

An Enterprise Architecture Framework for the Governance of AI Build and Digital Transformation

Design, Validation and Specification of the Intent-to-Runtime
Governance Framework (IRGF), Version 1.0

Dr. Rajan Kumar Upadhyay
August 2026

Executive Summary

Enterprise architecture frameworks were designed for systems whose behavior is specified in advance. Artificial intelligence systems are not. A model's outputs vary with its inputs, its grounding data, and its own retraining; an autonomous agent's actions are bounded by permissions rather than by program logic; and both change materially without any of the change events that conventional architecture governance is built to detect. At the same time, the standards that do govern AI credibly — the NIST AI Risk Management Framework, ISO/IEC 42001, and ISO/IEC 23894 — govern it as a management-system and risk concern, largely independently of the enterprise architecture in which the system sits and of the transformation program that funded it. Organisations are consequently governing the same AI system three times, through three vocabularies, with no structure that connects them.

This report asks whether enterprise architecture can be extended or reconceived to provide that connecting structure, and answers the question through a fourteen-chapter design-science sequence: problem definition, a structured landscape review of fifteen frameworks and standards, twenty-eight formal design requirements, a conceptual architecture, detailed governance models for AI build and digital transformation, a continuous assurance mechanism, an operating model, an artifact set, adversarial validation, a novelty assessment, an academic research model, and a consolidated specification.

The resulting framework — IRGF, the Intent-to-Runtime Governance Framework — is built on one metamodel expressed through three views: Structural, Lifecycle, and Control. At its center is an eight-entity Core Model: Intent, Capability, AI System, Decision, Risk, Control, Outcome, and Assurance. The name states the framework's scope, which is the chain it makes traceable, running from enterprise intent through architecture, risk classification and deployment authorization to runtime assurance. AI is established as a first-class architecture domain alongside Business, Data, Application, and Technology, with Integration, Platform, and Security treated as cross-cutting disciplines rather than peer domains. An eleven-stage AI build lifecycle is governed by five decision-critical gates whose intensity scales with a five-dimension risk score, subject to a safety-override floor that prevents low aggregate scores from masking a single catastrophic dimension. Transformation is governed at deliberately shallow depth — the framework claims visibility of, not ownership over, transformation and workforce change — and a control plane compares deployed architecture and observed AI behavior against what was approved, across eight named categories of drift.

The framework was then stress-tested rather than demonstrated. Eight enterprise scenarios, seven reviewer perspectives, and one adversarial red-team pass were run by independent reviewers instructed to find failure, producing fourteen revisions. That exercise, and the novelty assessment that followed it, materially narrowed the framework's claims. IRGF is positioned as a **complementary governance framework**: not a TOGAF replacement, not a next-generation enterprise architecture framework, and not a new architectural paradigm. It introduces no new AI risk taxonomy, no new AI management-system standard, and no new general IT governance model, deferring instead to the standards that already provide them. Its defensible contribution is narrower and consists of the connective mechanisms that make AI risk tier mechanically govern architecture-gate automation and control-enforcement permissibility, four specific original mechanisms, and the single coherent structure that allows one AI system to be traced from

strategic intent through architecture, risk tier, deployment authorization, and runtime assurance without switching frameworks mid-chain.

Two limitations are stated here rather than left to the closing pages. The framework has been demonstrated through simulated adversarial review, not through real organizational deployment, so its mechanisms are established as coherent and original but not as effective. And no systematic review of peer-reviewed literature was conducted; the landscape review compares the framework against named standards, which is a different and weaker evidentiary basis. The propositions set out in Chapter 13 are consequently offered as testable hypotheses, and the assumptions register in Appendix G is the intended starting point for the empirical work that would validate them.

About This Report

This report presents the full record of a structured research and design program, from problem definition through to a consolidated framework specification. It is organized as fourteen chapters corresponding to the stages of that program, followed by seven appendices and a reference list.

Three conventions are used throughout. Design decisions are recorded with identifiers of the form **D-*nn*** and are consolidated in Appendix A, so that any conclusion in the main text can be traced back to the reasoning and the chapter that produced it. Formal design requirements carry identifiers of the form **FR-*nn*** and are specified in Chapter 3. References to *the research brief* denote the terms of reference against which this work was commissioned; where the brief proposed a structure that the analysis subsequently rejected or narrowed, both the proposal and the reasoning for departing from it are retained rather than the proposal being quietly dropped.

Claims are deliberately graded. Statements about existing standards and frameworks are referenced to their published sources. Statements about the proposed framework are the output of this research and are marked as proposals, and where a mechanism has not been empirically tested, that is stated at the point of the claim rather than deferred to a limitations section.

Contents

Executive Summary	2
About This Report.....	4
List of Figures	6
List of Tables	7
Chapter 1. Problem Definition	9
Chapter 2. Framework and Standards Landscape	18
Chapter 3. Framework Design Requirements	31
Chapter 4. Conceptual Framework Architecture	41
Chapter 5. AI Build Governance Model	53
Chapter 6. Digital Transformation Governance Model	65
Chapter 7. Continuous Architecture Governance	73
Chapter 8. Governance Operating Model	82
Chapter 9. Framework Artifacts and Templates	92
Chapter 10. Framework Visualization and Identity	101
Chapter 11. Validation and Stress-Testing	115
Chapter 12. Novelty and Contribution Assessment	129
Chapter 13. Academic Research Model.....	138
Chapter 14. Final Framework Specification — IRGF Version 1.0.....	148
Appendix A — Consolidated Framework Design Log.....	165
Appendix B — Glossary	174
Appendix C — Diagram Index	175
Appendix D — Design Requirements	176
Appendix E — Version History.....	177
Appendix F — Open Research Questions	178
Appendix G — Assumptions Register.....	183
References.....	187

List of Figures

Figure 1. Core metamodel: entities and relationships	47
Figure 2. Overall conceptual framework	49
Figure 3. Framework layers	50
Figure 4. Strategy-to-technology traceability chain.....	51
Figure 5. The IRGF lifecycle wheel: eleven stages, five gates, and Risk and Governance at the hub	55
Figure 6. Transformation governance lifecycle (T0–T7) and its relationship to the AI build lifecycle	70
Figure 7. Enterprise architecture control plane	76
Figure 8. Exception governance flow	90
Figure 9. Artifact relationship map	99
Figure 10. Framework executive view.....	102
Figure 11. Framework practitioner view.....	103
Figure 12. Detailed reference model.....	104
Figure 13. Framework lifecycle: transformation and AI build cycles	105
Figure 14. Governance operating model.....	107
Figure 15. Continuous architecture control plane (consolidated view)	108
Figure 16. Signature diagram: the framework's layered dependency structure	110
Figure 17. Conceptual research model: constructs and relationships	141
Figure 18. The IRGF module map: twenty modules in seven groups, and what each depends on.....	149
Figure 19. AI agent governance chain	155

List of Tables

Table 1. TOGAF (The Open Group, 10th Edition).....	18
Table 2. NIST AI Risk Management Framework (AI RMF 1.0, 2023) + Generative AI Profile (NIST.AI.600-1, 2024).....	19
Table 3. ISO/IEC 42001:2023 — AI Management System.....	20
Table 4. COBIT (ISACA, 2019 with subsequent design-factor updates).....	21
Table 5. Condensed Profiles.....	22
Table 6. Comparative Coverage Matrix.....	25
Table 7. Prioritization.....	36
Table 8. Requirements Traceability Matrix	37
Table 9. Framework Dimensions	43
Table 10. Architecture Domains	44
Table 11. Defining AI Architecture	45
Table 12. The Governed AI Lifecycle.....	53
Table 13. Governance by AI Build Type	56
Table 14. AI Risk Classification	57
Table 15. Scoring methodology — a two-axis matrix, not a weighted sum	58
Table 16. AI Assurance Model.....	60
Table 17. AI Change Classification	61
Table 18. AI Lifecycle RACI (Provisional)	62
Table 19. Minimum Required Evidence	63
Table 20. Transformation Traceability Chain	65
Table 21. Portfolio Governance	67
Table 22. Transformation Governance Questions.....	68
Table 23. Transformation Governance Lifecycle (T0–T7) and Its Relationship to the AI Build Lifecycle (S0–S10).....	69
Table 24. Transformation Scorecard.....	71
Table 25. Architecture Telemetry	73
Table 26. Architecture Drift Taxonomy.....	74
Table 27. Control Plane Boundaries — the Human vs. Automated Decision Boundary.....	77
Table 28. Policy-as-Code	78
Table 29. Evidence-as-Code.....	78
Table 30. Avoiding Committee Proliferation.....	83
Table 31. Governance Levels.....	84
Table 32. Role Catalogue	84
Table 33. Decision Rights Matrix	86
Table 34. AI Build Lifecycle RACI (finalized)	87
Table 35. Transformation Governance Lifecycle RACI (new).....	88
Table 36. Escalation Model.....	88
Table 37. Candidate Artifact Disposition.....	92
Table 38. AI Use-Case Canvas.....	93
Table 39. Architecture Decision Record (AI-extended)	94

Table 40. Risk Classification Record.....	94
Table 41. Agent Card	95
Table 42. AI System Card (Derived View).....	96
Table 43. Remaining Primary Records (condensed).....	96
Table 44. Derived Views (compiled, never independently authored — full set).....	97
Table 45. Optional Artifact Set	97
Table 46. Name Development.....	111
Table 47. Cross-scenario summary table	115
Table 48. Reviewer Critiques (11.4)	117
Table 49. Recommended Revisions	121
Table 50. Component-by-Component Novelty Classification	129
Table 51. Revised Gap Matrix (updated from Chapter 2).....	134
Table 52. Constructs.....	140
Table 53. Positioning within Design Science Research Methodology (DSRM).....	143
Table 54. Reference Architectures	157
Table 55. Maturity Model	158
Table 56. Organizational Adoption Roadmap.....	160
Table 57. TOGAF Relationship	160
Table 58. Integration with AI Governance Standards.....	161

Chapter 1. Problem Definition

This chapter establishes the problem space. It traces the evolution of Enterprise Architecture, examines how AI and digital transformation alter its founding assumptions, distinguishes deficiencies that existing standards already address from those that remain open, and states the resulting research problem. No framework, dimension, lifecycle, or governance mechanism is proposed here; framework design begins in Chapter 4.

A note on epistemic status

Before proceeding, a methodological commitment. This analysis distinguishes three kinds of claims throughout. **Established facts** are matters of public record: standard versions, publication dates, regulatory status. These are treated as reliable as of August 2026, with sources noted where verified via search, not recalled from training data. **Practitioner consensus** reflects widely repeated positions in EA and AI-governance literature and industry commentary, directionally credible but not empirically settled. **Author propositions** are this analysis's own interpretive claims, the arguments being built toward a research problem, and are flagged as such. Where a claim about TOGAF (The Open Group, 2022), NIST, or ISO materials could not be verified, it is marked as an assumption requiring validation rather than stated as fact.

1. Evolution of Enterprise Architecture

Enterprise Architecture (EA) emerged from a narrow problem. Large organizations in the 1980s could not relate their proliferating information systems back to the business functions those systems were meant to serve. Zachman's 1987 framework addressed this with a classification schema: a matrix of perspectives (planner, owner, designer, builder, subcontractor) against interrogatives (what, how, where, who, when, why). The schema was descriptive, not prescriptive. It told organizations *what to document*, not *what to do*.

The Open Group Architecture Framework (TOGAF), first published in 1995 and now in its tenth edition, shifted EA from classification toward method. The Architecture Development Method (ADM) is a cyclical, phase-based process: Preliminary, Architecture Vision, Business, Information Systems, Technology Architecture, Opportunities & Solutions, Migration Planning, Implementation Governance, and Architecture Change Management. It is designed to move an enterprise from a baseline architecture to a target architecture in a controlled, traceable way. TOGAF 10 (published 2022) reorganized the standard into separately versionable parts and introduced more explicit guidance on digital-enterprise concerns. Its structural logic of sequential phases culminating in governed change remains intact from earlier editions. (*Verified: TOGAF 10th edition is the current published standard as of this writing; see The Open Group publication listings.*)

Alongside TOGAF, frameworks such as the U.S. Federal Enterprise Architecture Framework (FEAF; Federal Chief Information Officers Council, 2013), Gartner's EA methodology, and modeling notations such as ArchiMate (The Open Group, 2026) developed in parallel. Each emphasized different aspects of the same underlying assumption: that an enterprise's technology

estate is a knowable, largely deterministic set of applications, data stores, integrations, and infrastructure components, and that its behavior can be documented, planned, and governed through periodic review cycles.

For roughly three decades this assumption was reasonable. Applications executed defined logic. Data flowed through defined pipelines. Integration behavior was, in principle, traceable to code. Architecture governance built on top of this assumption (architecture review boards, reference models, standards catalogues, target-state roadmaps) worked because the artifact being governed did not change its own behavior between review cycles.

This is the premise the remainder of this analysis interrogates: does that assumption still hold, and if not, precisely where does it break down?

2. Why AI Changes Enterprise Architecture

Three properties of contemporary AI systems are structurally different from the deterministic-systems assumption underlying classical EA, and each has direct architectural consequences.

Probabilism replaces determinism. A conventional application, given the same input and the same code version, produces the same output. A large language model, given the same prompt, may not — the result depends on temperature, context window contents, model version, or the state of an external knowledge base it retrieves from at inference time. Architecture documentation that describes a system as a fixed set of components and interfaces cannot fully describe a system whose *behavior* is not fixed even when its *components* are unchanged. This is not merely a testing inconvenience. It changes what “the architecture” even refers to: is it the pipeline, the model weights, the prompt, the retrieved context, or the emergent output distribution?

The system boundary becomes porous and time-varying. A foundation model consumed via API is simultaneously an application dependency, like any external service, and something categorically different. Its internal logic is opaque. Its behavior can change when the vendor updates the model without the enterprise’s knowledge or consent, and its outputs can be influenced by data the enterprise never directly controls: training data, safety tuning, system-level guardrails set by the vendor. Traditional technology architecture treats third-party dependencies as integration points with contracted interfaces (APIs, SLAs). It has comparatively little machinery for governing a dependency whose *reasoning process* is unknown and whose *behavioral drift* is not versioned in any way the consuming enterprise controls.

Agency introduces action, not just computation. Where a traditional application computes and returns a result for a human or downstream system to act on, an AI agent can be architected to take the action itself: executing a transaction, modifying a record, calling further tools, or communicating externally. This converts an architecture question (“what does this component compute?”) into a governance question (“what is this component authorized to *do*, and to whom is that authority delegated?”). Classical EA has well-developed thinking about data flow and integration. Its thinking about delegated authority, machine identity, and bounded autonomy is comparatively underdeveloped, because until recently the enterprise’s software did not autonomously act.

Taken together, these properties mean that AI does not merely add a new technology category to an existing architecture taxonomy (the way, say, mobile or cloud did). It challenges the underlying epistemic assumption on which the discipline’s governance mechanisms (periodic review, static documentation, point-in-time approval) were built: that architecture is a knowable, stable, documentable structure.

3. Why Digital Transformation Changes Enterprise Architecture

Independent of AI specifically, digital transformation has been eroding the operating assumptions of classical EA for over a decade. It is worth treating as a distinct causal force, not folding it into “the AI problem.”

Three shifts are most consequential. First, the move from **project-based to product-based and platform-based delivery** means architecture decisions are increasingly made continuously, by distributed product teams, rather than periodically, by a central architecture function. TOGAF’s ADM cycle presumes a project has a beginning and an architecture governance checkpoint; a persistent product team shipping weekly does not fit that cadence cleanly. Second, the **decentralization of technology decision rights**, enabled by cloud self-service, SaaS procurement outside IT, and low-code/no-code tooling, means architecture decisions are made in more places, by more people, and more often than centralized architecture functions can review. Third, transformation initiatives increasingly cut *across* the domains EA traditionally separates. A single “AI-enabled customer experience transformation” simultaneously touches business capability redesign, process change, data architecture, application modernization, and workforce change. It is typically governed, if at all, by a transformation or change-management function that has limited structural connection to the architecture function.

