uuid" authority_scope: "review_and_accept" risk_class: "medium" evidence_pointer: "evidence-chain-uuid" privacy_treatment: "no_pii" status: "active" created_at: "2026-05-10T10:00:00Z" updated_at: "2026-05-10T10:00:00Z" closure_reason: null # Object-specific fields human_role_name: "Customer Support Manager" mas_responsibility_scope: "Review and accept agent-drafted customer responses" acceptance_required: true escalation_policy: "Escalate to Senior Manager if customer dispute" review_frequency: "daily"</pre>

MRO-02: Delegated Authority Boundary

Illustrative non-normative template

```
# MRO-02: Delegated Authority Boundary
object_id: "mro02-uuid-example"
object_type: "MRO-02"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "draft_only"
risk_class: "high"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "no_pii"
status: "active"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:00:00Z"
closure_reason: null

# Object-specific fields
authority_grant_type: "explicit_grant"
tool_action_permissions: ["read_crm", "draft_email"]
boundary_violation_policy: "alert_and_block"
reauthorization_trigger: "model_version_change"
drift_detection_enabled: true
```

MRO-03: Agent Role is not Human Role

Illustrative non-normative template

```
# MRO-03: Agent Role is not Human Role
object_id: "mro03-uuid-example"
object_type: "MRO-03"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "agent_responsibility_surface"
risk_class: "medium"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "no_pii"
status: "active"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:00:00Z"
closure_reason: null

# Object-specific fields
agent_role_name: "Customer Support Agent"
agent_responsibility_surface: "Draft responses; recommend actions; escalate disputes"
human_role_distinction: "Human role: Review, accept, and send responses; resolve disputes"
accountability_mapping: "Agent recommends; human accepts; human accountable"
role_confusion_risk: "low"
```

MRO-04: Accepted Outcome Compliance

Illustrative non-normative template

```
# MRO-04: Accepted Outcome Compliance
object_id: "mro04-uuid-example"
object_type: "MRO-04"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
```

```
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "outcome_acceptance"
risk_class: "high"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "redact_pii"
status: "accepted"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:15:00Z"
closure_reason: null

# Object-specific fields
outcome_description: "Customer refund request approved"
acceptance_status: "accepted"
accepted_by_role: "Customer Support Manager"
acceptance_timestamp: "2026-05-10T10:15:00Z"
rejection_reason: null
escalation_path: "Senior Manager if customer disputes"
```

MRO-05: Tool-Action Liability Boundary

Illustrative non-normative template

```
# MRO-05: Tool-Action Liability Boundary
object_id: "mro05-uuid-example"
object_type: "MRO-05"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "tool_action"
risk_class: "high"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "redact_pii"
status: "active"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:00:00Z"
closure_reason: null

# Object-specific fields
tool_name: "CRM API"
action_type: "update_customer_record"
liability_assignment: "responsibility_boundary_recorded"
tool_provider_agreement: "agreement-uuid"
authorization_record: "authorization-uuid"
action_reversibility: "reversible_with_audit_trail"
```

MRO-06: Responsibility Transfer Across Agents

Illustrative non-normative template

```
# MRO-06: Responsibility Transfer Across Agents
object_id: "mro06-uuid-example"
object_type: "MRO-06"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "handoff"
risk_class: "medium"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "no_pii"
status: "active"
```

```
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:00:00Z"
closure_reason: null

# Object-specific fields
source_agent_id: "agent-1-uuid"
target_agent_id: "agent-2-uuid"
transfer_timestamp: "2026-05-10T10:00:00Z"
constraint_inheritance: ["no_refund_over_1000", "require_manager_approval"]
handoff_validation: "handoff_check_recorded"
responsibility_acceptance: "accepted_by_agent_2"
```

MRO-07: Authority Drift

Illustrative non-normative template

```
# MRO-07: Authority Drift
object_id: "mro07-uuid-example"
object_type: "MRO-07"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "draft_only"
risk_class: "high"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "no_pii"
status: "drift_detected"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T11:00:00Z"
closure_reason: null

# Object-specific fields
authority_baseline: "draft_only"
current_authority_scope: "draft_and_send"
drift_detected: true
drift_magnitude: "high"
reauthorization_required: true
drift_alert_policy: "alert_governance_team"
```

MRO-08: MAS Evidence Partitioning

Illustrative non-normative template

```
# MRO-08: MAS Evidence Partitioning
object_id: "mro08-uuid-example"
object_type: "MRO-08"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "evidence_partition"
risk_class: "medium"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "selective_disclosure"
status: "active"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:00:00Z"
closure_reason: null

# Object-specific fields
partition_key: "customer-uuid"
partition_type: "customer"
evidence_export_policy: "selective_disclosure_only"
```

```
selective_disclosure_enabled: true
partition_boundary: "customer_data_only"
cross_partition_access_policy: "deny"
```

MRO-09: Cross-Project Reuse Compliance

Illustrative non-normative template

```
# MRO-09: Cross-Project Reuse Compliance
object_id: "mro09-uuid-example"
object_type: "MRO-09"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "reuse_validation"
risk_class: "high"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "context_reset"
status: "reuse_review_recorded"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:00:00Z"
closure_reason: null

# Object-specific fields
source_project_id: "project-1-uuid"
target_project_id: "project-2-uuid"
reuse_type: "prompt_template"
reset_validation: "reset_check_recorded"
reauthorization_required: true
context_boundary_check: "context_boundary_check_recorded"
```

MRO-10: Privacy / GDPR Lifecycle Mapping

Illustrative non-normative template

```
# MRO-10: Privacy / GDPR Lifecycle Mapping
object_id: "mro10-uuid-example"
object_type: "MRO-10"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "privacy_mapping"
risk_class: "high"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "gdpr_lifecycle_mapping_required"
status: "active"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:00:00Z"
closure_reason: null

# Object-specific fields
data_flow_record: "data-flow-uuid"
retention_policy: "30_days"
data_subject_rights_workflow: "rights_workflow_mapping_required"
privacy_impact_assessment: "pia-uuid"
gdpr_article_mapping: ["Article 6 mapping required", "Article 17 mapping required"]
data_minimization_policy: "collect_only_necessary"
```


MRO-11: Privacy-Preserving Third-Party Validation

Illustrative non-normative template

```
# MRO-11: Privacy-Preserving Third-Party Validation
object_id: "mro11-uuid-example"
object_type: "MRO-11"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "validation"
risk_class: "medium"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "hash_based_validation"
status: "validation_recorded"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:30:00Z"
closure_reason: null

# Object-specific fields
validation_protocol: "evidence_based_validation_pattern"
disclosure_profile: "minimal_disclosure"
redaction_profile: "redact_all_pii"
hash_manifest: "hash-manifest-uuid"
verdict_hash: "sha256-verdict-hash"
validation_provider_agreement: "agreement-uuid"
```

MRO-12: Evidence Minimization and Selective Disclosure

Illustrative non-normative template

```
# MRO-12: Evidence Minimization and Selective Disclosure
object_id: "mro12-uuid-example"
object_type: "MRO-12"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "evidence_minimization"
risk_class: "medium"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "minimized"
status: "active"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:00:00Z"
closure_reason: null

# Object-specific fields
retention_policy: "30_days"
disclosure_policy: "selective_disclosure_only"
minimization_rules: ["remove_pii", "aggregate_metrics"]
selective_disclosure_log: "disclosure-log-uuid"
evidence_expiry: "2026-06-10T10:00:00Z"
disclosure_justification: "audit_request"
```

MRO-13: Data Subject Rights vs Evidence Retention

Illustrative non-normative template

```
# MRO-13: Data Subject Rights vs Evidence Retention
object_id: "mro13-uuid-example"
object_type: "MRO-13"
```

```

lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "data_subject_rights"
risk_class: "high"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "erasure_exception"
status: "reconciled"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:00:00Z"
closure_reason: null

# Object-specific fields
data_subject_request_type: "erasure"
evidence_retention_requirement: "legal_hold"
reconciliation_policy: "pseudonymize_instead_of_erase"
legal_hold_status: "active"
erasure_exception: "legal_obligation"
retention_justification: "ongoing_dispute"

```

MRO-14: Third-Party Processor / Subprocessor Chain

Illustrative non-normative template

```

# MRO-14: Third-Party Processor / Subprocessor Chain
object_id: "mro14-uuid-example"
object_type: "MRO-14"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "processor_chain"
risk_class: "high"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "processor_agreement"
status: "active"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:00:00Z"
closure_reason: null

# Object-specific fields
processor_name: "Cloud AI Provider"
processor_role: "model_inference"
subprocessor_chain: ["Vector DB Provider", "Monitoring Platform"]
data_processing_agreement: "dpa-uuid"
responsibility_assignment: "processor_responsibility_reference"
processor_liability: "gdpr_article_28_mapping_required"

```

MRO-15: Vendor / Model / Runtime Substitution Conformance

Illustrative non-normative template

```

# MRO-15: Vendor / Model / Runtime Substitution Conformance
object_id: "mro15-uuid-example"
object_type: "MRO-15"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "substitution_validation"
risk_class: "high"
evidence_pointer: "evidence-chain-uuid"

```

```
privacy_treatment: "no_pii"
status: "conformance_check_recorded"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T10:00:00Z"
closure_reason: null

# Object-specific fields
substitution_type: "model_provider"
previous_vendor: "Vendor A"
new_vendor: "Vendor B"
conformance_validation: "conformance_check_recorded"
regression_testing: "regression_check_recorded"
evidence_integrity_check: "evidence_integrity_check_recorded"
```

MRO-16: Incident, Dispute, and Remediation Closure

Illustrative non-normative template

```
# MRO-16: Incident, Dispute, and Remediation Closure
object_id: "mro16-uuid-example"
object_type: "MRO-16"
lifecycle_id: "lifecycle-uuid"
project_id: "project-uuid"
object_version: 1
related_human_role_id: "human-role-uuid"
related_agent_role_id: "agent-role-uuid"
authority_scope: "incident_closure"
risk_class: "high"
evidence_pointer: "evidence-chain-uuid"
privacy_treatment: "redact_pii"
status: "closed"
created_at: "2026-05-10T10:00:00Z"
updated_at: "2026-05-10T12:00:00Z"
closure_reason: "remediation_complete"

# Object-specific fields
incident_type: "prompt_bug"
affected_outcomes: ["outcome-1-uuid", "outcome-2-uuid"]
remediation_action: "prompt_fixed_and_redeployed"
closure_status: "closed"
responsible_owner: "Engineering Manager"
customer_impact_assessment: "low_impact_2_customers_notified"
```

Appendix B — Scorecard Template

Version: v0.3.2-FRC-R3

B.0 Scorecard Boundary

This appendix provides reusable RCCS-T / RCCS-M / ALCS scorecard templates for system evaluation and gap analysis. These templates are analytical tools, not legal compliance scores, certifications, or regulatory approvals. The three-profile view is primary; optional composite views are worksheet aids only.

Boundary statement:

- **Scorecards are analytical tools only.** RCCS-T measures traditional regulatory compliance coverage, RCCS-M measures MRO-adjusted regulatory coverage, and ALCS measures lifecycle responsibility object maturity. They do not prove legal compliance, regulatory approval, or certification.
 - **High scores do not guarantee compliance.** A high RCCS-T, RCCS-M, or ALCS score indicates that a system provides primitives for regulatory compliance coverage or lifecycle responsibility objects. It does not prove that the system is legally compliant, operationally effective, or enterprise-ready.
 - **Low scores do not prove non-compliance.** A low RCCS-T, RCCS-M, or ALCS score indicates gaps in traditional regulatory coverage, MRO-adjusted coverage, or lifecycle responsibility object maturity. It does not prove that the system is non-compliant, unsafe, or unsuitable for deployment.
 - **Scores are context-independent.** RCCS-T, RCCS-M, and ALCS scores measure system capability, not organizational practice. Organizations must interpret scores within their legal, regulatory, and operational context.
 - **Scores require evidence.** All scores must be supported by evidence. The evidence level (**L1** - **L5**) affects the score multiplier. Scores without evidence are not valid.
 - **Scores are not product rankings.** Scorecards compare systems by responsibility semantics, not product quality, market leadership, or procurement suitability.
-

B.1 Dimension-Level Scoring Template

The following template supports dimension-level scoring for RCCS-T, RCCS-M, and ALCS dimensions.

RCCS-T and RCCS-M use the same dimension family but different scoring lenses. Scorecards should retain both values where evidence permits: RCCS-T for traditional governance/control coverage, RCCS-M for MRO-adjusted lifecycle responsibility object expression. The justification field should state which lens is being scored and whether the evidence supports the lens directly or only through analytical inference.

Table T-B-01: RCCS-T / RCCS-M / ALCS Dimension Scorecard Template

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Example System

Profile / Lens	RCCS-T
Dimension	RCCS-04 : Record-Keeping
Raw Score 0-5	4
Evidence Level	L2 : Product Docs
Multiplier	0.85
Adjusted Score	68.0
Justification	System provides audit trail API and log export capability documented in official product docs
Score-Delta Note	Strong traditional record-keeping surface
Boundary Note	Score measures capability, not organizational practice; does not prove legal compliance

Example System

Profile / Lens	RCCS-M
Dimension	RCCS-04 : Record-Keeping
Raw Score 0-5	2
Evidence Level	L5 : Author Inference
Multiplier	0.35
Adjusted Score	14.0
Justification	Public evidence does not show records partitioned by role, authority boundary, agent/tool action, privacy boundary, and accepted outcome
Score-Delta Note	Traditional logs do not automatically satisfy MRO evidence partitioning
Boundary Note	RCCS-M is author analytical where object semantics are inferred

Example System

Profile / Lens	ALCS
Dimension	ALCS-02 : Delegated Authority Boundary
Raw Score 0-5	3
Evidence Level	L3 : Audit Report
Multiplier	0.75
Adjusted Score	45.0
Justification	System supports authority grant records and boundary violation alerts per third-party audit report
Score-Delta Note	Lifecycle conformance differs from RCCS-M regulatory coverage
Boundary Note	Score measures capability, not operational enforcement; does not prove regulatory approval

Example System

Profile / Lens	RCCS-T
Dimension	RCCS-06 : Human Oversight
Raw Score 0-5	2
Evidence Level	L4 : Vendor Claim
Multiplier	0.55
Adjusted Score	22.0
Justification	Vendor claims human-in-the-loop capability but no public documentation or audit evidence
Score-Delta Note	Weak evidence limits both RCCS-T and RCCS-M confidence
Boundary Note	Score reflects vendor claim only; requires validation; does not prove human oversight is operational

Scoring rubric:

- **0:** No capability; no evidence
- **1:** Minimal capability; weak evidence
- **2:** Partial capability; some evidence
- **3:** Moderate capability; documented evidence
- **4:** Strong capability; validated evidence
- **5:** Complete capability demonstrated by implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts. Legal text may establish a regulatory baseline but does not by itself prove implementation.

Evidence level multipliers:

- **L1 (Binding Legal Text):** 1.00

- **L2 (Product Docs):** 0.85
- **L3 (Audit Report):** 0.75
- **L4 (Vendor Claim):** 0.55
- **L5 (Author Inference):** 0.35

B.2 Evidence-Level Adjustment Template

The following template supports evidence-level adjustment for dimension scores.

Evidence-level adjustment policy:

1. **Start with raw score (0-5)** based on system capability assessment
2. **Identify evidence level (L1 - L5)** based on source type
3. **Apply multiplier** to calculate adjusted score using normalized formula: $(\text{Raw Score} / 5) \times 100 \times \text{Evidence Multiplier}$
4. **Document justification** explaining why the raw score and evidence level were assigned
5. **Add boundary note** clarifying what the score does and does not prove

Example:

Illustrative non-normative template

```
dimension: RCCS-04 Record-Keeping
profile_lens: RCCS-T
raw_score: 4
evidence_level: L2
multiplier: 0.85
adjusted_score: 68.0 # (4/5) × 100 × 0.85
justification: "System provides audit trail API and log export capability documented in official product docs"
score_delta_note: "RCCS-M may be lower unless records are partitioned by authority, agent/tool action, evidence pointer, privacy boundary, and accepted outcome"
boundary_note: "Score measures capability, not organizational practice; does not prove legal compliance"
```

B.3 Three-Profile and Optional Composite Template

The following template supports the primary three-profile view and optional composite calculations. A single composite score should not be used as the main result display because it can hide the difference between traditional governance coverage and MRO-adjusted lifecycle coverage.

Table T-B-02: Three-Profile Analytical Score Worksheet

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Example System A

RCCS-T 0-100	65.0
RCCS-M 0-100	72.0
ALCS 0-100	55.0
Optional Traditional Composite	59.0
Optional Agentic Composite	61.8
Interpretation Boundary	Analytical worksheet view; not legal compliance proof or procurement ranking

Example System B

RCCS-T 0-100	82.0
RCCS-M 0-100	60.0
ALCS 0-100	58.0
Optional Traditional Composite	67.6
Optional Agentic Composite	58.8
Interpretation Boundary	Analytical worksheet view; not legal compliance proof or deployment readiness threshold

Example System C

RCCS-T 0-100	48.0
RCCS-M 0-100	78.0
ALCS 0-100	80.0
Optional Traditional Composite	67.2
Optional Agentic Composite	79.2
Interpretation Boundary	Analytical worksheet view; not vendor ranking or certification signal

Optional composite formulas:

Illustrative non-normative template

Traditional Composite = (RCCS-T × 0.40) + (ALCS × 0.60)
Agentic Composite = (RCCS-M × 0.40) + (ALCS × 0.60)

Interpretation guidance:

- **RCCS-T:** Traditional regulatory and governance coverage
- **RCCS-M:** MRO-adjusted regulatory coverage; author analytical and forward-looking

- **ALCS:** Lifecycle responsibility conformance maturity
- **Optional composites:** Secondary worksheet views only; never a ranking, procurement selection signal, legal compliance proof, certification signal, regulatory approval signal, or procurement recommendation

Any numerical bands are analytical interpretation bands only. They are not legal compliance thresholds, deployment readiness thresholds, procurement thresholds, or certification cutoffs.

B.4 Score Delta Record Template

The following template supports tracking score changes over time.

Table T-B-03: Score Delta Record

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

RCCS-04 : Record-Keeping	
Previous Adjusted Score 0-100	54.4
Updated Adjusted Score 0-100	68.0
Reason for Change	System added evidence export API with selective disclosure capability
Evidence Change	L2 product docs updated with new API documentation
Reviewer	Governance Team
Date	2026-05-10

ALCS-02 : Delegated Authority Boundary	
Previous Adjusted Score 0-100	40.5
Updated Adjusted Score 0-100	52.5
Reason for Change	System added authority drift detection and reauthorization workflow
Evidence Change	L3 audit report recorded new capability
Reviewer	Compliance Team
Date	2026-05-10

RCCS-06: Human Oversight	
Previous Adjusted Score 0-100	13.2
Updated Adjusted Score 0-100	34.0
Reason for Change	Vendor published official documentation for human-in-the-loop capability
Evidence Change	Evidence level upgraded from L4 (vendor claim) to L2 (product docs)
Reviewer	Governance Team
Date	2026-05-10

Score delta policy:

- **Track all score changes** with reason for change, evidence change, reviewer, and date
- **Require reviewer approval** for all score changes
- **Document evidence change** explaining why the score changed (new capability, new evidence, evidence level upgrade, etc.)
- **Preserve score history** to support audit trail and continuous improvement

B.5 Negative Control Template

The following template supports negative control to prevent overclaim.

Table T-B-04: Negative Control Checklist

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

"System provides human oversight"	
Adjacent Capability	System has approval button in UI
Why It Does Not Count	Approval button ≠ human-role-to-MAS responsibility mapping; no role assignment records; no acceptance records; no escalation workflow
Required Evidence	Human-role-to-MAS responsibility mapping (MRO-01); acceptance records (MRO-04); escalation workflow
Reviewer Note	Approval button is UI feature, not lifecycle governance object

"System provides evidence chain"

Adjacent Capability	System has logs
Why It Does Not Count	Logs ≠ partitioned evidence chain; no evidence export capability; no selective disclosure; no evidence integrity validation
Required Evidence	MAS evidence partitioning (MRO-08); evidence export capability; selective disclosure; evidence integrity validation
Reviewer Note	Logs are execution records, not lifecycle responsibility evidence

"System provides authority boundary"

Adjacent Capability	System has access control
Why It Does Not Count	Access control ≠ delegated authority boundary; no authority grant records; no boundary violation alerts; no drift detection
Required Evidence	Delegated authority boundary (MRO-02); authority grant records; boundary violation alerts; drift detection
Reviewer Note	Access control is security feature, not lifecycle governance object

"System provides accepted outcome compliance"

Adjacent Capability	Orchestrator marks task complete
Why It Does Not Count	Task completion ≠ accepted outcome; no accountable human role; no acceptance records; no review workflow
Required Evidence	Accepted outcome compliance (MRO-04); human-role-to-MAS responsibility mapping (MRO-01); acceptance records; review workflow
Reviewer Note	Execution completion ≠ accepted outcome

"System provides privacy-preserving validation"

Adjacent Capability	System has data export
Why It Does Not Count	Data export ≠ privacy-preserving validation; no disclosure profile; no redaction profile; no hash manifests; no verdict hashes
Required Evidence	Privacy-preserving third-party validation (MRO-11); disclosure profile; redaction profile; hash manifests; verdict hashes
Reviewer Note	Data export is data management feature, not privacy-preserving validation protocol

Negative control policy:

- **Distinguish adjacent capabilities from lifecycle responsibility objects.** Many systems have features that are adjacent to lifecycle governance but do not constitute lifecycle responsibility objects.
- **Require explicit evidence for lifecycle responsibility objects.** Approval buttons, logs, access control, task completion, and data export are not lifecycle responsibility objects

unless they include the required fields and workflows.

- **Prevent overclaim.** Negative control prevents scoring systems for adjacent capabilities that do not meet lifecycle responsibility object requirements.
- **Document why adjacent capabilities do not count.** Negative control checklist explains why specific capabilities do not count toward RCCS-T/RCCS-M/ALCS scores.

Appendix C — References and Source Notes

Version: v0.3.2-FRC-R3

C.0 Source Boundary

This appendix provides a source register and citation guidance for the Global AI Compliance White Paper 2026. Phase 1D-11 closes the high-risk citation/access/source-pointer blockers by replacing inaccessible sources, narrowing claims, downgrading lifecycle interpretations to L5, and retaining explicit non-blocking limitations where appropriate.

Boundary statement:

- **This appendix is a source register and claim-binding pointer, not legal proof.** The source register lists sources used in the white paper and their current validation status. Phase 1D-11 completed high-risk source closure for public research edition preparation.
- **Citation source pointers are classified by claim support.** L2 product documentation sources support visible capability surfaces only. Lifecycle-object interpretations remain L5 analytical unless the source directly supports the lifecycle object.
- **Evidence levels affect confidence.** L1 sources (binding legal text, formal standards) establish baseline obligations but do not prove implementation. L2 sources (official product/protocol documentation) support documented capability surfaces but do not prove deployment practice, audit success, legal compliance, or lifecycle responsibility-object implementation. L3 - L5 sources have lower or different confidence and must be marked appropriately.
- **Source validation is not legal compliance proof.** Validated sources support the white paper's analytical framework. They do not prove legal compliance, regulatory approval, or certification.

C.1 Evidence Level Policy

The following table defines evidence levels and their permitted use.

Table T-C-01: Evidence Level Policy

L1	
Level	L1
Source Type	Binding legal text (EU AI Act, GDPR, Colorado AI Act, etc.)
Permitted Use	Regulatory baseline; legal obligation statements
Confidence	Highest
Limitation	Legal text requires legal interpretation; does not prove organizational compliance

L1	
Level	L1
Source Type	Formal standards (NIST AI RMF, ISO/IEC 42001, Singapore IMDA, W3C PROV, W3C VC)
Permitted Use	Framework baseline; standard requirements
Confidence	Highest
Limitation	Standards are voluntary unless mandated by regulation; adoption does not prove compliance

L2	
Level	L2
Source Type	Official product documentation (vendor docs, protocol specs, GitHub repos)
Permitted Use	System capability claims; feature descriptions
Confidence	High
Limitation	Product docs describe capability, not operational effectiveness; require validation; may change without notice

L3	
Level	L3
Source Type	Third-party audit reports, academic papers, industry reports
Permitted Use	Supplemental confidence; independent assessment
Confidence	Medium
Limitation	Audit reports are point-in-time assessments; academic papers may not reflect current state; industry reports may have bias

L4	
Level	L4
Source Type	Vendor claims, marketing materials, blog posts
Permitted Use	Positioning only; not capability claims
Confidence	Low
Limitation	Vendor claims require validation; marketing materials may overstate capability; blog posts are not authoritative

L5	
Level	L5
Source Type	Author inference, analytical framework, logical derivation
Permitted Use	Framework design; gap analysis; analytical tools
Confidence	Lowest
Limitation	Author inference must be marked clearly; does not replace evidence; requires organizational interpretation

Evidence level multipliers for RCCS-T/RCCS-M/ALCS scoring:

- L1 : 1.00
- L2 : 0.85
- L3 : 0.75
- L4 : 0.55
- L5 : 0.35

C.2 Required Source Groups

The white paper uses five source groups: L1 Binding Legal/Regulatory Sources, L1 Formal Standards, L2 Official Product Documentation, L3 Third-Party Reports, and L5 Author Inference.