The consequence is that even absent any AI component, digital transformation has already been straining EA’s governance model: its cadence, its centralization, and its scope boundaries. AI intensifies this strain rather than creating it from nothing. That distinction matters for locating the research problem accurately (see Section 5).

4. Governance Problems Created by AI-Enabled Transformation

Synthesizing Sections 2 and 3, the governance problems that AI-enabled digital transformation creates or sharply intensifies can be grouped into five clusters. These are practitioner-observed patterns, not empirically quantified findings, drawn from the AI-governance and EA commentary literature and from current standards-development activity. Where an academic literature base exists it is comparatively young; most substantive treatment postdates 2022–2023.

Lifecycle governance gaps. Conventional application governance is weakest exactly where AI systems need it most: after deployment. A deployed application’s logic does not change on its own. A deployed model, prompt, or agent can drift in behavior, be silently updated by a vendor, or be exposed to inputs its validation never anticipated. Post-deployment monitoring in classical EA (capacity, uptime, incident management) is operationally mature. Monitoring for behavioral drift, degradation in output quality, or emergent misuse is not part of that toolkit.

Traceability gaps. Where does a given AI output come from? Tracing an LLM output back to the data that grounded it (in a RAG pipeline), the prompt that shaped it, the model version that

generated it, and the policy that authorized the use case is a distinct practice from conventional data lineage. Data architecture practices built for structured pipelines do not natively address it.

Autonomy and delegation gaps. As argued in Section 2, when software can act rather than merely compute, governance must define authority boundaries, escalation thresholds, and reversibility requirements. Few enterprises have an architectural concept equivalent to “the maximum authority this component may exercise without human sign-off,” expressed and enforced consistently across applications, agents, and integrations.

Third-party and supply-chain governance gaps. Enterprises are increasingly dependent on foundation models they do not control and cannot fully audit. Vendor model updates, changes to safety tuning, or discontinuation of a model version are architectural events with enterprise-wide consequences. Yet there is no mature practice, analogous to dependency management in software supply chains, for foundation-model dependency risk.

Organizational and accountability gaps. AI initiatives are frequently sponsored and governed outside the architecture function altogether: by data science teams, innovation labs, or business units procuring AI SaaS directly. The accountability structures EA relies on (architecture review boards, standards enforcement) may simply not be in the approval path for a meaningful share of enterprise AI activity.

These clusters are the raw material from which a research gap can be constructed. They must first be tested against what current standards already address (Sections 6–7) before any gap claim is credible.

5. Potential Limitations of TOGAF and Conventional EA

Structural limitations must be separated here from implementation failures that are not TOGAF’s fault, because the literature and much industry commentary conflate the two too often.

Likely structural limitations — features of the standard’s own design, not just of how organizations use it: TOGAF’s core content metamodel and ADM were designed around business, data, application, and technology architecture as the four foundational domains. AI is not a first-class architecture domain in the standard’s core structure, and concepts central to AI governance (model risk, prompt and context governance, agent autonomy, behavioral drift) have no native home in the metamodel. The ADM’s phase-gated cycle assumes architecture change is well-approximated as a sequence of discrete initiatives; it does not natively model the continuous, small-grained, high-frequency change that product teams and AI systems generate. Implementation Governance (Phase G) and Architecture Change Management (Phase H) provide governance touchpoints, but are comparatively lightly specified relative to the earlier design-oriented phases. That is consistent with TOGAF’s origins in a period when architecture *design* was the harder problem and architecture *runtime assurance* was not yet recognized as equally important.

Likely implementation failures, not structural limitations — for which TOGAF should not be blamed: architecture documentation going stale is an execution failure of governance practice, not a defect in the ADM, which includes change-management phases for this reason. Architecture review boards becoming bureaucratic bottlenecks is an operating-model choice, not a TOGAF requirement; the standard does not mandate any particular review cadence or board structure. The

absence of AI-specific content in TOGAF is arguably not a “failure” at all, given that the 10th edition’s core structure predates the current wave of generative AI adoption. It is better described as a coverage gap than a defect.

This distinction matters for the eventual research contribution: a framework that merely re-implements disciplined architecture governance (which TOGAF, used well, already supports) is not a contribution. A framework earns its place only where it addresses a *structural* limitation, something TOGAF’s own design cannot accommodate without extension, and not an organizational failure to use TOGAF properly.

6. Problems Already Addressed by Existing Standards

A candidate research gap is only credible if it is checked against the current standards landscape, since TOGAF plus an assembly of adjacent standards may already solve a given problem.

Model and AI-system risk management is directly addressed by the **NIST AI Risk Management Framework (National Institute of Standards and Technology [NIST], 2023)** (AI RMF 1.0, 2023) and its 2024 **Generative AI Profile** (NIST.AI.600-1). These provide a structured Govern–Map–Measure–Manage approach to identifying and mitigating AI risk, including generative-AI-specific risks such as confabulation, content provenance, and value-chain risk. (*Verified via NIST publication records.*) Organizational AI management systems (policy, roles, lifecycle controls, and continual improvement for AI specifically) are addressed by **ISO/IEC 42001 (International Organization for Standardization [ISO] & International Electrotechnical Commission [IEC], 2023a):2023**, the first certifiable AI management system standard, structured analogously to ISO/IEC 27001 (ISO & IEC, 2022) for information security. (*Verified via ISO’s standard catalogue.*) AI-specific risk assessment methodology sits with **ISO/IEC 23894 (ISO & IEC, 2023b)**, which extends ISO 31000 (International Organization for Standardization, 2018)’s general risk-management guidance to AI systems. General IT governance, covering decision rights, value delivery, and accountability for technology investment, is well established by **COBIT (ISACA, 2018)** and **ISO/IEC 38500 (ISO & IEC, 2024)**. Information security is comprehensively addressed by **ISO/IEC 27001** and the **NIST Cybersecurity Framework (NIST, 2024)**, both of which practitioners are increasingly extending, though not yet through formal reissue, to address AI-specific security concerns such as prompt injection and model exfiltration. Data governance (stewardship, quality, lineage) is addressed by **DAMA-DMBOK (DAMA International, 2017)** and, for decentralized architectures, by emerging data-mesh governance practice. General enterprise risk management is addressed by **ISO 31000**.

Regulatory obligations are also evolving quickly and should not be treated as settled background. As of August 2026, the **EU AI Act (European Parliament and Council of the European Union, 2024)** remains the most comprehensive binding AI-specific regulation. Its implementation timeline has itself become a live governance-drift example worth noting for the framework problem: reporting in 2026 indicates the EU has agreed, through a “Digital Omnibus” legislative package, to defer certain high-risk AI system obligations that were originally due to apply from August 2026. (*This is based on current news reporting rather than a primary legal text reviewed directly in this session; the precise revised dates and scope of the deferral should be treated as an assumption requiring validation before being cited in any later chapter.*) This regulatory volatility is itself relevant evidence for the “regulatory drift” governance problem named in

Section 4. If binding compliance deadlines can shift within a single year, an architecture governance framework that treats regulatory obligations as a static input at design time is already structurally exposed.

Taken together, these standards mean that model risk assessment, AI management-system structure, IT governance, security, and data governance are **not** open problems in the sense of lacking any authoritative treatment. Any new framework that simply re-describes what NIST AI RMF, ISO 42001, or COBIT already do is duplicative, not original, and Section 7 of the project’s design discipline (avoid framework duplication) would correctly reject it.

7. Problems That Remain Insufficiently Addressed

What the standards in Section 6 largely do *not* do is **integrate** across the enterprise-architecture, AI-build, and digital-transformation domains through a shared architectural structure. This is the more defensible territory for a research gap. Each standard is domain-scoped by design. NIST AI RMF and ISO 42001 govern AI risk and AI management systems, but neither is an enterprise architecture method, and neither specifies how AI risk classification should connect to business-capability maps, technology architecture decisions, or transformation portfolios. COBIT governs IT decision rights broadly but has no AI-specific or architecture-specific content. TOGAF governs architecture structure and process but, per Section 5, has no native AI domain.

More specifically, four gaps recur across the literature and standards reviewed here without a clearly owned solution. **Cross-domain traceability**: a mechanism connecting a specific AI system, through its architecture, to the business capability it serves, the risk tier it has been assigned, the policy that governs it, and the outcome it is meant to produce, maintained as a live structure rather than reconstructed on demand. **Continuous architecture assurance**: verifying that the architecture actually running in production still matches what was approved, and that AI-specific properties (drift, degradation, emergent behavior) are part of that ongoing verification and not a one-time approval gate. **Agent-specific architectural governance**: identity, permission boundaries, and action authority for autonomous components, addressed piecemeal by security frameworks (identity and access management) but not as an integrated architectural concern connecting business authority delegation to technical enforcement. **Transformation-to-architecture-to-AI accountability chains**: a consistent answer to “who is accountable for this outcome” that holds across a transformation initiative, its architecture decisions, and the AI systems it deploys, instead of three separate accountability structures (transformation governance, architecture governance, AI governance) that may not be reconciled.

It is not yet established that these integration gaps require a *new enterprise architecture framework* to close, as opposed to (a) an extension to TOGAF’s metamodel, (b) an orchestration layer that references existing standards without replacing them, or (c) better organizational practice using tools that already exist. This is the central question Chapter 1 must leave open rather than resolve.

8. Candidate Research Gap

Based on Sections 4–7, the candidate research gap is not “AI governance is inadequate” (it is reasonably well served by NIST AI RMF and ISO 42001) and not “TOGAF is obsolete” (its architecture-development method remains structurally sound for its original scope). The

candidate gap is narrower and more defensible: **there is no widely adopted architectural mechanism that integrates enterprise architecture, AI build governance, and digital transformation governance into one traceable, continuously assured structure, such that an enterprise can consistently answer, for any given AI-enabled capability, why it exists, what it depends on, what authority it holds, what risk it carries, and whether its current runtime state still matches what was approved.** Existing standards each govern a slice of this question well; none governs the connective structure across slices.

9. Proposed Research Problem

The research problem, provisionally, is to determine whether and how Enterprise Architecture — as a discipline, not necessarily as TOGAF specifically — can be extended, overlaid, or reconceived to provide this integrating structure. It must do so without duplicating what NIST AI RMF, ISO 42001, ISO 23894, COBIT, or TOGAF itself already do well, and without introducing governance overhead that a risk-proportionate design should avoid. This framing keeps open, per the research brief's instruction, whether the eventual answer is an extension of TOGAF, an overlay on it, a complementary framework, a next-generation EA framework, or a genuinely new paradigm. That determination requires the landscape comparison and gap matrix reserved for Chapter 2, and should not be prejudged here.

10. Preliminary Research Question

How can Enterprise Architecture be extended or reconceptualized to provide an integrated, continuously assured governance structure connecting AI build, AI-enabled systems, and enterprise digital transformation — without duplicating the domain-specific governance already provided by existing AI-risk, IT-governance, and architecture standards?

This restates the project's primary research question with one addition: the non-duplication constraint. Section 7 shows it must be a first-class design requirement and not an afterthought, given how much of the adjacent ground is already well covered.

11. Why This Research Could Constitute an Original Contribution

If the analysis above holds, originality would not come from inventing new AI-risk taxonomy, new AI management-system requirements, or a new IT-governance model. Those exist and are reasonably mature. It would come, if anywhere, from three narrower and more testable claims. The first is that AI systems and AI-enabled transformation require architecture to be treated as a **continuously observed** property rather than a periodically documented one, an extension of TOGAF-style governance and not a replacement for it. The second is that connecting AI risk classification, architecture decisions, and transformation outcomes through a **single traceable structure** is not currently offered by any one existing standard, even though each standard covers part of it. The third is that autonomous AI agents require an architectural governance treatment of authority, delegation, and enforcement that does not yet have a settled home in either EA or AI-governance literature. Each of these is a testable proposition, not a certainty. Each must survive the Section 6 duplication test in more detail (COBIT and ISO 38500 already address decision rights broadly; NIST's evolving agentic-AI profile work may already be moving into agent governance) before being accepted as an actual gap. That closer test is reserved for Chapter 2.

Preliminary Research Gap

Existing standards (TOGAF, NIST AI RMF, ISO/IEC 42001, ISO/IEC 23894, COBIT) each govern architecture, AI risk, or IT decision rights competently within their own scope. But no widely adopted framework integrates enterprise architecture, AI build governance, and digital transformation governance into one continuously assured, traceable structure connecting business intent, architecture decisions, AI system risk and authority, and measured outcomes.

Preliminary Research Question

How can Enterprise Architecture be extended or reconceptualized to provide an integrated, continuously assured governance structure connecting AI build, AI-enabled systems, and enterprise digital transformation — without duplicating the domain-specific governance already provided by existing AI-risk, IT-governance, and architecture standards?

Candidate Contribution

Provisionally, and pending the Chapter 2 landscape comparison: an architectural integration and continuous-assurance mechanism spanning EA, AI-build governance, and transformation governance — not a new AI-risk taxonomy, not a replacement IT-governance model, and not a rewrite of TOGAF’s design method. The strongest candidate sub-contributions are (a) treating architecture conformance as continuously observable rather than periodically documented, (b) a cross-domain traceability structure linking capability, architecture, AI system, risk tier, and outcome, and (c) an architectural (not merely operational) treatment of AI agent authority and delegation. All three require the duplication test in Chapter 2 before being accepted.

Assumptions Requiring Validation

The characterization of TOGAF 10’s metamodel as lacking AI as a first-class domain is based on the standard’s known public structure. It should be verified against the primary text directly before being asserted in the final specification. The claim about the EU AI Act’s 2026 “Digital Omnibus” deferral rests on August 2026 secondary reporting, not a primary legal source reviewed in this session, and its exact scope and dates need direct verification. The assumption that NIST’s agentic-AI profile work and existing identity/access-management security practice do not already substantively cover agent authority governance needs direct testing against current NIST and security-standard material, not assertion. The assumption that continuous architecture assurance is unaddressed, as opposed to addressed under a different name in DevOps/platform-engineering or cloud security-posture-management practice, also needs testing. “Policy-as-code” and cloud configuration drift detection are mature commercial practices that may already partially solve this for infrastructure, if not for AI-specific properties. Finally, the assumption that a single integrated framework is preferable to a lighter-weight orchestration layer referencing existing standards has not been tested against practitioner or academic opinion. It is a live question, not a conclusion.

Questions That Must Be Answered Before Framework Design

Does TOGAF's ADM and content metamodel, read closely instead of characterized from general knowledge, already accommodate an AI domain through extension mechanisms it explicitly provides, making a "new framework" unnecessary? Where exactly does NIST's AI RMF Generative AI Profile and any emerging agentic-AI guidance stop, and does that boundary leave the agent-governance gap claimed in Section 7 genuinely open? Is the traceability gap identified in Section 7 a governance problem (no framework specifies it) or a tooling problem (no product implements it, even though frameworks like ArchiMate could in principle model it)? If it is a tooling problem, does that change what kind of research contribution — framework versus reference implementation versus methodology — is appropriate? What is the actual empirical basis, beyond practitioner commentary, for claiming organizations are struggling with these integration problems? Is there survey, case-study, or industry-analyst evidence to cite in Chapter 2, or does this remain a plausible-but-unvalidated premise that the research design itself must test? Should the eventual framework position itself as EA extension, EA overlay, complementary governance layer, next-generation EA, or new paradigm? What evidence from the Chapter 2 landscape comparison would distinguish these positions from each other rather than treating them as rhetorical alternatives?

Chapter 2. Framework and Standards Landscape

Chapter 1 identified a preliminary research gap: no widely adopted framework integrates enterprise architecture, AI build governance, and digital transformation governance into one continuously assured, traceable structure. This chapter tests that claim against the actual standards landscape, before any framework component is designed. It closes with a quality test asking whether the research problem could be solved simply by using existing frameworks more effectively.