L1: Binding Legal/Regulatory Sources

Source ID	Source Name	Type	Used In	Current Status
EU-AI-ACT	EU AI Act (Regulation (EU) 2024/1689) Articles 9, 10, 11, 12, 13, 14, 15, 72	Legal text	Chapter 4	✓ Referenced
GDPR	GDPR (Regulation (EU) 2016/679) Article 5, Article 17, Article 28	Legal text	Chapter 4	✓ Referenced
COLORADO-AI-ACT	Colorado AI Act (SB25B-004)	Legal text	Chapter 4	✓ Referenced

L1: Formal Standards

Source ID	Source Name	Type	Used In	Current Status
NIST-AI-RMF	NIST AI Risk Management Framework 1.0	Standard	Chapter 4	✓ Referenced
ISO-42001	ISO/IEC 42001:2023 AI Management System	Standard	Chapter 4	✓ Referenced
SINGAPORE-IMDA	Singapore IMDA Model AI Governance Framework for Agentic AI	Standard	Chapter 4	✓ Referenced
W3C-PROV	W3C PROV-DM: The PROV Data Model	Protocol spec	Chapter 4	✓ Referenced
W3C-VC	W3C Verifiable Credentials Data Model	Protocol spec	Chapter 4	✓ Referenced

L2: Official Product Documentation (Claim-Level Reviewed)

IBM-WATSONX-GOVERNANCE-DOCS				
Source ID	IBM-WATSONX-GOVERNANCE-DOCS			
Source Name	IBM watsonx.governance Official Documentation			
Type	Product docs			
Used In	Chapter 12			
Current Status	✓ Entry point located; ALCS object mappings remain L5 unless exact docs support them			

MICROSOFT-AZURE-AI-FOUNDRY-DOCS				
Source ID	MICROSOFT-AZURE-AI-FOUNDRY-DOCS			
Source Name	Microsoft Azure AI Foundry Official Documentation			
Type	Product docs			
Used In	Chapter 12			
Current Status	✓ Entry point located; ALCS object mappings remain L5 unless exact docs support them			

MICROSOFT-RESPONSIBLE-AI-DOCS				
Source ID	MICROSOFT-RESPONSIBLE-AI-DOCS			
Source Name	Microsoft Responsible AI Documentation			
Type	Product docs			
Used In	Chapter 12			
Current Status	● Entry point/source family located; exact responsible-AI page pinning remains			

AWS-BEDROCK-DOCS

Source ID	AWS-BEDROCK-DOCS
Source Name	AWS Bedrock Official Documentation
Type	Product docs
Used In	Chapter 12
Current Status	✓ Entry point located; lifecycle-object mappings remain L5 unless exact docs support them

AWS-BEDROCK-GUARDRAILS-DOCS

Source ID	AWS-BEDROCK-GUARDRAILS-DOCS
Source Name	AWS Bedrock Guardrails Official Documentation
Type	Product docs
Used In	Chapter 12
Current Status	✓ Entry point located; guardrails are not delegated-authority proof

AWS-AGENTCORE-DOCS

Source ID	AWS-AGENTCORE-DOCS
Source Name	AWS AgentCore Official Documentation
Type	Product docs
Used In	Chapter 12
Current Status	✓ Entry point located; ALCS object mappings remain analytical

GOOGLE-VERTEX-AI-DOCS

Source ID	GOOGLE-VERTEX-AI-DOCS
Source Name	Google Vertex AI Official Documentation
Type	Product docs
Used In	Chapter 12
Current Status	✓ Entry point located; lifecycle-object mappings remain L5 unless exact docs support them

GOOGLE-ADK-DOCS

Source ID	GOOGLE-ADK-DOCS
Source Name	Google Agent Development Kit (ADK) Official Documentation
Type	Product docs
Used In	Chapter 12
Current Status	✓ Entry point located; ADK primitives are not governance objects by default

GOOGLE-MODEL-ARMOR-DOCS	
Source ID	GOOGLE-MODEL-ARMOR-DOCS
Source Name	Google Model Armor Official Documentation
Type	Product docs
Used In	Chapter 12
Current Status	✓ Entry point located; safety inspection is not lifecycle responsibility proof

LANGGRAPH-DOCS	
Source ID	LANGGRAPH-DOCS
Source Name	LangGraph Official Documentation
Type	Product docs
Used In	Chapter 12
Current Status	✓ Entry point located; orchestration mappings remain analytical

LANGSMITH-DOCS	
Source ID	LANGSMITH-DOCS
Source Name	LangSmith Official Documentation
Type	Product docs
Used In	Chapter 12
Current Status	✓ Entry point located; observability mappings remain analytical

OPENAI-AGENTS-SDK-DOCS	
Source ID	OPENAI-AGENTS-SDK-DOCS
Source Name	OpenAI Agents SDK Official Documentation
Type	Product docs
Used In	Chapter 12
Current Status	✓ Accessible official Python/JS SDK docs replace inaccessible platform guide for retained SDK-surface claims; platform guide 403 retained as access note

OPENAI-TOOLS-DOCS	
Source ID	OPENAI-TOOLS-DOCS
Source Name	OpenAI Tools Official Documentation
Type	Product docs
Used In	Chapter 12
Current Status	✓ Narrowed to accessible SDK primitive claims only; lifecycle-governance mapping remains analytical

CREWAI-DOCS	
Source ID	CREWAI-DOCS
Source Name	CrewAI Official Documentation
Type	Product docs
Used In	Chapter 12
Current Status	✓ Entry point located; agent role labels are not human accountability roles

CREWAI-GITHUB	
Source ID	CREWAI-GITHUB
Source Name	CrewAI GitHub Repository
Type	Product docs
Used In	Chapter 12
Current Status	✓ Repository entry point located; exact code/API claims require page-level pinning

MPLP-DOCS	
Source ID	MPLP-DOCS
Source Name	MPLP (Multi-Agent Lifecycle Protocol) Official Documentation
Type	Protocol docs
Used In	Chapters 12, 13
Current Status	✓ Entry point located; protocol mapping remains L2 + L5 analytical interpretation

MPLP-SPEC	
Source ID	MPLP-SPEC
Source Name	MPLP Protocol Specification
Type	Protocol docs
Used In	Chapter 13
Current Status	✓ Bound through MPLP docs entry point; version/citation rendering remains

VALIDATION-LAB-METHODOLOGY	
Source ID	VALIDATION-LAB-METHODOLOGY
Source Name	Validation Lab Evidence-Based Validation Methodology
Type	Protocol docs
Used In	Chapter 14
Current Status	✓ Methodology page cited as public-surface, non-certifying evidence adjudication example only

L5: Author Inference

Source ID	Source Name	Type	Used In	Current Status
AUTHOR-INFERENCE-MRO	Sixteen Missing Regulatory Objects (Author Inference)	Analytical framework	Chapter 6	✓ Marked as L5
AUTHOR-INFERENCE-RCCS	RCCS Analytical Framework (Author Inference)	Analytical framework	Chapter 7	✓ Marked as L5
AUTHOR-INFERENCE-RCCS-M	RCCS-M — MRO-Adjusted Regulatory Compliance Coverage Score (Author Inference)	Analytical framework	Chapters 6-9, Evaluation Results	✓ Marked as L5; forward-looking author analytical model
AUTHOR-INFERENCE-ALCS	ALCS Analytical Framework (Author Inference)	Analytical framework	Chapter 8	✓ Marked as L5
AUTHOR-INFERENCE-COMPOSITE	Composite Scoring Method (Author Inference)	Analytical framework	Chapter 9	✓ Marked as L5
AUTHOR-INFERENCE-SYSTEM-MAPPING	System Mapping Methodology (Author Inference)	Analytical framework	Chapters 11, 12	✓ Marked as L5
AUTHOR-INFERENCE-FAILURE-SCENARIOS	Enterprise Failure Scenarios (Author Inference)	Analytical framework	Chapter 15	✓ Marked as L5

C.3 Source Register

The following table provides the complete source register for the white paper.

Table T-C-02: Source Register

Wide source table split into consecutive column groups; all source rows and columns are preserved.

Continuation 1 of 2

Source ID	Source Name	Evidence Level
EU-AI-ACT	EU AI Act (Regulation (EU) 2024/1689)	L1
GDPR	GDPR (Regulation (EU) 2016/679)	L1
COLORADO-AI-ACT	Colorado AI Act (SB25B-004)	L1
NIST-AI-RMF	NIST AI RMF 1.0	L1
ISO-42001	ISO/IEC 42001:2023	L1
SINGAPORE-IMDA	Singapore IMDA Model AI Governance Framework for Agentic AI	L1
W3C-PROV	W3C PROV-DM	L1
W3C-VC	W3C Verifiable Credentials	L1
IBM-WATSONX-GOVERNANCE-DOCS	IBM watsonx.governance Docs	L2
MICROSOFT-AZURE-AI-FOUNDRY-DOCS	Microsoft Azure AI Foundry Docs	L2
MICROSOFT-RESPONSIBLE-AI-DOCS	Microsoft Responsible AI Docs	L2
AWS-BEDROCK-DOCS	AWS Bedrock Docs	L2
AWS-BEDROCK-GUARDRAILS-DOCS	AWS Bedrock Guardrails Docs	L2
AWS-AGENTCORE-DOCS	AWS AgentCore Docs	L2
GOOGLE-VERTEX-AI-DOCS	Google Vertex AI Docs	L2

Source ID	Source Name	Evidence Level
GOOGLE-ADK-DOCS	Google ADK Docs	L2
GOOGLE-MODEL-ARMOR-DOCS	Google Model Armor Docs	L2
LANGGRAPH-DOCS	LangGraph Docs	L2
LANGSMITH-DOCS	LangSmith Docs	L2
OPENAI-AGENTS-SDK-DOCS	OpenAI Agents SDK Docs	L2
OPENAI-TOOLS-DOCS	OpenAI Tools Docs	L2
CREWAI-DOCS	CrewAI Docs	L2
CREWAI-GITHUB	CrewAI GitHub	L2
MPLP-DOCS	MPLP Docs	L2
MPLP-SPEC	MPLP Spec	L2
VALIDATION-LAB-METHODOLOGY	Validation Lab Methodology	L2
AUTHOR-INFERENCE-MRO	Sixteen MROs (Author Inference)	L5
AUTHOR-INFERENCE-RCCS	RCCS-T / traditional RCCS framework (Author Inference)	L5
AUTHOR-INFERENCE-RCCS-M	RCCS-M — MRO-Adjusted Regulatory Compliance Coverage Score (Author Inference)	L5
AUTHOR-INFERENCE-ALCS	ALCS (Author Inference)	L5
AUTHOR-INFERENCE-COMPOSITE	Composite Scoring (Author Inference)	L5
AUTHOR-INFERENCE-SYSTEM-MAPPING	System Mapping (Author Inference)	L5
AUTHOR-INFERENCE-FAILURE-SCENARIOS	Failure Scenarios (Author Inference)	L5

Continuation 2 of 2

Source ID	Used In	Current Status	Publication Follow-up Action
EU-AI-ACT	Chapter 4	✓ Referenced	Verify article citations
GDPR	Chapter 4	✓ Referenced	Verify article citations
COLORADO-AI-ACT	Chapter 4	✓ Referenced	Verify citation
NIST-AI-RMF	Chapter 4	✓ Referenced	Verify citation
ISO-42001	Chapter 4	✓ Referenced	Verify citation
SINGAPORE-IMDA	Chapter 4	✓ Referenced	Verify citation
W3C-PROV	Chapter 4	✓ Referenced	Verify citation
W3C-VC	Chapter 4	✓ Referenced	Verify citation
IBM-WATSONX-GOVERNANCE-DOCS	Chapter 12	Closed for high-risk citation review; official entry points support narrowed capability-surface claims	Preserve narrowed claim wording and final citation formatting
MICROSOFT-AZURE-AI-FOUNDRY-DOCS	Chapter 12	Closed for high-risk citation review; Microsoft Learn entry point supports narrowed platform-surface claims	Preserve narrowed claim wording and final citation formatting
MICROSOFT-RESPONSIBLE-AI-DOCS	Chapter 12	Closed as source-family support for retained responsible-AI surface claims	Keep lifecycle-object scoring analytical unless exact source language directly supports the object
AWS-BEDROCK-DOCS	Chapter 12	Closed for high-risk citation review; official entry points support narrowed Bedrock capability-surface claims	Preserve narrowed claim wording and final citation formatting
AWS-BEDROCK-GUARDRAILS-DOCS	Chapter 12	Closed for high-risk citation review; guardrail claims narrowed to document-	Preserve authority-boundary caveat

Source ID	Used In	Current Status	Publication Follow-up Action
		ed safety/control surfaces	
AWS-AGENTCORE-DOCS	Chapter 12	Closed for high-risk citation review; AgentCore claims narrowed to documented runtime/agent surfaces	Preserve lifecycle-object mapping as analytical where needed
GOOGLE-VERTEX-AI-DOCS	Chapter 12	Closed for high-risk citation review; Vertex AI claims narrowed to documented platform surfaces	Preserve narrowed claim wording and final citation formatting
GOOGLE-ADK-DOCS	Chapter 12	Closed for high-risk citation review; ADK claims narrowed to documented agent-development primitives	Preserve lifecycle-object mapping as analytical where needed
GOOGLE-MODEL-ARMOR-DOCS	Chapter 12	Closed for high-risk citation review; Model Armor claims narrowed to documented safety inspection surfaces	Preserve lifecycle-governance caveat
LANGGRAPH-DOCS	Chapter 12	Closed for high-risk citation review; LangGraph claims narrowed to documented orchestration surfaces	Preserve narrowed claim wording and final citation formatting
LANGSMITH-DOCS	Chapter 12	Closed for high-risk citation review; LangSmith claims narrowed to documented observability/evaluation surfaces	Preserve lifecycle-object mapping as analytical where needed
OPENAI-AGENTS-SDK-DOCS	Chapter 12	Accessible official Python/JS SDK docs validated; platform guide 403 is a non-blocking access note	Use accessible official SDK docs for retained agents/tools/handoffs/guardrails/tracing/session claims
OPENAI-TOOLS-DOCS	Chapter 12	Narrowed to SDK primitive claims supported by accessible official docs	Keep liability-boundary and lifecycle-governance mapping analytical
CREWAI-DOCS	Chapter 12	Closed for high-risk citation review; CrewAI claims narrowed to documented orchestration surfaces	Preserve human-accountability caveat
CREWAI-GITHUB	Chapter 12	Repository entry point located; exact code/API claims require page-level pinning	Pin exact repository paths only for code/API-specific claims
MPLP-DOCS	Chapters 12, 13	Closed for protocol identity claims; GAIC MRO/ALCS mapping remains author analytical mapping where not directly stated	Preserve conflict-of-interest boundary
MPLP-SPEC	Chapter 13	Closed through MPLP docs entry point for protocol identity support	Preserve version/citation details during final proof
VALIDATION-LAB-METHODOLOGY	Chapter 14	Methodology page cited and boundary narrowed	Retain as public-surface non-certifying example only
AUTHOR-INFERENCE-MRO	Chapter 6	✓ Marked as L5	Verify L5 marking in text
AUTHOR-INFERENCE-RCCS	Chapter 7	✓ Marked as L5	Verify L5 marking in text
AUTHOR-INFERENCE-RCCS-M	Chapters 6-9, Evaluation Results	✓ Marked as L5 ; author analytical, forward-looking, not current law	Preserve boundary wording and MPLP COI
AUTHOR-INFERENCE-ALCS	Chapter 8	✓ Marked as L5	Verify L5 marking in text
AUTHOR-INFERENCE-COMPOSITE	Chapter 9	✓ Marked as L5	Verify L5 marking in text
AUTHOR-INFERENCE-SYSTEM-MAPPING	Chapters 11, 12	✓ Marked as L5	Verify L5 marking in text
AUTHOR-INFERENCE-FAILURE-SCENARIOS	Chapter 15	✓ Marked as L5	Verify L5 marking in text

C.4 Citation Closure Ledger

The following table lists Phase 1D-11 source closure results after claim-level source review.

Table T-C-03: Citation Revalidation Queue

IBM-WATSONX-GOVERNANCE-DOCS	
Source ID	IBM-WATSONX-GOVERNANCE-DOCS
URL / Source Pointer	https://www.ibm.com/products/watsonx-governance ; https://www.ibm.com/docs/en/watsonx
Closure Result	Official entry points accessible; governance capability claims narrowed to documented surfaces
Remaining Limitation	ALCS lifecycle-object mapping remains L5 where not directly documented
Release Status	Non-blocking

MICROSOFT-AZURE-AI-FOUNDRY-DOCS	
Source ID	MICROSOFT-AZURE-AI-FOUNDRY-DOCS
URL / Source Pointer	https://learn.microsoft.com/en-us/azure/foundry/
Closure Result	Official entry point accessible; platform/agent claims narrowed
Remaining Limitation	ALCS lifecycle-object mapping remains L5 where not directly documented
Release Status	Non-blocking

AWS-BEDROCK-DOCS / GUARDRAILS / AGENTCORE	
Source ID	AWS-BEDROCK-DOCS / GUARDRAILS / AGENTCORE
URL / Source Pointer	AWS official docs URLs
Closure Result	Official entry points accessible; guardrail/runtime claims narrowed
Remaining Limitation	Guardrails/runtime features are not delegated-authority proof
Release Status	Non-blocking

GOOGLE-VERTEX-AI-DOCS / ADK / MODEL ARMOR	
Source ID	GOOGLE-VERTEX-AI-DOCS / ADK / MODEL ARMOR
URL / Source Pointer	Google official docs URLs
Closure Result	Official entry points accessible; safety/agent claims narrowed
Remaining Limitation	Model safety and ADK primitives are not lifecycle responsibility proof
Release Status	Non-blocking

LANGGRAPH-DOCS / LANGSMITH-DOCS

Source ID	LANGGRAPH-DOCS / LANGSMITH-DOCS
URL / Source Pointer	LangChain official docs URLs
Closure Result	Official entry points accessible; orchestration/observability claims narrowed
Remaining Limitation	Lifecycle governance requires implementer integration
Release Status	Non-blocking

OPENAI-AGENTS-SDK-DOCS / OPENAI-TOOLS-DOCS

Source ID	OPENAI-AGENTS-SDK-DOCS / OPENAI-TOOLS-DOCS
URL / Source Pointer	https://openai.github.io/openai-agents-python/ ; https://openai.github.io/openai-agents-js/
Closure Result	Inaccessible platform guide replaced by accessible official SDK docs for SDK claims
Remaining Limitation	Platform guide HTTP 403 retained as access note only
Release Status	Non-blocking

CREWAI-DOCS / CREWAI-GITHUB

Source ID	CREWAI-DOCS / CREWAI-GITHUB
URL / Source Pointer	https://docs.crewai.com/ ; https://github.com/crewAIInc/crewAI
Closure Result	Official docs/repository accessible; orchestration claims narrowed
Remaining Limitation	Human accountability/lifecycle governance remains implementer-built
Release Status	Non-blocking

MPLP-DOCS / MPLP-SPEC

Source ID	MPLP-DOCS / MPLP-SPEC
URL / Source Pointer	https://docs.mplp.io/docs/introduction/mplp-v1.0-protocol-overview
Closure Result	Protocol identity source-bound; conflict-of-interest boundary preserved
Remaining Limitation	GAIC MRO/ALCS mapping remains author analytical mapping
Release Status	Non-blocking

VALIDATION-LAB-METHODOLOGY

Source ID	VALIDATION-LAB-METHODOLOGY
URL / Source Pointer	https://lab.mplp.io/methodology
Closure Result	Closed as public-surface non-certifying methodology example
Remaining Limitation	Does not support certification, endorsement, audit opinion, or legal compliance proof
Release Status	Non-blocking

C.5 Source Notes and Revalidation Policy

Source Notes

- **L1 sources (legal text and formal standards)** are referenced in Chapter 4 and provide the regulatory baseline for the white paper. These sources are authoritative but require legal interpretation. The white paper does not provide legal advice or legal interpretation.
- **L2 sources (official product documentation)** are referenced in Chapters 12, 13, and 14 and support documented product/protocol capability surfaces. Phase 1D-11 records closure dispositions for high-risk source pointers. **L2** sources do not prove deployment practice, audit success, lifecycle responsibility-object implementation, or legal compliance.
- **L5 sources (author inference)** are used in Chapters 6-9, 11-12, and 15 to define the analytical framework (MROs, RCCS-T, RCCS-M, ALCS, optional composite views, System Mapping, Failure Scenarios). These sources are marked as author inference and do not replace evidence-based analysis.
- **AUTHOR-INFERENCE-RCCS-M** marks the MRO-adjusted regulatory coverage model. RCCS-M is an author analytical, forward-looking model grounded in MRO analysis. It is not current law, certification, legal compliance proof, a regulator-approved benchmark, a procurement benchmark, a vendor ranking, or an official standard.
- **MPLP and Validation Lab sources** require special attention due to author conflict of interest. The author of this white paper is also the creator of MPLP and Validation Lab. Phase 1C follow-up verified that MPLP and Validation Lab remain positioned as examples, not required or exclusive solutions; public research edition QA must preserve that boundary.

Revalidation Policy

Public research edition source governance must:

1. **Keep L2 product documentation tied to narrowed capability-surface claims** and preserve **L5** mapping labels where lifecycle support is analytical
2. **Verify all L1 legal/regulatory citations** for accuracy and completeness
3. **Mark all L5 author inference** clearly in the text
4. **Preserve MPLP and Validation Lab boundary discipline** and keep the Validation Lab methodology citation limited to public-surface non-certifying support
5. **Remove or mark as inference** any unsupported claims
6. **Maintain the claim evidence register** documenting all source validation and downgrade actions

Appendix D — Expanded Cross-System MRO Mapping

Version: v0.3.2-FRC-R3

D.0 Mapping Boundary

This appendix provides an expanded cross-system MRO mapping across the eight representative systems evaluated in Chapters 11-12. This mapping is an analytical tool, not a product ranking, certification, or procurement recommendation. Phase 1C follow-up located official source entry points and added claim-level source binding, but the mapping remains **revalidated provisional** because many MRO-to-system cells are analytical interpretations rather than direct product-documentation claims.

Boundary statement:

- **This mapping is analytical only.** The mapping compares systems by responsibility semantics, not product quality, market leadership, or procurement suitability.
 - **This mapping is not a product ranking.** Systems are not ranked by score or quality. The mapping identifies where systems appear to provide lifecycle responsibility object primitives and where gaps may exist.
 - **This mapping is not certification.** The presence of lifecycle responsibility object primitives does not certify a system as compliant, approved, or enterprise-ready.
 - **This mapping is not a procurement recommendation.** Organizations must evaluate systems within their specific legal, regulatory, and operational context. This mapping provides analytical input, not procurement guidance.
 - **This mapping is based on source-bound public evidence and analytical interpretation.** Official product/protocol entry points were located during Phase 1C follow-up. L2 evidence supports documented capability surfaces; MRO cell-level mapping remains L5 analytical interpretation where official documentation does not directly define the lifecycle responsibility object.
 - **This mapping uses qualitative values.** The mapping uses qualitative values (Strong object-level mapping, Strong workflow/feature mapping, Partial or adjacent capability, Weak or implementer-built only, Not visible in public evidence, explicit non-blocking limitation where applicable) rather than numeric scores to avoid creating a false precision ranking.
-

D.1 How to Read the Expanded MRO Mapping

The expanded MRO mapping (Table T-D-01) maps all sixteen Missing Regulatory Objects across all eight representative systems.

Qualitative values:

- **Strong object-level mapping:** System appears to provide explicit lifecycle responsibility object primitives with protocol-level or API-level support based on available documentation. Example: MPLP appears to provide explicit role, context, plan, confirm, and trace semantics. **Revalidated provisional; mapping remains L5 unless the source directly defines the object.**
- **Strong workflow/feature mapping:** System appears to provide workflow or feature primitives that may support lifecycle responsibility objects with configuration or integration based on available documentation. Example: IBM watsonx.governance appears to provide model governance workflows that may support lifecycle responsibility objects. **Revalidated provisional; adjacent workflow evidence is not lifecycle-object proof.**
- **Partial or adjacent capability:** System appears to provide adjacent capabilities that are related to lifecycle responsibility objects but may not fully satisfy the object requirements based on available documentation. Example: Approval button is not the same as human-role-to-MAS responsibility mapping. **Revalidated provisional; keep as adjacent unless exact source support is pinned.**
- **Weak or implementer-built only:** Lifecycle responsibility object primitives are not visible in current public documentation; implementers may need to build lifecycle governance on top of the system. Example: OpenAI Agents SDK provides execution primitives, while enterprise lifecycle governance objects require additional design. **Revalidated provisional; this does not prove absence of private or implementation-specific capability.**
- **Not visible in public evidence:** Lifecycle responsibility object primitives are not visible in public product documentation or protocol specifications. This does not prove the capability does not exist; it means public evidence is not available.
- **Explicit non-blocking limitation:** Mapping remains analytical because official sources support adjacent capability surfaces but do not directly state the lifecycle responsibility object.

Reading the table:

- **Rows:** Sixteen Missing Regulatory Objects (MRO-01 through MRO-16)
- **Columns:** Eight representative systems (MPLP, IBM, Microsoft, AWS, Google, LangGraph/LangSmith, OpenAI, CrewAI)
- **Cells:** Qualitative mapping value for each MRO × System combination
- **Notes column:** Additional context or boundary notes

D.2 Expanded Cross-System MRO Mapping

Expanded Cross-System MRO Mapping is a high-comparison appendix matrix. It is rendered as landscape split panels with the MRO ID repeated so long system names wrap horizontally rather than collapsing into vertical single-character columns.

All cell values remain provisional analytical mappings in fixed source order. This matrix is not a vendor ranking, procurement recommendation, certification, regulatory approval signal, or legal compliance proof.

Note: This table is very wide (10 columns) and requires landscape layout. Phase 1C layout audit documented the risk; Phase 1D rendering QA remains required.

Table T-D-01: Expanded Cross-System MRO Mapping

Wide source table split into consecutive column groups; all source rows and columns are preserved.

Continuation 1 of 3

MRO ID	MRO Name	MPLP	IBM watsonx.governance
MRO-01	Human Role to MAS Responsibility Mapping	Strong object-level mapping (provisional)	Strong workflow/feature mapping (provisional)
MRO-02	Delegated Authority Boundary	Strong object-level mapping (provisional)	Partial or adjacent capability (provisional)
MRO-03	Agent Role is not Human Role	Strong object-level mapping (provisional)	Strong workflow/feature mapping (provisional)
MRO-04	Accepted Outcome Compliance	Strong object-level mapping (provisional)	Strong workflow/feature mapping (provisional)
MRO-05	Tool-Action Liability Boundary	Strong object-level mapping (provisional)	Partial or adjacent capability (provisional)
MRO-06	Responsibility Transfer Across Agents	Strong object-level mapping (provisional)	Partial or adjacent capability (provisional)
MRO-07	Authority Drift	Strong object-level mapping (provisional)	Strong workflow/feature mapping (provisional)
MRO-08	MAS Evidence Partitioning	Strong object-level mapping (provisional)	Strong workflow/feature mapping (provisional)
MRO-09	Cross-Project Reuse Compliance	Strong object-level mapping (provisional)	Partial or adjacent capability (provisional)
MRO-10	Privacy / GDPR Lifecycle Mapping	Strong object-level mapping (provisional)	Strong workflow/feature mapping (provisional)
MRO-11	Privacy-Preserving Third-Party Validation	Strong object-level mapping (provisional)	Partial or adjacent capability (provisional)
MRO-12	Evidence Minimization and Selective Disclosure	Strong object-level mapping (provisional)	Strong workflow/feature mapping (provisional)
MRO-13	Data Subject Rights vs Evidence Retention	Strong object-level mapping (provisional)	Strong workflow/feature mapping (provisional)
MRO-14	Third-Party Processor / Subprocessor Chain	Partial or adjacent capability (provisional)	Strong workflow/feature mapping (provisional)
MRO-15	Vendor / Model / Runtime Substitution Conformance	Strong object-level mapping (provisional)	Strong workflow/feature mapping (provisional)
MRO-16	Incident, Dispute, and Remediation Closure	Strong object-level mapping (provisional)	Strong workflow/feature mapping (provisional)

Continuation 2 of 3

MRO ID	Microsoft Azure AI Foundry	AWS Bedrock / Guardrails / AgentCore	Google Vertex AI / ADK / Model Armor
MRO-01	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)
MRO-02	Partial or adjacent capability (provisional)	Strong workflow/feature mapping (provisional)	Partial or adjacent capability (provisional)
MRO-03	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)
MRO-04	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)

MRO ID	Microsoft Azure AI Foundry	AWS Bedrock / Guardrails / AgentCore	Google Vertex AI / ADK / Model Armor
MRO-05	Partial or adjacent capability (provisional)	Strong workflow/feature mapping (provisional)	Partial or adjacent capability (provisional)
MRO-06	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)
MRO-07	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)
MRO-08	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)
MRO-09	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)
MRO-10	Strong workflow/feature mapping (provisional)	Partial or adjacent capability (provisional)	Strong workflow/feature mapping (provisional)
MRO-11	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)
MRO-12	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)
MRO-13	Strong workflow/feature mapping (provisional)	Partial or adjacent capability (provisional)	Strong workflow/feature mapping (provisional)
MRO-14	Strong workflow/feature mapping (provisional)	Strong workflow/feature mapping (provisional)	Strong workflow/feature mapping (provisional)
MRO-15	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)
MRO-16	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)	Partial or adjacent capability (provisional)

Continuation 3 of 3

MRO ID	LangGraph / LangSmith	OpenAI Agents SDK	CrewAI
MRO-01	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-02	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-03	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Partial or adjacent capability (provisional)
MRO-04	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-05	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-06	Strong workflow/feature mapping (provisional)	Weak or implementer-built only (provisional)	Partial or adjacent capability (provisional)
MRO-07	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-08	Strong workflow/feature mapping (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-09	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-10	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-11	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-12	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-13	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-14	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-15	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)
MRO-16	Partial or adjacent capability (provisional)	Weak or implementer-built only (provisional)	Weak or implementer-built only (provisional)

D.3 Pattern-Level Interpretation

The expanded MRO mapping reveals several provisional patterns across system categories. **All mappings remain revalidated provisional after Phase 1C claim-level source binding.**

Pattern 1: Lifecycle protocol paths (MPLP) provisionally show strong object-level mapping across most MROs.

Based on available protocol documentation, MPLP appears to provide explicit protocol-level semantics for lifecycle responsibility objects. However, protocol-level fit does not prove deployment conformance, operational effectiveness, or enterprise adoption. MPLP requires implementation validation, enterprise integration, and organizational adoption. Phase 1C follow-up located the official MPLP documentation entry point; mapping strength remains L5 analytical interpretation where not directly stated in protocol text.

Pattern 2: Enterprise AI governance platforms (IBM watsonx.governance) provisionally show strong workflow/feature mapping across governance-related MROs.

Based on available product documentation, IBM watsonx.governance appears to provide model governance workflows, audit trails, data governance, and incident management. These workflows may support lifecycle responsibility objects with configuration or integration. However, workflow/feature mapping does not automatically provide lifecycle responsibility semantics. Organizations must map workflows to lifecycle objects. Phase 1C follow-up located official IBM entry points; MRO cell mappings remain provisional.

Pattern 3: Cloud AI platforms (Microsoft, AWS, Google) provisionally show strong workflow/feature mapping for data governance and processor chain MROs, but partial or adjacent capability for lifecycle-specific MROs.

Based on available product documentation, cloud AI platforms appear to provide data governance, processor chain documentation, and compliance workflows. However, these platforms are designed for model governance and AI platform operations, not necessarily lifecycle responsibility-object governance. Lifecycle responsibility objects may require additional integration or custom implementation. Phase 1C follow-up located official Microsoft, AWS, and Google entry points; MRO cell mappings remain provisional.

Pattern 4: Agent orchestration frameworks (LangGraph/LangSmith) provisionally show strong workflow/feature mapping for evidence partitioning and responsibility transfer, but partial or adjacent capability for other MROs.

Based on available product documentation, LangGraph appears to provide orchestration primitives and LangSmith appears to provide tracing/observability primitives. These primitives may be useful for lifecycle governance but do not automatically provide lifecycle responsibility semantics. Organizations must build lifecycle governance on top of orchestration primitives. Phase 1C follow-up located official LangGraph/LangSmith entry points; MRO cell mappings remain provisional.

Pattern 5: Developer agent SDKs (OpenAI Agents SDK, CrewAI) provisionally show weak or implementer-built only for most MROs.

Based on available product documentation, developer agent SDKs appear to provide execution primitives but not final lifecycle governance primitives. Organizations must build lifecycle governance on top of execution primitives. This is not a weakness of the SDKs; it reflects their design purpose (execution, not governance). Phase 1C follow-up located official OpenAI Agents SDK and CrewAI entry points; MRO cell mappings remain provisional.

Pattern 6: The field gap is not lack of AI tools. The field gap is lack of explicit lifecycle responsibility objects.

All eight systems provide useful primitives for agentic AI. However, based on available public evidence, many lifecycle responsibility object primitives remain adjacent, implementer-built, or analytically inferred. Organizations must build lifecycle governance by integrating multiple systems, configuring workflows, or implementing custom lifecycle governance layers. **This pattern remains revalidated provisional after Phase 1C claim-level source binding.**

D.4 Phase 1C Claim-Level Source Binding Note

This appendix is based on public product documentation and protocol specifications available as of the white paper publication date. Phase 1D-11 completed high-risk source closure for Appendix D by narrowing source-supported claims and retaining L5 analytical labels where official sources do not directly define lifecycle responsibility objects. Remaining final proofing work is to:

1. **Preserve narrowed citation wording** for high-risk MRO cell claims
2. **Preserve L5 analytical status** for MRO mappings not directly stated in official documentation
3. **Retain provisional wording** for all cross-system cells until publication approval
4. **Verify all qualitative mapping values** during final proofing
5. **Mark uncertain mappings** as explicit non-blocking limitations if exact lifecycle-object support cannot be pinned

Phase 1D wide-table evaluation: Table T-D-01 is very wide (10 columns) and may require splitting, rotation, or alternative layout for readability in PDF format.

Appendix E — RCCS-T Detailed Rubrics and RCCS-M Adjustment Matrix

Version: v0.3.2-FRC-R3

E.0 RCCS-T / RCCS-M Rubric Boundary

This appendix provides detailed scoring rubrics for all ten RCCS (Regulatory Compliance Coverage Score) dimensions. The existing detailed rubrics serve as **RCCS-T baseline rubrics**: they evaluate traditional regulatory and governance coverage primitives such as risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, and monitoring.

RCCS-M uses the same ten dimensions and weights but applies MRO adjustment tests. RCCS-M evaluates whether the traditional regulatory obligation can be represented as Missing Regulatory Objects and lifecycle responsibility semantics. High RCCS-T does not automatically imply high RCCS-M.

These rubrics are analytical tools for evaluating system capability. They are not legal compliance scores, certifications, regulatory approvals, legal compliance proof, regulator-approved benchmarks, procurement benchmarks, vendor rankings, or procurement selection signals.

Boundary statement:

- **RCCS-T measures traditional regulatory compliance coverage primitives.** RCCS-T evaluates whether a system provides primitives that can support familiar regulatory compliance coverage requirements. It does not prove legal compliance, regulatory approval, certification, audit readiness, or operational effectiveness.
- **RCCS-M measures MRO-adjusted regulatory coverage.** RCCS-M evaluates whether those same obligations can be expressed through lifecycle responsibility objects such as authority boundaries, responsibility mappings, accepted outcome states, partitioned evidence, privacy lifecycle maps, substitution conformance records, and incident/dispute/remediation closure records.
- **High RCCS scores do not prove legal compliance.** A high RCCS score indicates that a system provides primitives for regulatory compliance coverage. It does not prove that the system is legally compliant, operationally effective, or enterprise-ready.
- **Low RCCS scores do not prove non-compliance.** A low RCCS score indicates gaps in regulatory compliance coverage primitives. It does not prove that the system is non-compliant, unsafe, or unsuitable for deployment.

- **RCCS scores are context-independent.** RCCS scores measure system capability, not organizational practice. Organizations must interpret scores within their legal, regulatory, and operational context.
- **RCCS scores require evidence.** All scores must be supported by evidence. The evidence level (L1 - L5) affects the score multiplier using the canonical formula from Phase 1B-8C.
- **RCCS scores are not product rankings.** Rubrics compare systems by regulatory compliance coverage primitives and MRO-adjusted lifecycle object expression, not product quality, market leadership, or procurement suitability.

E.1 How to Use RCCS-T and RCCS-M Rubrics

Scoring process:

1. **Assess RCCS-T raw capability (0-5)** based on traditional governance/control coverage against the detailed rubric
2. **Assess RCCS-M raw capability (0-5)** using the RCCS-M adjustment matrix in Table T-E-11
3. **Identify evidence level (L1 - L5)** based on source type for each lens
4. **Apply evidence multiplier** using canonical formula: $\text{Dimension Score} = (\text{Raw Score} / 5) \times 100 \times \text{Evidence Confidence Multiplier}$
5. **Document justification** explaining why the raw score and evidence level were assigned
6. **Add boundary note** clarifying what the score does and does not prove

Canonical evidence multipliers (from Phase 1B-8C):

- L1 (Binding Legal Text): 1.00
- L2 (Product Docs): 0.85
- L3 (Audit Report): 0.75
- L4 (Vendor Claim): 0.55
- L5 (Author Inference): 0.35

Scoring scale:

- **0:** No capability; no evidence
- **1:** Minimal capability; weak evidence
- **2:** Partial capability; some evidence
- **3:** Moderate capability; documented evidence
- **4:** Strong capability; validated evidence
- **5:** Complete capability demonstrated by implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts. Legal text may establish a regulatory baseline but does not by itself prove implementation.

In score-5 rows, legal references identify the regulatory baseline only. They do not count as implementation evidence unless paired with implementation artifacts, formal audit evidence,

protocol-level evidence, or validated operational records.

Negative controls:

Each rubric includes negative controls to prevent overclaim. Negative controls distinguish adjacent capabilities from regulatory compliance coverage primitives. For RCCS-M, negative controls also prevent ordinary feature support from being treated as lifecycle responsibility object expression.

E.2 RCCS Detailed Rubrics

RCCS-01: Risk Management

Core question: Does the system provide primitives for AI risk identification, assessment, mitigation, and monitoring?

What counts: Risk assessment frameworks, risk registers, risk mitigation controls, risk monitoring dashboards, risk escalation workflows

What does not count: General software risk management, non-AI-specific risk controls, risk management without AI lifecycle integration

Table T-E-01: RCCS-01 Risk Management Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No risk management capability Absence of risk management primitives Required Evidence None Example Evidence Artifact No risk assessment, no risk register, no risk controls Negative Control N/A Boundary Note Absence of risk management primitives
1	Minimal risk management Vendor claim requires validation Required Evidence Vendor claim or marketing material Example Evidence Artifact Risk management mentioned in marketing docs but no implementation details Negative Control Risk management mentioned ≠ risk management implemented Boundary Note Vendor claim requires validation

2

Partial risk management

Partial coverage; gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Risk assessment template exists but no risk register or monitoring

Negative Control

Risk template ≠ risk assessment workflow

Boundary Note

Partial coverage; gaps exist

3

Moderate risk management

Moderate coverage; monitoring gaps exist

Required Evidence

Product documentation with risk assessment and register

Example Evidence Artifact

Risk assessment workflow, risk register, basic mitigation controls

Negative Control

Risk register ≠ risk monitoring or escalation

Boundary Note

Moderate coverage; monitoring gaps exist

4

Strong risk management

Strong coverage; escalation gaps may exist

Required Evidence

Product documentation with risk assessment, register, mitigation, and monitoring

Example Evidence Artifact

Risk assessment workflow, risk register, mitigation controls, risk monitoring dashboard

Negative Control

Risk monitoring ≠ risk escalation or continuous improvement

Boundary Note

Strong coverage; escalation gaps may exist

5

Complete capability demonstrated by implementation evidence

Complete capability requires implementation evidence; legal text alone insufficient

Required Evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts

Example Evidence Artifact

Legal baseline: EU AI Act Article 9 risk management system requirement; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record

Negative Control

Legal text may establish regulatory baseline but does not prove system implementation

Boundary Note

Complete capability requires implementation evidence; legal text alone insufficient

RCCS-02: Data Governance

Core question: Does the system provide primitives for data governance including data quality, data lineage, data access control, and data retention?

What counts: Data quality checks, data lineage tracking, data access control, data retention policies, data subject rights workflows

What does not count: General database management, data storage without governance, data access control without lifecycle integration

Table T-E-02: RCCS-02 Data Governance Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No data governance capability Absence of data governance primitives Required Evidence None Example Evidence Artifact No data quality checks, no data lineage, no data access control Negative Control N/A Boundary Note Absence of data governance primitives
1	Minimal data governance Vendor claim requires validation Required Evidence Vendor claim or marketing material Example Evidence Artifact Data governance mentioned in marketing docs but no implementation details Negative Control Data governance mentioned ≠ data governance implemented Boundary Note Vendor claim requires validation
2	Partial data governance Partial coverage; lineage and retention gaps exist Required Evidence Product documentation with partial coverage Example Evidence Artifact Data quality checks exist but no data lineage or retention policies Negative Control Data quality ≠ data lineage or retention Boundary Note Partial coverage; lineage and retention gaps exist

3

Moderate data governance

Moderate coverage; retention and subject rights gaps exist

Required Evidence

Product documentation with data quality, lineage, and access control

Example Evidence Artifact

Data quality checks, data lineage tracking, data access control

Negative Control

Data access control ≠ data retention or subject rights

Boundary Note

Moderate coverage; retention and subject rights gaps exist

4

Strong data governance

Strong coverage; subject rights gaps may exist

Required Evidence

Product documentation with data quality, lineage, access control, and retention

Example Evidence Artifact

Data quality checks, data lineage tracking, data access control, data retention policies

Negative Control

Data retention ≠ data subject rights workflows

Boundary Note

Strong coverage; subject rights gaps may exist

5

Complete capability demonstrated by implementation evidence

Complete capability requires implementation evidence; legal text alone insufficient

Required Evidence

Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts

Example Evidence Artifact

Legal baseline: GDPR Article 5 data governance requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record

Negative Control

Legal text may establish regulatory baseline but does not prove system implementation

Boundary Note

Complete capability requires implementation evidence; legal text alone insufficient

RCCS-03: Documentation

Core question: Does the system provide primitives for AI system documentation including model cards, system cards, and technical documentation?

What counts: Model cards, system cards, technical documentation, API documentation, deployment documentation

What does not count: General software documentation, marketing materials, documentation without AI lifecycle integration

Table T-E-03: RCCS-03 Documentation Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No documentation capability Absence of documentation primitives Required Evidence None Example Evidence Artifact No model cards, no system cards, no technical documentation Negative Control N/A Boundary Note Absence of documentation primitives
1	Minimal documentation Vendor claim requires validation Required Evidence Vendor claim or marketing material Example Evidence Artifact Documentation mentioned in marketing docs but no implementation details Negative Control Documentation mentioned ≠ documentation implemented Boundary Note Vendor claim requires validation
2	Partial documentation Partial coverage; system card and deployment gaps exist Required Evidence Product documentation with partial coverage Example Evidence Artifact Model card template exists but no system card or deployment documentation Negative Control Model card ≠ system card or deployment documentation Boundary Note Partial coverage; system card and deployment gaps exist
3	Moderate documentation Moderate coverage; deployment and API gaps exist Required Evidence Product documentation with model cards and system cards Example Evidence Artifact Model card template, system card template, basic technical documentation Negative Control System card ≠ deployment documentation or API documentation Boundary Note Moderate coverage; deployment and API gaps exist

4	Strong documentation Strong coverage; deployment and lifecycle gaps may exist Required Evidence Product documentation with model cards, system cards, and technical documentation Example Evidence Artifact Model card template, system card template, technical documentation, API documentation Negative Control API documentation ≠ deployment documentation or lifecycle documentation Boundary Note Strong coverage; deployment and lifecycle gaps may exist
5	Complete capability demonstrated by implementation evidence Complete capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 11 technical documentation requirement; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove system implementation Boundary Note Complete capability requires implementation evidence; legal text alone insufficient

RCCS-04: Record-Keeping

Core question: Does the system provide primitives for record-keeping including audit trails, logs, and evidence export?

What counts: Audit trails, logs, evidence export capability, log retention policies, log integrity validation

What does not count: General application logs, logs without lifecycle integration, logs without export capability

Table T-E-04: RCCS-04 Record-Keeping Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No record-keeping capability

Absence of record-keeping primitives

Required Evidence

None

Example Evidence Artifact

No audit trails, no logs, no evidence export

Negative Control

N/A

Boundary Note

Absence of record-keeping primitives

1

Minimal record-keeping

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Record-keeping mentioned in marketing docs but no implementation details

Negative Control

Record-keeping mentioned ≠ record-keeping implemented

Boundary Note

Vendor claim requires validation

2

Partial record-keeping

Partial coverage; export and retention gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Audit trails exist but no evidence export or retention policies

Negative Control

Audit trails ≠ evidence export or retention

Boundary Note

Partial coverage; export and retention gaps exist

3

Moderate record-keeping

Moderate coverage; integrity and disclosure gaps exist

Required Evidence

Product documentation with audit trails and evidence export

Example Evidence Artifact

Audit trails, evidence export capability, basic log retention policies

Negative Control

Evidence export ≠ log integrity validation or selective disclosure

Boundary Note

Moderate coverage; integrity and disclosure gaps exist

4	Strong record-keeping Strong coverage; disclosure and minimization gaps may exist Required Evidence Product documentation with audit trails, evidence export, and retention Example Evidence Artifact Audit trails, evidence export capability, log retention policies, log integrity validation Negative Control Log integrity ≠ selective disclosure or evidence minimization Boundary Note Strong coverage; disclosure and minimization gaps may exist
5	Complete capability demonstrated by implementation evidence Complete capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 12 record-keeping requirement; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove system implementation Boundary Note Complete capability requires implementation evidence; legal text alone insufficient

RCCS-05: Transparency

Core question: Does the system provide primitives for transparency including explainability, interpretability, and disclosure?

What counts: Explainability tools, interpretability methods, disclosure mechanisms, transparency reports

What does not count: General UI transparency, transparency without AI lifecycle integration, transparency without disclosure capability

Table T-E-05: RCCS-05 Transparency Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No transparency capability

Absence of transparency primitives

Required Evidence

None

Example Evidence Artifact

No explainability, no interpretability, no disclosure

Negative Control

N/A

Boundary Note

Absence of transparency primitives

1

Minimal transparency

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Transparency mentioned in marketing docs but no implementation details

Negative Control

Transparency mentioned ≠ transparency implemented

Boundary Note

Vendor claim requires validation

2

Partial transparency

Partial coverage; interpretability and disclosure gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Explainability tools exist but no interpretability or disclosure mechanisms

Negative Control

Explainability ≠ interpretability or disclosure

Boundary Note

Partial coverage; interpretability and disclosure gaps exist

3

Moderate transparency

Moderate coverage; reporting and selective disclosure gaps exist

Required Evidence

Product documentation with explainability and interpretability

Example Evidence Artifact

Explainability tools, interpretability methods, basic disclosure mechanisms

Negative Control

Disclosure mechanisms ≠ transparency reports or selective disclosure

Boundary Note

Moderate coverage; reporting and selective disclosure gaps exist

4	Strong transparency Strong coverage; selective disclosure and minimization gaps may exist Required Evidence Product documentation with explainability, interpretability, and disclosure Example Evidence Artifact Explainability tools, interpretability methods, disclosure mechanisms, transparency reports Negative Control Transparency reports ≠ selective disclosure or evidence minimization Boundary Note Strong coverage; selective disclosure and minimization gaps may exist
5	Complete capability demonstrated by implementation evidence Complete capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 13 transparency requirement; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove system implementation Boundary Note Complete capability requires implementation evidence; legal text alone insufficient

RCCS-06: Human Oversight

Core question: Does the system provide primitives for human oversight including human-in-the-loop, human-on-the-loop, and human review workflows?

What counts: Human-in-the-loop workflows, human-on-the-loop monitoring, human review workflows, escalation mechanisms

What does not count: General approval buttons, human oversight without lifecycle integration, human oversight without accountability mapping

Table T-E-06: RCCS-06 Human Oversight Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No human oversight capability

Absence of human oversight primitives

Required Evidence

None

Example Evidence Artifact

No human-in-the-loop, no human-on-the-loop, no human review

Negative Control

N/A

Boundary Note

Absence of human oversight primitives

1

Minimal human oversight

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Human oversight mentioned in marketing docs but no implementation details

Negative Control

Human oversight mentioned ≠ human oversight implemented

Boundary Note

Vendor claim requires validation

2

Partial human oversight

Partial coverage; responsibility mapping and escalation gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Approval button exists but no human-role-to-MAS responsibility mapping or escalation

Negative Control

Approval button ≠ human-role-to-MAS responsibility mapping

Boundary Note

Partial coverage; responsibility mapping and escalation gaps exist

3

Moderate human oversight

Moderate coverage; escalation and accountability gaps exist

Required Evidence

Product documentation with human-in-the-loop and human-on-the-loop

Example Evidence Artifact

Human-in-the-loop workflows, human-on-the-loop monitoring, basic human review workflows

Negative Control

Human review ≠ escalation mechanisms or accountability mapping

Boundary Note

Moderate coverage; escalation and accountability gaps exist

4	Strong human oversight Strong coverage; accountability and accepted outcome gaps may exist Required Evidence Product documentation with human-in-the-loop, human-on-the-loop, and escalation Example Evidence Artifact Human-in-the-loop workflows, human-on-the-loop monitoring, human review workflows, escalation mechanisms Negative Control Escalation mechanisms ≠ accountability mapping or accepted outcome compliance Boundary Note Strong coverage; accountability and accepted outcome gaps may exist
5	Complete capability demonstrated by implementation evidence Complete capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 14 human oversight requirement; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove system implementation Boundary Note Complete capability requires implementation evidence; legal text alone insufficient

RCCS-07: Security

Core question: Does the system provide primitives for security including access control, authentication, authorization, and security monitoring?

What counts: Access control, authentication, authorization, security monitoring, security incident response

What does not count: General application security, security without AI lifecycle integration, security without lifecycle evidence

Table T-E-07: RCCS-07 Security Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No security capability

Absence of security primitives

Required Evidence

None

Example Evidence Artifact

No access control, no authentication, no authorization

Negative Control

N/A

Boundary Note

Absence of security primitives

1

Minimal security

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Security mentioned in marketing docs but no implementation details

Negative Control

Security mentioned ≠ security implemented

Boundary Note

Vendor claim requires validation

2

Partial security

Partial coverage; authentication and authorization gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Access control exists but no authentication or authorization for AI lifecycle

Negative Control

Access control ≠ authentication or authorization

Boundary Note

Partial coverage; authentication and authorization gaps exist

3

Moderate security

Moderate coverage; incident response and lifecycle evidence gaps exist

Required Evidence

Product documentation with access control, authentication, and authorization

Example Evidence Artifact

Access control, authentication, authorization, basic security monitoring

Negative Control

Security monitoring ≠ security incident response or lifecycle evidence

Boundary Note

Moderate coverage; incident response and lifecycle evidence gaps exist

4	Strong security Strong coverage; lifecycle evidence and authority boundary gaps may exist Required Evidence Product documentation with access control, authentication, authorization, and monitoring Example Evidence Artifact Access control, authentication, authorization, security monitoring, security incident response Negative Control Security incident response ≠ lifecycle evidence or delegated authority boundary Boundary Note Strong coverage; lifecycle evidence and authority boundary gaps may exist
5	Complete capability demonstrated by implementation evidence Complete capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Baseline source: NIST AI RMF security requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal or standards text may establish regulatory baseline but does not prove system implementation Boundary Note Complete capability requires implementation evidence; legal text alone insufficient

RCCS-08: Accountability

Core question: Does the system provide primitives for accountability including role assignment, responsibility mapping, and accountability tracking?

What counts: Role assignment, responsibility mapping, accountability tracking, accountability escalation

What does not count: General user management, accountability without lifecycle integration, accountability without accepted outcome compliance

Table T-E-08: RCCS-08 Accountability Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No accountability capability

Absence of accountability primitives

Required Evidence

None

Example Evidence Artifact

No role assignment, no responsibility mapping, no accountability tracking

Negative Control

N/A

Boundary Note

Absence of accountability primitives

1

Minimal accountability

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Accountability mentioned in marketing docs but no implementation details

Negative Control

Accountability mentioned ≠ accountability implemented

Boundary Note

Vendor claim requires validation

2

Partial accountability

Partial coverage; responsibility mapping and tracking gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Role assignment exists but no responsibility mapping or accountability tracking

Negative Control

Role assignment ≠ responsibility mapping or accountability tracking

Boundary Note

Partial coverage; responsibility mapping and tracking gaps exist

3

Moderate accountability

Moderate coverage; escalation and accepted outcome gaps exist

Required Evidence

Product documentation with role assignment and responsibility mapping

Example Evidence Artifact

Role assignment, responsibility mapping, basic accountability tracking

Negative Control

Accountability tracking ≠ accountability escalation or accepted outcome compliance

Boundary Note

Moderate coverage; escalation and accepted outcome gaps exist

4	Strong accountability Strong coverage; accepted outcome and human-role-to-MAS gaps may exist Required Evidence Product documentation with role assignment, responsibility mapping, and tracking Example Evidence Artifact Role assignment, responsibility mapping, accountability tracking, accountability escalation Negative Control Accountability escalation ≠ accepted outcome compliance or human-role-to-MAS mapping Boundary Note Strong coverage; accepted outcome and human-role-to-MAS gaps may exist
5	Complete capability demonstrated by implementation evidence Complete capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 26 accountability requirement; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove system implementation Boundary Note Complete capability requires implementation evidence; legal text alone insufficient

RCCS-09: Contestability

Core question: Does the system provide primitives for contestability including dispute resolution, appeal mechanisms, and remediation workflows?

What counts: Dispute resolution workflows, appeal mechanisms, remediation workflows, contestability tracking

What does not count: General customer support, contestability without lifecycle integration, contestability without remediation closure

Table T-E-09: RCCS-09 Contestability Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No contestability capability

Absence of contestability primitives

Required Evidence

None

Example Evidence Artifact

No dispute resolution, no appeal mechanisms, no remediation workflows

Negative Control

N/A

Boundary Note

Absence of contestability primitives

1

Minimal contestability

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Contestability mentioned in marketing docs but no implementation details

Negative Control

Contestability mentioned ≠ contestability implemented

Boundary Note

Vendor claim requires validation

2

Partial contestability

Partial coverage; appeal and remediation gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Dispute resolution workflow exists but no appeal mechanisms or remediation workflows

Negative Control

Dispute resolution ≠ appeal mechanisms or remediation workflows

Boundary Note

Partial coverage; appeal and remediation gaps exist

3

Moderate contestability

Moderate coverage; closure and tracking gaps exist

Required Evidence

Product documentation with dispute resolution and appeal mechanisms

Example Evidence Artifact

Dispute resolution workflows, appeal mechanisms, basic remediation workflows

Negative Control

Remediation workflows ≠ remediation closure or contestability tracking

Boundary Note

Moderate coverage; closure and tracking gaps exist

4	Strong contestability Strong coverage; closure gaps may exist Required Evidence Product documentation with dispute resolution, appeal, and remediation Example Evidence Artifact Dispute resolution workflows, appeal mechanisms, remediation workflows, contestability tracking Negative Control Contestability tracking ≠ remediation closure or incident/dispute/remediation closure Boundary Note Strong coverage; closure gaps may exist
5	Complete capability demonstrated by implementation evidence Complete capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: GDPR Article 22 right to contest automated decision-making; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove system implementation Boundary Note Complete capability requires implementation evidence; legal text alone insufficient

RCCS-10: Monitoring

Core question: Does the system provide primitives for monitoring including performance monitoring, drift detection, and continuous improvement?

What counts: Performance monitoring, drift detection, continuous improvement workflows, monitoring dashboards

What does not count: General application monitoring, monitoring without AI lifecycle integration, monitoring without drift detection

Table T-E-10: RCCS-10 Monitoring Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No monitoring capability

Absence of monitoring primitives

Required Evidence

None

Example Evidence Artifact

No performance monitoring, no drift detection, no continuous improvement

Negative Control

N/A

Boundary Note

Absence of monitoring primitives

1

Minimal monitoring

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Monitoring mentioned in marketing docs but no implementation details

Negative Control

Monitoring mentioned ≠ monitoring implemented

Boundary Note

Vendor claim requires validation

2

Partial monitoring

Partial coverage; drift detection and continuous improvement gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Performance monitoring exists but no drift detection or continuous improvement

Negative Control

Performance monitoring ≠ drift detection or continuous improvement

Boundary Note

Partial coverage; drift detection and continuous improvement gaps exist

3

Moderate monitoring

Moderate coverage; dashboards and authority drift gaps exist

Required Evidence

Product documentation with performance monitoring and drift detection

Example Evidence Artifact

Performance monitoring, drift detection, basic continuous improvement workflows

Negative Control

Continuous improvement ≠ monitoring dashboards or authority drift detection

Boundary Note

Moderate coverage; dashboards and authority drift gaps exist

4	Strong monitoring Strong coverage; authority drift and lifecycle drift gaps may exist Required Evidence Product documentation with performance monitoring, drift detection, and continuous improvement Example Evidence Artifact Performance monitoring, drift detection, continuous improvement workflows, monitoring dashboards Negative Control Monitoring dashboards ≠ authority drift detection or lifecycle drift detection Boundary Note Strong coverage; authority drift and lifecycle drift gaps may exist
5	Complete capability demonstrated by implementation evidence Complete capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 72 post-market monitoring requirement; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove system implementation Boundary Note Complete capability requires implementation evidence; legal text alone insufficient

E.3 RCCS-M Adjustment Matrix

RCCS-M uses the same ten RCCS dimensions as RCCS-T, but each dimension is tested for lifecycle responsibility object expression. Table T-E-11 is the reusable adjustment matrix for scoring RCCS-M. It aligns with Table T-07-01A in Chapter 7 and provides the practical evidence tests evaluators should apply.