Note on analytical depth. Seventeen-criterion profiles are produced in full only for the four frameworks most structurally adjacent to the candidate integration problem — TOGAF (The Open Group, 2022), NIST AI RMF (NIST, 2023), ISO/IEC 42001 (ISO & IEC, 2023a), and COBIT (ISACA, 2018). These are the frameworks a “why not simply use X” challenge would most plausibly invoke. The remaining frameworks receive a condensed six-point profile covering purpose, scope, lifecycle model, principal strength, principal limitation, and relevance to the candidate gap. This is a documented scoping decision, not an omission. It is recorded in Appendix A and revisited where the stress-testing in Chapter 11 suggests a framework was under-weighted.

A. Framework Landscape Analysis

A.1 Enterprise Architecture Frameworks

TOGAF (The Open Group, 10th Edition)

Table 1.

Criterion	Assessment
1. Original purpose	Provide a repeatable method (the ADM) for developing, governing, and maintaining enterprise architecture aligned to business goals.
2. Primary governance scope	Architecture development and change governance across business, data, application, and technology domains.
3. Unit of analysis	The architecture artifact/deliverable (baseline, target, transition architectures) produced within an ADM cycle.
4. Lifecycle/process model	Cyclical, phase-gated ADM (Preliminary → Vision → Business → Information Systems → Technology → Opportunities & Solutions → Migration Planning → Implementation Governance → Change Management), wrapped by Requirements Management.
5. Architecture coverage	Strong for business, data, application, technology. No native AI, agent, or model domain in the core content metamodel.
6. AI coverage	Not explicitly addressed in the standard’s core structure. TOGAF’s extension mechanisms (content metamodel extensions) could in principle accommodate an AI domain, but this is not provided out of the box. (<i>Assumption requiring validation — see Chapter 1 carry-forward.</i>)
7. Data governance coverage	Data Architecture phase covers data entities, lifecycle, and quality at a conceptual level; does not reach data lineage, consent, or AI-grounding concerns in operational depth.
8. Risk coverage	Risk Management is referenced as a supporting technique (ADM guidelines) but is not a fully developed risk

Criterion	Assessment
	taxonomy; TOGAF assumes integration with a dedicated risk framework.
9. Transformation coverage	Business Architecture and Opportunities & Solutions phases touch transformation planning; TOGAF does not itself provide portfolio management, benefits realization, or operating-model design depth.
10. Runtime governance	Implementation Governance (Phase G) exists but is comparatively lightly specified; oriented toward project-based implementation, not continuous runtime assurance.
11. Continuous monitoring capability	Weak. Architecture Change Management (Phase H) is designed around periodic re-entry into the ADM cycle, not continuous telemetry.
12. Decision rights	Architecture Governance concepts (Architecture Board, compliance review) are described structurally but staffing/authority is left to the adopting organization.
13. Organizational roles	Defines architecture roles conceptually (architects by domain) but not AI-specific or transformation-specific roles.
14. Required artifacts	Extensive: architecture vision, definition documents, roadmaps, ADRs-equivalent, compliance assessments.
15. Strengths	Mature, widely adopted, method-rich, extensible metamodel, strong traceability discipline for architecture change.
16. Limitations	Document-centric bias; cycle time assumes discrete initiatives; no first-class AI/agent concept; governance intensity is not natively risk-tiered.
17. Relevance to proposed framework	High — likely foundation layer for architecture domain structure and ADM-style governance discipline, but insufficient alone for AI-specific and continuous-assurance requirements.

NIST AI Risk Management Framework (AI RMF 1.0, 2023) + Generative AI Profile (NIST.AI.600-1, 2024)

Table 2.

Criterion	Assessment
1. Original purpose	Voluntary, risk-based guidance to help organizations manage risks of AI systems across their lifecycle.
2. Primary governance scope	AI risk management specifically: trustworthiness characteristics (valid/reliable, safe, secure/resilient, accountable/transparent, explainable/interpretable, privacy-enhanced, fair).
3. Unit of analysis	The AI system and its risk profile, assessed through four functions.
4. Lifecycle/process model	Govern, Map, Measure, Manage — a continuous, non-linear set of functions rather than a phase-gated cycle.
5. Architecture coverage	None — NIST AI RMF is explicitly not an architecture framework; it does not address business, data, or technology architecture structure.
6. AI coverage	Strong. The Generative AI Profile adds GenAI-specific risks (confabulation, data privacy, information integrity, harmful bias, dangerous/violent content, intellectual property, value chain/component integration).
7. Data governance coverage	Partial — addresses data-related AI risk (representativeness, privacy) but is not a data architecture or lineage standard.
8. Risk coverage	Strong for AI-specific risk; does not integrate with general enterprise risk taxonomies (ISO 31000 (International Organization for Standardization, 2018)-style) natively, though it is designed to be compatible.

Criterion	Assessment
9. Transformation coverage	None.
10. Runtime governance	Manage function includes ongoing risk monitoring and response, but implementation detail (telemetry, drift detection mechanics) is left to the adopter.
11. Continuous monitoring capability	Conceptually present (Manage function expects ongoing monitoring) but not operationally specified — no reference architecture for how monitoring should be implemented.
12. Decision rights	Govern function addresses organizational accountability structures at a policy level, not a detailed RACI.
13. Organizational roles	General (AI actors across the lifecycle) rather than enterprise-specific roles.
14. Required artifacts	Lightweight by design — profiles, risk documentation — not a heavy artifact regime.
15. Strengths	Authoritative, flexible, GenAI-current, strong trustworthiness taxonomy, US government-recognized reference point.
16. Limitations	Not an architecture method; does not connect AI risk to business capability, architecture decisions, or transformation portfolios; agentic AI coverage is still maturing at the time of writing.
17. Relevance to proposed framework	High for AI risk taxonomy and lifecycle function language — the proposed framework should reference/adopt AI RMF risk concepts rather than reinvent them, and focus its own contribution on connecting this risk assessment to architecture and business structure.

ISO/IEC 42001:2023 — AI Management System

Table 3.

Criterion	Assessment
1. Original purpose	Certifiable management-system standard (Plan-Do-Check-Act, structurally parallel to ISO 27001 (ISO & IEC, 2022)) for organizations that develop, provide, or use AI products/services.
2. Primary governance scope	Organizational AI management: policy, objectives, roles, resources, AI risk and impact assessment processes, continual improvement.
3. Unit of analysis	The organization's AI management system as a whole, not individual AI systems.
4. Lifecycle/process model	Management-system cycle (establish, implement, maintain, continually improve) rather than an AI-build lifecycle.
5. Architecture coverage	None.
6. AI coverage	Strong at the management-system level (policy, competence, AI system impact assessment per Annex); does not specify architecture-level controls (model routing, RAG, agent design).
7. Data governance coverage	Referenced through data-for-AI system requirements (Annex controls) but not a full data governance standard.
8. Risk coverage	Requires AI risk assessment as part of the management system; does not itself define the risk methodology in the depth ISO 23894 (ISO & IEC, 2023b) does.
9. Transformation coverage	None.
10. Runtime governance	Continual improvement and monitoring are management-system requirements, but implementation mechanics are left open, consistent with ISO management-system standards generally.
11. Continuous monitoring capability	Present as a requirement, not as a specified mechanism.

Criterion	Assessment
12. Decision rights	Requires defined roles and responsibilities (top management commitment, AI policy ownership) but not an enterprise-architecture-integrated decision-rights model.
13. Organizational roles	AI management roles required; not integrated with EA, transformation, or business-capability roles.
14. Required artifacts	AI policy, objectives, risk/impact assessments, documented processes — auditable for certification.
15. Strengths	First certifiable AI management standard; structurally interoperable with ISO 27001; credible external assurance mechanism (third-party certification).
16. Limitations	Organization-level, not system- or architecture-level; does not address agent autonomy, architecture traceability, or continuous technical assurance.
17. Relevance to proposed framework	High as the organizational “management system” layer the proposed framework can reference for policy/accountability scaffolding, while the new framework’s contribution focuses on architecture-level and cross-domain integration ISO 42001 does not attempt.

COBIT (ISACA, 2019 with subsequent design-factor updates)

Table 4.

Criterion	Assessment
1. Original purpose	Governance and management framework for enterprise IT, built around governance objectives and design factors.
2. Primary governance scope	IT governance broadly; value delivery, risk optimization, resource optimization, stakeholder alignment.
3. Unit of analysis	Governance and management objectives (40 objectives across Evaluate-Direct-Monitor and Align-Plan-Organize etc. domains).
4. Lifecycle/process model	Governance system design using design factors (strategy, risk profile, compliance requirements, IT role, sourcing model, etc.), not a build lifecycle.
5. Architecture coverage	Includes an “Managed Enterprise Architecture” management objective at a governance level, but not a method for producing architecture (that gap is explicitly why organizations pair COBIT with TOGAF).
6. AI coverage	Not natively specified in the core 2019 objectives; ISACA has published supplementary AI-focused guidance separately, but AI is not integrated into the core governance objective set as of this analysis. (<i>Assumption requiring validation — check for a formal COBIT AI extension.</i>)
7. Data governance coverage	A dedicated data-management objective exists at a governance level; not an operational data architecture standard.
8. Risk coverage	Strong for IT risk governance generally (risk optimization is a core governance objective); not AI-risk-specific.
9. Transformation coverage	Governance objectives touch program/project and portfolio management but COBIT is not a transformation methodology.
10. Runtime governance	Monitor, Evaluate, and Assess objectives exist; COBIT is a mature source for governance monitoring structure, but not for technical runtime telemetry.
11. Continuous monitoring capability	Governance-level monitoring is well specified; technical continuous assurance is not COBIT’s focus.
12. Decision rights	This is COBIT’s strongest area — governance vs. management distinction, RACI-style accountability, is

Criterion	Assessment
	core to the framework's design.
13. Organizational roles	Strong, general-purpose IT governance roles; not AI- or architecture-domain-specific.
14. Required artifacts	Governance system design outputs, policy documents, performance management artifacts.
15. Strengths	Most mature IT decision-rights and accountability framework available; widely adopted; explicitly designed to integrate with other frameworks (including TOGAF) rather than replace them.
16. Limitations	Not architecture-specific, not AI-specific; risk of overlap/redundancy if the new framework reinvents governance-objective structure COBIT already provides well.
17. Relevance to proposed framework	High for governance/decision-rights vocabulary and structure — the proposed framework's operating model (Chapter 8) should align with, not duplicate, COBIT's governance-vs-management distinction.

A.2 Condensed Profiles

Table 5.

Framework	Purpose / Scope	Lifecycle Model	Key Strength	Key Limitation	Relevance to Candidate Gap
Zachman Framework (Zachman, 1987)	Classification schema relating stakeholder perspectives to interrogatives (what/how/where/who/when/why)	None — descriptive taxonomy, not a process	Rigorous, durable ontology for classifying architecture artifacts	Provides no method, no governance process, no AI or risk content; largely descriptive today	Low direct relevance; useful only as a classification reference, not a governance mechanism
FEAF (Federal EA Framework; Federal Chief Information Officers Council, 2013)	US federal reference-model structure for cross-agency architecture alignment	Reference-model-based, not cyclical	Strong performance/business/data/service/technology reference model alignment	Public-sector-specific origin; limited native AI or continuous-assurance content	Low-moderate; reference-model pattern (not content) may inform capability-to-architecture mapping
ArchiMate (The Open Group, 2026) 3.2 (The Open Group)	Modeling notation for enterprise architecture across business, application, technology, and (since 3.x) motivation, strategy, physical, and implementation layers	N/A — notation, not governance process	Precise, tool-supported modeling language with an extensible metamodel; increasingly used to model AI/data elements informally by practitioners	Not a governance framework; provides no risk, decision-rights, or lifecycle content	Moderate — plausible notation layer for expressing the proposed metamodel (Chapter 4), not a governance substitute
Capability-based planning (general practice)	Plans investment and architecture around business capabilities rather than projects or systems	Capability maturity assessment → gap → roadmap	Strong strategy-to-investment traceability logic	Not standardized; implementations vary widely in rigor	High conceptually — the strategy-to-capability-to-initiative chain in Chapter 1's traceability argument draws directly on this practice
ISO/IEC 23894:2023	AI-specific extension of ISO	Risk-management	Ties AI risk into standard	Guidance-level, not prescriptive;	Moderate — a reference the

Framework	Purpose / Scope	Lifecycle Model	Key Strength	Key Limitation	Relevance to Candidate Gap
	31000 risk management guidance	process (establish context, assess, treat, monitor) applied to AI	enterprise risk-management vocabulary	no architecture or lifecycle-gate content	proposed AI risk model (Chapter 5) should align to rather than diverge from
OECD AI Principles (Organisation for Economic Co-operation and Development, 2024) (2019, updated May 2024)	High-level intergovernmental principles for trustworthy AI (inclusive growth, human rights, transparency, robustness, accountability)	Principles, not a process	Broad international legitimacy; the 2024 update explicitly addresses general-purpose and generative AI	Principles-level only — no operational mechanism	Low-moderate; useful as a principles cross-check (Chapter 4 principles should not contradict OECD language), not a governance mechanism
Model-risk-management practice (e.g., banking SR 11-7-style approaches)	Sector practice (originating in financial-services regulatory guidance) for validating and governing statistical/ML models	Model development → independent validation → ongoing monitoring → periodic revalidation	Mature, battle-tested approach to model validation and independent challenge	Sector-specific origin; not natively extended to generative AI, agents, or enterprise architecture integration	Moderate-high — the independent-validation and revalidation discipline is directly relevant to the proposed AI assurance model (Chapter 5)
ISO/IEC 38500 (ISO & IEC, 2024):2015	Governance of IT standard — six principles (responsibility, strategy, acquisition, performance, conformance, human behavior)	Evaluate-Direct-Monitor (same model COBIT builds on)	Concise, principle-based governance-of-IT standard	Very high-level; no architecture, AI, or lifecycle detail	Low-moderate; principle language may inform framework principles (Chapter 4)
ITIL 4 (AXELOS, 2019)	IT service management practice (service value system, value streams, practices)	Service value chain (plan, improve, engage, design & transition, obtain/build, deliver & support)	Mature operational service-management discipline, including a “monitoring and event management” practice relevant to runtime observability	Service-management orientation, not architecture governance; AI content is emerging, not core	Low-moderate; runtime/event-management practice patterns are relevant background for the continuous-assurance concept (Chapter 7)
DAMA-DMBOK2 (DAMA International, 2017)	Comprehensive data management body of knowledge (11 knowledge areas including governance, quality, metadata, architecture)	Function-based, not phase-gated	Most complete data governance reference available; strong on lineage and metadata concepts	Not AI-specific; does not natively address model-grounding or AI lineage as a distinct concept	High for data governance content the new framework should reference rather than reinvent; a gap remains specifically at the <i>AI-data-model</i> lineage seam
Data mesh governance (practitioner concept, Dehghani et al.)	Decentralized, domain-oriented data product ownership with federated computational governance	N/A — organizational/architectural pattern	Directly relevant pattern for decentralized data and (by extension) decentralized AI-product ownership	Not a formal standard; implementation maturity varies significantly across organizations	Moderate-high as a structural analogy for federated AI/architecture governance (Chapter 8)
NIST	Voluntary	Continuous	The 2024 Govern	Cybersecurity-	Moderate — a

Framework	Purpose / Scope	Lifecycle Model	Key Strength	Key Limitation	Relevance to Candidate Gap
Cybersecurity Framework 2.0 (NIST, 2024) (2024)	cybersecurity risk-management framework, now including a Govern function alongside Identify, Protect, Detect, Respond, Recover	functions, not phase-gated	function addition strengthens alignment with enterprise governance and supply-chain risk; well suited as a model for how a Govern function could sit alongside architecture functions	scoped; does not address AI-specific risk (model drift, hallucination) or architecture traceability	structural precedent (a distinct “Govern” function) worth learning from for the proposed framework’s own governance layer
ISO/IEC 27001:2022	Certifiable information security management system standard	PDCA management-system cycle, same family as ISO 42001	Mature, widely certified, strong control-catalogue (Annex A)	Security-scoped; does not address AI-specific or architecture-specific concerns	Moderate; the proposed framework’s security-architecture domain should reference, not duplicate, ISO 27001 controls
Zero Trust Architecture (NIST SP 800-207; NIST, 2020)	Security architecture principle set: never trust, always verify, least privilege, continuous verification	Architectural principle set, not a lifecycle	Strong conceptual foundation for machine-identity and least-privilege thinking directly applicable to AI agents	Security-architecture-scoped; does not address business/AI risk classification or architecture traceability	High as a direct architectural precedent for agent identity and permission boundaries (Chapter 5/12), likely to be explicitly cited rather than reinvented
ISO 31000:2018	Generic enterprise risk-management guidance (principles, framework, process)	Iterative risk process (establish context, assess, treat, monitor/review)	Foundational, sector-agnostic risk vocabulary that ISO 23894 and most AI risk approaches build on	Generic — provides no AI, architecture, or technology-specific content	High as the upstream risk vocabulary the proposed AI risk model should stay compatible with
Digital transformation / product-operating-model / platform-operating-model practice (e.g., capability maturity models, agile-transformation and product-operating-model literature)	Practitioner and academic guidance on operating-model change, product-centric delivery, and platform strategy	Varies; typically capability maturity assessment → target operating model → roadmap	Directly addresses the transformation traceability and portfolio-governance requirements Chapter 1 identified as EA’s weak connection point	Fragmented — no single authoritative standard; quality and rigor vary widely by source, and much of it is consultancy material rather than peer-reviewed or standards-body work	High conceptually, but treated cautiously per the instruction not to over-credit generic consultancy diagrams as formal frameworks

B. Comparative Coverage Matrix

Ratings: **S** = Strong, **P** = Partial, **W** = Weak, **N** = Not Explicitly Addressed. Columns group closely related frameworks to keep the matrix legible: **EA-Mod** = Zachman/ArchiMate/FEAF as a group; **Gov** = COBIT + ISO/IEC 38500; **Sec** = ISO/IEC 27001 + NIST CSF 2.0 + Zero Trust;

Data = DAMA-DMBOK2 + data mesh; **DT** = digital-transformation/product-operating-model practice.