Table T-E-11: RCCS-M Adjustment Matrix

Methodology matrix rendered as semantic row cards to preserve long evidence and boundary text; all source rows are retained.

RCCS-01 Risk Management	
RCCS-M Adjustment Question	Are risks linked to lifecycle authority, evidence, mitigation, monitoring, and incident/dispute/remediation closure objects?
Related MROs	MRO-02 , MRO-07 , MRO-08 , MRO-16
Required MRO Evidence	Risk object, authority boundary, mitigation record, evidence pointer, monitoring/closure record
Negative Control	Risk register or dashboard without lifecycle linkage

RCCS-02 Data Governance

RCCS-M Adjustment Question	Are data flows mapped across agents, tools, memory, evidence stores, privacy boundaries, reuse contexts, and processor/subprocessor chains?
Related MROs	MRO-08 , MRO-10 , MRO-12 , MRO-13 , MRO-14
Required MRO Evidence	Lifecycle data-flow map, selective disclosure profile, retention/erasure policy, processor chain record
Negative Control	Dataset lineage without agent/runtime lifecycle mapping

RCCS-03 Documentation

RCCS-M Adjustment Question	Does documentation bind intent, plan, authority, role, evidence, accepted outcome, privacy boundary, and substitution context?
Related MROs	MRO-01 , MRO-02 , MRO-04 , MRO-08 , MRO-15
Required MRO Evidence	Versioned lifecycle documentation, object schema, trace manifest, substitution record
Negative Control	Static model card or system card without lifecycle responsibility objects

RCCS-04 Record-Keeping

RCCS-M Adjustment Question	Are records partitioned by role, authority boundary, agent/tool action, evidence pointer, privacy boundary, accepted outcome, and closure state?
Related MROs	MRO-08 , MRO-12 , MRO-13 , MRO-16
Required MRO Evidence	Partitioned evidence chain, evidence pointer, disclosure policy, closure record
Negative Control	Generic logs without lifecycle responsibility partitioning

RCCS-05 Transparency

RCCS-M Adjustment Question	Can disclosure be tied to lifecycle evidence, role, authority, accepted outcome, privacy boundary, selective disclosure, and dispute path?
Related MROs	MRO-04 , MRO-08 , MRO-11 , MRO-12 , MRO-16
Required MRO Evidence	Evidence-linked explanation, disclosure profile, redacted validation pack, dispute replay record
Negative Control	Explanation UI without lifecycle evidence or selective disclosure

RCCS-06 Human Oversight

RCCS-M Adjustment Question	Is oversight bound to human-role responsibility, delegated authority scope, escalation, drift detection, and accepted outcome authority?
Related MROs	MRO-01 , MRO-02 , MRO-03 , MRO-04 , MRO-07
Required MRO Evidence	Human-role map, authority grant, escalation path, acceptance state, drift alert
Negative Control	Generic approve/reject UI without responsibility or authority scope

RCCS-07 Security

RCCS-M Adjustment Question	Are security controls linked to delegated authority, tool-action liability, lifecycle evidence, privacy boundaries, and selective disclosure rules?
Related MROs	MRO-02 , MRO-05 , MRO-08 , MRO-10 , MRO-12
Required MRO Evidence	Delegated authority record, tool-action liability object, evidence partition, privacy boundary
Negative Control	Access control treated as delegated authority proof

RCCS-08 Accountability

RCCS-M Adjustment Question	Are accountability relationships represented as human/agent responsibility mappings, transfer records, accepted outcome states, and liability records?
Related MROs	MRO-01 , MRO-03 , MRO-04 , MRO-05 , MRO-06
Required MRO Evidence	Responsibility map, agent role boundary, transfer record, acceptance record, liability record
Negative Control	User role assignment without responsibility continuity

RCCS-09 Contestability

RCCS-M Adjustment Question	Can challenged outcomes be replayed with evidence, reviewer role, accepted/rejected state, remediation action, selective disclosure, and closure acceptance?
Related MROs	MRO-04 , MRO-08 , MRO-12 , MRO-13 , MRO-16
Required MRO Evidence	Dispute replay pack, evidence manifest, remediation record, closure acceptance
Negative Control	Customer support ticket without lifecycle replay or closure

RCCS-10 Monitoring

RCCS-M Adjustment Question	Does monitoring detect authority drift, lifecycle drift, substitution change, privacy events, incidents, reuse/revalidation needs, and remediation closure?
Related MROs	MRO-07 , MRO-09 , MRO-10 , MRO-15 , MRO-16
Required MRO Evidence	Drift event, reuse/revalidation record, privacy event record, substitution conformance record, incident closure
Negative Control	General observability dashboard without responsibility-object monitoring

Examples:

- Generic logs can support RCCS-T Record-Keeping but not RCCS-M unless partitioned by authority, agent/tool action, evidence pointer, privacy boundary, accepted outcome, and closure state.
- Approval UI can support RCCS-T Human Oversight but not RCCS-M unless bound to human-role responsibility, delegated scope, escalation, drift detection, and accepted outcome authority.

RCCS-M remains author analytical, forward-looking, and MRO-adjusted. It is not current law, certification, legal compliance proof, a regulator-approved benchmark, a procurement benchmark, a vendor ranking, or an official standard.

E.4 RCCS Evidence Requirements

Evidence hierarchy:

- L1 (Binding Legal Text):** EU AI Act, GDPR, Colorado AI Act, formal standards (NIST AI RMF, ISO/IEC 42001)
- L2 (Product Docs):** Official product documentation, protocol specifications, GitHub repositories
- L3 (Audit Report):** Third-party audit reports, academic papers, industry reports
- L4 (Vendor Claim):** Vendor claims, marketing materials, blog posts
- L5 (Author Inference):** Author inference, analytical framework, logical derivation

Evidence multiplier application:

Illustrative non-normative template

Dimension Score = (Raw Score / 5) × 100 × Evidence Confidence Multiplier

Where:
Raw Score = 0–5 based on capability assessment
Evidence Confidence Multiplier = L1=1.00, L2=0.85, L3=0.75, L4=0.55, L5=0.35

Example:

Illustrative non-normative template

RCCS-04 Record-Keeping
Raw Score: 4 (Strong capability)
Evidence Level: L2 (Product documentation)
Evidence Multiplier: 0.85
Dimension Score: (4/5) × 100 × 0.85 = 68.0

E.5 RCCS Scoring Notes and Negative Controls

Negative controls prevent overclaim:

1. Risk management mentioned ≠ risk management implemented
- Vendor marketing materials may mention risk management without providing implementation details
 - Require product documentation or audit evidence to validate implementation
1. Data quality ≠ data lineage or retention

- Data quality checks do not automatically provide data lineage tracking or retention policies
- Require explicit evidence of lineage and retention capabilities

1. Model card ≠ system card or deployment documentation

- Model cards document model-level information but do not provide system-level or deployment-level documentation
- Require explicit evidence of system cards and deployment documentation

1. Audit trails ≠ evidence export or retention

- Audit trails provide execution records but do not automatically provide evidence export or retention capabilities
- Require explicit evidence of export and retention capabilities

1. Explainability ≠ interpretability or disclosure

- Explainability tools provide post-hoc explanations but do not automatically provide interpretability methods or disclosure mechanisms
- Require explicit evidence of interpretability and disclosure capabilities

1. Approval button ≠ human-role-to-MAS responsibility mapping

- Approval buttons provide UI controls but do not automatically provide human-role-to-MAS responsibility mapping or accountability tracking
- Require explicit evidence of responsibility mapping and accountability tracking

1. Access control ≠ authentication or authorization

- Access control provides basic security but does not automatically provide authentication or authorization for AI lifecycle
- Require explicit evidence of authentication and authorization capabilities

1. Role assignment ≠ responsibility mapping or accountability tracking

- Role assignment provides user management but does not automatically provide responsibility mapping or accountability tracking
- Require explicit evidence of responsibility mapping and accountability tracking

1. Dispute resolution ≠ appeal mechanisms or remediation workflows

- Dispute resolution workflows provide basic contestability but do not automatically provide appeal mechanisms or remediation workflows
- Require explicit evidence of appeal and remediation capabilities

1. Performance monitoring ≠ drift detection or continuous improvement

- Performance monitoring provides basic observability but does not automatically provide drift detection or continuous improvement workflows

- Require explicit evidence of drift detection and continuous improvement capabilities

Boundary discipline:

- RCCS measures regulatory compliance coverage primitives, not legal compliance
- High RCCS scores do not prove legal compliance, regulatory approval, or certification
- Low RCCS scores do not prove non-compliance or unsuitability
- RCCS scores are context-independent; organizations must interpret within their legal, regulatory, and operational context
- RCCS scores require evidence; evidence level affects score multiplier
- RCCS scores are not product rankings; they compare systems by regulatory compliance coverage primitives
- RCCS-M is author analytical, forward-looking, and MRO-adjusted; it is not current law, certification, legal compliance proof, a regulator-approved benchmark, a procurement benchmark, or a vendor ranking

Appendix F — ALCS Detailed Scoring Rubrics

Version: v0.3.2-FRC-R3

F.0 ALCS Rubric Boundary

This appendix provides detailed scoring rubrics for all fifteen ALCS (Agentic Lifecycle Conformance Score) dimensions. These rubrics are analytical tools for evaluating system capability to support agentic lifecycle responsibility object maturity. They are not legal compliance scores, certifications, or regulatory approvals.

Boundary statement:

- **ALCS measures agentic lifecycle responsibility object maturity.** ALCS evaluates whether a system provides primitives that can support agentic lifecycle responsibility tracking, delegation, and accountability. It does not prove legal compliance, regulatory approval, certification, audit readiness, or operational effectiveness.
 - **High ALCS scores do not prove legal compliance.** A high ALCS score indicates that a system provides primitives for agentic lifecycle responsibility management. It does not prove that the system is legally compliant, operationally effective, or enterprise-ready.
 - **Low ALCS scores do not prove non-compliance.** A low ALCS score indicates gaps in agentic lifecycle responsibility primitives. It does not prove that the system is non-compliant, unsafe, or unsuitable for deployment.
 - **ALCS scores are context-independent.** ALCS scores measure system capability, not organizational practice. Organizations must interpret scores within their legal, regulatory, and operational context.
 - **ALCS scores require evidence.** All scores must be supported by evidence. The evidence level (L1 - L5) affects the score multiplier using the canonical formula from Phase 1B-8C.
 - **ALCS scores are not product rankings.** Rubrics compare systems by agentic lifecycle responsibility primitives, not product quality, market leadership, or procurement suitability.
-

F.1 How to Use ALCS Rubrics

Scoring process:

1. **Assess raw capability (0-5)** based on system capability against the rubric
2. **Identify evidence level (L1 - L5)** based on source type

3. **Apply evidence multiplier** using canonical formula: $\text{Dimension Score} = (\text{Raw Score} / 5) \times 100 \times \text{Evidence Confidence Multiplier}$
4. **Document justification** explaining why the raw score and evidence level were assigned
5. **Add boundary note** clarifying what the score does and does not prove

Canonical evidence multipliers (from Phase 1B-8C):

- **L1 (Binding Legal Text):** 1.00
- **L2 (Product Docs):** 0.85
- **L3 (Audit Report):** 0.75
- **L4 (Vendor Claim):** 0.55
- **L5 (Author Inference):** 0.35

Scoring scale:

- **0:** No capability; no evidence
- **1:** Minimal capability; weak evidence
- **2:** Partial capability; some evidence
- **3:** Moderate capability; documented evidence
- **4:** Strong capability; validated evidence
- **5:** Complete capability demonstrated by implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts. Legal text may establish a regulatory baseline but does not by itself prove implementation.

In score-5 rows, legal references identify the regulatory baseline only. They do not count as implementation evidence unless paired with implementation artifacts, formal audit evidence, protocol-level evidence, or validated operational records.

Negative controls:

Each rubric includes negative controls to prevent overclaim. Negative controls distinguish adjacent capabilities from agentic lifecycle responsibility primitives.

F.2 ALCS Detailed Rubrics

ALCS-01: Human-Role-to-MAS Responsibility Mapping

Core question: Does the system provide primitives for mapping human roles to multi-agent system (MAS) responsibilities?

What counts: Human role definitions, MAS responsibility definitions, role-to-responsibility mapping, responsibility tracking, responsibility audit trails

What does not count: General user management, role assignment without MAS integration, responsibility tracking without lifecycle integration

Table T-F-01: ALCS-01 **Human-Role-to-MAS Responsibility Mapping Detailed Rubric**

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0	No human-role-to-MAS mapping capability Absence of human-role-to-MAS mapping primitives Required Evidence None Example Evidence Artifact No human role definitions, no MAS responsibility definitions, no mapping Negative Control N/A Boundary Note Absence of human-role-to-MAS mapping primitives
1	Minimal human-role-to-MAS mapping Vendor claim requires validation Required Evidence Vendor claim or marketing material Example Evidence Artifact Human-role-to-MAS mapping mentioned in marketing docs but no implementation details Negative Control Mapping mentioned ≠ mapping implemented Boundary Note Vendor claim requires validation
2	Partial human-role-to-MAS mapping Partial coverage; MAS responsibility and mapping gaps exist Required Evidence Product documentation with partial coverage Example Evidence Artifact Human role definitions exist but no MAS responsibility definitions or mapping Negative Control Human roles ≠ MAS responsibilities or mapping Boundary Note Partial coverage; MAS responsibility and mapping gaps exist
3	Moderate human-role-to-MAS mapping Moderate coverage; tracking and audit gaps exist Required Evidence Product documentation with human roles and MAS responsibilities Example Evidence Artifact Human role definitions, MAS responsibility definitions, basic role-to-responsibility mapping Negative Control Role-to-responsibility mapping ≠ responsibility tracking or audit trails Boundary Note Moderate coverage; tracking and audit gaps exist

4	Strong human-role-to-MAS mapping Strong coverage; audit and accepted outcome gaps may exist Required Evidence Product documentation with human roles, MAS responsibilities, and tracking Example Evidence Artifact Human role definitions, MAS responsibility definitions, role-to-responsibility mapping, responsibility tracking Negative Control Responsibility tracking ≠ responsibility audit trails or accepted outcome compliance Boundary Note Strong coverage; audit and accepted outcome gaps may exist
5	Complete human-role-to-MAS mapping demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 14 human oversight requirement with MAS integration; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-02: Delegated Authority Boundary

Core question: Does the system provide primitives for defining and enforcing delegated authority boundaries between humans and agents?

What counts: Authority boundary definitions, authority delegation workflows, authority enforcement mechanisms, authority audit trails

What does not count: General permission systems, authority without lifecycle integration, authority without delegation tracking

Table T-F-02: ALCS-02 Delegated Authority Boundary Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No delegated authority boundary capability

Absence of delegated authority boundary primitives

Required Evidence

None

Example Evidence Artifact

No authority boundary definitions, no delegation workflows, no enforcement

Negative Control

N/A

Boundary Note

Absence of delegated authority boundary primitives

1

Minimal delegated authority boundary

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Delegated authority mentioned in marketing docs but no implementation details

Negative Control

Authority mentioned ≠ authority implemented

Boundary Note

Vendor claim requires validation

2

Partial delegated authority boundary

Partial coverage; delegation and enforcement gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Authority boundary definitions exist but no delegation workflows or enforcement

Negative Control

Authority definitions ≠ delegation workflows or enforcement

Boundary Note

Partial coverage; delegation and enforcement gaps exist

3

Moderate delegated authority boundary

Moderate coverage; audit and drift detection gaps exist

Required Evidence

Product documentation with authority boundaries and delegation

Example Evidence Artifact

Authority boundary definitions, authority delegation workflows, basic enforcement mechanisms

Negative Control

Enforcement mechanisms ≠ authority audit trails or authority drift detection

Boundary Note

Moderate coverage; audit and drift detection gaps exist

4	Strong delegated authority boundary Strong coverage; drift detection and transfer tracking gaps may exist Required Evidence Product documentation with authority boundaries, delegation, and enforcement Example Evidence Artifact Authority boundary definitions, authority delegation workflows, enforcement mechanisms, authority audit trails Negative Control Authority audit trails ≠ authority drift detection or responsibility transfer tracking Boundary Note Strong coverage; drift detection and transfer tracking gaps may exist
5	Complete delegated authority boundary demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 14 human oversight with authority boundary requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-03: Agent Role vs Human Role Distinction

Core question: Does the system provide primitives for distinguishing agent roles from human roles in lifecycle workflows?

What counts: Agent role definitions, human role definitions, role distinction mechanisms, role-based access control, role audit trails

What does not count: General role-based access control, role distinction without lifecycle integration, role distinction without accountability tracking

Table T-F-03: ALCS-03 Agent Role vs Human Role Distinction Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No agent-human role distinction capability

Absence of agent-human role distinction primitives

Required Evidence

None

Example Evidence Artifact

No agent role definitions, no human role definitions, no distinction mechanisms

Negative Control

N/A

Boundary Note

Absence of agent-human role distinction primitives

1

Minimal agent-human role distinction

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Agent-human role distinction mentioned in marketing docs but no implementation details

Negative Control

Distinction mentioned ≠ distinction implemented

Boundary Note

Vendor claim requires validation

2

Partial agent-human role distinction

Partial coverage; human role and distinction gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Agent role definitions exist but no human role definitions or distinction mechanisms

Negative Control

Agent roles ≠ human roles or distinction mechanisms

Boundary Note

Partial coverage; human role and distinction gaps exist

3

Moderate agent-human role distinction

Moderate coverage; access control and audit gaps exist

Required Evidence

Product documentation with agent roles, human roles, and distinction

Example Evidence Artifact

Agent role definitions, human role definitions, basic role distinction mechanisms

Negative Control

Role distinction ≠ role-based access control or role audit trails

Boundary Note

Moderate coverage; access control and audit gaps exist

4	Strong agent-human role distinction Strong coverage; audit and accountability gaps may exist Required Evidence Product documentation with agent roles, human roles, distinction, and access control Example Evidence Artifact Agent role definitions, human role definitions, role distinction mechanisms, role-based access control Negative Control Role-based access control ≠ role audit trails or accountability tracking Boundary Note Strong coverage; audit and accountability gaps may exist
5	Complete agent-human role distinction demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 14 human oversight with role distinction requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-04: Accepted Outcome Compliance

Core question: Does the system provide primitives for tracking accepted outcomes and ensuring compliance with accepted outcome boundaries?

What counts: Accepted outcome definitions, outcome acceptance workflows, outcome compliance tracking, outcome audit trails

What does not count: General output validation, outcome tracking without acceptance workflows, outcome tracking without compliance enforcement

Table T-F-04: **ALCS-04** Accepted Outcome Compliance Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No accepted outcome compliance capability

Absence of accepted outcome compliance primitives

Required Evidence

None

Example Evidence Artifact

No accepted outcome definitions, no acceptance workflows, no compliance tracking

Negative Control

N/A

Boundary Note

Absence of accepted outcome compliance primitives

1

Minimal accepted outcome compliance

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Accepted outcome compliance mentioned in marketing docs but no implementation details

Negative Control

Compliance mentioned ≠ compliance implemented

Boundary Note

Vendor claim requires validation

2

Partial accepted outcome compliance

Partial coverage; acceptance and compliance gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Accepted outcome definitions exist but no acceptance workflows or compliance tracking

Negative Control

Outcome definitions ≠ acceptance workflows or compliance tracking

Boundary Note

Partial coverage; acceptance and compliance gaps exist

3

Moderate accepted outcome compliance

Moderate coverage; audit and enforcement gaps exist

Required Evidence

Product documentation with outcome definitions and acceptance workflows

Example Evidence Artifact

Accepted outcome definitions, outcome acceptance workflows, basic compliance tracking

Negative Control

Compliance tracking ≠ outcome audit trails or compliance enforcement

Boundary Note

Moderate coverage; audit and enforcement gaps exist

4	Strong accepted outcome compliance Strong coverage; enforcement and transfer tracking gaps may exist Required Evidence Product documentation with outcome definitions, acceptance, and compliance Example Evidence Artifact Accepted outcome definitions, outcome acceptance workflows, compliance tracking, outcome audit trails Negative Control Outcome audit trails ≠ compliance enforcement or responsibility transfer tracking Boundary Note Strong coverage; enforcement and transfer tracking gaps may exist
5	Complete accepted outcome compliance demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 14 human oversight with outcome acceptance requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-05: Tool-Action Liability Boundary

Core question: Does the system provide primitives for defining and tracking tool-action liability boundaries?

What counts: Tool-action liability definitions, liability boundary enforcement, liability tracking, liability audit trails

What does not count: General tool usage tracking, liability tracking without boundary enforcement, liability tracking without lifecycle integration

Table T-F-05: **ALCS-05** Tool-Action Liability Boundary Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No tool-action liability boundary capability

Absence of tool-action liability boundary primitives

Required Evidence

None

Example Evidence Artifact

No liability definitions, no boundary enforcement, no liability tracking

Negative Control

N/A

Boundary Note

Absence of tool-action liability boundary primitives

1

Minimal tool-action liability boundary

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Tool-action liability mentioned in marketing docs but no implementation details

Negative Control

Liability mentioned ≠ liability implemented

Boundary Note

Vendor claim requires validation

2

Partial tool-action liability boundary

Partial coverage; enforcement and tracking gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Liability definitions exist but no boundary enforcement or liability tracking

Negative Control

Liability definitions ≠ boundary enforcement or tracking

Boundary Note

Partial coverage; enforcement and tracking gaps exist

3

Moderate tool-action liability boundary

Moderate coverage; audit and transfer tracking gaps exist

Required Evidence

Product documentation with liability definitions and boundary enforcement

Example Evidence Artifact

Tool-action liability definitions, liability boundary enforcement, basic liability tracking

Negative Control

Liability tracking ≠ liability audit trails or responsibility transfer tracking

Boundary Note

Moderate coverage; audit and transfer tracking gaps exist

4	Strong tool-action liability boundary Strong coverage; transfer tracking and accepted outcome gaps may exist Required Evidence Product documentation with liability definitions, enforcement, and tracking Example Evidence Artifact Tool-action liability definitions, liability boundary enforcement, liability tracking, liability audit trails Negative Control Liability audit trails ≠ responsibility transfer tracking or accepted outcome compliance Boundary Note Strong coverage; transfer tracking and accepted outcome gaps may exist
5	Complete tool-action liability boundary demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 26 accountability with tool-action liability requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-06: Responsibility Transfer Across Agents

Core question: Does the system provide primitives for tracking responsibility transfer across agents in multi-agent workflows?

What counts: Responsibility transfer definitions, transfer workflows, transfer tracking, transfer audit trails

What does not count: General workflow orchestration, transfer tracking without responsibility mapping, transfer tracking without lifecycle integration

Table T-F-06: ALCS-06 Responsibility Transfer Across Agents Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No responsibility transfer capability

Absence of responsibility transfer primitives

Required Evidence

None

Example Evidence Artifact

No transfer definitions, no transfer workflows, no transfer tracking

Negative Control

N/A

Boundary Note

Absence of responsibility transfer primitives

1

Minimal responsibility transfer

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Responsibility transfer mentioned in marketing docs but no implementation details

Negative Control

Transfer mentioned ≠ transfer implemented

Boundary Note

Vendor claim requires validation

2

Partial responsibility transfer

Partial coverage; workflow and tracking gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Transfer definitions exist but no transfer workflows or transfer tracking

Negative Control

Transfer definitions ≠ transfer workflows or tracking

Boundary Note

Partial coverage; workflow and tracking gaps exist

3

Moderate responsibility transfer

Moderate coverage; audit and accountability gaps exist

Required Evidence

Product documentation with transfer definitions and workflows

Example Evidence Artifact

Responsibility transfer definitions, transfer workflows, basic transfer tracking

Negative Control

Transfer tracking ≠ transfer audit trails or cross-agent accountability

Boundary Note

Moderate coverage; audit and accountability gaps exist

4	Strong responsibility transfer Strong coverage; accountability and evidence partitioning gaps may exist Required Evidence Product documentation with transfer definitions, workflows, and tracking Example Evidence Artifact Responsibility transfer definitions, transfer workflows, transfer tracking, transfer audit trails Negative Control Transfer audit trails ≠ cross-agent accountability or MAS evidence partitioning Boundary Note Strong coverage; accountability and evidence partitioning gaps may exist
5	Complete responsibility transfer demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 26 accountability with multi-agent responsibility requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-07: Authority Drift Detection

Core question: Does the system provide primitives for detecting authority drift where agents exceed delegated authority boundaries?

What counts: Authority drift definitions, drift detection mechanisms, drift alerting, drift audit trails

What does not count: General anomaly detection, drift detection without authority boundary integration, drift detection without lifecycle integration

Table T-F-07: ALCS-07 Authority Drift Detection Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No authority drift detection capability

Absence of authority drift detection primitives

Required Evidence

None

Example Evidence Artifact

No drift definitions, no drift detection, no drift alerting

Negative Control

N/A

Boundary Note

Absence of authority drift detection primitives

1

Minimal authority drift detection

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Authority drift detection mentioned in marketing docs but no implementation details

Negative Control

Drift detection mentioned ≠ drift detection implemented

Boundary Note

Vendor claim requires validation

2

Partial authority drift detection

Partial coverage; detection and alerting gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Drift definitions exist but no drift detection mechanisms or drift alerting

Negative Control

Drift definitions ≠ drift detection or alerting

Boundary Note

Partial coverage; detection and alerting gaps exist

3

Moderate authority drift detection

Moderate coverage; audit and remediation gaps exist

Required Evidence

Product documentation with drift definitions and detection

Example Evidence Artifact

Authority drift definitions, drift detection mechanisms, basic drift alerting

Negative Control

Drift alerting ≠ drift audit trails or drift remediation workflows

Boundary Note

Moderate coverage; audit and remediation gaps exist

4	Strong authority drift detection Strong coverage; remediation and continuous monitoring gaps may exist Required Evidence Product documentation with drift definitions, detection, and alerting Example Evidence Artifact Authority drift definitions, drift detection mechanisms, drift alerting, drift audit trails Negative Control Drift audit trails ≠ drift remediation workflows or continuous monitoring Boundary Note Strong coverage; remediation and continuous monitoring gaps may exist
5	Complete authority drift detection demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 72 post-market monitoring with authority drift requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-08: MAS Evidence Partitioning

Core question: Does the system provide primitives for partitioning evidence by agent, role, and responsibility in multi-agent systems?

What counts: Evidence partitioning definitions, partitioning mechanisms, partitioned evidence storage, partitioned evidence retrieval

What does not count: General evidence storage, evidence partitioning without agent/role/responsibility mapping, evidence partitioning without lifecycle integration

Table T-F-08: ALCS-08 MAS Evidence Partitioning Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No MAS evidence partitioning capability

Absence of MAS evidence partitioning primitives

Required Evidence

None

Example Evidence Artifact

No partitioning definitions, no partitioning mechanisms, no partitioned storage

Negative Control

N/A

Boundary Note

Absence of MAS evidence partitioning primitives

1

Minimal MAS evidence partitioning

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

MAS evidence partitioning mentioned in marketing docs but no implementation details

Negative Control

Partitioning mentioned ≠ partitioning implemented

Boundary Note

Vendor claim requires validation

2

Partial MAS evidence partitioning

Partial coverage; mechanism and storage gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Partitioning definitions exist but no partitioning mechanisms or partitioned storage

Negative Control

Partitioning definitions ≠ partitioning mechanisms or storage

Boundary Note

Partial coverage; mechanism and storage gaps exist

3

Moderate MAS evidence partitioning

Moderate coverage; retrieval and disclosure gaps exist

Required Evidence

Product documentation with partitioning definitions and mechanisms

Example Evidence Artifact

Evidence partitioning definitions, partitioning mechanisms, basic partitioned evidence storage

Negative Control

Partitioned storage ≠ partitioned evidence retrieval or selective disclosure

Boundary Note

Moderate coverage; retrieval and disclosure gaps exist

4	Strong MAS evidence partitioning Strong coverage; disclosure and minimization gaps may exist Required Evidence Product documentation with partitioning definitions, mechanisms, and storage Example Evidence Artifact Evidence partitioning definitions, partitioning mechanisms, partitioned evidence storage, partitioned evidence retrieval Negative Control Partitioned retrieval ≠ selective disclosure or evidence minimization Boundary Note Strong coverage; disclosure and minimization gaps may exist
5	Complete MAS evidence partitioning demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: GDPR Article 5 data minimization with MAS evidence requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-09: Cross-Project Reuse Compliance

Core question: Does the system provide primitives for ensuring compliance when reusing agents, models, or workflows across projects?