Table 6. Comparative Coverage Matrix

Capability	TO GA F	EA- Mod	Gov	NIS T AI RM F	ISO 4200 1	ISO 238 94	Sec	Dat a	ISO 310 00	DT	Proposed Need
Business strategy alignment	P	W	S	N	W	N	N	N	N	S	Integration point, not new content
Capability planning	P	W	W	N	N	N	N	N	N	S	Integration point
Business architecture	S	P	W	N	N	N	N	N	N	P	None — reuse TOGAF
Data architecture	S	P	N	N	W	N	N	S	N	W	None — reuse TOGAF/DAMA
Application architecture	S	P	W	N	N	N	N	N	N	W	None — reuse TOGAF
Technology architecture	S	P	W	N	N	N	P	N	N	W	None — reuse TOGAF
AI architecture (model/prompt/agent/RAG)	N	W	N	N	W	N	N	N	N	N	Gap — Category D
Model governance	N	N	W	S	P	P	N	N	N	N	Integrate NIST AI RMF; architectural connective tissue missing
AI development lifecycle	N	N	N	P	P	N	N	N	N	N	Gap — Category C/D
AI risk classification	N	N	N	S	P	S	N	N	P	N	Integrate ISO 23894/NIST; enterprise-architecture tiering missing
Generative AI governance	N	N	N	S	P	W	N	N	N	N	Integrate NIST GenAI Profile
AI agent governance	N	N	N	W	W	W	P (Zero Trust)	N	N	N	Gap — Category D , though Zero Trust supplies identity/permission precedent
AI autonomy governance	N	N	N	W	W	W	W	N	N	N	Gap — Category D
Human oversight	N	N	W	S	P	P	N	N	N	N	Integrate NIST AI RMF
AI assurance (post-deployment)	N	N	W	P	P	W	N	N	N	N	Gap — Category C
Deployment authorization	W	N	S	W	P	W	N	N	N	W	Integrate COBIT decision-rights pattern; risk-tiered gating missing
Continuous monitoring (technical)	W	N	W	W	W	W	S	N	N	N	Gap — Category D , though security tooling patterns are reusable
Model drift	N	N	N	W	W	W	N	N	N	N	Gap — Category D
Architecture drift	W	N	W	N	N	N	P	N	N	N	Gap — Category

Capability	TO GA F	EA- Mod	Gov	NIS T AI RM F	ISO 4200 1	ISO 238 94	Sec	Dat a	ISO 310 00	DT	Proposed Need
											C/D
Data lineage	W	N	N	N	W	N	N	S	N	N	Integrate DAMA; extend to AI- grounding lineage
AI lineage (model↔data↔pro mpt↔output)	N	N	N	W	W	N	N	W	N	N	Gap — Category D
Decision traceability (strategy→system)	W	N	P	N	N	N	N	N	N	P	Gap — Category C; fragments exist, none connect end to end
Runtime controls (technical enforcement)	N	N	W	W	W	N	S	N	N	N	Integrate security practice (policy-as- code)
Transformation portfolio governance	W	N	P	N	N	N	N	N	N	S	Integration point — reuse DT practice, connect to architecture
Benefits realization	N	N	W	N	N	N	N	N	N	P	Integration point
Architecture governance (process)	S	N	P	N	N	N	N	N	N	N	None — reuse TOGAF
Regulatory governance	W	N	P	P	P	P	P	P	N	N	Fragmented (Category B) — orchestration opportunity, not new content
Policy enforcement	W	N	S	W	P	N	S	N	N	N	Integration point — reuse security policy-as-code patterns
Architecture-as- code	N	N	N	N	N	N	P	N	N	N	Gap — Category D
Policy-as-code	N	N	N	N	N	N	P	N	N	N	Extend security practice into architecture/AI domain — Category C
Continuous assurance (composite)	W	N	W	P	W	W	P	N	N	N	Gap — Category D; the most consequential single gap identified

Selected judgment rationale, for the ratings most likely to be contested: TOGAF is rated **P**, not **W**, on decision traceability because Requirements Management and the ADM's phase-to-phase deliverables do establish some traceability. The weakness is that this traceability does not extend past architecture into AI risk or transformation benefit. COBIT is rated **S** on business strategy alignment and deployment authorization because governance-objective design and the EDM (Evaluate-Direct-Monitor) model are built precisely for this, even though COBIT specifies neither architecture nor AI content. Zero Trust is rated **P**, not **N**, on AI agent governance and continuous monitoring: its least-privilege and continuous-verification principles transfer directly to machine identities, even though the standard itself was not written with AI agents in mind. That is flagged as an extension opportunity, not an existing solution.

C. Integration Map

The landscape analysis supports a clear division of labor, not a single-framework replacement. Architecture domain structure and change-governance method should be inherited from **TOGAF**, with its content metamodel extended (not replaced) to add an AI domain. AI-specific risk taxonomy and lifecycle risk functions should be inherited from **NIST AI RMF and its Generative AI Profile**, referenced directly rather than re-derived. Organizational AI management-system scaffolding (policy, accountability, certification-grade process) should be inherited from **ISO/IEC 42001**, with the proposed framework treating an organization's ISO 42001 posture as an input, not a component to reproduce. General AI risk methodology should stay compatible with **ISO/IEC 23894 and ISO 31000**, so that enterprise risk reporting does not fork into an incompatible AI-specific taxonomy. IT governance decision-rights structure should be inherited from **COBIT and ISO/IEC 38500**, particularly the governance/management distinction and the Evaluate-Direct-Monitor pattern. Data governance and lineage practice should be inherited from **DAMA-DMBOK2**, extended at the AI-grounding seam (model↔data↔prompt↔output) DAMA does not cover. Security architecture, identity, and least-privilege enforcement patterns, all of them relevant to agent governance, should be inherited from **Zero Trust Architecture, ISO/IEC 27001, and NIST CSF 2.0**. The 2024 Govern function of NIST CSF 2.0 is a useful structural precedent for how a governance layer can sit alongside operational functions without duplicating them. Transformation traceability logic (capability-based planning, product/platform operating models) should be drawn from **digital-transformation practice**, treated cautiously given its fragmentation and lower standardization.

What none of these frameworks does, individually or as a simple sum, is hold these seven inheritances together. No single structure lets a given enterprise query them for a specific AI-enabled capability — the “single traceable structure” named as the candidate gap in Chapter 1. That remains the open question this chapter must now test, not assume.

D. Gap Taxonomy

Architecture gaps. AI (models, prompts, RAG, agents) has no first-class position in the TOGAF content metamodel or in ArchiMate's standard element set; architecture-as-code has no standards-level treatment comparable to policy-as-code in security practice.

AI lifecycle gaps. No framework connects use-case qualification through architecture decision, build, assurance, deployment authorization, and retirement as a single governed sequence that is also architecture-aware. NIST AI RMF's functions are lifecycle-adjacent but not an architecture-integrated gate sequence, and ISO 42001's PDCA cycle operates at the management-system level, not the individual-system level.

Governance gaps. Decision rights for AI-specific events (autonomy increase, model swap, risk-tier change) are addressed neither by COBIT's general IT governance objectives nor by TOGAF's Architecture Board concept. No framework specifies where AI-specific governance sits relative to existing architecture governance and IT governance bodies.

Transformation gaps. No standards-body framework connects transformation portfolio decisions to architecture decisions to AI risk in a single traceable chain; this connective tissue currently exists, where it exists at all, as ad hoc organizational practice.

Runtime gaps. Continuous assurance — verifying that deployed architecture and deployed AI behavior still match what was approved — is the single most consistently “Not Explicitly Addressed / Weak” capability across the entire matrix (Section B). Security tooling (Sec column) is the closest existing practice, but it monitors for security events, not architecture or AI-risk conformance.

Organizational gaps. Federating decision rights across architecture, AI, security, data, and transformation without creating committee proliferation is not solved by any single reviewed framework. Each framework assumes its own governance body exists, without addressing how that body relates to the others.

Integration gaps. This is the umbrella category: every gap above is, at root, a *connective* gap. The individual components (risk taxonomy, management-system process, governance objectives, architecture method, security controls) exist in mature form. What is absent is the structure that keeps them mutually traceable for a specific AI-enabled business capability over time.

E. Candidate Innovation Areas

Eight areas survive the landscape review as plausible sources of original contribution, each classified against the project’s novelty criteria. **AI as a first-class enterprise architecture domain:** Extension (of TOGAF’s metamodel, not invention of AI concepts, which NIST/ISO already define). **Strategy-to-agent traceability structure:** New Contribution candidate, since no reviewed framework provides this end-to-end chain, though every link in the chain individually exists elsewhere. **Continuous architecture and AI conformance assurance (“control plane” concept):** New Contribution candidate, and the strongest single candidate given how consistently Section B rated this capability Weak or Not Addressed. It must nonetheless be checked in Chapter 12 against security posture-management and DevOps/platform-engineering practice, which may already partially solve the infrastructure slice of this problem under a different name. **AI agent architectural governance (identity, permission, delegation):** Extension of Zero Trust principles into a business-authority context Zero Trust itself does not address. **Risk-tiered, architecture-integrated governance gates:** Integration of NIST/ISO risk methodology with COBIT-style decision rights and TOGAF-style architecture gates, not a new risk taxonomy. **Federated governance operating model spanning EA, AI, security, and transformation:** Integration, drawing structurally on data-mesh’s federated-computational-governance pattern and NIST CSF 2.0’s Govern function precedent. **AI-to-data lineage extension:** Extension of DAMA-DMBOK2’s lineage concept to cover model grounding, not a new lineage methodology. **Transformation-to-architecture-to-outcome traceability:** Integration of capability-based planning practice with architecture decision records, addressing a connective gap rather than inventing new transformation methodology.

Four areas identified in Chapter 1 are explicitly **not** carried forward as innovation claims. The landscape review shows them adequately handled elsewhere: a new AI risk taxonomy (NIST AI

RMF and ISO 23894 already provide this credibly); a new AI management-system standard (ISO 42001 already provides this and is achieving certification-body traction); a new general IT governance/decision-rights model (COBIT already provides this maturely); and a new information-security control catalogue for AI (ISO 27001, NIST CSF 2.0, and Zero Trust already provide the relevant primitives, even if not yet AI-labeled).

F. Revised Research Gap

The Chapter 1 gap stated: *no widely adopted framework integrates enterprise architecture, AI build governance, and digital transformation governance into one continuously assured, traceable structure*. The landscape review sharpens this rather than overturning it. The revision: **the gap is not the absence of AI governance, architecture method, or IT-governance content (all three exist in mature, standards-body form), but the absence of (a) AI as a first-class, continuously observable element of the enterprise architecture metamodel, and (b) a connective traceability and assurance structure that keeps business capability, architecture decision, AI risk tier, delegated authority, and runtime behavior mutually consistent over time. Without those two elements, the seventeen questions in the project's ultimate design goal cannot be answered consistently by any single existing framework, or by any realistic combination of them.** This is narrower than the Chapter 1 formulation, and it excludes territory (risk taxonomy, management-system design, general IT governance) that Chapter 1 left ambiguous.

G. Framework Boundary

The new framework WILL govern: the architectural representation of AI as a domain (what belongs in the enterprise architecture repository for models, prompts, RAG, agents, and their relationships to business capability); the traceability structure connecting strategy, capability, transformation initiative, architecture decision, AI system, risk tier, and outcome; risk-tiered, architecture-integrated governance gates for AI-enabled capabilities; architectural governance of AI agent authority, permission, and delegation; and the mechanism (telemetry, drift detection, conformance comparison) for continuous assurance that deployed architecture and AI behavior match what was approved.

The new framework WILL NOT attempt to replace: AI-specific risk assessment methodology (defers to NIST AI RMF / ISO 23894 / ISO 31000); organizational AI management-system requirements (defers to ISO 42001); general IT governance objectives and decision-rights theory (defers to COBIT / ISO 38500); information security controls, identity, and access management (defers to ISO 27001 / NIST CSF 2.0 / Zero Trust architecture); data quality, stewardship, and general lineage methodology (defers to DAMA-DMBOK2); or the core TOGAF ADM's business/data/application/technology architecture method (extends, does not replace, its metamodel and phase structure).

Chapter 2 Quality Test

Can the research problem be solved simply by using existing frameworks more effectively?

Partially. An enterprise that rigorously implemented TOGAF, COBIT, NIST AI RMF, and ISO 42001 together, with careful cross-referencing, would close most of the Category A and Category B gaps in Section D through better integration practice alone. That is an implementation and organizational-discipline problem, not a framework-absence problem, and no new framework is needed to solve it. Two things would still be missing, even with disciplined use of all four frameworks. The first is a shared metamodel that makes AI a first-class, jointly-referenceable architecture element across those frameworks. The second is a continuous conformance-assurance mechanism connecting architecture-as-approved to architecture-as-deployed to AI-behavior-as-observed. That residual — Category D in Section D, and the two elements named in the revised gap (Section F) — is the part of the research problem that existing frameworks, however well combined, do not currently provide a mechanism for. This is a materially narrower claim than the Chapter 1 formulation implied, and the framework design from Chapter 4 onward should be held to it.

Chapter 3. Framework Design Requirements

This chapter converts the revised research gap (Chapter 2, Section F) and the framework boundary (Chapter 2, Section G) into testable design requirements. No diagrams, layers, or metamodel are designed here; that is the subject of Chapter 4. Every requirement traces to a named gap from Chapter 1 or Chapter 2. Requirements that do not survive the scope test in Section 3.6 are recorded as rejected.

3.1–3.2 Requirements by Category

Each requirement is stated with its rationale, source, and validation method. “Related Framework Component” is left blank, as instructed, since architecture has not yet been designed.

Strategic Requirements

FR-01 — Strategy-to-AI-System Traceability. The framework shall enable traceability between enterprise objectives, business capabilities, transformation initiatives, AI use cases, deployed AI systems, and measurable outcomes. *Rationale:* AI investment can become disconnected from strategic priorities; Chapter 1 named this as ultimate design goal question 1–2. *Source:* Chapter 1 candidate gap; capability-based planning practice (Chapter 2, A.2). *Validation:* Case-study traceability analysis: can an auditor walk from a live AI system back to the strategic objective it serves, and back again, without gaps?