What counts: Reuse compliance definitions, reuse tracking, reuse validation, reuse audit trails

What does not count: General code reuse, reuse tracking without compliance validation, reuse tracking without lifecycle integration

Table T-F-09: **ALCS-09** Cross-Project Reuse Compliance Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No cross-project reuse compliance capability

Absence of cross-project reuse compliance primitives

Required Evidence

None

Example Evidence Artifact

No reuse compliance definitions, no reuse tracking, no reuse validation

Negative Control

N/A

Boundary Note

Absence of cross-project reuse compliance primitives

1

Minimal cross-project reuse compliance

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Cross-project reuse compliance mentioned in marketing docs but no implementation details

Negative Control

Reuse compliance mentioned ≠ reuse compliance implemented

Boundary Note

Vendor claim requires validation

2

Partial cross-project reuse compliance

Partial coverage; tracking and validation gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Reuse compliance definitions exist but no reuse tracking or reuse validation

Negative Control

Reuse definitions ≠ reuse tracking or validation

Boundary Note

Partial coverage; tracking and validation gaps exist

3

Moderate cross-project reuse compliance

Moderate coverage; audit and enforcement gaps exist

Required Evidence

Product documentation with reuse definitions and tracking

Example Evidence Artifact

Reuse compliance definitions, reuse tracking, basic reuse validation

Negative Control

Reuse validation ≠ reuse audit trails or compliance enforcement

Boundary Note

Moderate coverage; audit and enforcement gaps exist

4	Strong cross-project reuse compliance Strong coverage; enforcement and substitution gaps may exist Required Evidence Product documentation with reuse definitions, tracking, and validation Example Evidence Artifact Reuse compliance definitions, reuse tracking, reuse validation, reuse audit trails Negative Control Reuse audit trails ≠ compliance enforcement or vendor substitution conformance Boundary Note Strong coverage; enforcement and substitution gaps may exist
5	Complete cross-project reuse compliance demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 43 conformity assessment with reuse requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-10: Privacy/GDPR Lifecycle Mapping

Core question: Does the system provide primitives for mapping privacy and GDPR requirements to AI lifecycle stages?

What counts: Privacy lifecycle mapping definitions, GDPR requirement mapping, lifecycle stage integration, privacy audit trails

What does not count: General privacy controls, privacy mapping without lifecycle integration, privacy mapping without GDPR-specific requirements

Table T-F-10: **ALCS-10** Privacy/GDPR Lifecycle Mapping Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No privacy/GDPR lifecycle mapping capability

Absence of privacy/GDPR lifecycle mapping primitives

Required Evidence

None

Example Evidence Artifact

No privacy mapping definitions, no GDPR requirement mapping, no lifecycle integration

Negative Control

N/A

Boundary Note

Absence of privacy/GDPR lifecycle mapping primitives

1

Minimal privacy/GDPR lifecycle mapping

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Privacy/GDPR lifecycle mapping mentioned in marketing docs but no implementation details

Negative Control

Mapping mentioned ≠ mapping implemented

Boundary Note

Vendor claim requires validation

2

Partial privacy/GDPR lifecycle mapping

Partial coverage; GDPR and lifecycle gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Privacy mapping definitions exist but no GDPR requirement mapping or lifecycle integration

Negative Control

Privacy definitions ≠ GDPR mapping or lifecycle integration

Boundary Note

Partial coverage; GDPR and lifecycle gaps exist

3

Moderate privacy/GDPR lifecycle mapping

Moderate coverage; audit and data subject rights gaps exist

Required Evidence

Product documentation with privacy mapping and GDPR requirements

Example Evidence Artifact

Privacy lifecycle mapping definitions, GDPR requirement mapping, basic lifecycle stage integration

Negative Control

Lifecycle integration ≠ privacy audit trails or data subject rights workflows

Boundary Note

Moderate coverage; audit and data subject rights gaps exist

4	Strong privacy/GDPR lifecycle mapping Strong coverage; data subject rights and minimization gaps may exist Required Evidence Product documentation with privacy mapping, GDPR requirements, and lifecycle integration Example Evidence Artifact Privacy lifecycle mapping definitions, GDPR requirement mapping, lifecycle stage integration, privacy audit trails Negative Control Privacy audit trails ≠ data subject rights workflows or evidence minimization Boundary Note Strong coverage; data subject rights and minimization gaps may exist
5	Complete privacy/GDPR lifecycle mapping demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: GDPR Article 25 data protection by design with AI lifecycle requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-11: Privacy-Preserving Third-Party Validation

Core question: Does the system provide primitives for third-party validation while preserving privacy and minimizing evidence disclosure?

What counts: Privacy-preserving validation mechanisms, selective disclosure controls, third-party validation workflows, validation audit trails

What does not count: General third-party access, validation without privacy controls, validation without selective disclosure

Table T-F-11: **ALCS-11 Privacy-Preserving Third-Party Validation Detailed Rubric**

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No privacy-preserving third-party validation capability

Absence of privacy-preserving third-party validation primitives

Required Evidence

None

Example Evidence Artifact

No privacy-preserving validation, no selective disclosure, no validation workflows

Negative Control

N/A

Boundary Note

Absence of privacy-preserving third-party validation primitives

1

Minimal privacy-preserving third-party validation

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Privacy-preserving validation mentioned in marketing docs but no implementation details

Negative Control

Validation mentioned ≠ validation implemented

Boundary Note

Vendor claim requires validation

2

Partial privacy-preserving third-party validation

Partial coverage; disclosure and workflow gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Privacy-preserving validation mechanisms exist but no selective disclosure or validation workflows

Negative Control

Privacy mechanisms ≠ selective disclosure or workflows

Boundary Note

Partial coverage; disclosure and workflow gaps exist

3

Moderate privacy-preserving third-party validation

Moderate coverage; audit and minimization gaps exist

Required Evidence

Product documentation with privacy mechanisms and selective disclosure

Example Evidence Artifact

Privacy-preserving validation mechanisms, selective disclosure controls, basic third-party validation workflows

Negative Control

Validation workflows ≠ validation audit trails or evidence minimization

Boundary Note

Moderate coverage; audit and minimization gaps exist

4	Strong privacy-preserving third-party validation Strong coverage; minimization and zero-knowledge gaps may exist Required Evidence Product documentation with privacy mechanisms, disclosure, and workflows Example Evidence Artifact Privacy-preserving validation mechanisms, selective disclosure controls, third-party validation workflows, validation audit trails Negative Control Validation audit trails ≠ evidence minimization or zero-knowledge proofs Boundary Note Strong coverage; minimization and zero-knowledge gaps may exist
5	Complete privacy-preserving third-party validation demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: GDPR Article 5 data minimization with third-party validation requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-12: Evidence Minimization and Selective Disclosure

Core question: Does the system provide primitives for evidence minimization and selective disclosure to reduce privacy risk?

What counts: Evidence minimization policies, selective disclosure mechanisms, disclosure controls, minimization audit trails

What does not count: General data minimization, minimization without lifecycle integration, minimization without selective disclosure controls

Table T-F-12: **ALCS-12** Evidence Minimization and Selective Disclosure Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No evidence minimization capability

Absence of evidence minimization primitives

Required Evidence

None

Example Evidence Artifact

No minimization policies, no selective disclosure, no disclosure controls

Negative Control

N/A

Boundary Note

Absence of evidence minimization primitives

1

Minimal evidence minimization

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Evidence minimization mentioned in marketing docs but no implementation details

Negative Control

Minimization mentioned ≠ minimization implemented

Boundary Note

Vendor claim requires validation

2

Partial evidence minimization

Partial coverage; disclosure and control gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Minimization policies exist but no selective disclosure mechanisms or disclosure controls

Negative Control

Minimization policies ≠ selective disclosure or controls

Boundary Note

Partial coverage; disclosure and control gaps exist

3

Moderate evidence minimization

Moderate coverage; audit and retention gaps exist

Required Evidence

Product documentation with minimization policies and selective disclosure

Example Evidence Artifact

Evidence minimization policies, selective disclosure mechanisms, basic disclosure controls

Negative Control

Disclosure controls ≠ minimization audit trails or retention enforcement

Boundary Note

Moderate coverage; audit and retention gaps exist

4	Strong evidence minimization Strong coverage; retention and data subject rights gaps may exist Required Evidence Product documentation with minimization policies, disclosure, and controls Example Evidence Artifact Evidence minimization policies, selective disclosure mechanisms, disclosure controls, minimization audit trails Negative Control Minimization audit trails ≠ retention enforcement or data subject rights integration Boundary Note Strong coverage; retention and data subject rights gaps may exist
5	Complete evidence minimization demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: GDPR Article 5 data minimization requirement; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-13: Data Subject Rights vs Evidence Retention

Core question: Does the system provide primitives for balancing data subject rights (erasure, portability) with evidence retention requirements?

What counts: Data subject rights workflows, evidence retention policies, rights-retention conflict resolution, conflict audit trails

What does not count: General data subject rights, rights workflows without evidence retention integration, rights workflows without conflict resolution

Table T-F-13: ALCS-13 Data Subject Rights vs Evidence Retention Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No data subject rights vs evidence retention capability

Absence of data subject rights vs evidence retention primitives

Required Evidence

None

Example Evidence Artifact

No data subject rights workflows, no evidence retention policies, no conflict resolution

Negative Control

N/A

Boundary Note

Absence of data subject rights vs evidence retention primitives

1

Minimal data subject rights vs evidence retention

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Data subject rights vs evidence retention mentioned in marketing docs but no implementation details

Negative ControlRights-retention mentioned $\neq$ rights-retention implemented**Boundary Note**

Vendor claim requires validation

2

Partial data subject rights vs evidence retention

Partial coverage; retention and conflict resolution gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Data subject rights workflows exist but no evidence retention policies or conflict resolution

Negative ControlRights workflows $\neq$ retention policies or conflict resolution**Boundary Note**

Partial coverage; retention and conflict resolution gaps exist

3

Moderate data subject rights vs evidence retention

Moderate coverage; audit and legal basis gaps exist

Required Evidence

Product documentation with rights workflows and retention policies

Example Evidence Artifact

Data subject rights workflows, evidence retention policies, basic rights-retention conflict resolution

Negative ControlConflict resolution $\neq$ conflict audit trails or legal basis tracking**Boundary Note**

Moderate coverage; audit and legal basis gaps exist

4	Strong data subject rights vs evidence retention Strong coverage; legal basis and exemption tracking gaps may exist Required Evidence Product documentation with rights workflows, retention policies, and conflict resolution Example Evidence Artifact Data subject rights workflows, evidence retention policies, rights-retention conflict resolution, conflict audit trails Negative Control Conflict audit trails ≠ legal basis tracking or regulatory exemption tracking Boundary Note Strong coverage; legal basis and exemption tracking gaps may exist
5	Complete data subject rights vs evidence retention demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: GDPR Article 17 right to erasure with legal basis exemptions; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-14: Third-Party Processor/Subprocessor Chain

Core question: Does the system provide primitives for tracking third-party processor and subprocessor chains in AI lifecycle workflows?

What counts: Processor chain definitions, processor tracking, processor compliance validation, processor audit trails

What does not count: General vendor management, processor tracking without compliance validation, processor tracking without lifecycle integration

Table T-F-14: **ALCS-14** Third-Party Processor/Subprocessor Chain Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No third-party processor chain capability

Absence of third-party processor chain primitives

Required Evidence

None

Example Evidence Artifact

No processor chain definitions, no processor tracking, no compliance validation

Negative Control

N/A

Boundary Note

Absence of third-party processor chain primitives

1

Minimal third-party processor chain

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Third-party processor chain mentioned in marketing docs but no implementation details

Negative Control

Processor chain mentioned ≠ processor chain implemented

Boundary Note

Vendor claim requires validation

2

Partial third-party processor chain

Partial coverage; tracking and validation gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Processor chain definitions exist but no processor tracking or compliance validation

Negative Control

Processor definitions ≠ processor tracking or validation

Boundary Note

Partial coverage; tracking and validation gaps exist

3

Moderate third-party processor chain

Moderate coverage; audit and notification gaps exist

Required Evidence

Product documentation with processor definitions and tracking

Example Evidence Artifact

Processor chain definitions, processor tracking, basic processor compliance validation

Negative Control

Compliance validation ≠ processor audit trails or subprocessor notification

Boundary Note

Moderate coverage; audit and notification gaps exist

4	Strong third-party processor chain Strong coverage; notification and impact assessment gaps may exist Required Evidence Product documentation with processor definitions, tracking, and validation Example Evidence Artifact Processor chain definitions, processor tracking, processor compliance validation, processor audit trails Negative Control Processor audit trails ≠ subprocessor notification or data transfer impact assessment Boundary Note Strong coverage; notification and impact assessment gaps may exist
5	Complete third-party processor chain demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: GDPR Article 28 processor requirements with subprocessor chain tracking; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

ALCS-15: Vendor/Model/Runtime Substitution Conformance

Core question: Does the system provide primitives for ensuring conformance when substituting vendors, models, or runtimes?

What counts: Substitution conformance definitions, substitution tracking, substitution validation, substitution audit trails

What does not count: General configuration management, substitution tracking without conformance validation, substitution tracking without lifecycle integration

Table T-F-15: **ALCS-15** Vendor/Model/Runtime Substitution Conformance Detailed Rubric

Rubric rendered as score cards to preserve evidence readability; all source rows are retained.

0

No vendor/model/runtime substitution conformance capability

Absence of vendor/model/runtime substitution conformance primitives

Required Evidence

None

Example Evidence Artifact

No substitution conformance definitions, no substitution tracking, no substitution validation

Negative Control

N/A

Boundary Note

Absence of vendor/model/runtime substitution conformance primitives

1

Minimal vendor/model/runtime substitution conformance

Vendor claim requires validation

Required Evidence

Vendor claim or marketing material

Example Evidence Artifact

Substitution conformance mentioned in marketing docs but no implementation details

Negative Control

Substitution mentioned ≠ substitution implemented

Boundary Note

Vendor claim requires validation

2

Partial vendor/model/runtime substitution conformance

Partial coverage; tracking and validation gaps exist

Required Evidence

Product documentation with partial coverage

Example Evidence Artifact

Substitution conformance definitions exist but no substitution tracking or substitution validation

Negative Control

Substitution definitions ≠ substitution tracking or validation

Boundary Note

Partial coverage; tracking and validation gaps exist

3

Moderate vendor/model/runtime substitution conformance

Moderate coverage; audit and reuse compliance gaps exist

Required Evidence

Product documentation with substitution definitions and tracking

Example Evidence Artifact

Substitution conformance definitions, substitution tracking, basic substitution validation

Negative Control

Substitution validation ≠ substitution audit trails or cross-project reuse compliance

Boundary Note

Moderate coverage; audit and reuse compliance gaps exist

4	Strong vendor/model/runtime substitution conformance Strong coverage; reuse compliance and re-partitioning gaps may exist Required Evidence Product documentation with substitution definitions, tracking, and validation Example Evidence Artifact Substitution conformance definitions, substitution tracking, substitution validation, substitution audit trails Negative Control Substitution audit trails ≠ cross-project reuse compliance or evidence re-partitioning Boundary Note Strong coverage; reuse compliance and re-partitioning gaps may exist
5	Complete vendor/model/runtime substitution conformance demonstrated by implementation evidence Complete ALCS capability requires implementation evidence; legal text alone insufficient Required Evidence Implementation evidence, formal audit evidence, protocol-level evidence, or validated operational artifacts Example Evidence Artifact Legal baseline: EU AI Act Article 43 conformity assessment with substitution requirements; implementation evidence: audit report, operational artifact, protocol evidence, or validated workflow record Negative Control Legal text may establish regulatory baseline but does not prove ALCS implementation Boundary Note Complete ALCS capability requires implementation evidence; legal text alone insufficient

F.3 ALCS Evidence Requirements

Evidence hierarchy:

- L1 (Binding Legal Text):** EU AI Act, GDPR, Colorado AI Act, formal standards (NIST AI RMF, ISO/IEC 42001)
- L2 (Product Docs):** Official product documentation, protocol specifications, GitHub repositories
- L3 (Audit Report):** Third-party audit reports, academic papers, industry reports
- L4 (Vendor Claim):** Vendor claims, marketing materials, blog posts
- L5 (Author Inference):** Author inference, analytical framework, logical derivation

Evidence multiplier application:

Illustrative non-normative template
Dimension Score = (Raw Score / 5) × 100 × Evidence Confidence Multiplier Where:

Raw Score = 0–5 based on capability assessment

Evidence Confidence Multiplier = L1=1.00, L2=0.85, L3=0.75, L4=0.55, L5=0.35

Example:

Illustrative non-normative template

ALCS-01 Human-Role-to-MAS Responsibility Mapping

Raw Score: 3 (Moderate capability)

Evidence Level: L2 (Product documentation)

Evidence Multiplier: 0.85

Dimension Score: $(3/5) \times 100 \times 0.85 = 51.0$

F.4 ALCS Scoring Notes and Negative Controls

Negative controls prevent overclaim:

1. Human-role-to-MAS mapping mentioned ≠ mapping implemented

- Vendor marketing materials may mention human-role-to-MAS mapping without providing implementation details
- Require product documentation or audit evidence to validate implementation

1. Authority definitions ≠ delegation workflows or enforcement

- Authority boundary definitions do not automatically provide delegation workflows or enforcement mechanisms
- Require explicit evidence of delegation and enforcement capabilities

1. Agent roles ≠ human roles or distinction mechanisms

- Agent role definitions do not automatically provide human role definitions or distinction mechanisms
- Require explicit evidence of human roles and distinction capabilities

1. Outcome definitions ≠ acceptance workflows or compliance tracking

- Accepted outcome definitions do not automatically provide acceptance workflows or compliance tracking
- Require explicit evidence of acceptance and compliance capabilities

1. Liability definitions ≠ boundary enforcement or tracking

- Tool-action liability definitions do not automatically provide boundary enforcement or liability tracking
- Require explicit evidence of enforcement and tracking capabilities

1. Transfer definitions ≠ transfer workflows or tracking

- Responsibility transfer definitions do not automatically provide transfer workflows or transfer tracking
- Require explicit evidence of workflows and tracking capabilities

1. Drift definitions ≠ drift detection or alerting

- Authority drift definitions do not automatically provide drift detection mechanisms or drift alerting
- Require explicit evidence of detection and alerting capabilities

1. Partitioning definitions ≠ partitioning mechanisms or storage

- Evidence partitioning definitions do not automatically provide partitioning mechanisms or partitioned storage
- Require explicit evidence of mechanisms and storage capabilities

1. Reuse definitions ≠ reuse tracking or validation

- Cross-project reuse compliance definitions do not automatically provide reuse tracking or reuse validation
- Require explicit evidence of tracking and validation capabilities

1. Privacy definitions ≠ GDPR mapping or lifecycle integration

- Privacy lifecycle mapping definitions do not automatically provide GDPR requirement mapping or lifecycle integration
- Require explicit evidence of GDPR mapping and lifecycle integration capabilities

1. Privacy mechanisms ≠ selective disclosure or workflows

- Privacy-preserving validation mechanisms do not automatically provide selective disclosure controls or validation workflows
- Require explicit evidence of disclosure and workflow capabilities

1. Minimization policies ≠ selective disclosure or controls

- Evidence minimization policies do not automatically provide selective disclosure mechanisms or disclosure controls
- Require explicit evidence of disclosure and control capabilities

1. Rights workflows ≠ retention policies or conflict resolution

- Data subject rights workflows do not automatically provide evidence retention policies or conflict resolution
- Require explicit evidence of retention and conflict resolution capabilities

1. Processor definitions ≠ processor tracking or validation

- Processor chain definitions do not automatically provide processor tracking or compliance validation
- Require explicit evidence of tracking and validation capabilities

1. **Substitution definitions ≠ substitution tracking or validation**

- Vendor/model/runtime substitution conformance definitions do not automatically provide substitution tracking or substitution validation
- Require explicit evidence of tracking and validation capabilities

Boundary discipline:

- ALCS measures agentic lifecycle responsibility object maturity, not legal compliance
- High ALCS scores do not prove legal compliance, regulatory approval, or certification
- Low ALCS scores do not prove non-compliance or unsuitability
- ALCS scores are context-independent; organizations must interpret within their legal, regulatory, and operational context
- ALCS scores require evidence; evidence level affects score multiplier
- ALCS scores are not product rankings; they compare systems by agentic lifecycle responsibility primitives

ALCS vs RCCS-T / RCCS-M distinction:

- **RCCS-T** measures traditional regulatory compliance coverage primitives (risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, monitoring)
- **RCCS-M** measures whether regulatory obligations can be expressed through Missing Regulatory Objects and lifecycle responsibility semantics
- **ALCS** measures agentic lifecycle responsibility object maturity (human-role-to-MAS mapping, delegated authority boundary, agent-human role distinction, accepted outcome compliance, tool-action liability boundary, responsibility transfer, authority drift detection, MAS evidence partitioning, cross-project reuse compliance, privacy/GDPR lifecycle mapping, privacy-preserving third-party validation, evidence minimization, data subject rights vs evidence retention, third-party processor chain, vendor/model/runtime substitution conformance)
- **RCCS-T** focuses on traditional regulatory compliance coverage capabilities
- **RCCS-M** focuses on MRO-adjusted regulatory coverage
- **ALCS** focuses on agentic lifecycle responsibility tracking and accountability capabilities
- All three profiles are analytical tools, not legal compliance scores

Optional composite worksheet formulas:

Illustrative non-normative template

Traditional Composite = $0.40 \times \text{RCCS-T} + 0.60 \times \text{ALCS}$
 Agentic Composite = $0.40 \times \text{RCCS-M} + 0.60 \times \text{ALCS}$

Where:

RCCS-T = Average of traditional RCCS dimension scores (0-100)
RCCS-M = MRO-adjusted regulatory coverage score (0-100)
ALCS = Average of 15 ALCS dimension scores (0-100)
Optional composites = Weighted worksheet views (0-100), not the main result display

Example:

Illustrative non-normative template

RCCS-T Adjusted Score: 65.0 (across 10 dimensions, 0-100)
RCCS-M Adjusted Score: 72.0 (MRO-adjusted, 0-100)
ALCS Adjusted Score: 55.0 (across 15 dimensions, 0-100)
Traditional Composite: $0.40 \times 65.0 + 0.60 \times 55.0 = 26.0 + 33.0 = 59.0$
Agentic Composite: $0.40 \times 72.0 + 0.60 \times 55.0 = 28.8 + 33.0 = 61.8$

Appendix G — Expanded System-by-System Assessment

Version: v0.3.2-FRC-R3

G.0 Assessment Boundary and Provisional Status

This appendix provides expanded system-by-system assessments for eight representative AI governance and agentic systems. These assessments are **revalidated provisional** after Phase 1C follow-up. Official source entry points were located for the required **L2** product/protocol groups and material claims were either source-bound, narrowed, downgraded to **L5** analytical interpretation, or retained as explicit lifecycle gaps. The assessments remain provisional because public documentation does not by itself prove deployed lifecycle responsibility-object implementation, audit readiness, legal compliance, procurement suitability, or operational effectiveness.

CRITICAL BOUNDARY STATEMENT:

- **All assessments remain provisional.** Phase 1C claim-level review located official source entry points and narrowed claims, but did not produce final vendor assessments.
- **L2 product documentation is source-bound, not final proof.** Product documentation can support documented capability surfaces; it does not prove deployment practice, audit success, or lifecycle responsibility-object implementation.
- **This is not a product ordering.** Systems are assessed by capability posture, not ordered by quality, market position, or procurement suitability.
- **This is not a procurement recommendation.** Organizations must evaluate systems within their own legal, regulatory, operational, and business context.
- **This is not certification.** Assessments measure capability posture against analytical frameworks (RCCS-T/RCCS-M/ALCS), not legal compliance, regulatory approval, or certification status.
- **Numerical profiles are not in Appendix G.** Numeric RCCS-T / RCCS-M / ALCS profiles appear only in the Evaluation Results section. Appendix G remains qualitative, provisional, and no-score.
- **Assessments are context-independent.** Organizations must interpret assessments within their legal, regulatory, and operational context.
- **Low provisional posture does not prove non-compliance or poor product quality.** It indicates gaps in publicly visible evidence for specific capability primitives.
- **High provisional posture does not prove legal compliance, enterprise readiness, or operational effectiveness.** It indicates stronger publicly visible evidence for specific

capability primitives.

Phase 1C follow-up result:

1. Official **L2** source entry points were located for the eight assessed systems and required source groups.
2. Product/protocol capability claims were bound to source IDs where official documentation supports the surface capability.
3. Lifecycle responsibility-object mappings that are not directly stated in official documentation remain **L5** analytical interpretation.
4. Unsupported or adjacent ALCS claims remain boundary-limited as "requires implementer integration", "not visible as lifecycle responsibility object", or "requires exact source support".
5. No numerical scores, ordering, procurement recommendation, certification claim, or legal compliance proof was introduced.

G.1 How to Read the System Assessments

Assessment structure:

Each system assessment uses a qualitative provisional posture framework across key assessment areas. Assessments do not compute numerical scores, ordered comparisons, or composite metrics. After Phase 1C source binding, they remain revalidated provisional and non-ranking.