FR-02 — Transformation-to-Capability Traceability. The framework shall connect digital transformation initiatives to the specific business capabilities they are intended to change, including target maturity state. *Rationale:* Chapter 1, Section 3, identified transformation’s decentralization as straining EA’s traditional scope boundary. Capability-based planning (Chapter 2) provides the pattern, but no framework currently connects it to AI-specific execution. *Source:* Chapter 1 Section 3; Chapter 2 Integration Map. *Validation:* Expert review of whether the capability model resolves ambiguity in real transformation-portfolio decisions.

FR-03 — Outcome and Value Traceability. The framework shall require that major architecture and AI investments be linked to a defined, measurable business outcome, tracked from baseline to realized value. *Rationale:* Ultimate design goal question 16; avoids architecture governance becoming a compliance exercise disconnected from value. *Source:* Research terms of reference; Chapter 2 Section D, “transformation gaps.” *Validation:* Benefits-realization audit against a sample of AI investments in a case study.

Architecture Requirements

FR-04 — AI as a First-Class Architecture Domain. The framework shall define AI (models, prompts/context, RAG and grounding, agents, and their orchestration) as an explicit architecture domain within the enterprise architecture repository, positioned alongside (not replacing) business, data, application, and technology architecture. *Rationale:* Chapter 2 rated this the clearest “Not Explicitly Addressed” gap in TOGAF’s (The Open Group, 2022) core metamodel (Section B, row “AI architecture”). *Source:* Chapter 2 Gap Taxonomy, “architecture gaps”; Chapter 2 Candidate Innovation Area 1. *Validation:* Test whether the extended metamodel can

represent three real AI systems of different types (predictive model, RAG copilot, autonomous agent) without ad hoc extension.

FR-05 — Extension of, Not Replacement for, TOGAF’s Content Metamodel. Where the framework introduces new architecture concepts, it shall express them as an extension of TOGAF’s existing content metamodel and ADM phase structure, unless a specific incompatibility is identified and documented. *Rationale:* Chapter 2’s Framework Boundary states that the ADM’s business/data/application/technology method is inherited, not replaced. Duplicating TOGAF would fail the non-duplication requirement (project Section 7). *Source:* Chapter 2 Section G; Framework Design Log D-01. *Validation:* Direct comparison of the extended metamodel against the published TOGAF 10 content metamodel to confirm compatibility, once the primary text is reviewed (Assumptions Register item 1).

FR-06 — Reference Architecture Patterns for Common AI Configurations. The framework shall provide a small set of pre-approved reference architecture patterns (e.g., enterprise RAG, LLM gateway, human-in-the-loop agent) that solution teams can adopt without a full bespoke architecture review. *Rationale:* Reduces governance friction while preserving control quality, consistent with the “maximum governance effectiveness, minimum friction” design principle. *Source:* Research terms of reference; Chapter 2 Framework Boundary. *Validation:* Measure time-to-approval for pattern-conformant solutions versus bespoke solutions in a pilot.

FR-07 — AI-Aware Architecture Decision Records. The framework shall require that architecture decisions involving AI capture AI-specific decision criteria (model/vendor choice, risk tier, autonomy level, data sensitivity) using an extension of standard architecture decision record (ADR) practice, not a parallel documentation track. *Rationale:* Prevents AI decisions from being made and recorded outside the architecture governance system entirely, which Chapter 1 (Section 4, “organizational and accountability gaps”) identified as a live risk. *Source:* Chapter 1 Section 4; Chapter 2 Section B, “decision traceability” row. *Validation:* Sample audit of whether AI-related architecture decisions in a pilot organization are actually captured in the extended ADR format.

AI Requirements

FR-08 — Risk-Tiered AI Classification, Referencing Not Replacing Existing Taxonomies. The framework shall classify AI systems into risk tiers using a methodology explicitly compatible with, and cross-referenced to, NIST AI RMF (NIST, 2023) and ISO/IEC 23894 (ISO & IEC, 2023b), rather than an independently invented taxonomy. *Rationale:* Chapter 2 rated AI risk classification “Strong” under NIST AI RMF/ISO 23894; inventing a competing taxonomy would be duplicative (Candidate Innovation Areas, excluded list). *Source:* Chapter 2 Section E, excluded innovation claim; Chapter 2 Integration Map. *Validation:* Crosswalk table showing every framework risk tier maps cleanly to NIST AI RMF trustworthiness characteristics and ISO 23894 risk criteria.

FR-09 — Architecture-Integrated AI Governance Gates. The framework shall define governance gates for AI systems that are triggered by, and recorded within, architecture governance checkpoints (not a separate AI-only approval track), with gate intensity varying by risk tier. *Rationale:* Chapter 2 Section B rated “deployment authorization” Weak-to-Partial across all reviewed frameworks when AI-specific; this is a connective gap. *Source:* Chapter 2 Gap

Taxonomy, “governance gaps”; the research terms of reference. *Validation*: Pilot test whether low-risk AI use cases pass through fewer/faster gates than high-risk ones, evidencing risk-proportionality, not uniform bureaucracy.

FR-10 — AI Change Classification Tied to Architecture Re-Approval Thresholds. The framework shall classify AI system changes (model version, provider, prompt, RAG source, autonomy level, user population, purpose) into minor/material/major tiers, each with a defined re-approval requirement. *Rationale*: Without this, every AI change either requires full re-governance (excessive friction) or none (governance decay). Project Section 10, Stage 8 names this problem. *Source*: Research terms of reference, Stage 8; Chapter 1 Section 4, “lifecycle governance gaps.” *Validation*: Test the classification against real change logs from a deployed generative AI system to see if the tiering produces sensible, non-arbitrary re-approval triggers.

FR-11 — Explicit Human Oversight Points as Architecture Elements. The framework shall require that human oversight mechanisms (human-in-the-loop, human-on-the-loop approval points, override/kill-switch capability) be represented as explicit elements in the AI system’s architecture, not merely described in policy documents. *Rationale*: Ultimate design goal question 6–8 require the framework to answer what decisions/actions a system can take and at what autonomy level. Policy text alone cannot answer this. *Source*: Research terms of reference; Chapter 2 Zero Trust precedent. *Validation*: Confirm oversight points are traceable to specific architecture diagram elements in a pilot AI agent design.

FR-12 — Agent Authority and Delegation Boundaries as Architectural Constructs. The framework shall require that any AI agent’s permitted actions, tool access, spending/transaction limits, and escalation thresholds be explicitly modeled as part of its architecture, using identity and least-privilege concepts consistent with Zero Trust Architecture (NIST, 2020). *Rationale*: Chapter 2 rated AI agent governance Weak/Not-Addressed across nearly every reviewed framework except Zero Trust’s transferable principles (Section B). *Source*: Chapter 2 Candidate Innovation Area 4; the research terms of reference. *Validation*: Test whether the modeled authority boundary can be technically enforced (not just documented) in a reference implementation.

Data Requirements

FR-13 — AI Grounding Lineage. The framework shall extend standard data lineage practice (per DAMA-DMBOK2 (DAMA International, 2017)) to trace the relationship between a governed dataset or knowledge source, the model or RAG pipeline that uses it, and the outputs it grounds. *Rationale*: Chapter 2 Section B rated “AI lineage (model↔data↔prompt↔output)” Weak/Not-Addressed across every reviewed framework, including DAMA itself. *Source*: Chapter 2 Gap Taxonomy, “architecture gaps”; Candidate Innovation Area 7. *Validation*: Test whether a specific AI output can be traced back to its grounding data source within a defined time bound in a pilot RAG system.

FR-14 — Data Sensitivity Linked to AI Risk Tiering. The framework shall require that the data sensitivity classification of inputs used by an AI system be a direct input to that system’s risk tier (FR-08), not assessed independently. *Rationale*: Prevents a scenario where an AI system is classified low-risk on business-impact grounds while processing highly sensitive data, a known failure mode noted in AI-governance commentary. *Source*: Research terms of reference; Chapter

2 Integration Map (DAMA/data governance inheritance). *Validation*: Cross-check risk-tier outcomes against data-sensitivity classification for a sample of use cases to confirm no high-sensitivity/low-tier mismatches survive review.

Risk Requirements

FR-15 — Risk-Proportional Governance Intensity. The framework shall scale governance gate count, evidence requirements, and approval authority level in direct proportion to an AI system's assigned risk tier. *Rationale*: Direct response to the anti-bureaucracy requirement in the research terms of reference and Chapter 1's identification of "insufficient autonomy controls" alongside "over-governance" as twin failure modes. *Source*: Research terms of reference; Chapter 1 Section 4. *Validation*: Compare governance cycle time across risk tiers in a pilot; Tier 1 (minimal risk) should show materially lower overhead than Tier 3–4.

FR-16 — Control-to-Risk-to-Standard Mapping. The framework shall maintain an explicit, auditable mapping from each identified AI risk to the control(s) that mitigate it and the external standard(s) (NIST AI RMF, ISO 23894, ISO 27001 (ISO & IEC, 2022), etc.) those controls satisfy. *Rationale*: Supports ultimate design goal questions 9–11 (what risks, what policies, what controls) and prevents controls from being invented ad hoc per system. *Source*: Chapter 2 Integration Map; the research terms of reference. *Validation*: Confirm every Tier 3+ system in a pilot has a complete, non-empty control mapping before deployment authorization.

FR-17 — Material-Change and Risk-Tier-Change Escalation. The framework shall define an escalation path triggered automatically when a change classified under FR-10 as material/major, or a risk reassessment, would move a system to a higher risk tier. *Rationale*: Static point-in-time risk assessment is insufficient for systems that can drift in behavior or scope after deployment (Chapter 1 Section 2, "probabilism"). *Source*: Chapter 1 Section 2 and Section 4; Chapter 2 Section B, "model drift" and "AI assurance" rows. *Validation*: Simulate a risk-tier-triggering change in a pilot and confirm the escalation path fires without manual intervention to notice it.

Transformation Requirements

FR-18 — Portfolio-Level Architecture and AI Dependency Visibility. The framework shall make architecture and AI-system dependencies visible at the transformation-portfolio level, so investment sequencing decisions account for shared platform, data, and model dependencies. *Rationale*: Chapter 2 rated "transformation portfolio governance" as an integration point where digital-transformation practice is strong but disconnected from architecture visibility. *Source*: Chapter 2 Integration Map and Gap Taxonomy, "transformation gaps." *Validation*: Test whether portfolio prioritization decisions in a pilot correctly surface a shared-dependency conflict that would otherwise have been missed.

FR-19 — Benefits Realization Linked to Architecture and AI Investment Records. The framework shall require that benefit tracking (FR-03) reference the specific architecture and AI system records it depends on, rather than being maintained as an independent transformation-office artifact. *Rationale*: Avoids the fragmentation Chapter 1 named between architecture governance and transformation governance functions. *Source*: Chapter 1 Section 3; the research terms of reference. *Validation*: Confirm a sampled benefit record can be traced bidirectionally to its architecture/AI dependency and back.

FR-20 — Visibility Into, Not Ownership Of, Operating-Model and Workforce Change. The framework shall provide visibility hooks connecting architecture/AI changes to workforce and operating-model impact assessments owned by change-management functions, without the framework itself attempting to govern change management. *Rationale:* Applies the Section 3.6 scope test proactively. Workforce transformation governance belongs to organizational change management, not EA. But EA needs enough visibility to avoid deploying AI systems whose operating-model prerequisites are unmet. *Source:* Chapter 1 Section 3; Section 3.6 quality test (this chapter). *Validation:* Confirm the framework specifies only an interface/visibility requirement here, not a workforce-transformation methodology of its own.

Operational (Runtime) Requirements

FR-21 — Continuous Architecture-to-Deployment Conformance Checking. The framework shall define a mechanism for comparing the architecture and AI configuration actually running in production against the architecture and configuration that was approved, on an ongoing rather than point-in-time basis. *Rationale:* This is the single most consistently unaddressed capability identified in Chapter 2 (Section B, “continuous assurance” row), and the leading candidate for novel contribution (Chapter 2, Candidate Innovation Area 2). It is also the item most likely to fail the Chapter 12 anti-novelty test against DevOps/security posture-management practice, per the Assumptions Register. *Source:* Chapter 2 Section D, “runtime gaps”; Chapter 2 Framework Design Log D-03. *Validation:* Introduce an undocumented configuration change in a test environment and measure whether the mechanism detects the deviation, and within what latency.

FR-22 — Multi-Type Drift Detection. The framework shall define, at minimum, architecture drift, AI model/behavior drift, and data drift as distinct monitored categories, each with an owner and a response expectation. *Rationale:* the research terms of reference named at least nine drift types; Chapter 2 confirmed this capability is weak or absent across every reviewed framework. *Source:* Research terms of reference; Chapter 2 Section B. *Validation:* Confirm each drift category has a distinct, testable detection signal in a reference implementation, not a single undifferentiated “monitoring” catch-all.

FR-23 — Telemetry Interface Requirements, Not Telemetry Tooling. The framework shall specify *what* architectural and AI-behavioral evidence must be observable (the interface/requirement) without prescribing *which* monitoring, APM, or MLOps tooling must be used to produce it. *Rationale:* Directly enforces the Section 3.6 scope boundary. Telemetry tooling implementation is an engineering/MLOps concern; defining what EA governance needs to see is legitimately an EA concern. *Source:* Section 3.6 quality test; Chapter 2 Framework Boundary. *Validation:* Confirm the specification can be satisfied by at least two structurally different tooling stacks (e.g., a cloud-native observability stack and an MLOps-specific platform) without modification to the requirement itself.

Governance Requirements

FR-24 — Federated Decision Rights Without Committee Proliferation. The framework shall define a governance operating model in which architecture, AI, security, data, and transformation decision rights are federated under shared principles and escalation rules, rather than requiring a separate standing committee for each domain. *Rationale:* Chapter 2 Gap Taxonomy named “organizational gaps” as unresolved by any single reviewed framework; the research terms of reference warns against board proliferation. *Source:* Research terms of reference; Chapter 2

Integration Map (data-mesh federated-governance and NIST CSF 2.0 (NIST, 2024) Govern-function precedents). *Validation*: Compare the number of distinct approval bodies a Tier 2 AI use case must pass through under the federated model versus a naive “one committee per domain” baseline.

FR-25 — Explicit, Non-Orphanable Ownership. The framework shall require that every AI system, agent, and significant architecture component have a named accountable owner at all times, with no permitted “unowned” state. *Rationale*: Ultimate design goal question 3 (“who owns it?”); Chapter 1 named accountability gaps as a distinct cluster. *Source*: Chapter 1 Section 4, “organizational and accountability gaps”; the research terms of reference. *Validation*: Audit a live architecture/AI repository in a pilot for orphaned (ownerless) entries after framework adoption.

FR-26 — Exception and Waiver Governance. The framework shall define a formal, time-bound, and auditable process for granting exceptions to architecture or AI governance requirements, distinct from silent non-compliance. *Rationale*: Rigid frameworks accumulate silent workarounds; an explicit exception path is a known mitigation and is consistent with the project’s over-governance-avoidance principle. *Source*: Research terms of reference; Chapter 2 Section G. *Validation*: In a pilot review, confirm that exceptions are time-bound and appear in periodic governance reporting instead of persisting indefinitely.

Implementation Requirements

FR-27 — Minimum Viable Implementation Path. The framework shall define a reduced-scope implementation (a defined subset of principles, gates, roles, and artifacts) usable by a mid-sized organization without a large architecture department. *Rationale*: The research terms of reference requires this; a framework only usable by large enterprises fails a stated design goal. *Source*: Research terms of reference. *Validation*: Reserved for Chapter 14; test with a simulated mid-sized-organization case study.

FR-28 — Automation- and Policy-as-Code-Compatible Control Expression. Where a control can plausibly be expressed as a machine-checkable rule (naming conventions, approved model lists, data-residency constraints, agent permission ceilings), the framework shall express it in a form compatible with policy-as-code implementation, without mandating a specific tooling vendor. *Rationale*: Supports the continuous-assurance mechanism (FR-21–23) and the governance-speed-through-automation principle. *Source*: Research brief Sections 13 and 35; Chapter 2 Section C (security policy-as-code precedent). *Validation*: Confirm a sample of Tier 1–2 controls can be encoded as executable policy rules in a proof-of-concept without loss of governance intent.