Assessment areas:

1. **Traditional Regulatory Compliance Coverage Primitives (RCCS-T-aligned):** Risk management, data governance, documentation, record-keeping, transparency, human oversight, security, accountability, contestability, monitoring
2. **MRO-Adjusted Regulatory Coverage (RCCS-M-aligned):** Whether traditional governance obligations are expressible through Missing Regulatory Objects such as authority boundaries, accepted outcomes, evidence partitioning, responsibility transfer, privacy lifecycle mapping, substitution conformance, and remediation closure
3. **Agentic Lifecycle Responsibility Objects (ALCS-aligned):** Human-role-to-MAS mapping, delegated authority boundary, agent-human role distinction, accepted outcome compliance, tool-action liability boundary, responsibility transfer, authority drift detection, MAS evidence partitioning, cross-project reuse compliance, privacy/GDPR lifecycle mapping, privacy-preserving validation, evidence minimization, data subject rights vs retention, processor chain tracking, vendor/model/runtime substitution conformance

Provisional posture levels:

- **Strong provisional posture:** Multiple capability primitives visible in public product documentation with implementation details

- **Moderate provisional posture:** Some capability primitives visible in public product documentation with partial implementation details
- **Partial provisional posture:** Limited capability primitives visible in public product documentation with minimal implementation details
- **Weak / implementer-built posture:** Few or no capability primitives visible in public product documentation; capabilities may exist but require implementer-built solutions
- **Not visible in public evidence:** No publicly visible evidence for capability primitives; does not prove absence, only lack of public documentation

Evidence types:

- **L1 (Binding Legal Text):** EU AI Act, GDPR, formal standards
- **L2 (Product Docs):** Official product documentation, protocol specifications, GitHub repositories
- **L3 (Audit Report):** Third-party audit reports, academic papers
- **L4 (Vendor Claim):** Vendor claims, marketing materials
- **L5 (Author Inference):** Author inference, analytical framework

What assessments do NOT provide:

- Numerical scores or ordered comparisons
- Composite metrics or weighted averages
- Procurement recommendations or vendor preferences
- Legal compliance proof or certification status
- Regulatory approval or conformity assessment
- Market endorsement or quality ratings
- "Best" or "worst" designations
- Market tier classifications

G.2 System Assessment Profiles

G.2.1 MPLP (Multi-Agent Lifecycle Protocol)

Table T-G-01: MPLP Expanded Assessment

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	No formal risk assessment framework visible
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Data Governance

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Data governance primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Documentation

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Strong provisional posture
Visible Evidence Type	L2 (Protocol spec, GitHub)
Main Lifecycle Gap	Protocol documentation strong; implementation documentation varies
Claim-Level Support Status	source-bound at entry-point level; implementation examples remain deployment-context dependent

Record-Keeping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Strong provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Evidence pack structure visible; export/retention require implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Transparency

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Transparency primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Human Oversight

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Strong provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Human-role-to-MAS mapping visible in protocol; enforcement requires implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Security

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Security primitives not protocol-level; require implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Accountability

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Strong provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Responsibility mapping visible in protocol; tracking requires implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Contestability

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Contestability primitives not protocol-level; require implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Monitoring

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Monitoring primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Human-Role-to-MAS Mapping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Strong provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Protocol-level mapping visible; implementation varies
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Delegated Authority Boundary

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Strong provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Protocol-level boundary visible; enforcement requires implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Agent-Human Role Distinction

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Strong provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Protocol-level distinction visible; implementation varies
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Accepted Outcome Compliance

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Strong provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Protocol-level outcome tracking visible; compliance enforcement requires implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Tool-Action Liability Boundary

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Protocol-level liability tracking visible; boundary enforcement requires implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Responsibility Transfer

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Strong provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Protocol-level transfer tracking visible; implementation varies
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Authority Drift Detection

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Drift detection not protocol-level; requires implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

MAS Evidence Partitioning

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Strong provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Protocol-level evidence partitioning visible; storage/retrieval require implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Cross-Project Reuse Compliance

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Reuse tracking not protocol-level; requires implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Privacy/GDPR Lifecycle Mapping

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Privacy primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Privacy-Preserving Validation

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Privacy-preserving validation not protocol-level; requires implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Evidence Minimization

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Minimization primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Data Subject Rights vs Retention

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Rights-retention conflict resolution not protocol-level; requires implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Processor Chain Tracking

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Processor chain tracking not protocol-level; requires implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

Vendor/Model/Runtime Substitution

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Protocol spec)
Main Lifecycle Gap	Substitution conformance tracking requires implementer integration
Claim-Level Support Status	source-bound at entry-point level; implementation conformance unresolved

MPLP Boundary Note:

MPLP is positioned as a lifecycle protocol path with limited traditional RCCS-T product-platform coverage and strong provisional RCCS-M / ALCS semantic alignment at the protocol specification level. Protocol fit does not prove implementation conformance, legal compliance, enterprise readiness, adoption, or operational effectiveness. MPLP is not required, not exclusive, not certified, and not regulator-approved. Organizations may use MPLP, alternative protocols, or implementer-built solutions based on their requirements.

Provisional Status: REVALIDATED PROVISIONAL — Official MPLP documentation entry point located. Protocol identity is L2-supported; protocol-to-MRO/ALCS mapping remains L5 analytical interpretation unless directly stated in protocol text. Implementation conformance remains unresolved.

G.2.2 IBM watsonx.governance

Table T-G-02: IBM watsonx.governance Expanded Assessment

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Risk assessment framework visible; AI-specific risk integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Governance

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Data governance framework visible; lifecycle integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Documentation

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Model cards and factsheets visible; system cards require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Record-Keeping

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Audit records visible; evidence export capability requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Transparency

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Explainability tools visible; interpretability integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human Oversight

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Approval workflows visible; human-role-to-MAS mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Security

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Enterprise security controls visible; AI lifecycle integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accountability

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Role assignment visible; responsibility tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Contestability

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Dispute resolution primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Monitoring

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Model monitoring visible; drift detection requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human-Role-to-MAS Mapping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Role definitions visible; MAS responsibility mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Delegated Authority Boundary

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Authority boundary primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Agent-Human Role Distinction

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Role distinction visible; lifecycle integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accepted Outcome Compliance

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Outcome compliance primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Tool-Action Liability Boundary

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Liability boundary primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Responsibility Transfer

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Transfer tracking primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Authority Drift Detection

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Drift detection primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

MAS Evidence Partitioning

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Evidence partitioning visible; MAS-specific partitioning requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Cross-Project Reuse Compliance

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Reuse compliance primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy/GDPR Lifecycle Mapping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Privacy controls visible; GDPR lifecycle mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy-Preserving Validation

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Privacy-preserving validation primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Evidence Minimization

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Data minimization visible; evidence minimization requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Subject Rights vs Retention

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Data subject rights visible; conflict resolution requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Processor Chain Tracking

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Processor chain tracking primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Vendor/Model/Runtime Substitution

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Model substitution visible; conformance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Provisional Status: REVALIDATED PROVISIONAL — Official IBM source entry points located. RCCS-adjacent AI governance claims are L2-supported at source-entry level; ALCS-specific lifecycle responsibility-object claims remain L5 unless exact IBM documentation directly supports the object.

G.2.3 Microsoft Azure AI Foundry

Table T-G-03: Microsoft Azure AI Foundry Expanded Assessment

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management	
Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Risk monitoring visible; formal risk assessment framework requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Governance	
Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Data governance controls visible; lifecycle integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Documentation	
Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Extensive documentation visible; model cards and system cards require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Record-Keeping	
Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Logging and audit records visible; evidence export capability requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Transparency

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Transparency tools visible; explainability integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human Oversight

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Human oversight workflows visible; human-role-to-MAS mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Security

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Enterprise security controls visible; AI lifecycle integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accountability

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Role assignment visible; accountability tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Contestability

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Escalation workflows visible; formal contestability requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Monitoring

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Monitoring and metrics visible; drift detection requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human-Role-to-MAS Mapping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Role mapping visible; MAS responsibility mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Delegated Authority Boundary

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Authority controls visible; boundary enforcement requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Agent-Human Role Distinction

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Role distinction visible; lifecycle integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accepted Outcome Compliance

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Outcome validation visible; compliance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Tool-Action Liability Boundary

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Action definitions visible; liability boundary tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Responsibility Transfer

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Workflow handoff visible; responsibility transfer tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Authority Drift Detection

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Drift detection primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

MAS Evidence Partitioning

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Evidence partitioning visible; MAS-specific partitioning requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Cross-Project Reuse Compliance

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Model versioning visible; reuse compliance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy/GDPR Lifecycle Mapping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Privacy controls visible; GDPR lifecycle mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy-Preserving Validation

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Privacy-preserving validation primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Evidence Minimization

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Log filtering visible; evidence minimization requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Subject Rights vs Retention

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Retention policies visible; conflict resolution requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Processor Chain Tracking

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Shared responsibility model visible; processor chain tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Vendor/Model/Runtime Substitution

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Model selection and versioning visible; substitution conformance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Provisional Status: REVALIDATED PROVISIONAL — Official Microsoft source entry points located. Cloud AI platform, agent, tracing, evaluation, and monitoring claims are L2-supported at source-entry level; ALCS-specific lifecycle responsibility-object claims remain L5 unless exact Microsoft documentation directly supports the object.

G.2.4 AWS Bedrock / Guardrails / AgentCore

Table T-G-04: AWS Bedrock / Guardrails / AgentCore Expanded Assessment

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Guardrails visible; formal risk assessment framework requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Governance

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Data governance controls visible; lifecycle integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Documentation

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Extensive AWS documentation visible; model cards and system cards require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Record-Keeping

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	CloudWatch logging and audit records visible; evidence export capability requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Transparency

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Trace logs visible; explainability integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human Oversight

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Action group approval visible; human-role-to-MAS mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Security

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	IAM integration and encryption visible; AI lifecycle security requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accountability

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	IAM roles visible; accountability tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Contestability

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Contestability primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Monitoring

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	CloudWatch monitoring visible; drift detection requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human-Role-to-MAS Mapping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	IAM role mapping visible; MAS responsibility mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Delegated Authority Boundary

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Action group permissions visible; boundary enforcement requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Agent-Human Role Distinction

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Agent definitions and IAM roles visible; distinction mechanisms require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accepted Outcome Compliance

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Guardrails for outcome validation visible; compliance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Tool-Action Liability Boundary

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Action group definitions visible; liability boundary tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Responsibility Transfer

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Agent orchestration visible; responsibility transfer tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Authority Drift Detection

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Drift detection primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

MAS Evidence Partitioning

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Per-agent CloudWatch logs visible; MAS evidence partitioning requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Cross-Project Reuse Compliance

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Agent versioning visible; reuse compliance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy/GDPR Lifecycle Mapping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	AWS compliance controls visible; GDPR lifecycle mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy-Preserving Validation

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Privacy-preserving validation primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Evidence Minimization

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Log filtering visible; evidence minimization requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Subject Rights vs Retention

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Retention policies visible; conflict resolution requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Processor Chain Tracking

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	AWS shared responsibility model visible; processor chain tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Vendor/Model/Runtime Substitution

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Model selection and versioning visible; substitution conformance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Provisional Status: REVALIDATED PROVISIONAL — Official AWS source entry points located for Bedrock, Guardrails, and AgentCore. Guardrail, runtime, security, and observability claims are L2-supported at source-entry level; ALCS-specific lifecycle responsibility-object claims remain L5 unless exact AWS documentation directly supports the object.

G.2.5 Google Vertex AI / ADK / Model Armor

Table T-G-05: Google Vertex AI / ADK / Model Armor Expanded Assessment

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Model monitoring visible; formal risk assessment framework requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Governance

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Data governance controls visible; lifecycle integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Documentation

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Extensive Google Cloud documentation visible; model cards and system cards require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Record-Keeping

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Cloud Logging and audit records visible; evidence export capability requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Transparency

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Conversation logs visible; explainability integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human Oversight

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Human handoff visible; human-role-to-MAS mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Security

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	IAM integration and encryption visible; AI lifecycle security requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accountability

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	IAM roles visible; accountability tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Contestability

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Contestability primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Monitoring

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Cloud Monitoring visible; drift detection requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human-Role-to-MAS Mapping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	IAM role mapping visible; MAS responsibility mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Delegated Authority Boundary

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Tool permissions visible; boundary enforcement requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Agent-Human Role Distinction

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Agent definitions and IAM roles visible; distinction mechanisms require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accepted Outcome Compliance

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Response validation visible; compliance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Tool-Action Liability Boundary

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Tool definitions visible; liability boundary tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Responsibility Transfer

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Agent orchestration visible; responsibility transfer tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Authority Drift Detection

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Drift detection primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

MAS Evidence Partitioning

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Per-agent Cloud Logging visible; MAS evidence partitioning requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Cross-Project Reuse Compliance

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Agent versioning visible; reuse compliance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy/GDPR Lifecycle Mapping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Google Cloud compliance controls visible; GDPR lifecycle mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy-Preserving Validation

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Privacy-preserving validation primitives require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Evidence Minimization

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Log filtering visible; evidence minimization requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Subject Rights vs Retention

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Retention policies visible; conflict resolution requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Processor Chain Tracking

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Google Cloud shared responsibility model visible; processor chain tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Vendor/Model/Runtime Substitution

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Model selection and versioning visible; substitution conformance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Provisional Status: REVALIDATED PROVISIONAL — Official Google source entry points located for Vertex AI, ADK, and Model Armor. Agent-development, model safety, evaluation, and platform claims are L2-supported at source-entry level; ALCS-specific lifecycle responsibility-object claims remain L5 unless exact Google documentation directly supports the object.

G.2.6 LangGraph / LangSmith

Table T-G-06: LangGraph / LangSmith Expanded Assessment

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Callback mechanisms visible; formal risk assessment framework requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Governance

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Data loaders visible; formal data governance framework requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Documentation

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Extensive API documentation visible; model cards and system cards require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Record-Keeping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	LangSmith tracing visible; formal audit trail and evidence export require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Transparency

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Graph visualization visible; explainability integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human Oversight

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Human-in-the-loop via interrupts visible; formal oversight framework requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Security

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	API key management visible; formal security framework requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accountability

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Node naming visible; formal accountability tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Contestability

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Contestability primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Monitoring

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	LangSmith monitoring visible; drift detection requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human-Role-to-MAS Mapping

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Node definitions visible; formal human-role-to-MAS mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Delegated Authority Boundary

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Tool definitions visible; formal boundary enforcement requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Agent-Human Role Distinction

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Node types visible; formal distinction mechanisms require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accepted Outcome Compliance

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Output validation visible; formal compliance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Tool-Action Liability Boundary

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Tool definitions visible; formal liability boundary tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Responsibility Transfer

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Graph edges visible; formal responsibility transfer tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Authority Drift Detection

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Drift detection primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

MAS Evidence Partitioning

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	LangSmith per-node tracing visible; formal MAS evidence partitioning requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Cross-Project Reuse Compliance

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Graph serialization visible; formal reuse compliance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy/GDPR Lifecycle Mapping

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Privacy/GDPR lifecycle mapping primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy-Preserving Validation

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Privacy-preserving validation primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Evidence Minimization

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Evidence minimization primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Subject Rights vs Retention

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Rights-retention conflict resolution primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Processor Chain Tracking

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Processor chain tracking primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Vendor/Model/Runtime Substitution

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Model abstraction visible; formal substitution conformance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Provisional Status: REVALIDATED PROVISIONAL — Official LangGraph and LangSmith source entry points located. Orchestration, persistence, human-in-the-loop, tracing, and evaluation/observability claims are L2-supported at source-entry level; ALCS-specific lifecycle responsibility-object claims remain L5 unless exact documentation directly supports the object.

G.2.7 OpenAI Agents SDK

Table T-G-07: OpenAI Agents SDK Expanded Assessment

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Usage monitoring visible; formal risk assessment framework requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Governance

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	File management visible; formal data governance framework requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Documentation

Provisional RCCS Posture	Strong provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Extensive API documentation and model cards visible; system cards require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Record-Keeping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Thread history visible; formal audit trail and evidence export require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Transparency

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Run steps visible; explainability integration requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human Oversight

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Function calling approval patterns visible; formal oversight framework requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Security

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	API key management and SOC 2 compliance visible; formal AI lifecycle security requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accountability

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Assistant naming visible; formal accountability tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Contestability

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Contestability primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Monitoring

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Usage metrics visible; drift detection requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human-Role-to-MAS Mapping

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Function calling visible; formal human-role-to-MAS mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Delegated Authority Boundary

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Function definitions visible; formal boundary enforcement requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Agent-Human Role Distinction

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Assistant/user message distinction visible; formal distinction mechanisms require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accepted Outcome Compliance

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Output validation visible; formal compliance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Tool-Action Liability Boundary

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Function calling visible; formal liability boundary tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Responsibility Transfer

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Thread management visible; formal responsibility transfer tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Authority Drift Detection

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Drift detection primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

MAS Evidence Partitioning

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Per-thread history visible; formal MAS evidence partitioning requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Cross-Project Reuse Compliance

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Assistant definitions visible; formal reuse compliance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy/GDPR Lifecycle Mapping

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Data retention controls visible; formal GDPR lifecycle mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy-Preserving Validation

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Privacy-preserving validation primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Evidence Minimization

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Evidence minimization primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Subject Rights vs Retention

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Data deletion visible; formal conflict resolution requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Processor Chain Tracking

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Processor chain tracking primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Vendor/Model/Runtime Substitution

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs)
Main Lifecycle Gap	Model selection visible; formal substitution conformance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Provisional Status: REVALIDATED PROVISIONAL — Official OpenAI Agents SDK source entry point located, with platform docs requiring final citation rendering/access check where applicable. Agent execution, tools, handoffs, guardrails, sessions, and tracing claims are L2 - supported at source-entry level; ALCS-specific lifecycle responsibility-object claims remain L5 unless exact documentation directly supports the object.

G.2.8 CrewAI

Table T-G-08: CrewAI Expanded Assessment

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Risk Management

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Risk management primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Governance

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Data governance primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Documentation

Provisional RCCS Posture	Moderate provisional posture
Provisional ALCS Posture	Moderate provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	API documentation visible; model cards and system cards require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Record-Keeping

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Task execution logs visible; formal audit trail and evidence export require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Transparency

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Transparency primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human Oversight

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Human oversight primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Security

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Security primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accountability

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Agent roles and task assignment visible; formal accountability tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Contestability

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Contestability primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Monitoring

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Monitoring primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Human-Role-to-MAS Mapping

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Agent roles and task delegation visible; formal human-role-to-MAS mapping requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Delegated Authority Boundary

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Authority boundary primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Agent-Human Role Distinction

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Agent role definitions visible; formal distinction mechanisms require validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Accepted Outcome Compliance

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Outcome compliance primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Tool-Action Liability Boundary

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Liability boundary primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Responsibility Transfer

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Task delegation across agents visible; formal responsibility transfer tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Authority Drift Detection

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Drift detection primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

MAS Evidence Partitioning

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Per-agent task logs visible; formal MAS evidence partitioning requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Cross-Project Reuse Compliance

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Reuse compliance primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy/GDPR Lifecycle Mapping

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Privacy/GDPR lifecycle mapping primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Privacy-Preserving Validation

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Privacy-preserving validation primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Evidence Minimization

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Evidence minimization primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Data Subject Rights vs Retention

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Rights-retention conflict resolution primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Processor Chain Tracking

Provisional RCCS Posture	Weak / implementer-built posture
Provisional ALCS Posture	Weak / implementer-built posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Processor chain tracking primitives require implementer integration
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Vendor/Model/Runtime Substitution

Provisional RCCS Posture	Partial provisional posture
Provisional ALCS Posture	Partial provisional posture
Visible Evidence Type	L2 (Product docs, GitHub)
Main Lifecycle Gap	Model configuration visible; formal substitution conformance tracking requires validation
Claim-Level Support Status	source-bound at entry-point level; exact lifecycle-object support not visible unless directly stated

Provisional Status: REVALIDATED PROVISIONAL — Official CrewAI documentation and repository entry points located. Agent, task, crew/flow, tool, and orchestration claims are L2 - supported at source-entry level; ALCS-specific lifecycle responsibility-object claims remain L5 unless exact CrewAI documentation directly supports the object.

G.3 Cross-System Pattern Summary

Table T-G-09: Cross-System Pattern Summary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

RCCS Coverage

Observed Pattern	Enterprise cloud platforms show stronger RCCS provisional posture due to integrated monitoring, logging, security, and governance controls
Systems with More Visible Posture	Microsoft Azure AI Foundry, AWS Bedrock, Google Vertex AI, IBM watsonx.governance
Systems with More Limited Visible Posture	CrewAI, LangGraph/LangSmith, OpenAI Agents SDK
Claim-Level Support Status	source-bound at entry-point level; integration support remains deployment-context dependent

ALCS Coverage

Observed Pattern	MPLP shows stronger visible ALCS provisional posture at protocol specification level; enterprise platforms show moderate visible posture; open-source frameworks show more limited visible posture
Systems with More Visible Posture	MPLP (protocol-level), Microsoft Azure AI Foundry, AWS Bedrock, Google Vertex AI
Systems with More Limited Visible Posture	CrewAI, LangGraph/LangSmith, OpenAI Agents SDK
Claim-Level Support Status	source-bound at entry-point level; ALCS implementation conformance unresolved

Documentation

Observed Pattern	All systems show strong to moderate documentation provisional posture; model cards and system cards require validation across all systems
Systems with More Visible Posture	All systems
Systems with More Limited Visible Posture	None
Claim-Level Support Status	source-bound at entry-point level; system-card support remains product- and deployment-dependent

Record-Keeping

Observed Pattern	Enterprise platforms show strong record-keeping provisional posture with cloud logging integration; open-source frameworks show partial to moderate posture
Systems with More Visible Posture	Microsoft Azure AI Foundry, AWS Bedrock, Google Vertex AI, IBM watsonx.governance
Systems with More Limited Visible Posture	CrewAI
Claim-Level Support Status	source-bound at entry-point level; evidence export support remains deployment-context dependent

Human Oversight

Observed Pattern	Microsoft Azure AI Foundry shows stronger visible human oversight provisional posture; MPLP shows strong protocol-level posture; other systems show moderate to more limited visible posture
Systems with More Visible Posture	Microsoft Azure AI Foundry, MPLP (protocol-level)
Systems with More Limited Visible Posture	CrewAI, LangGraph/LangSmith
Claim-Level Support Status	source-bound at entry-point level; lifecycle responsibility-object support remains L5 unless directly stated

Security

Observed Pattern	Enterprise platforms show strong security provisional posture with IAM integration; open-source frameworks show weak to partial posture
Systems with More Visible Posture	Microsoft Azure AI Foundry, AWS Bedrock, Google Vertex AI, IBM watsonx.governance
Systems with More Limited Visible Posture	CrewAI, LangGraph/LangSmith, OpenAI Agents SDK
Claim-Level Support Status	source-bound at entry-point level; AI lifecycle security integration remains deployment-context dependent

Accountability

Observed Pattern	All systems show partial to moderate accountability provisional posture; formal accountability tracking requires validation across all systems
Systems with More Visible Posture	None
Systems with More Limited Visible Posture	None
Claim-Level Support Status	source-bound at entry-point level; accountability tracking remains L5 unless directly stated

Contestability

Observed Pattern	All systems show weak to partial contestability provisional posture; contestability primitives require implementer integration or validation
Systems with More Visible Posture	None
Systems with More Limited Visible Posture	All systems
Claim-Level Support Status	source-bound at entry-point level; contestability support remains implementation-context dependent

Authority Drift Detection

Observed Pattern	All systems show weak to partial authority drift detection provisional posture; drift detection primitives require implementer integration or validation
Systems with More Visible Posture	None
Systems with More Limited Visible Posture	All systems
Claim-Level Support Status	source-bound at entry-point level; authority drift support remains L5 unless directly stated

Privacy/GDPR Lifecycle Mapping

Observed Pattern	Enterprise platforms show moderate privacy provisional posture; open-source frameworks show more limited visible posture; formal GDPR lifecycle mapping requires validation
Systems with More Visible Posture	Microsoft Azure AI Foundry, AWS Bedrock, Google Vertex AI, IBM watsonx.governance
Systems with More Limited Visible Posture	CrewAI, LangGraph/LangSmith, OpenAI Agents SDK
Claim-Level Support Status	source-bound at entry-point level; GDPR lifecycle-object support remains L5 unless directly stated

Implementer-Built Gap

Observed Pattern	Open-source frameworks (CrewAI, LangGraph/LangSmith) show significant implementer-built gaps across multiple assessment areas; enterprise platforms show fewer gaps
Systems with More Visible Posture	Microsoft Azure AI Foundry, AWS Bedrock, Google Vertex AI, IBM watsonx.governance
Systems with More Limited Visible Posture	CrewAI, LangGraph/LangSmith
Claim-Level Support Status	source-bound at entry-point level; platform-provided primitive boundaries remain provisional

Key Observations (Provisional):

- Enterprise platforms show stronger RCCS provisional posture** due to integrated cloud platform capabilities (monitoring, logging, security, governance)
- MPLP shows stronger visible ALCS provisional posture at protocol specification level** but requires implementer integration for operational deployment
- All systems show gaps in contestability and authority drift detection** indicating industry-wide gaps in these capability primitives
- Open-source frameworks show significant implementer-built gaps** requiring organizations to build additional governance layers
- All assessments are provisional** after Phase 1C source binding and require final publication proofing before any finalized assessment posture

CRITICAL REMINDER: These patterns are provisional observations after Phase 1C claim-level source binding. They are not final assessment postures, product ordering, procurement recommendations, or certification.

G.4 Claim-Level Source Support Note

Appendix G has completed Phase 1C claim-level source binding and remains revalidated provisional.

Remaining finalization scope:

1. **Pin exact source pages for each source-bound claim** — Source entry points have been located, but some claims still need exact page-level citation pinning.
2. **Preserve L5 mapping discipline** — Product documentation supporting adjacent platform features must not be treated as direct ALCS lifecycle responsibility-object proof.
3. **Retain no-score/no-ranking posture** — Qualitative postures remain analytical and provisional.
4. **Perform final citation rendering QA in Phase 1D** — No PDF citation rendering has been generated in this phase.

Systems not assessed in Appendix G:

The following systems were mentioned in earlier phases but are not assessed in Appendix G. They may be considered for future versions:

- LangChain (standalone framework; LangGraph/LangSmith assessed instead as representative LangChain ecosystem system)
- AutoGen (not included in representative system set)
- Semantic Kernel (not included in representative system set)
- Microsoft Copilot Studio (Microsoft Azure AI Foundry assessed instead as representative Microsoft system)
- Anthropic Claude API (not included in representative system set)
- OpenAI Assistants API (OpenAI Agents SDK assessed instead as representative OpenAI system)
- Amazon Bedrock Agents (AWS Bedrock / Guardrails / AgentCore assessed instead as comprehensive AWS system)
- Google Vertex AI Agent Builder (Google Vertex AI / ADK / Model Armor assessed instead as comprehensive Google system)

Validation Lab Note:

Validation Lab is mentioned in Chapter 14 as a non-certifying evidence adjudication example. It is not assessed in Appendix G as it is not positioned as a required, exclusive, or certified validation path. Organizations may use Validation Lab, alternative validation approaches, or implementer-built validation solutions based on their requirements.

Appendix H — Enterprise Procurement Scorecard

Version: v0.3.2-FRC-R3

H.0 Procurement Boundary

This appendix provides an enterprise procurement-oriented checklist and scorecard template for evaluating agentic AI systems using RCCS-T, RCCS-M, ALCS, and lifecycle governance requirements.

This appendix is a buyer diligence aid. It is not a procurement recommendation, vendor ranking, legal compliance determination, certification checklist, market endorsement, or substitute for human review. It helps enterprise buyers ask better governance questions before adopting agentic AI systems.

The scorecard may reference RCCS-T, RCCS-M, and ALCS dimensions as analytical prompts. It must not be read as proof that a vendor, platform, protocol, or implementation is legally compliant, operationally effective, enterprise-ready, certified, or regulator-approved. RCCS-M remains author analytical and forward-looking, not current law or a procurement benchmark.

Appendix G may be used only as an example of qualitative provisional assessment categories. Appendix G provisional postures must not be reused as vendor selection advice without enterprise-specific review, page-level citation support, and accountable buyer judgment.

H.1 How Enterprises Should Use This Scorecard

Enterprises should use this scorecard during early diligence, request-for-information review, request-for-proposal design, architecture review, and internal governance review. The scorecard is intended to structure questions, evidence requests, and review boundaries.

Recommended use sequence:

1. Identify the intended agentic AI use case, deployment context, and lifecycle risk profile.
2. Select the relevant RCCS-T, RCCS-M, and ALCS dimensions for that use case.
3. Request evidence from vendors or implementation teams using the evidence request template.
4. Record gaps, red flags, and unresolved assumptions.
5. Escalate material gaps to legal, security, privacy, risk, procurement, and business owners.
6. Defer final evidence interpretation to publication / enterprise review.

This appendix does not tell an enterprise which system to buy. It supports disciplined questioning and evidence collection.

H.2 Procurement Due Diligence Questions

Table T-H-01: Enterprise Procurement Scorecard

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Governance scope	
Procurement Question	What agentic workflows, tools, models, and runtime components are covered by the governance controls?
Related RCCS-T/RCCS-M/ALCS	RCCS-03 , ALCS-15
Required Vendor Evidence	Scope statement, architecture diagram, runtime inventory
Red Flag	Governance claims cover only model behavior while agent workflow remains undefined
Buyer Note	Confirm whether controls apply to the full lifecycle, not only model invocation

Risk management	
Procurement Question	How are agentic risks identified, assessed, mitigated, and monitored over time?
Related RCCS-T/RCCS-M/ALCS	RCCS-01 , RCCS-10
Required Vendor Evidence	Risk register template, monitoring policy, escalation workflow
Red Flag	Risk management is described only as a general platform feature
Buyer Note	Require AI-agent-specific risk handling evidence

Data governance	
Procurement Question	How are input data, context data, generated evidence, and retained records governed?
Related RCCS-T/RCCS-M/ALCS	RCCS-02 , ALCS-10 , ALCS-13
Required Vendor Evidence	Data flow map, retention policy, access control description
Red Flag	Retention and deletion duties are described without lifecycle evidence handling
Buyer Note	Verify privacy and retention conflicts during publication / enterprise review

Documentation

Procurement Question	What documentation exists for model use, agent role, tool access, and deployment configuration?
Related RCCS-T/RCCS-M/ALCS	RCCS-03 , ALCS-03 , ALCS-15
Required Vendor Evidence	System card, model card, deployment record, configuration manifest
Red Flag	Documentation exists for models but not agents or tool actions
Buyer Note	Separate model documentation from agent lifecycle documentation

Record keeping

Procurement Question	What records are created for plans, tool actions, authority grants, evidence packs, reviews, and accepted outcomes?
Related RCCS-T/RCCS-M/ALCS	RCCS-04 , ALCS-08 , ALCS-12
Required Vendor Evidence	Log schema, evidence export sample, retention rule
Red Flag	Logs cannot be connected to lifecycle responsibility objects
Buyer Note	Ask whether records are exportable and reviewable

Human oversight

Procurement Question	How are human roles mapped to agent responsibilities, review points, and escalation authority?
Related RCCS-T/RCCS-M/ALCS	RCCS-06 , ALCS-01 , ALCS-02
Required Vendor Evidence	Responsibility matrix, review workflow, escalation rule
Red Flag	Human oversight is described as a generic approval button
Buyer Note	Require role-to-responsibility mapping

Authority boundaries

Procurement Question	How are delegated authority limits defined, enforced, and changed?
Related RCCS-T/RCCS-M/ALCS	ALCS-02 , ALCS-07
Required Vendor Evidence	Authority policy, change log, exception workflow
Red Flag	Agents can acquire or use tools without visible authority boundaries
Buyer Note	Require authority drift review before deployment

Tool action liability

Procurement Question	How are tool calls, side effects, and responsibility transfers recorded?
ReLated RCCS-T/RCCS-M/ALCS	ALCS-05 , ALCS-06
Required Vendor Evidence	Tool registry, action log schema, handoff record
Red Flag	Tool execution is visible but responsibility assignment is not
Buyer Note	Require traceability from tool action to accountable review owner

Contestability and remediation

Procurement Question	How can contested outputs, incidents, and remediation actions be recorded and closed?
ReLated RCCS-T/RCCS-M/ALCS	RCCS-09 , MRD-16
Required Vendor Evidence	Dispute workflow, incident record, remediation closure template
Red Flag	No clear path for dispute intake or closure evidence
Buyer Note	Align with enterprise incident and dispute processes

Vendor/model/runtime substitution

Procurement Question	How are model, tool, vendor, and runtime changes evaluated before reuse or substitution?
ReLated RCCS-T/RCCS-M/ALCS	ALCS-09 , ALCS-15
Required Vendor Evidence	Change approval record, substitution policy, version manifest
Red Flag	Substitution can occur without lifecycle impact review
Buyer Note	Require substitution conformance evidence

H.3 Lifecycle Governance Procurement Checklist

The following checklist organizes lifecycle governance diligence. Buyers may adapt the checklist for internal approval gates, but should preserve the distinction between evidence request, evidence review, and final enterprise decision.

Checklist areas:

1. Lifecycle scope and system boundary
2. Human role and responsibility mapping
3. Delegated authority and tool boundary
4. Evidence creation, retention, minimization, and export
5. Privacy and data subject rights handling
6. Monitoring, incident, dispute, and remediation closure
7. Vendor, model, tool, and runtime substitution control

8. Evidence level and source quality review

The checklist should be completed by a cross-functional review group. Procurement alone should not decide lifecycle governance sufficiency.

H.4 Evidence Request Template

Table T-H-02: Vendor Evidence Request Template

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

System boundary	
Requested Artifact	Product scope statement and deployment architecture
Minimum Acceptable Evidence	Official documentation or implementation description
Preferred Evidence	Versioned architecture record tied to deployed configuration
Publication / Enterprise Review Note	Review documentation currency before publication or adoption

Human responsibility	
Requested Artifact	Role-to-responsibility mapping
Minimum Acceptable Evidence	Documented workflow or responsibility matrix
Preferred Evidence	Evidence of review workflow execution and exception handling
Publication / Enterprise Review Note	Confirm mapping against active enterprise roles

Delegated authority	
Requested Artifact	Authority policy and enforcement description
Minimum Acceptable Evidence	Product documentation or implementation policy
Preferred Evidence	Validated operational artifact showing grant, denial, and change handling
Publication / Enterprise Review Note	Verify whether enforcement is built-in or implementer-defined

Tool actions

Requested Artifact	Tool registry and action record schema
Minimum Acceptable Evidence	Tool list and logging description
Preferred Evidence	Exportable action records with accountability linkage
Publication / Enterprise Review Note	Validate record completeness before publication or adoption

Evidence handling

Requested Artifact	Evidence pack, log export, or audit record sample
Minimum Acceptable Evidence	Sample artifact with field definitions
Preferred Evidence	Hashable or reviewable evidence artifact with minimization controls
Publication / Enterprise Review Note	Confirm no sensitive evidence over-disclosure

Privacy lifecycle

Requested Artifact	Data flow, retention, and deletion policy
Minimum Acceptable Evidence	Privacy documentation and retention statement
Preferred Evidence	Mapped workflow for retention conflict and data subject request handling
Publication / Enterprise Review Note	Review against current privacy requirements before publication or adoption

Monitoring

Requested Artifact	Monitoring dashboard or alert workflow
Minimum Acceptable Evidence	Monitoring documentation
Preferred Evidence	Operational monitoring artifact tied to lifecycle risk events
Publication / Enterprise Review Note	Confirm monitoring covers agent lifecycle, not only model metrics

Substitution control

Requested Artifact	Versioning and change approval record
Minimum Acceptable Evidence	Change management description
Preferred Evidence	Substitution review artifact covering vendor, model, runtime, and tool changes
Publication / Enterprise Review Note	Review version history and current product state

H.5 Red Flag and Negative Control Checklist

Table T-H-03: Procurement Red Flag Checklist

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Governance evidence covers only model output

Why It Matters	Agentic failures can arise from planning, delegation, tool use, or handoff
Related Lifecycle Risk	Model-centric review misses lifecycle responsibility gaps
Required Clarification	Clarify whether lifecycle events are governed
Boundary	Red flag does not prove non-compliance

Human oversight is described without role mapping

Why It Matters	Oversight without responsibility mapping may not support accountability
Related Lifecycle Risk	Review duty unclear after agent action
Required Clarification	Request role-to-responsibility artifact
Boundary	Requires enterprise context review

Authority limits are not visible

Why It Matters	Agents may operate beyond intended delegated scope
Related Lifecycle Risk	Authority drift and unauthorized tool use
Required Clarification	Request authority boundary and enforcement evidence
Boundary	Absence in public docs does not prove absence in product

Logs are not exportable or interpretable

Why It Matters	Evidence may not support review, audit, or incident reconstruction
Related Lifecycle Risk	Evidence gap during dispute or remediation
Required Clarification	Request export format and retention rule
Boundary	Exportability must be tested in implementation context

Privacy retention conflict is not addressed

Why It Matters	Evidence retention may conflict with privacy obligations
Related Lifecycle Risk	Data subject rights vs evidence retention conflict
Required Clarification	Request retention conflict workflow
Boundary	Legal interpretation remains outside this appendix

Vendor substitution lacks impact review

Why It Matters	Model or runtime changes may change lifecycle risk
Related Lifecycle Risk	Reuse and substitution conformance gap
Required Clarification	Request change review template
Boundary	This checklist does not approve substitutions

Claims rely only on marketing material

Why It Matters	Evidence confidence is lower than product docs or validated artifacts
Related Lifecycle Risk	Overstated capability posture
Required Clarification	Request higher-confidence evidence
Boundary	Evidence level must be recorded explicitly

Appendix G posture is treated as a purchase conclusion

Why It Matters	Appendix G is provisional and not procurement advice
Related Lifecycle Risk	Premature selection based on provisional source posture
Required Clarification	Require enterprise-specific review, page-level citation support, and accountable buyer judgment before final use
Boundary	Appendix G categories are illustrative only

H.6 Procurement Decision Boundary

Table T-H-04: Buyer Interpretation Boundary

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

RFI/RFP design	
What This Scorecard Supports	Structured governance questions and evidence requests
What It Does Not Support	Automatic vendor filtering or vendor ordering
Required Human Review	Procurement, risk, legal, privacy, security

Architecture review	
What This Scorecard Supports	Identification of lifecycle governance evidence gaps
What It Does Not Support	Proof of production readiness
Required Human Review	Architecture, security, platform owner

Risk review	
What This Scorecard Supports	Mapping of visible evidence to RCCS-T/RCCS-M/ALCS prompts
What It Does Not Support	Legal compliance determination
Required Human Review	Risk, legal, compliance

Privacy review	
What This Scorecard Supports	Identification of retention and data subject rights questions
What It Does Not Support	Privacy law conclusion
Required Human Review	Privacy, legal, data governance

Contract negotiation	
What This Scorecard Supports	Evidence requirements and control expectations
What It Does Not Support	Contract sufficiency determination
Required Human Review	Legal, procurement, business owner

Deployment gate

What This Scorecard Supports	Checklist for unresolved assumptions
What It Does Not Support	Certification or approval
Required Human Review	Enterprise governance board or equivalent owner

Procurement decisions remain enterprise decisions. This appendix helps buyers ask better questions; it does not replace accountable human judgment.

Appendix I — Source Audit and Revalidation Register

Version: v0.3.2-FRC-R3

I.0 Source Audit Boundary

This appendix provides a source audit register for the completed Phase 1D-11 source closure record and public research edition handoff. It tracks source classes, citation status, evidence level, claim-binding status, and remaining non-blocking limitations.

This appendix was created before Phase 1C as a revalidation workflow. Phase 1C cleanup updated the citation inventory and source coverage matrix with URL/entry-point review status. Phase 1C follow-up then added the claim evidence register and reviewed high-risk product/protocol claims at claim level. Appendix G is now revalidated provisional, not final.

Source audit work in this appendix is procedural. Phase 1D-11 closed high-risk citation/access/source-pointer blockers by replacing inaccessible sources, narrowing claims, downgrading lifecycle interpretations to **L5**, and preserving explicit limitations.

I.1 Source Classes and Evidence Levels

The source audit register uses the canonical evidence levels:

- **L1 (Binding Legal Text):** 1.00
- **L2 (Product Docs):** 0.85
- **L3 (Audit Report):** 0.75
- **L4 (Vendor Claim):** 0.55
- **L5 (Author Inference):** 0.35

Evidence levels describe source confidence for analytical scoring. They do not make a source legally sufficient, current, complete, or operationally validated. Evidence levels may change if source type, source currency, or claim support changes during future source review.

I.2 Source Audit Register

Table T-I-01: Source Audit Register

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

EU-AI-ACT	
Source Name	EU AI Act articles referenced in baseline chapters
Source Class	Legal / regulatory text
Evidence Level	L1
Used In	Chapters 4, 7, 8, Appendices E/F
Current Status	Referenced; citation rendering pending global check
Publication Follow-up Action	Verify article references and citation formatting

GDPR	
Source Name	GDPR articles referenced in privacy and data governance sections
Source Class	Legal / regulatory text
Evidence Level	L1
Used In	Chapters 4, 6, 8, Appendices E/F
Current Status	Referenced; citation rendering pending global check
Publication Follow-up Action	Verify article references and privacy mapping language

NIST-AI-RMF	
Source Name	NIST AI RMF 1.0
Source Class	Formal standard
Evidence Level	L1
Used In	Chapter 4, RCCS security/risk mapping
Current Status	Referenced; final citation check pending
Publication Follow-up Action	Verify source coverage and citation formatting

ISO-IEC-42001

Source Name	ISO/IEC 42001
Source Class	Formal standard
Evidence Level	L1
Used In	Chapter 4, management system baseline
Current Status	Referenced; final citation check pending
Publication Follow-up Action	Verify source coverage and citation formatting

IMDA-AGENTIC-AI

Source Name	Singapore IMDA Model AI Governance Framework for Agentic AI
Source Class	Formal governance framework
Evidence Level	L1
Used In	Chapter 4 and comparative baseline
Current Status	Referenced; final citation check pending
Publication Follow-up Action	Verify citation details and framework naming

W3C-PROV

Source Name	W3C PROV
Source Class	Protocol / provenance standard
Evidence Level	L1
Used In	Chapter 4, evidence/provenance framing
Current Status	Referenced; final citation check pending
Publication Follow-up Action	Verify citation details and relevance language

W3C-VC

Source Name	W3C Verifiable Credentials
Source Class	Protocol / evidence standard
Evidence Level	L1
Used In	Chapter 4, evidence framing
Current Status	Referenced; final citation check pending
Publication Follow-up Action	Verify citation details and relevance language

MPLP-DOCS

Source Name	MPLP documentation
Source Class	Protocol documentation
Evidence Level	L2 / L5
Used In	Chapters 12, 13, Appendix G, claim register
Current Status	Official docs entry point located; mapping remains analytical
Publication Follow-up Action	Keep exact URL pinned; do not treat mapping as implementation proof

MPLP-SPEC

Source Name	MPLP protocol specification
Source Class	Protocol documentation
Evidence Level	L2 / L5
Used In	Chapters 12, 13, Appendix G, claim register
Current Status	Source-bound through MPLP docs entry point
Publication Follow-up Action	Validate version/citation rendering during public research edition QA

VALIDATION-LAB

Source Name	Validation Lab methodology / public surface
Source Class	Validation methodology source
Evidence Level	L2 / L5
Used In	Chapter 14, Appendix K, claim register
Current Status	Methodology page located and narrowed to public-surface non-certifying support
Publication Follow-up Action	Keep non-certifying boundary; do not treat as audit opinion, endorsement, or legal compliance proof

IBM-WATSONX-GOV

Source Name	IBM watsonx.governance documentation
Source Class	Product documentation
Evidence Level	L2 / L5
Used In	Chapters 11, 12, Appendix G, claim register
Current Status	Official entry points located; ALCS mappings remain L5 where not directly documented
Publication Follow-up Action	Pin exact docs pages for final citation

MS-AZURE-FOUNDRY

Source Name	Microsoft Azure AI Foundry documentation
Source Class	Product documentation
Evidence Level	L2 / L5
Used In	Chapters 11, 12, Appendix G, claim register
Current Status	Official entry point located; ALCS mappings remain L5 where not directly documented
Publication Follow-up Action	Pin exact Microsoft Learn pages for final citation

AWS-BEDROCK-STACK

Source Name	AWS Bedrock / Guardrails / AgentCore documentation
Source Class	Product documentation
Evidence Level	L2 / L5
Used In	Chapters 11, 12, Appendix G, claim register
Current Status	Official entry points located; ALCS mappings remain L5 where not directly documented
Publication Follow-up Action	Pin exact AWS pages for final citation

GOOGLE-VERTEX-STACK

Source Name	Google Vertex AI / ADK / Model Armor documentation
Source Class	Product documentation
Evidence Level	L2 / L5
Used In	Chapters 11, 12, Appendix G, claim register
Current Status	Official entry points located; ALCS mappings remain L5 where not directly documented
Publication Follow-up Action	Pin exact Google pages for final citation

LANGGRAPH-LANGSMITH

Source Name	LangGraph / LangSmith documentation
Source Class	Product documentation / repository documentation
Evidence Level	L2 / L5
Used In	Chapters 11, 12, Appendix G, claim register
Current Status	Official entry points located; ALCS mappings remain L5 where not directly documented
Publication Follow-up Action	Pin exact LangChain docs pages for final citation

OPENAI-AGENTS-SDK

Source Name	OpenAI Agents SDK documentation
Source Class	Product documentation
Evidence Level	L2 / L5
Used In	Chapters 11, 12, Appendix G, claim register
Current Status	Accessible official Python/JS SDK docs located; platform guide HTTP 403 retained as access note
Publication Follow-up Action	Use accessible official SDK docs for SDK-surface claims only

CREWAI-DOCS

Source Name	CrewAI documentation and repository references
Source Class	Product documentation / repository documentation
Evidence Level	L2 / L5
Used In	Chapters 11, 12, Appendix G, claim register
Current Status	Official docs and repository entry points located; ALCS mappings remain L5 where not directly documented
Publication Follow-up Action	Pin exact CrewAI docs pages for final citation

CLAIM-EVIDENCE-REGISTER

Source Name	Claim evidence register
Source Class	Internal governance ledger
Evidence Level	L5
Used In	Phase 1C follow-up
Current Status	PHASE 1C CLAIM-LEVEL REVIEW COMPLETE — PROVISIONAL SOURCE BINDING
Publication Follow-up Action	Maintain as source-to-claim audit ledger

CITATION-INVENTORY

Source Name	Citation inventory ledger
Source Class	Internal governance ledger
Evidence Level	L5
Used In	Appendix C, Appendix I
Current Status	PHASE 1D-11 CITATION SOURCE CLOSURE COMPLETE — HIGH-RISK BLOCKERS CLOSED
Publication Follow-up Action	Maintain closure ledger and explicit limitations

SOURCE-COVERAGE-MATRIX

Source Name	Source coverage matrix
Source Class	Internal governance ledger
Evidence Level	L5
Used In	Appendix I, Phase 1C carryover
Current Status	PHASE 1C CLAIM-LEVEL REVIEWED — PROVISIONAL SOURCE BINDING COMPLETE
Publication Follow-up Action	Citation rendering and page-level source pinning remain part of public research edition maintenance

I.3 Source Review and Publication Handoff Workflow

Source review and publication handoff should follow a controlled workflow:

1. Freeze the active Markdown source set for validation.
2. Confirm the citation inventory status and source coverage matrix status.
3. Verify each L1 legal or standards citation for citation rendering and relevance.
4. Revalidate every L2 product documentation source for currency, scope, and claim support.
5. Confirm MPLP documentation, Validation Lab methodology, and OpenAI SDK replacement source status.
6. Reclassify evidence levels only when documentation supports the change.
7. Record all source changes, narrowed claims, L5 downgrades, and explicit limitations in Appendix I or a publication QA report.

Phase 1D-11 completed the high-risk source closure path. It did not convert L2 documentation into audit proof, legal compliance proof, certification evidence, regulatory approval, procurement recommendation, or final vendor assessment evidence.

I.4 URL and Citation Verification Queue

Table T-I-02: URL Verification Queue

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

MPLP-DOCS	
URL / Source Pointer	https://docs.mplp.io/docs/introduction/mplp-v1.0-protocol-overview
Used In	Chapters 12, 13, Appendix G
Verification Need	Official docs accessible; protocol identity source-bound
Priority	Non-blocking
Notes	MRO/ALCS mapping remains L5 analytical where not directly stated

MPLP-SPEC	
URL / Source Pointer	https://docs.mplp.io/docs/introduction/mplp-v1.0-protocol-overview
Used In	Chapters 12, 13, Appendix G
Verification Need	Bound through protocol overview
Priority	Non-blocking
Notes	Do not treat protocol mapping as implementation proof

VALIDATION-LAB	
URL / Source Pointer	https://lab.mplp.io/methodology
Used In	Chapter 14, Appendix K
Verification Need	Methodology page accessible; narrowed to public-surface non-certifying example
Priority	Non-blocking
Notes	No certification, endorsement, audit opinion, or compliance proof

IBM-WATSONX-GOV	
URL / Source Pointer	IBM official entry points
Used In	Chapters 11, 12, Appendix G
Verification Need	Accessible; claims narrowed to capability surfaces
Priority	Non-blocking
Notes	ALCS mapping remains L5 where not directly documented

MS-AZURE-FOUNDRY

URL / Source Pointer	https://learn.microsoft.com/en-us/azure/foundry/
Used In	Chapters 11, 12, Appendix G
Verification Need	Accessible; claims narrowed to capability surfaces
Priority	Non-blocking
Notes	ALCS mapping remains L5 where not directly documented

AWS-BEDROCK-STACK

URL / Source Pointer	AWS official Bedrock / Guardrails / AgentCore docs
Used In	Chapters 11, 12, Appendix G
Verification Need	Accessible; claims narrowed to capability surfaces
Priority	Non-blocking
Notes	Guardrails/runtime features are not delegated-authority proof

GOOGLE-VERTEX-STACK

URL / Source Pointer	Google official Vertex / ADK / Model Armor docs
Used In	Chapters 11, 12, Appendix G
Verification Need	Accessible; claims narrowed to capability surfaces
Priority	Non-blocking
Notes	Safety/agent primitives are not lifecycle responsibility proof

LANGGRAPH-LANGSMITH

URL / Source Pointer	LangChain official docs
Used In	Chapters 11, 12, Appendix G
Verification Need	Accessible; claims narrowed to orchestration/observability surfaces
Priority	Non-blocking
Notes	Lifecycle governance requires implementer integration

OPENAI-AGENTS-SDK

URL / Source Pointer	https://openai.github.io/openai-agents-python/ ; https://openai.github.io/openai-agents-js/
Used In	Chapters 11, 12, Appendix G
Verification Need	Accessible official SDK docs replace platform guide 403 for retained SDK claims
Priority	Non-blocking
Notes	Platform guide HTTP 403 retained as access note only

CREWAI-DOCS

URL / Source Pointer	https://docs.crewai.com/ ; https://github.com/crewAIInc/crewAI
Used In	Chapters 11, 12, Appendix G
Verification Need	Accessible; claims narrowed to orchestration surfaces
Priority	Non-blocking
Notes	Accountability/lifecycle governance remains implementer-built

L1 -LEGAL-STANDARDS

URL / Source Pointer	Citation source pointers / reference notes
Used In	Chapters 4, 7, 8, Appendices E/F
Verification Need	Verify citation rendering and article/standard references
Priority	Medium
Notes	Does not change legal interpretation

FIGURE-TABLE-REFERENCES

URL / Source Pointer	Internal references
Used In	Entire source
Verification Need	Verify table and figure references render correctly
Priority	Medium
Notes	Coordinate with Phase 1D layout QA

Phase 1D-11 located, validated, replaced, or narrowed the high-priority L2 groups. Final human proofing may refine citations, but no high-risk source-pointer remains unresolved.

I.5 Evidence Level Reclassification Rules

Table T-I-03: Evidence Level Reclassification Rules

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

L5 author inference	
Trigger for Reclassification	Official product documentation directly supports the claim
New Level	L2
Required Documentation	Source URL, publication/access date, quoted claim summary, mapped claim location
Reviewer Note	Do not reclassify based on assumption

L4 vendor claim	
Trigger for Reclassification	Claim is found in official technical documentation with implementation detail
New Level	L2
Required Documentation	Official documentation reference and implementation detail
Reviewer Note	Marketing-only material remains L4

L2 product documentation	
Trigger for Reclassification	Documentation no longer exists, changed materially, or no longer supports the claim
New Level	L4 or L5
Required Documentation	Change note, archived pointer if available, unsupported claim record
Reviewer Note	Downgrade evidence rather than preserving stale confidence

L2 product documentation	
Trigger for Reclassification	Third-party audit report validates implementation detail
New Level	L3
Required Documentation	Audit report reference, scope, date, and control mapping
Reviewer Note	Audit scope must match claim scope

L3 audit report	
Trigger for Reclassification	Audit scope does not cover the claim or is not accessible
New Level	L4 or L5
Required Documentation	Gap explanation and reviewer note
Reviewer Note	Do not infer coverage from report existence

L1 legal text	
Trigger for Reclassification	Legal text establishes regulatory baseline but not system implementation
New Level	L1 for baseline only
Required Documentation	Citation and claim boundary note
Reviewer Note	Do not use legal text alone as implementation evidence

Any level	
Trigger for Reclassification	Source is unverifiable during source review
New Level	Lower confidence level or unresolved
Required Documentation	Verification attempt log and unresolved issue record
Reviewer Note	Preserve uncertainty explicitly

I.6 Source Coverage Matrix Validation Note

Table T-I-04: Source Coverage Matrix Validation Checklist

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Legal baseline claims	
Expected Source Type	L1 legal / regulatory text
Current Status	Referenced in baseline chapters
Gap	Citation rendering and article mapping not globally validated
Publication Follow-up Action	Verify all legal citations and claim boundaries

Standards baseline claims

Expected Source Type	L1 formal standards
Current Status	Referenced in baseline chapters
Gap	Citation rendering and standard references not globally validated
Publication Follow-up Action	Verify all standards citations

Product capability claims

Expected Source Type	L2 product documentation plus L5 mapping where needed
Current Status	Phase 1D-11 closure applied
Gap	No high-risk source-pointer remains unresolved
Publication Follow-up Action	Preserve narrowed claims and L5 downgrades during publication QA

MPLP protocol mapping

Expected Source Type	L2 protocol docs plus disclosed author relationship
Current Status	Official docs entry point located; mapping remains L2 + L5
Gap	Version/citation rendering still requires final proofing
Publication Follow-up Action	Preserve conflict-of-interest and implementation-proof boundary

Validation Lab discussion

Expected Source Type	L2 public methodology page plus L5 methodology interpretation
Current Status	Phase 1D-11 closure applied
Gap	No high-risk source-pointer remains unresolved
Publication Follow-up Action	Keep non-certifying boundary and do not promote methodology claims

Appendix G provisional assessments

Expected Source Type	L2 product docs and L5 inference
Current Status	Revalidated provisional
Gap	Final assessment remains non-final and non-ranking
Publication Follow-up Action	Preserve no-score/no-ranking posture and page-level citation needs

Citation inventory

Expected Source Type	Internal ledger
Current Status	PHASE 1D-11 CITATION SOURCE CLOSURE COMPLETE — HIGH-RISK BLOCKERS CLOSED
Gap	Remaining limitations explicit
Publication Follow-up Action	Preserve closure ledger through final proof

Source coverage matrix

Expected Source Type	Internal ledger
Current Status	PHASE 1D-11 CITATION SOURCE CLOSURE COMPLETE — HIGH-RISK BLOCKERS CLOSED
Gap	Claim-label preservation verified in generated artifact
Publication Follow-up Action	Preserve source labels during final proof

The source coverage matrix has been updated through Phase 1D-11 as a coverage and closure ledger. Appendix G remains revalidated provisional, not final, and no score is a vendor ranking, procurement recommendation, certification, or legal compliance proof.

Appendix J — v0.3.2 Gap Closure Coverage Matrix

Version: v0.3.2-FRC-R3

J.0 Gap Closure Boundary

This appendix provides a governance closure matrix showing how known v0.3.1/v0.3.2 reconstruction gaps were addressed, structurally closed, corrected by later patch, historically deferred to Phase 1C, or deferred to Phase 1D publication QA.

This appendix does not claim that all issues are fully resolved. It distinguishes content closure from source validation, editorial cleanup, and publication layout QA.