3.4 Prioritization

Table 7.

ID	Priority	Complexity	Key Dependencies
FR-01	Mandatory	Medium	FR-02, FR-03
FR-02	Mandatory	Medium	FR-01
FR-03	Important	Medium	FR-01, FR-19

ID	Priority	Complexity	Key Dependencies
FR-04	Mandatory	High	FR-05
FR-05	Mandatory	High	—
FR-06	Important	Medium	FR-04
FR-07	Important	Low	FR-04, FR-05
FR-08	Mandatory	Medium	—
FR-09	Mandatory	High	FR-08, FR-04
FR-10	Mandatory	Medium	FR-08
FR-11	Mandatory	Medium	FR-04, FR-12
FR-12	Mandatory	High	FR-04
FR-13	Important	High	FR-04
FR-14	Important	Low	FR-08
FR-15	Mandatory	Medium	FR-08, FR-09
FR-16	Important	Medium	FR-08
FR-17	Mandatory	Medium	FR-10, FR-08
FR-18	Important	Medium	FR-02
FR-19	Important	Low	FR-03, FR-18
FR-20	Optional	Low	FR-02
FR-21	Mandatory	High	FR-04, FR-23
FR-22	Mandatory	High	FR-21
FR-23	Mandatory	Medium	FR-21
FR-24	Mandatory	High	FR-08, FR-09
FR-25	Mandatory	Low	FR-04
FR-26	Important	Low	FR-09, FR-24
FR-27	Mandatory	Medium	All above
FR-28	Optional	Medium	FR-21, FR-23

Complexity is a design-effort estimate, not a governance-burden estimate. FR-25 (ownership) is low-complexity to specify but organizationally hard to enforce, which is a Chapter 8/11 concern, not a Chapter 3 one.

3.5 Requirements Traceability Matrix

Table 8.

ID	Requirement (short)	Gap Addressed (Chapter 2 §D)	Existing Standard Leveraged	New Mechanism Needed?	Priority
FR-01	Strategy→AI traceability	Integration gaps	Capability-based planning	Yes	Mandatory
FR-02	Transformation→capability traceability	Transformation gaps	Capability-based planning	Yes	Mandatory
FR-03	Outcome/value traceability	Transformation gaps	DT practice	Yes	Important
FR-04	AI as first-class architecture domain	Architecture gaps	TOGAF (extended)	Yes	Mandatory
FR-05	Extend, don't replace, TOGAF metamodel	Architecture gaps	TOGAF	No (constraint, not new mechanism)	Mandatory
FR-06	Reference AI architecture patterns	Architecture gaps	—	Yes	Important

ID	Requirement (short)	Gap Addressed (Chapter 2 §D)	Existing Standard Leveraged	New Mechanism Needed?	Priority
FR-07	AI-aware ADRs	Governance gaps	TOGAF ADR practice (extended)	Partial	Important
FR-08	Risk-tiered AI classification	AI lifecycle gaps	NIST AI RMF / ISO 23894	No — reference existing	Mandatory
FR-09	Architecture-integrated AI gates	Governance gaps	COBIT EDM pattern (ISACA, 2018) + TOGAF	Yes	Mandatory
FR-10	AI change classification	AI lifecycle gaps	—	Yes	Mandatory
FR-11	Human oversight as architecture elements	AI lifecycle gaps	NIST AI RMF (Govern)	Yes	Mandatory
FR-12	Agent authority/delegation boundaries	Architecture gaps	Zero Trust (extended)	Yes	Mandatory
FR-13	AI grounding lineage	Architecture gaps	DAMA-DMBOK2 (extended)	Yes	Important
FR-14	Data sensitivity → risk tier link	Integration gaps	DAMA-DMBOK2	Partial	Important
FR-15	Risk-proportional governance intensity	Governance gaps	—	Yes	Mandatory
FR-16	Control-to-risk-to-standard mapping	Integration gaps	NIST/ISO/COBIT (referenced)	Partial	Important
FR-17	Material-change escalation	Runtime gaps	—	Yes	Mandatory
FR-18	Portfolio dependency visibility	Transformation gaps	DT practice	Yes	Important
FR-19	Benefits realization linkage	Transformation gaps	DT practice	Partial	Important
FR-20	Operating-model visibility hook	Organizational gaps	—	Yes (minimal)	Optional
FR-21	Continuous conformance checking	Runtime gaps	— (nearest: security posture mgmt)	Yes	Mandatory
FR-22	Multi-type drift detection	Runtime gaps	—	Yes	Mandatory
FR-23	Telemetry interface (not tooling)	Runtime gaps	Security/ITIL (AXELOS, 2019) monitoring practice	Yes	Mandatory
FR-24	Federated decision rights	Organizational gaps	COBIT + data mesh + NIST CSF 2.0 Govern	Yes	Mandatory
FR-25	Non-orphanable ownership	Organizational gaps	COBIT (RACI)	Partial	Mandatory
FR-26	Exception/waiver governance	Governance gaps	—	Yes	Important
FR-27	Minimum viable implementation	(Implementation, cross-cutting)	—	Yes	Mandatory
FR-28	Policy-as-code compatible controls	Runtime gaps	Security policy-as-code practice	Partial	Optional

3.6 Quality Test — Scope Discipline

Each requirement above was tested against: *does this belong in an Enterprise Architecture governance framework, or does it belong to software engineering, model engineering, cybersecurity operations, data-science implementation, or project management?* Several candidate requirements considered during drafting were rejected or narrowed on this basis. They are recorded here, per the continuity rules.

Rejected — belongs to ML/model engineering, not EA governance: “The framework shall specify fine-tuning hyperparameter standards” and “the framework shall define prompt-engineering style guidelines.” These are legitimate practices, but they are model-build engineering decisions. EA’s proper interest is limited to whether such changes trigger the change-classification and re-approval mechanism (FR-10), not how they are technically performed.

Rejected — belongs to software engineering: “The framework shall define code review and testing standards for AI-generated code.” Code quality practice belongs to software engineering. EA’s legitimate interest, retained as FR-07/FR-10, is limited to whether AI-generated code that materially changes a governed system triggers architecture review, not the code review method itself.

Rejected — belongs to security operations: “The framework shall define incident response runbooks for AI security incidents.” Incident response procedure is a security-operations matter (ISO 27001/NIST CSF territory, per Chapter 2’s Integration Map). EA’s legitimate interest, retained as part of FR-22/FR-23, is limited to ensuring AI-relevant incidents are visible to architecture governance, not owning the response process.

Rejected — belongs to project management: “The framework shall define AI project scheduling and resourcing templates.” This is delivery/project-management practice. Nothing in Chapter 1 or Chapter 2’s gap analysis identified project scheduling as an architecture governance gap, so no EA-scoped version of this requirement was retained at all.

Narrowed, not rejected: the original candidate “the framework shall govern workforce transformation and change management” was narrowed to FR-20 (visibility hook only). Chapter 1 Section 3 identifies workforce transformation as adjacent to, but not properly owned by, EA. Full ownership would duplicate organizational change-management practice that the research brief does not ask this framework to replace.

This boundary matters. A framework that silently absorbs software-engineering, MLOps, security-operations, or project-management scope stops being defensible as an *architecture governance* framework and becomes an unbounded catch-all — precisely the failure mode Chapter 2’s Quality Test warned against.

Chapter 3 Output

Final Design Requirements

Twenty-eight requirements (FR-01–FR-28) across nine categories, detailed above, each traced to a specific Chapter 1 or Chapter 2 finding.

Mandatory Requirements

FR-01, FR-02, FR-04, FR-05, FR-08, FR-09, FR-10, FR-11, FR-12, FR-15, FR-17, FR-21, FR-22, FR-23, FR-24, FR-25, FR-27: seventeen requirements without which the framework could not credibly claim to close the revised research gap (Chapter 2, Section F). Notably, the mandatory set spans all nine categories except Data and Implementation-automation, where FR-13/FR-14 and FR-28 were judged Important/Optional, not load-bearing.

Differentiating Requirements

The requirements least likely to be achievable by disciplined use of existing frameworks alone (i.e., those Chapter 2's Quality Test identified as the residual problem) are FR-04 (AI as first-class domain), FR-09 (architecture-integrated gates), FR-12 (agent authority as architecture), FR-13 (AI grounding lineage), FR-21 (continuous conformance), FR-22 (multi-type drift), and FR-24 (federated decision rights). These map directly to Chapter 2's Candidate Innovation Areas. They should receive the most design attention from Chapter 4 onward, and the most skeptical scrutiny in Chapter 12's anti-novelty test, especially FR-21/FR-22 given the open question about DevOps/security posture-management overlap.

Requirements Already Substantially Satisfied by Existing Standards

FR-08 (risk-tiered classification: NIST AI RMF/ISO 23894), FR-16 (control mapping: largely a referencing exercise across NIST/ISO/COBIT), and FR-25 (ownership: COBIT's RACI practice already covers this at the general IT-governance level) require the framework to *reference and integrate* rather than *invent*. These are retained as requirements because the *connection* to architecture is still missing, not because the underlying content is missing.

Requirements Requiring Genuinely New Mechanisms

FR-04, FR-06, FR-09 through FR-13, FR-15, FR-17, FR-18, FR-21 through FR-24, and FR-26 have no reviewed existing standard that substantially provides them (per the Traceability Matrix's "New Mechanism Needed?" column). This is a larger set than the "Differentiating Requirements" list above because some new mechanisms (e.g., FR-06 reference patterns, FR-18 portfolio visibility) are necessary engineering of the framework without themselves being the strongest novelty claims. That distinction is deliberate and will matter for the Chapter 12 novelty assessment.

Chapter 4. Conceptual Framework Architecture

This chapter translates the twenty-eight requirements from Chapter 3 into the framework’s first coherent structure: its fundamental shape, dimensions, architecture domains, AI Architecture definition, metamodel, and core model. Detailed processes, gates, roles, and artifacts follow in Chapters 5 to 9. No framework name is proposed at this stage, on the principle that naming should follow a matured conceptual architecture, not precede it.

4.1 Fundamental Framework Structure

The four candidate structures were tested against the Chapter 3 requirement set, specifically against which FRs each structure could represent without distortion.

A **purely layered structure** (Strategy → Capability → Transformation → Architecture → AI → Technology) represents FR-01, FR-02, and FR-03 (traceability requirements) cleanly, and it communicates well to executive audiences. It struggles, though, to represent FR-21–FR-23 (continuous conformance and drift detection) without implying a one-directional, periodic cadence — precisely the assumption Chapter 1 identified as a structural limitation of TOGAF’s ADM (The Open Group, 2022). A layered diagram is a static picture; the requirements it must satisfy include a non-static, always-on capability.

A **purely lifecycle structure** (Intent → Design → Build → Deploy → Operate → Adapt) represents the AI Requirements category (FR-08–FR-17) well, and it matches the research brief’s own language for AI build governance. But it offers no natural place for architecture domain structure (FR-04–FR-07) or governance operating-model concepts (FR-24–FR-26); both can only be bolted on as annotations to lifecycle stages instead of first-class structure.

A **pure matrix structure** (architecture domains × lifecycle stages × governance) can represent domain/stage/risk-tier intersections precisely, which is attractive for FR-09 and FR-15. A full matrix across five architecture domains, roughly nine lifecycle stages, and multiple risk tiers, however, produces a structure that is comprehensive on paper and unusable in practice. That is a documented risk, given the project’s warning against over-governance (Section 35) and the anticipated Chief Enterprise Architect reviewer critique in Chapter 11. The matrix is also weak at representing feedback loops (FR-17, FR-21), since a matrix cell does not naturally point back to an earlier cell.

A **pure control-system structure** (Intent → Execution → Telemetry → Assurance → Adaptation) directly represents the requirements Chapter 3 flagged as most differentiating (FR-21, FR-22, FR-17), and it matches the “adaptive system, not sequential methodology” ambition in the research brief (Section 28). Alone, however, it has no natural place for architecture domain structure or strategy-to-capability traceability. A control loop shows *that* something is being observed and corrected, not *what* the enterprise’s architecture actually consists of.

None of the four pure structures, alone, covers the mandatory requirement set from Chapter 3 without distortion or omission. This is not a case for arbitrarily combining all four decoratively. It is evidence that the requirements themselves ask three structurally different questions: what exists (a largely static reference structure), how it gets built (a staged sequence), and how it stays

correct (a continuous loop). A defensible framework should answer each with the structure suited to it, sharing one underlying metamodel instead of three disconnected models.

Structural decision: the framework is organized as **one shared metamodel expressed through three complementary views**, not as three separate frameworks stitched together:

1. **The Structural View** (Section 4.2–4.4): a largely static reference structure of dimensions and architecture domains, answering “what does the enterprise’s AI-enabled architecture consist of, and why does it exist.” This view does the work a layered structure does well.
2. **The Lifecycle View** (deferred in detail to Chapter 5, previewed here as the AI Lifecycle dimension): a staged sequence answering “how does a specific AI capability move from idea to retirement.” This view does the work a lifecycle structure does well.
3. **The Control View** (deferred in detail to Chapter 7, previewed here as the Assurance dimension): a continuous loop answering “is what’s running still what was approved, and what happens if not.” This view does the work a control-system structure does well.

All three views reference the same entities (Section 4.5), so a given AI system has exactly one record, visible through three lenses, not three duplicated records. This is a direct design response to the artifact-duplication risk the research brief warns against (Section 19, “avoid excessive documentation”; Section 9.3, artifact non-duplication). The decision is logged as D-11 below. It should be tested in Chapter 11 for whether practitioners actually experience it as “one model, three views” or as “three frameworks in a trenchcoat” — that distinction can only be judged empirically, not asserted here.

4.2 Framework Dimensions

The research brief proposes eight dimensions (Enterprise Intent, Enterprise Capabilities, Architecture, AI Lifecycle, Governance, Risk, Transformation, Continuous Assurance). Section 8 of the research brief instructs testing these rather than accepting them. Tested against the FR set, two of the eight collapse into cross-cutting properties, and one is scoped down rather than dropped.

Governance does not survive as a standing dimension. Every mandatory requirement that mentions governance (FR-09, FR-15, FR-24, FR-25, FR-26) attaches governance to something else: a capability decision, an architecture decision, a lifecycle gate, a runtime exception. There is no case in the requirement set of governance existing independently of the thing it governs. Treating it as a sixth peer dimension risks exactly the outcome Chapter 2 flagged when reviewing NIST CSF 2.0 (NIST, 2024): that standard’s 2024 addition of a Govern *function* is a precedent for governance-as-cross-cutting-concern, not governance-as-a-separate-layer. Governance is therefore retained as a **cross-cutting property** attached to entities in every other dimension, not as a sixth box on a diagram. Every capability, architecture domain, lifecycle stage, and assurance signal carries governance metadata: owner, decision rights, policy reference.

Risk does not survive as a standing dimension, for the same structural reason. FR-08 and FR-14 attach risk tiering to AI systems; FR-15 makes governance intensity a function of that tier; FR-17 escalates based on tier changes. Risk is always a property *of* something (an AI system, a

data flow, an architecture decision), never a freestanding layer. It is retained as the second **cross-cutting property**, propagated alongside governance.

Transformation survives, but at reduced depth. FR-20 narrowed the framework’s ownership of transformation and workforce change to a visibility interface, not full governance (Chapter 3, Section 3.6). Transformation is retained as a dimension because FR-02, FR-03, FR-18, and FR-19 need it as a first-class connective node between capability and architecture. It is documented as a **shallow-governance dimension**: the framework records and traces transformation initiatives without attempting to govern transformation delivery methodology, which correctly remains with change-management practice per Chapter 2’s Framework Boundary.

Enterprise Intent, Enterprise Capabilities, Architecture, AI Lifecycle, and Continuous Assurance survive as full dimensions, each required by at least one Mandatory FR. Each also represents a distinct kind of question (why / what capability / what structure / what sequence / what ongoing truth), and none can be collapsed into another without losing a Mandatory requirement’s home.

Revised dimension set: five functional dimensions, two cross-cutting properties.

Table 9. Framework Dimensions

#	Dimension / Property	Type	Primary FRs Served	Why It Survives / Why It Was Reduced
1	Enterprise Intent	Functional dimension	FR-01, FR-03	Anchors the “why”; no other dimension can substitute
2	Capability & Value	Functional dimension	FR-01, FR-02, FR-03, FR-18, FR-19	Connective node between strategy and execution; explicitly merges the project’s separate “Capabilities” and “Value” language since Chapter 3 treats value/outcome as inseparable from the capability it realizes
3	Architecture	Functional dimension	FR-04–FR-07, FR-12, FR-13	The structural core; detailed in 4.3–4.4
4	AI Lifecycle	Functional dimension	FR-08–FR-17	Sequential build-governance backbone; detailed in Chapter 5
5	Transformation (shallow)	Functional dimension, reduced depth	FR-02, FR-18–FR-20	Retained but explicitly interface-only per FR-20
—	Continuous Assurance	Functional dimension (control loop, not a static layer)	FR-21–FR-23	Detailed in Chapter 7; structurally different from the other four (loop, not layer)
—	Governance	Cross-cutting property	FR-09, FR-15, FR-24–FR-26	Collapsed from standing dimension; attaches to every entity in every dimension
—	Risk	Cross-cutting property	FR-08, FR-14–FR-17	Collapsed from standing dimension; attaches to every entity in every dimension

This yields six functional dimensions plus two cross-cutting properties, a reduction from the project’s original eight peer dimensions that is argued rather than asserted. Continuous Assurance is counted among the six but listed separately, because it behaves as a loop rather than a layer.

4.3 Architecture Domains

The research brief proposes extending TOGAF’s four domains (Business, Data, Application, Technology) into a ten-element chain: Business → Experience → Process → Data → AI → Application → Integration → Platform → Technology → Security. Tested against FR-04–FR-07

and against Chapter 2’s finding that TOGAF’s existing four domains remain structurally sound for their original scope (Chapter 1, Section 5), several of the proposed additions do not warrant peer status alongside Business, Data, Application, and Technology.

Experience Architecture and Process Architecture are, in mainstream TOGAF practice, already represented as artifacts *within* Business Architecture (value streams, business processes, stakeholder maps). Elevating them to independent peer domains would fragment a discipline TOGAF already handles adequately (Chapter 2, Section A.1: TOGAF rated Strong on business architecture) without closing any FR-identified gap. They are retained as **named sub-disciplines within Business Architecture**, not as primary domains.

Integration Architecture and Platform Architecture are retained as **cross-cutting supporting disciplines** rather than primary domains. In practice they exist to connect and host the primary domains: an integration layer or a platform has no purpose except in service of the data, application, AI, or technology components it carries. This positioning also avoids a specific overlap risk. FR-06’s reference patterns (LLM gateway, enterprise RAG) are simultaneously integration concerns and AI Architecture concerns, and forcing a single ownership boundary between two peer domains would recreate the kind of artificial boundary dispute Chapter 2 warned against. Treating Integration and Platform as cross-cutting means an LLM gateway pattern is authored once, within AI Architecture, with a defined interface to general Integration Architecture practice for its non-AI-specific aspects (authentication, network routing) — one authoritative home, not two competing ones.

Security Architecture is retained as **cross-cutting**, not a primary domain. That is consistent with Chapter 2’s finding that Zero Trust’s own operating principle (“security embedded everywhere,” not appended afterward) is the correct model. It is equally consistent with how ISO/IEC 27001 (ISO & IEC, 2022) and NIST CSF 2.0 already treat security as a concern that touches every other domain rather than a walled-off sixth pillar.

AI Architecture is added as a genuinely new **primary domain**, the one addition from the proposed ten-element chain that survives scrutiny. Chapter 2’s landscape review found no existing framework, TOGAF included, that treats it as first-class (this is FR-04, Mandatory).

Revised domain structure: five primary domains, three cross-cutting supporting disciplines.

Table 10. Architecture Domains

Category	Domains	Status
Primary	Business Architecture (incl. Experience, Process as sub-disciplines)	Inherited from TOGAF, extended
Primary	Data Architecture	Inherited from TOGAF, extended for AI grounding lineage (FR-13)
Primary	AI Architecture	New — the framework’s central structural contribution (FR-04)
Primary	Application Architecture	Inherited from TOGAF
Primary	Technology Architecture	Inherited from TOGAF
Cross-cutting	Integration Architecture	Supporting discipline spanning Data, AI, Application, Technology
Cross-cutting	Platform Architecture	Supporting discipline spanning Data, AI, Application, Technology
Cross-cutting	Security Architecture	Supporting discipline spanning all five primary

Category	Domains	Status
		domains

This is a materially more conservative domain structure than the research brief’s proposed ten-element chain. The reduction is the direct product of applying the same non-duplication and minimal-necessary-structure discipline used in Chapter 3’s FR scope test.

4.4 Defining AI Architecture

Definition (proposed). *AI Architecture is the enterprise architecture domain concerned with the structural decisions that determine how AI capability (predictive, generative, or agentic) is sourced, hosted, grounded, orchestrated, observed, and constrained across the enterprise, and how those decisions connect to business capability, data governance, and risk. It governs the existence, composition, and authority boundaries of AI components at the level an architecture review board would reasonably need to see; it does not govern the internal engineering of those components.*

This definition is narrower than the research brief’s illustrative component list (Section 9), which mixes architectural concerns (model routing, RAG existence, agent authority) with what are, on inspection, solution-engineering or data-science concerns (prompt wording, fine-tuning hyperparameters). Chapter 3’s FR-04 through FR-13 drew this line implicitly; Section 4.4 makes it explicit and exhaustive.

Table 11. Defining AI Architecture

AI Architecture Component	EA-Level Concern (governed by this framework)	Solution/Engineering-Level Concern (out of scope, per Chapter 3 §3.6)
Foundation models	Which model families/vendors the enterprise authorizes; portfolio and concentration risk	Internal model weights, training methodology
Hosting	Deployment pattern (SaaS API / private hosting / on-prem) and its sovereignty, cost, and continuity implications	Infrastructure sizing, autoscaling configuration
Model routing / gateway	Existence and governance policy of a gateway pattern (which models it may route to, what it logs)	Gateway code implementation, load-balancing algorithm
RAG / grounding	Existence of a grounding relationship between an AI system and a governed data source (the lineage link itself, FR-13)	Chunking strategy, embedding model choice, retrieval-ranking tuning
Vector databases / embeddings	That a vector store exists, what governed data flows into it, and its data-retention/classification policy	Specific vector database product, indexing algorithm, embedding dimensionality
Agent architecture	Existence of an agent, its authorized tools, action boundaries, and escalation thresholds (FR-12)	Agent’s internal planning/reasoning implementation, orchestration code
Orchestration / multi-agent composition	Which agents may invoke which other agents or tools, and the authority each delegation carries	Specific orchestration framework or library used
Knowledge systems	Which governed knowledge sources an AI system may draw on	Internal indexing/search implementation
Inference infrastructure	Not an EA-level concern in itself; visible only through the Platform cross-cutting discipline	Entirely engineering/platform-operations scope
AI observability	<i>What</i> must be observable to satisfy FR-23 (the interface requirement)	<i>Which</i> monitoring/APM/MLOps tool produces it
Guardrails	Existence and risk-tier binding of guardrail categories (e.g., “Tier 3 systems require content-safety and prompt-injection guardrails”)	The specific classifier, filter, or library implementing a guardrail

AI Architecture Component	EA-Level Concern (governed by this framework)	Solution/Engineering-Level Concern (out of scope, per Chapter 3 §3.6)
Evaluation	Existence and pass/fail binding of evaluation gates required before deployment (FR-09)	Specific benchmark suite, evaluation dataset composition
AI security	Policy requirement that AI-specific security concerns (prompt injection, model exfiltration) are addressed, as a cross-cutting Security Architecture concern applied to AI	Specific security control implementation (belongs to Security Architecture / ISO 27001 practice, per Chapter 2 Integration Map)
AI identity	Existence of a registered machine identity for each agent/system, consistent with Zero Trust principles (FR-12)	Specific IAM platform configuration
Human oversight	Existence and placement of oversight points in the architecture (FR-11)	UI/workflow implementation of the oversight interaction

The pattern across every row is consistent: **AI Architecture governs the existence, connection, and authority boundary of a component; it does not govern the component's internal construction.** This is the same boundary TOGAF already draws between enterprise and solution architecture for conventional applications. AI Architecture is not exempt from it merely because the underlying technology is newer. The point is worth stating, since the temptation to let AI governance sprawl into engineering territory is exactly the failure mode Chapter 3's quality test was designed to catch.

4.5 Framework Metamodel — Version 1

The entities below are drawn from the research brief's proposed list (Section 26), each retained, merged, or not instantiated as a separate entity based on the Chapter 3 requirement set. **Transformation** and **Regulation** are treated as entities, but with the shallow/reference-only status established in Section 4.2 and Chapter 2's Framework Boundary respectively: the framework references regulation and transformation initiatives without attempting to model their internal structure.

Core relationships (Entity → Relationship → Entity)

Strategy → sets → Objective Objective → requires → Capability Objective → realized through → Outcome Stakeholder → sponsors → Initiative Stakeholder → owns → Capability Capability → realized by → Value Stream Value Stream → composed of → Process Initiative → targets → Capability Initiative → is an instance of → Transformation Initiative → delivers → Product Product → enabled by → Application Product → enabled by → AI System Application → hosted on → Platform Application → exposes → API Application → consumes → Service AI Use Case → qualifies into → AI System (*the pre-approval concept becomes the governed entity only after passing FR-09 gates*) AI System → uses → Model AI System → uses → Agent AI System → grounded by → Data Product Data Product → composed of → Dataset Dataset → classified by → Policy (*data sensitivity, FR-14*) Model → trained or fine-tuned on → Dataset Agent → invokes → Tool Agent → constrained by → Policy (*authority boundary, FR-12*) Policy → implements → Control Policy → informed by → Regulation Control → mitigates → Risk Risk → attaches to → AI System (*and, by the same relationship type, to Application and Platform — Risk is a cross-cutting property realized as a relationship, not a domain-specific one*) Decision → authorizes → AI System (*and Application/Initiative — the governance-gate event, FR-09*) Decision → made by → Stakeholder (*in an accountable role*) Decision → references → Risk

Decision → references → Control Outcome → measured by → Metric Outcome → realizes → Objective (*closes the strategy loop, FR-03*) Metric → monitors → AI System (*and Application/Capability — the telemetry relationship underlying FR-21–FR-23*)

Diagram C — Core Metamodel

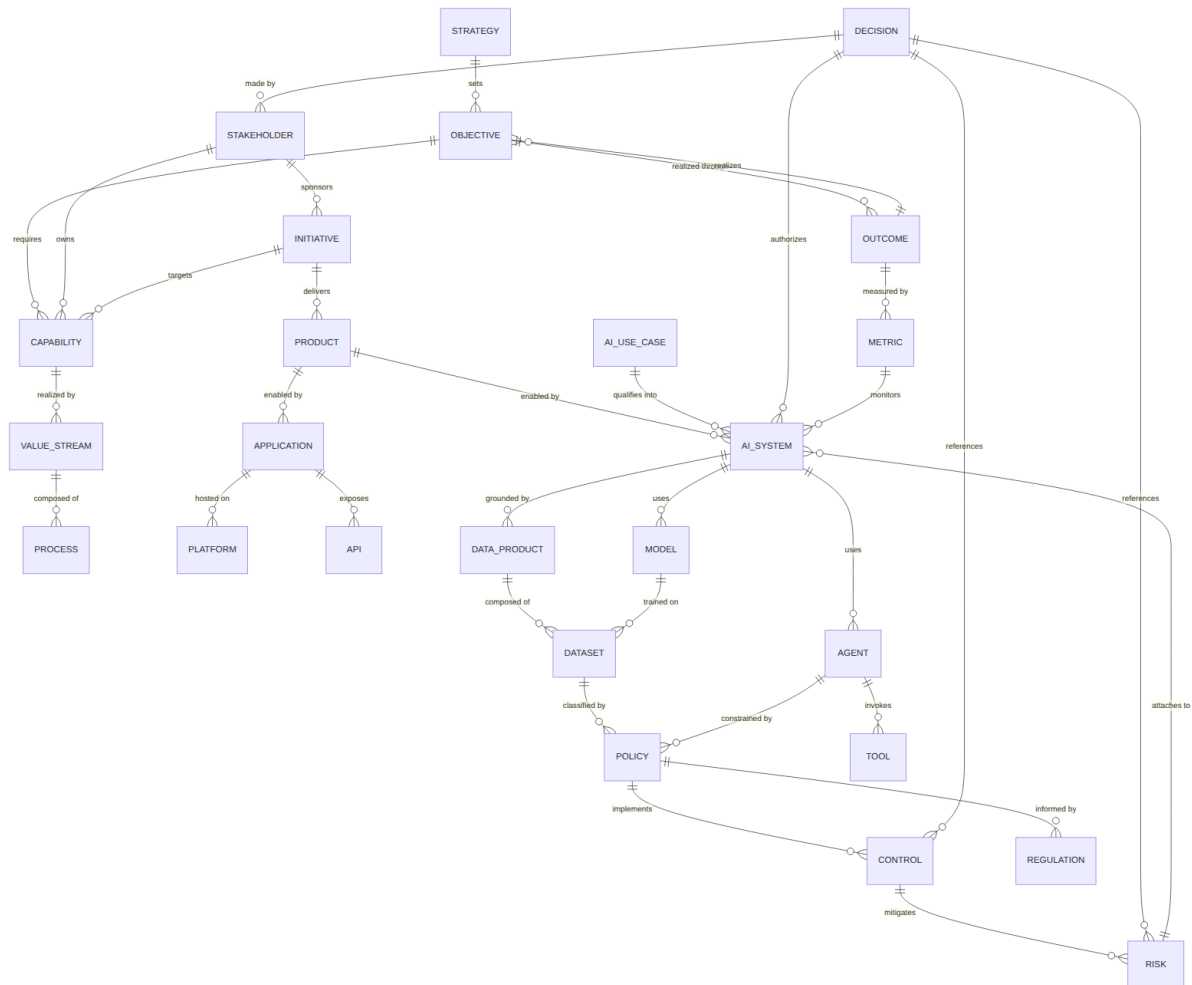


Figure 1. Core metamodel: entities and relationships

(Rendered as an entity-relationship diagram for compactness. Cardinalities shown are illustrative of the dominant pattern; most relationships are many-to-many in practice, e.g., an AI System may be grounded by multiple Data Products, and a Data Product may ground multiple AI Systems.)

4.6 Framework Core Model

The research brief's illustrative core model (Intent, Capability, Architecture, AI System, Risk, Control, Outcome, Assurance; 8 concepts) is a reasonable starting hypothesis, but it was tested rather than adopted directly, per the research brief's own instruction not to accept it automatically.

“Architecture” is removed as a standalone Core Model concept. It is too broad to serve as a minimal, testable unit. In the metamodel above, “architecture” is not itself an entity; it is expressed through Application, AI System, Platform, and the relationships among them. Including an undifferentiated “Architecture” node in the Core Model would be definitionally circular (the Core Model would contain “the thing the framework governs,” instead of the specific entity types that make governance operational).

“Decision” is added. Every mandatory requirement that involves a gate, an approval, an exception, or an authority boundary (FR-09, FR-11, FR-12, FR-17, FR-24, FR-26) resolves to a recorded Decision entity. Without a Decision concept, the Core Model has Risk and Control but no mechanism connecting them to an accountable act of authorization. That is precisely the “who approved it” question (ultimate design goal, question 12) the framework exists to answer. Decision is judged more load-bearing than the generic “Architecture” concept it replaces.

“Capability” is retained — without it, AI System has no connection back to Enterprise Intent, and FR-01/FR-02 (both Mandatory) fail immediately.