Closure status values:

- **Closed**
- **Structurally Closed**
- **Corrected by Later Patch**
- **Formerly Deferred to Phase 1C — Closed / Superseded by Phase 1C follow-up**
- **Deferred to Phase 1D**
- **Candidate Issue to Verify**

J.1 Gap Categories

Gap categories used in this appendix:

1. **Content reconstruction gaps:** Missing chapters, appendices, tables, or required sections.
2. **Methodology consistency gaps:** Evidence multipliers, scoring formulas, score-5 wording, and composite weights.
3. **Boundary discipline gaps:** Overclaim risk, vendor ordering, certification implication, legal compliance implication, and procurement misuse.
4. **Source governance gaps:** Citation inventory status, L2 documentation revalidation, evidence level annotation, URL validation, and source coverage matrix validation.
5. **Layout and publication gaps:** Wide-table handling, figure numbering, captions, page breaks, and final PDF QA.

J.2 Phase 1B Gap Closure Matrix

Table T-J-01: v0.3.2 Gap Closure Coverage Matrix

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

G-J-01	
Gap Description	R2 PDF reconstruction introduced layout and extraction instability
Origin Phase	R2 carryover
Closure Status	Structurally Closed
Closure Evidence	Markdown-first reconstruction strategy adopted in Phase 1A
Remaining Action	Phase 1D publication QA remains

G-J-02	
Gap Description	Body chapters 1-18 incomplete in clean source
Origin Phase	Phase 1A/1B
Closure Status	Closed
Closure Evidence	Chapters 1-18 filled across Phase 1A and Phase 1B-1 through 1B-7
Remaining Action	Phase 1C editorial consistency check completed; final proof remains

G-J-03	
Gap Description	Appendices A-D missing from clean source
Origin Phase	Phase 1B-8
Closure Status	Corrected by Later Patch
Closure Evidence	Appendices A-D filled and corrected by Phase 1B-8C
Remaining Action	Phase 1C source review completed; Phase 1D layout QA remains

G-J-04	
Gap Description	Appendix B evidence multipliers and composite weights inconsistent with canonical method
Origin Phase	Phase 1B-8
Closure Status	Corrected by Later Patch
Closure Evidence	Phase 1B-8C methodology correction
Remaining Action	Verified in Phase 1C global consistency pass; final proof remains

G-J-05

Gap Description	Appendix A compliance-sounding field values risked overclaim
Origin Phase	Phase 1B-8
Closure Status	Corrected by Later Patch
Closure Evidence	Phase 1B-8C neutralized field values
Remaining Action	Phase 1C template review completed; final proof remains

G-J-06

Gap Description	Appendix D vendor mappings needed provisional language
Origin Phase	Phase 1B-8
Closure Status	Corrected by Later Patch
Closure Evidence	Phase 1B-8C provisional mapping language; Phase 1D-11 source closure narrowed high-risk source pointers
Remaining Action	Preserve LS analytical labels and explicit limitations

G-J-07

Gap Description	Appendices E/F/G initially needed detailed rubrics and system assessments
Origin Phase	Phase 1B-9
Closure Status	Structurally Closed
Closure Evidence	Appendices E-G filled; E/F rubrics and G assessments created
Remaining Action	Phase 1C source validation completed; final proof remains

G-J-08

Gap Description	Appendix G scope drift and score/order overreach
Origin Phase	Phase 1B-9C
Closure Status	Corrected by Later Patch
Closure Evidence	Appendix G corrected to 8 representative systems plus T-G-09 qualitative summary; Phase 1D-11 closed high-risk source pointers
Remaining Action	Preserve non-ranking, provisional, source-qualified posture

G-J-09

Gap Description	Appendix E/F score-5 wording riskd implying legal text proves implementation
Origin Phase	Phase 1B-9C
Closure Status	Corrected by Later Patch
Closure Evidence	Appendix E/F score-5 rows hardened
Remaining Action	Verified in Phase 1C consistency pass; final proof remains

G-J-10

Gap Description	Phase 1B-9D governance ledgers stale after partial patch
Origin Phase	Phase 1B-9D
Closure Status	Corrected by Later Patch
Closure Evidence	SOURCE MASTER, Appendix G metadata, carryover list, and final report synchronized
Remaining Action	Continue with Phase 1B-10 completion record

G-J-11

Gap Description	Appendices H-K missing from clean source
Origin Phase	Phase 1B-10
Closure Status	Closed
Closure Evidence	Appendices H-K filled in Phase 1B-10
Remaining Action	Phase 1C source validation completed; Phase 1D layout QA remains

G-J-12

Gap Description	Citation Inventory remained partial after Phase 1B
Origin Phase	Phase 1B-3
Closure Status	Corrected by Later Patch
Closure Evidence	Phase 1D-11 citation/source closure ledger records source-bound, narrowed, replaced, and L5 -downgraded claims
Remaining Action	Preserve explicit limitations during public research edition maintenance

G-J-15

Gap Description	Claim-level L2 product/protocol source binding remained pending after Phase 1C cleanup
Origin Phase	Phase 1C follow-up
Closure Status	Corrected by Later Patch
Closure Evidence	Phase 1D-11 closed high-risk L2 source-pointer blockers; Appendix G remains revalidated provisional
Remaining Action	Keep Appendix G provisional; no final scores/rankings

G-J-13

Gap Description	Wide tables require layout evaluation
Origin Phase	R2 / Phase 1B
Closure Status	Corrected by Later Patch
Closure Evidence	Phase 1C recorded layout risk notes in table inventory
Remaining Action	Render and inspect tables in Phase 1D

G-J-14

Gap Description	PDF layout, figures, captions, and page breaks not regenerated
Origin Phase	Phase 1D
Closure Status	Deferred to Phase 1D
Closure Evidence	PDF generation intentionally deferred
Remaining Action	Generate and visually QA in Phase 1D

J.3 Phase 1C Items and Current Disposition

Table T-J-02: Phase 1C Items and Current Disposition

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Chapter 14/15 ALCS numbering consistency

Reason Deferred	Phase 1C taxonomy pass found naming and scenario mapping drift
Required Action	Corrected active-source ALCS/MRO references
Blocking Level	Closed for taxonomy; monitor in final proof
Owner Phase	Phase 1C

Chapter 17 roadmap-count inconsistency

Reason Deferred	Body chapter wording did not match Stage 0-to-7 table
Required Action	Corrected to Stage 0-to-7 wording
Blocking Level	Closed for editorial consistency
Owner Phase	Phase 1C

Chapter 18 contribution-count inconsistency

Reason Deferred	Contribution summary did not match T-18-01 row count
Required Action	Corrected to eight contributions
Blocking Level	Closed for editorial consistency
Owner Phase	Phase 1C

F-09 figure numbering gap

Reason Deferred	Figure inventory skipped F-09
Required Action	Reserved F-09 to avoid renumbering risk
Blocking Level	Closed for inventory; Phase 1D rendering decision remains
Owner Phase	Phase 1C / 1D

Citation inventory pre-cleanup status

Reason Deferred	Citation validation belongs to global cleanup
Required Action	Updated citation inventory to Phase 1D-11 citation/source closure complete
Blocking Level	Remaining limitations explicit and non-blocking
Owner Phase	Phase 1D-11

All L2 product docs required Phase 1C revalidation

Reason Deferred	Product docs may have changed and required claim-level review
Required Action	Official entry points located; material claims source-bound, narrowed, replaced, or downgraded to L5
Blocking Level	Appendix G remains provisional; no final scores/rankings
Owner Phase	Phase 1D-11

T-D-01 required layout evaluation

Reason Deferred	Very wide table may require split or landscape handling
Required Action	Layout risk documented in table inventory
Blocking Level	Render and inspect in Phase 1D
Owner Phase	Phase 1D

Wide table evaluation for T-13/T-14/T-15/T-17/T-18 and Appendix G

Reason Deferred	Several wide tables may not render in portrait layout
Required Action	Layout risk documented in table inventory
Blocking Level	Render and inspect in Phase 1D
Owner Phase	Phase 1D

MPLP/Validation Lab URL validation

Reason Deferred	URLs were not validated in Phase 1B
Required Action	MPLP docs/site and Validation Lab methodology page validated; mapping remains bounded
Blocking Level	Validation Lab supports public-surface non-certifying example only
Owner Phase	Phase 1D-11

Evidence level annotation

Reason Deferred	Evidence confidence must be visible and consistent
Required Action	Claim evidence register added L1 - L5 labels and downgrade dispositions
Blocking Level	Verify citations render with evidence labels in Phase 1D
Owner Phase	Phase 1D

Source coverage matrix validation

Reason Deferred	Coverage matrix is a governance ledger requiring global validation
Required Action	Matrix expanded and updated to claim-level source-binding status
Blocking Level	Maintain claim register during page-level citation pinning
Owner Phase	Phase 1C follow-up / Phase 1D

Appendix G provisional assessments required source revalidation before finalization

Reason Deferred	Appendix G uses qualitative provisional posture
Required Action	Appendix G marked revalidated provisional; material claims source-bound, narrowed, or downgraded
Blocking Level	Keep provisional until publication approval; no final scores/rankings
Owner Phase	Phase 1C follow-up

J.4 Deferred Phase 1D Items

Table T-J-03: Phase 1D Layout / Publication Items

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

PDF generation

Publication Risk	Formatting may drift from Markdown source
Required QA	Generate PDF from reviewed source
Expected Output	Publication candidate PDF
Owner Phase	Phase 1D

PDF generation

Publication Risk	Tables or figures may clip or wrap poorly
Required QA	Generate PDF from finalized source
Expected Output	Publication candidate PDF
Owner Phase	Phase 1D

Wide table rendering

Publication Risk	Landscape tables may be unreadable or clipped
Required QA	Inspect T-D-01, T-13/T-14/T-15/T-17/T-18, Appendix G, H-K tables
Expected Output	Wide-table remediation record
Owner Phase	Phase 1D

Figure readability

Publication Risk	Figure panels or generated figures may not be publication grade
Required QA	Verify figure resolution, labels, captions, and references
Expected Output	Figure QA record
Owner Phase	Phase 1D

Table captions and page breaks

Publication Risk	Captions and headers may split incorrectly
Required QA	Inspect page breaks, captions, repeated headers
Expected Output	Layout QA record
Owner Phase	Phase 1D

Citation rendering

Publication Risk	Citations may not render consistently in PDF
Required QA	Verify citation formatting after generation
Expected Output	Citation rendering QA record
Owner Phase	Phase 1D

J.5 Closure Interpretation Boundary

Closed and structurally closed items indicate that the Markdown source now contains the required content or correction. They do not mean that source revalidation, legal review, procurement review, certification, publication layout, or final rendering has been completed.

Phase 1D-11 citation/source closure has been completed for high-risk blockers. Remaining work is public research edition maintenance and proofing, with explicit non-blocking limitations preserved.

Appendix K — Editorial Remediation Record

Version: v0.3.2-FRC-R3

K.0 Editorial Remediation Boundary

This appendix records governance-relevant remediation decisions across Phase 1A and Phase 1B. It is not a line-by-line changelog. It preserves audit traceability for reconstruction strategy, methodology correction, boundary hardening, source governance, and deferred cleanup.

The record does not claim certification, legal compliance proof, regulator approval, procurement guidance, or vendor ranking.

K.1 Remediation Categories

Remediation categories:

1. **Structural reconstruction:** Creating clean Markdown source, filling chapters, and replacing unstable PDF extraction behavior.
 2. **Methodology correction:** Aligning evidence multipliers, scoring formulas, score-5 interpretation, and composite weighting.
 3. **Boundary hardening:** Preventing legal, procurement, certification, vendor ordering, or compliance-proof overclaim.
 4. **Source governance:** Preserving citation inventory status, L2 revalidation requirements, evidence level annotation, and source coverage validation.
 5. **Publication deferral:** Reserving PDF generation and layout QA for Phase 1D.
-

K.2 Phase 1A Remediation Record

Phase 1A established the clean source foundation:

- Rebuilt the document as Markdown-first source.
 - Filled core body chapters and methodology chapters.
 - Established RCCS, ALCS, evidence multiplier, and composite scoring discipline.
 - Preserved non-legal and non-certifying boundaries.
 - Deferred PDF generation until after source completion and cleanup.
-

K.3 Phase 1B Remediation Record

Table T-K-01: Editorial Remediation Record

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

R-K-01	
Phase	Phase 1A
Issue	R2 extraction and layout instability
Remediation	Adopted Markdown-first reconstruction strategy
Evidence / File Area	Reconstruction notes; SOURCE MASTER
Status	Closed

R-K-02	
Phase	Phase 1A
Issue	Core methodology needed stable scoring discipline
Remediation	Established canonical evidence multipliers and scoring formulas
Evidence / File Area	Chapters 7-9; Appendix B
Status	Closed

R-K-03	
Phase	Phase 1B-1 to 1B-7
Issue	Body chapters 11-15 and 17-18 missing from clean source
Remediation	Filled comparative field, mappings, validation pattern, failure scenarios, roadmap, and conclusion
Evidence / File Area	Sections 11-15, 17-18
Status	Closed

R-K-04	
Phase	Phase 1B-8
Issue	Appendices A-D missing from clean source
Remediation	Created templates, scorecards, source notes, and expanded MRO mapping
Evidence / File Area	Appendices A-D
Status	Closed

R-K-05

Phase	Phase 1B-8C
Issue	Appendix B methodology values inconsistent with canonical method
Remediation	Corrected evidence multipliers, adjusted score formula, composite weights, and Article 12 example
Evidence / File Area	Appendix B; Phase 1B-8C report
Status	Corrected by later patch

R-K-06

Phase	Phase 1B-8C
Issue	Appendix A and D contained overclaim risk
Remediation	Neutralized compliance-sounding template values and marked mappings provisional
Evidence / File Area	Appendices A and D
Status	Corrected by later patch

R-K-07

Phase	Phase 1B-9
Issue	Appendices E-G missing detailed rubric and assessment content
Remediation	Created RCCS rubrics, ALCS rubrics, and provisional system assessments
Evidence / File Area	Appendices E-G
Status	Structurally closed

R-K-08

Phase	Phase 1B-9C
Issue	Appendix G scope drift and score/order overreach
Remediation	Rewrote Appendix G using authorized 8 systems and qualitative provisional posture
Evidence / File Area	Appendix G; 1B-9C report
Status	Corrected by later patch

R-K-09

Phase	Phase 1B-9C
Issue	Appendix E/F score-5 wording risk implementation overclaim
Remediation	Hardened score-5 rows to require implementation evidence
Evidence / File Area	Appendices E/F
Status	Corrected by later patch

R-K-10

Phase	Phase 1B-9D
Issue	Governance ledgers stale after partial 1B-9D patch
Remediation	Synced Appendix G metadata, SOURCE MASTER, carryover list, and report
Evidence / File Area	SOURCE MASTER; 1B-9D report
Status	Closed

R-K-11

Phase	Phase 1B-10
Issue	Appendices H-K missing
Remediation	Filled procurement scorecard, source audit register, gap closure matrix, and editorial remediation record
Evidence / File Area	Appendices H-K
Status	Closed

R-K-12

Phase	Phase 1B-10
Issue	Phase 1C / 1D obligations needed final pre-cleanup ledger
Remediation	Preserved canonical carryover and publication deferrals
Evidence / File Area	Appendices I/J/K; completion report
Status	Closed

R-K-13

Phase	Phase 1C
Issue	Taxonomy and editorial drift remained after Phase 1B
Remediation	Corrected ALCS/MRO names, Stage 0-to-7 wording, eight-contribution wording, and F-09 reservation
Evidence / File Area	Chapters 14/15/17/18; figure inventory
Status	Closed for editorial cleanup

R-K-14

Phase	Phase 1C
Issue	Citation and coverage ledgers remained pre-cleanup
Remediation	Updated citation inventory and source coverage matrix with URL/entry-point review and unresolved claim-level validation queue
Evidence / File Area	Citation inventory; source coverage matrix
Status	Corrected by Phase 1C follow-up

R-K-15

Phase	Phase 1C
Issue	Wide-table risks were not fully documented for Phase 1D
Remediation	Added layout risk notes for requested wide-table set and H-K tables
Evidence / File Area	Table inventory
Status	Closed for layout-readiness audit

R-K-16

Phase	Phase 1C follow-up / Phase 1D-11
Issue	Claim-level L2 product/protocol source binding remained pending
Remediation	Created claim evidence register; Phase 1D-11 closed high-risk source pointers by official source replacement, claim narrowing, L5 downgrade, and explicit limitations
Evidence / File Area	Claim evidence register; citation inventory; source coverage matrix; Appendix G
Status	Closed for public research edition maintenance; Appendix G remains provisional

K.4 Boundary and Non-Claim Remediation Record

Table T-K-02: Boundary Hardening Record

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

RCCS-T/RCCS-M/ALCS non-legal-score boundary

Risk	Readers may treat analytical scores as legal determinations
Remediation Applied	Repeated boundary language across chapters and appendices; RCCS-M marked as author analytical and forward-looking
Remaining Check	Phase 1D-13B consistency review completed; verify in final proof
Phase	Phase 1A / 1B / 1D-13B

MPLP non-required / non-exclusive boundary

Risk	MPLP could be misread as required or sufficient by itself
Remediation Applied	Disclosed author relationship and framed MPLP as one protocol path
Remaining Check	Phase 1C wording review completed; verify in final proof
Phase	Phase 1B-4

Validation Lab non-certifying boundary

Risk	Validation Lab could be misread as an approval body
Remediation Applied	Framed as a non-certifying evidence adjudication example
Remaining Check	URL and boundary language reviewed at entry-point level; verify in final proof
Phase	Phase 1B-5

Appendix G no-score/no-ranking boundary

Risk	Provisional assessment could be misused as vendor ordering
Remediation Applied	Removed scores, ordered comparisons, and selection language; Phase 1C follow-up preserved revalidated provisional status
Remaining Check	Verify no score/ranking language in final proof
Phase	Phase 1B-9C / 1B-9D / 1C

Appendix E/F legal-text boundary

Risk	Legal text could be misread as implementation evidence
Remediation Applied	Hardened score-5 rows and top-level scale language
Remaining Check	Formula and rubric consistency reviewed; verify in final proof
Phase	Phase 1B-9C / 1B-9D

Appendix H procurement boundary

Risk	Buyer checklist could be misused as vendor selection advice
Remediation Applied	Added procurement-use boundary and required human review
Remaining Check	Verify no vendor ordering language in final proof
Phase	Phase 1B-10

Appendix I source audit boundary

Risk	Revalidation register could be misread as completed validation
Remediation Applied	Stated that source closure supports public research edition maintenance but not legal, audit, certification, or procurement conclusions
Remaining Check	Preserve explicit limitations during final proofing
Phase	Phase 1B-10 / Phase 1C / Phase 1D-11

Appendix J closure boundary

Risk	Gap closure could be misread as final publication readiness
Remediation Applied	Distinguished closed, structurally closed, completed Phase 1C follow-up, and Phase 1D/publication QA items
Remaining Check	Complete remaining publication QA work
Phase	Phase 1B-10 / Wave 7

K.5 Remaining Publication QA Actions

Table T-K-03: Remaining Publication QA Actions

System mapping rendered as profile cards to preserve long evidence and boundary text; all source rows are retained without rank ordering.

Source-bound product/protocol claim proofing

Reason	Phase 1D-11 closed high-risk blockers, but final human proofing should preserve narrowed wording
Target Phase	Publication-candidate review
Required Evidence	Closure ledger and rendered References / Sources section
Blocking Level	Non-blocking limitation

Validation Lab methodology-specific citation

Reason	Methodology page is pinned; support remains public-surface and non-certifying
Target Phase	Publication-candidate review
Required Evidence	Methodology URL plus boundary note
Blocking Level	Non-blocking limitation

Evidence level source-to-claim rendering check

Reason	Evidence confidence is tied to claims and rendered in artifacts
Target Phase	Publication-candidate review
Required Evidence	Citation/evidence rendering QA record
Blocking Level	Non-blocking limitation

Appendix G revalidated provisional proofread

Reason	Provisional postures now have source binding but remain non-final
Target Phase	Publication-candidate review
Required Evidence	Final no-score/no-ranking grep and citation check
Blocking Level	Non-blocking limitation

PDF generation and visual QA

Reason	HTML/PDF derivative generated through Phase 1D-11
Target Phase	Publication-candidate review
Required Evidence	HTML/PDF artifacts and visual QA report
Blocking Level	Non-blocking limitation

References / Sources

This section is a publication-facing source register. It summarizes source classes, evidence levels, and citation status. It does not convert product documentation into audit proof, legal compliance proof, certification, regulatory approval, procurement recommendation, or vendor endorsement.

How to Read Evidence Labels

CLOSED BY OFFICIAL SOURCE 0 entries are source-bound to official accessible sources.	REPLACED SOURCE 0 entries use an accessible official replacement for an access-limited pointer.
NARROWED CLAIM 0 entries support capability surfaces but not lifecycle-compliance proof.	DOWNGRADED TO L5 0 entries remain explicit author analytical mapping.
EXPLICIT LIMITATION 0 entries carry non-blocking release limitations.	UNRESOLVED BLOCKER 0 high-risk citation blockers remain after Phase 1D-12 closure.

Evidence labels express source confidence and claim-support posture. They do not prove legal compliance, certification, deployment readiness, product quality, or procurement suitability.

Evidence Level Counts

L1 9	L2 2	L2/L4 depending on claim 1	L2/L5 13	L2/L5 depending on claim scope 1
L4/L5 depending on claim 1				

Legal / Regulatory

EU-AI-ACT-2024-1689 Regulation (EU) 2024/1689, Artificial Intelligence Act Legal / regulatory text L1 Official EUR-Lex entry point identified: https://eur-lex.europa.eu/eli/reg/2024/1689/oj ; EC AI Act explorer also identified

GDPR-2016-679

Regulation (EU) 2016/679, General Data Protection Regulation

Legal / regulatory text L1

Official EUR-Lex entry point identified: <https://eur-lex.europa.eu/eli/reg/2016/679/oj>

COLORADO-SB24-205

Colorado SB24-205 Consumer Protections for Artificial Intelligence

Legal / regulatory text L1

Official Colorado General Assembly entry point identified: <https://leg.colorado.gov/bills/sb24-205>

COLORADO-SB25B-004

Colorado SB25B-004 amendment / delay materials

Legal / regulatory text L1

Official Colorado General Assembly bill materials identified

Standards / Frameworks

NIST-AI-RMF-1-0

NIST AI Risk Management Framework 1.0

Formal standard / framework L1

Official NIST publication entry point identified: <https://www.nist.gov/publications/artificial-intelligence-risk-management-framework-ai-rmf-10>

ISO-IEC-42001

ISO/IEC 42001:2023 AI management systems

Formal standard L1

Official ISO entry point identified: <https://www.iso.org/standard/42001>

IMDA-AGENTIC-AI-MGF

Singapore IMDA Model AI Governance Framework for Agentic AI

Formal governance framework L1

Official IMDA PDF entry point identified

W3C-PROV

W3C PROV provenance family

Formal web standard L1

Official W3C PROV overview entry point identified: <https://www.w3.org/TR/prov-overview/>

W3C-VC-DATA-MODEL

W3C Verifiable Credentials Data Model

Formal web standard

L1

Official W3C VC Data Model v2.0 entry point identified: <https://www.w3.org/TR/vc-data-model/>

Protocol / MPLP

MPLP-DOCS

MPLP documentation

Protocol documentation

L2

Official documentation entry point validated: <https://docs.mplp.io/docs/introduction/mplp-v1.0-protocol-overview>

MPLP-SITE

MPLP public positioning site

Protocol documentation / positioning

L4/L5 depending on claim

Public entry point validated: <https://www.mplp.io/>

Validation

VALIDATION-LAB

MPLP Validation Lab public surface

Validation methodology / evidence adjudication source

L2/L5 depending on claim scope

Public methodology page validated: <https://lab.mplp.io/methodology>

Product / Platform Documentation

IBM-WATSONX-GOVERNANCE-DOCS

IBM watsonx.governance documentation / official IBM pages

Product documentation

L2/L5

Official IBM entry points validated, including <https://www.ibm.com/products/watsonx-governance> and <https://www.ibm.com/docs/en/watsonx>

MICROSOFT-AZURE-AI-FOUNDRY-DOCS**Microsoft Azure AI Foundry / Microsoft Foundry documentation**

Product documentation

L2/L5

Official Microsoft Learn entry point validated via current redirect target: <https://learn.microsoft.com/en-us/azure/foundry/>

MICROSOFT-RESPONSIBLE-AI-DOCS**Microsoft Responsible AI documentation**

Product documentation / policy documentation

L2

Official Microsoft documentation source family identified and treated as supporting responsible-AI framing only

AWS-BEDROCK-DOCS**Amazon Bedrock documentation**

Product documentation

L2/L5

Official AWS documentation entry point validated:
<https://docs.aws.amazon.com/bedrock/latest/userguide/what-is-bedrock.html>

AWS-BEDROCK-GUARDRAILS-DOCS**Amazon Bedrock Guardrails documentation**

Product documentation

L2/L5

Official AWS Guardrails entry point validated:
<https://docs.aws.amazon.com/bedrock/latest/userguide/guardrails.html>

AWS-AGENTCORE-DOCS**Amazon Bedrock AgentCore documentation**

Product documentation

L2/L5

Official AWS AgentCore entry point validated: <https://docs.aws.amazon.com/bedrock-agentcore/latest/devguide/what-is-bedrock-agentcore.html>

GOOGLE-VERTEX-AI-DOCS**Google Vertex AI documentation**

Product documentation

L2/L5

Official Google Cloud documentation entry point validated via current redirect target:
<https://docs.cloud.google.com/vertex-ai/docs>

GOOGLE-ADK-DOCS**Google Agent Development Kit / ADK documentation**

Product documentation

L2/L5

Official ADK entry point validated via current redirect target: <https://adk.dev/>

GOOGLE-MODEL-ARMOR-DOCS**Google Model Armor documentation**

Product documentation

L2/L5

Official Google Cloud Model Armor entry point validated via current redirect target:
<https://docs.cloud.google.com/model-armor/overview>

LANGGRAPH-DOCS**LangGraph documentation**

Product / framework documentation

L2/L5

Official LangChain documentation entry point validated:
<https://docs.langchain.com/oss/python/langgraph/overview>

LANGSMITH-DOCS**LangSmith documentation**

Product documentation

L2/L5

Official LangSmith documentation entry point validated: <https://docs.langchain.com/langsmith/home>

OPENAI-AGENTS-SDK-DOCS**OpenAI Agents SDK documentation**

Product / SDK documentation

L2/L5

Official SDK docs validated: <https://openai.github.io/openai-agents-python/>; JavaScript docs validated:
<https://openai.github.io/openai-agents-js/>; OpenAI platform guide remains HTTP 403 by curl

OPENAI-TOOLS-DOCS**OpenAI tools documentation**

Product documentation

L2/L5

Accessible OpenAI Agents SDK docs used for tools/sessions/tracing references; platform guide remains HTTP 403 by curl

CREWAI-DOCS**CrewAI documentation**

Product / framework documentation

L2/L5

Official CrewAI documentation entry point validated: <https://docs.crewai.com/>

CREWAI - GITHUB**CrewAI GitHub repository**

Repository documentation

L2/L4 depending on claim

Official repository entry point validated: <https://github.com/crewAIInc/crewAI>

Source Status Notes

- L1/L2/L3/L4/L5 evidence levels indicate source confidence and claim support posture; they do not prove deployment compliance.
- OpenAI platform guide HTTP 403 is retained as an access note only; accessible official Python and JavaScript Agents SDK docs replace it for retained SDK-surface claims.
- Validation Lab methodology is cited as a public-surface non-certifying evidence adjudication example only; it is not certification, endorsement, audit opinion, or legal compliance proof.
- MPLP references carry a conflict-of-interest boundary: MPLP may be analyzed as one lifecycle protocol path, but this paper does not require, certify, or exclusively endorse MPLP.
- Product/platform documentation supports feature and capability surfaces. RCCS-T/RCCS-M/ALCS scoring remains provisional, evidence-qualified, and analytical.