“AI System” is retained as the specific governed unit, not generalized to an abstract “Governed Entity,” because the framework’s differentiating purpose (per Chapter 2’s revised research gap) is AI governance integration. The Core Model should reflect what makes this framework distinct, not the generic EA content it inherits from TOGAF. Non-AI entities (Application, Platform) remain in the full metamodel but are not core model-level, consistent with the framework extending rather than replacing TOGAF’s existing, adequate treatment of them.

Revised Core Model (8 concepts, composition changed from the research brief’s example):

Intent · Capability · AI System · Decision · Risk · Control · Outcome · Assurance

Each is tested for indispensability: remove any one, and a Mandatory FR becomes unanswerable. Remove Intent → FR-01 fails. Remove Capability → FR-01/FR-02 fail. Remove AI System → the framework has nothing to govern. Remove Decision → FR-09/FR-12/FR-24 fail (no recorded authorization). Remove Risk → FR-08/FR-15/FR-17 fail. Remove Control → FR-16 fails and Decision has nothing to reference as mitigation evidence. Remove Outcome → FR-03 fails and ultimate design goal question 16 is unanswerable. Remove Assurance → FR-21/FR-22/FR-23 fail and the framework degrades to a periodic-documentation model, reproducing the exact TOGAF limitation Chapter 1 identified.

4.7 Diagrams

Diagram A — Overall Conceptual Framework

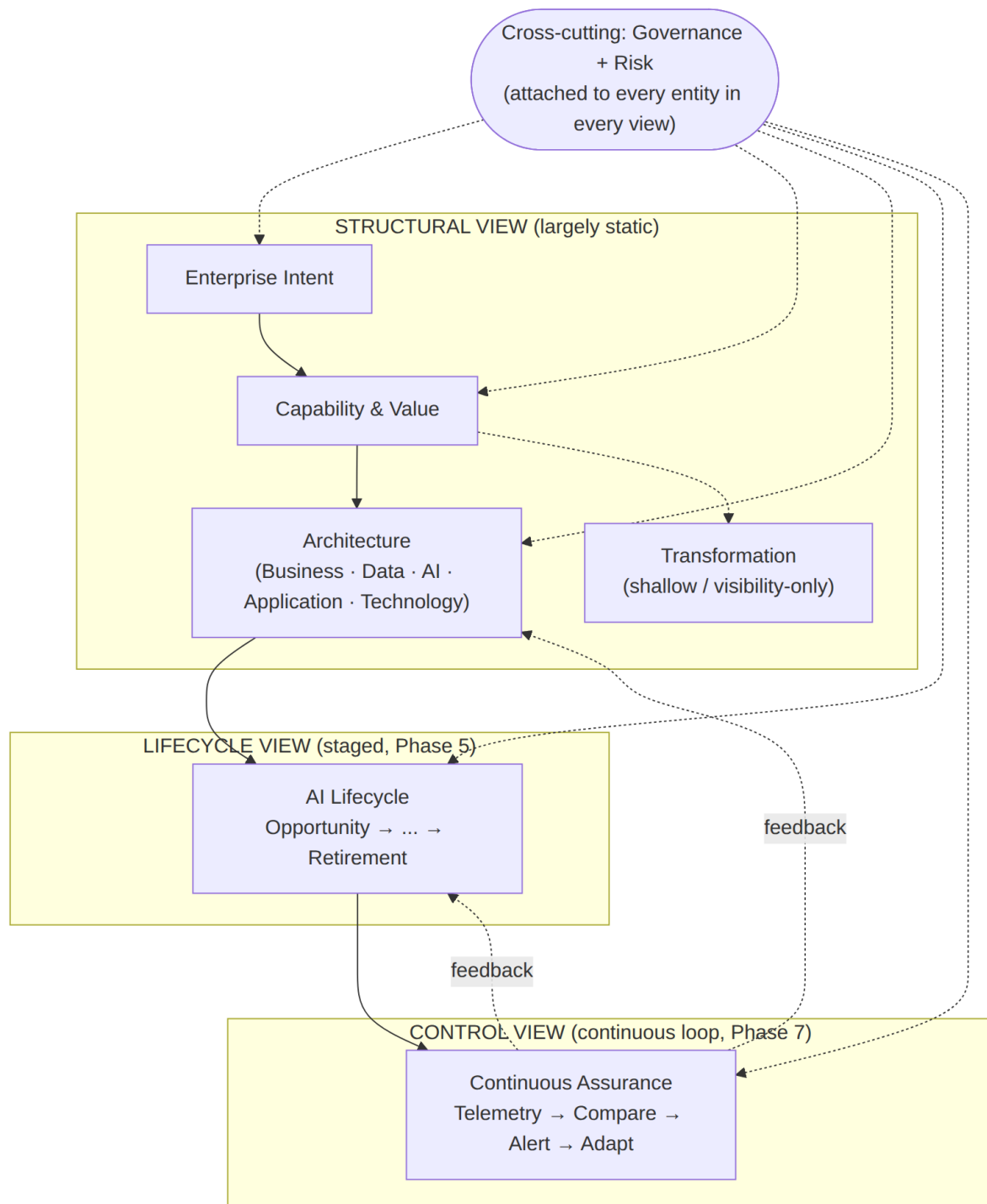


Figure 2. Overall conceptual framework

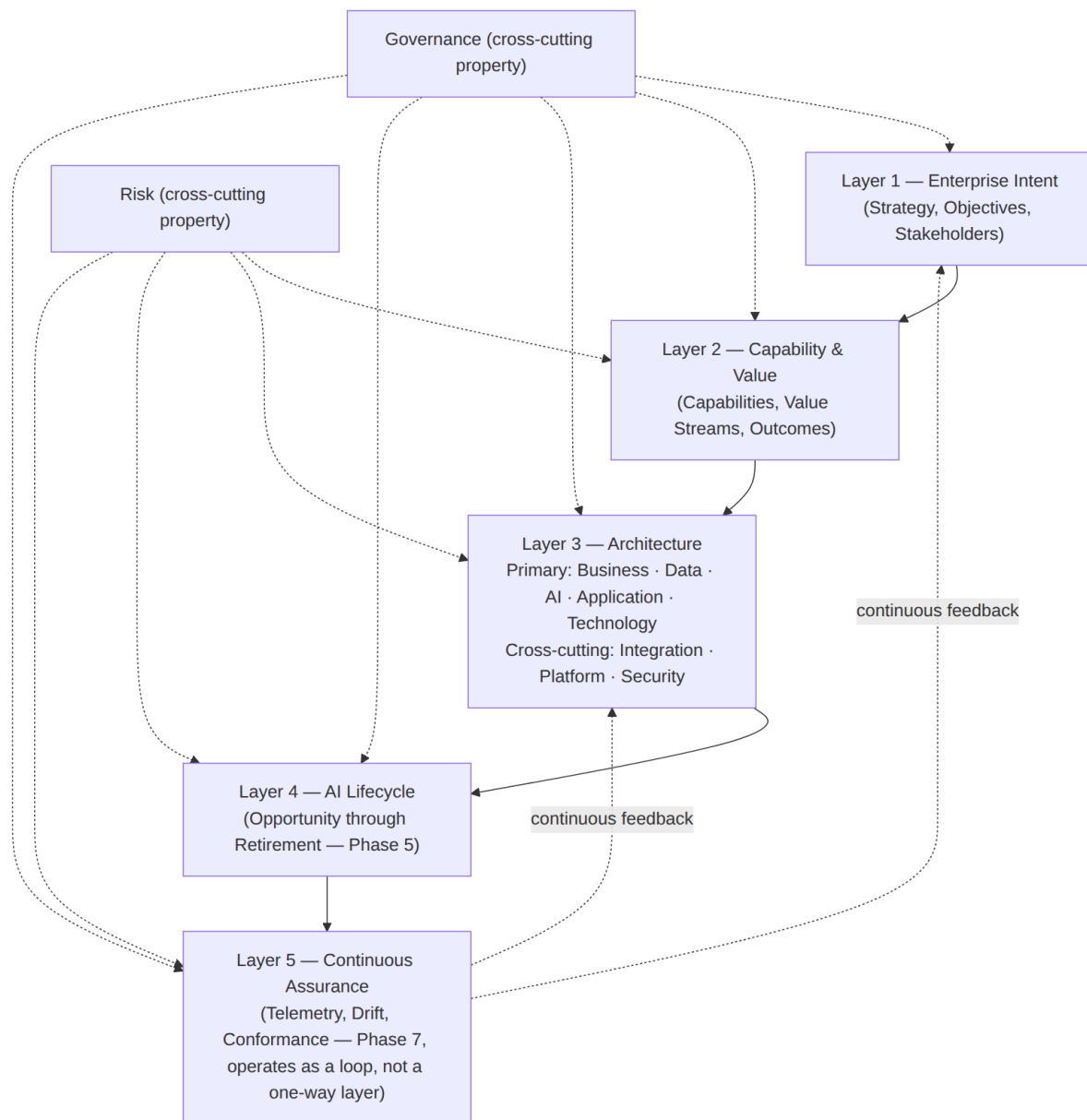
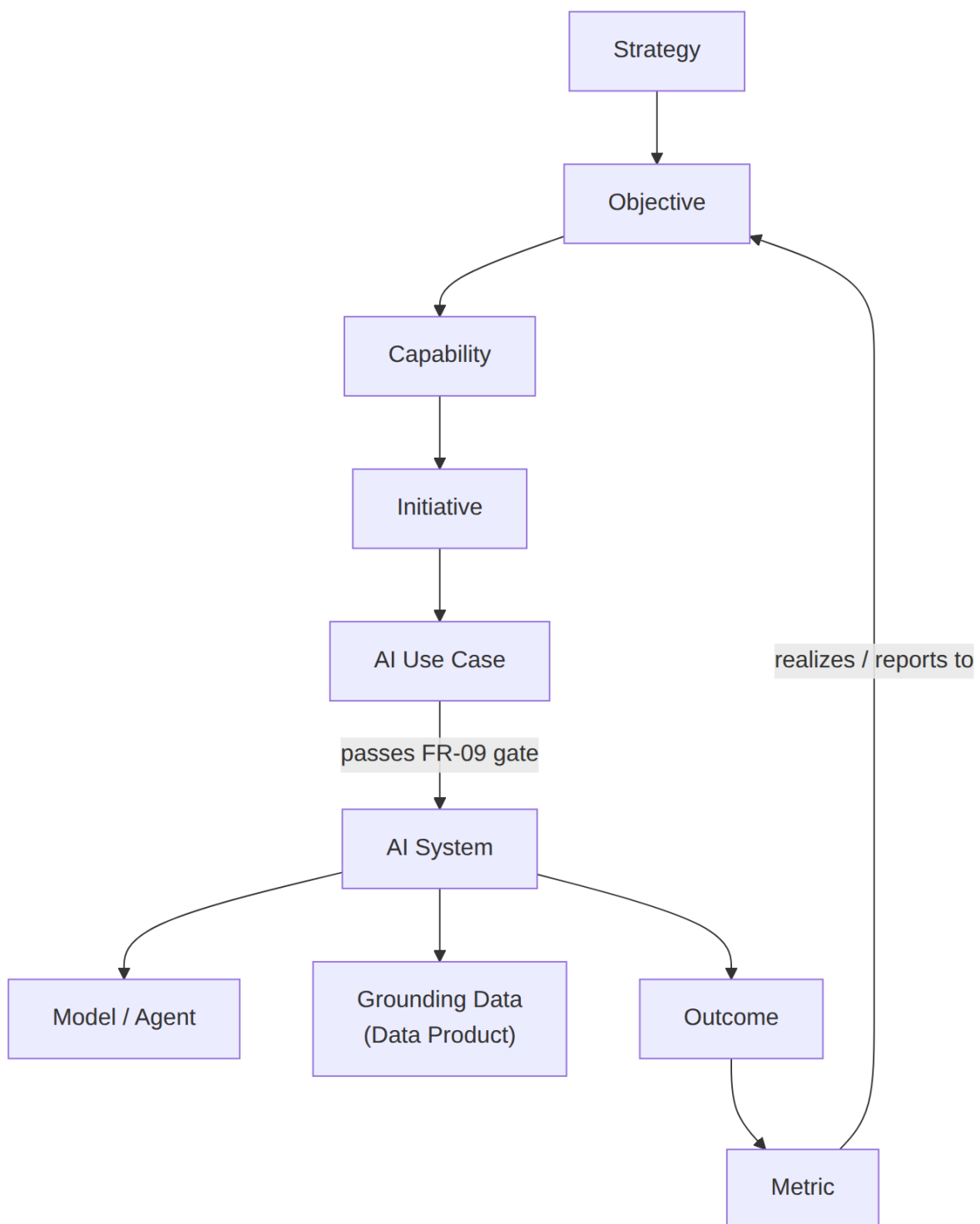
Diagram B — Framework Layers**Figure 3. Framework layers**

Diagram B shows Layer 5 feeding back into Layers 1 and 3 with a dashed loop instead of terminating the diagram, to avoid visually implying the periodic, one-directional cadence Chapter 1 identified as a structural limitation in conventional layered EA diagrams.

Diagram D — Strategy-to-Technology Traceability**Figure 4.** Strategy-to-technology traceability chain

(Diagram C — Core Metamodel — is provided in Section 4.5 above.)

Chapter 4 in Summary

Framework Architecture Version 0.1

One shared metamodel (Section 4.5) expressed through three complementary views (Structural, Lifecycle, Control) rather than three separate frameworks; six functional dimensions (Enterprise Intent, Capability & Value, Architecture, AI Lifecycle, Transformation [shallow], Continuous Assurance) plus two cross-cutting properties (Governance, Risk); five primary architecture domains (Business, Data, AI, Application, Technology) plus three cross-cutting supporting disciplines (Integration, Platform, Security); an 8-concept core model (Intent, Capability, AI System, Decision, Risk, Control, Outcome, Assurance) that differs from the research brief's illustrative core model by replacing the ambiguous "Architecture" node with the more load-bearing "Decision" node.

Core Components

Intent, Capability, AI System, Decision, Risk, Control, Outcome, Assurance (the Core Model, Section 4.6). None of these eight can be removed without a Mandatory FR becoming unanswerable.

Supporting Components

Value Stream, Process, Initiative, Product, Application, Platform, API, Service, Data Product, Dataset, Model, Agent, Tool, Policy, Regulation, Metric — present in the full metamodel (Section 4.5), necessary for completeness and for inheriting TOGAF's existing adequate treatment of non-AI architecture, but not core model-level.

Cross-Cutting Components

Governance and Risk (Section 4.2) attach to entities across all dimensions rather than existing as standing dimensions themselves; Integration, Platform, and Security Architecture (Section 4.3) attach across the five primary architecture domains instead of standing as peer domains.

Proposed Metamodel

As specified in Section 4.5 and Diagram C, with the caveat that cardinalities shown are illustrative of the dominant pattern, not exhaustively modeled (a full formal metamodel with complete cardinality and constraint specification is deferred to later, more implementation-oriented work beyond this research program's current chapter).

Design Decisions

Recorded in the Framework Design Log below (D-11 through D-15).

Questions Requiring Validation

Recorded in Open Research Questions and the Assumptions Register below.

Chapter 5. AI Build Governance Model

This chapter details the AI Lifecycle dimension and the risk and governance cross-cutting properties previewed in Chapter 4. It populates the Core Model concepts of Decision, Risk, Control, and Assurance with operational content: a governed lifecycle, a risk-scoring methodology, a gate model, an assurance model, and a change-classification model. The roles named in the responsibility assignment at Section 5.7 are provisional; the operating model and committee structure are settled in Chapter 8.

5.1 The Governed AI Lifecycle

The research brief offers two overlapping stage lists: a 9-stage list (Opportunity → Retirement, Section 10) and a 12-stage list (Opportunity → Retirement via separate Risk Assessment, Data Readiness, Validation, Runtime Operation, and Monitoring stages, Section 5.1). Reconciling the two instead of choosing one, three changes were made, each logged as a design decision.

Runtime Operation and Monitoring are merged into one concurrent state, not two sequential stages. Chapter 1’s critique of conventional EA was specifically that periodic, phase-gated thinking is a poor fit for AI systems whose behavior can change between reviews. Placing “Monitoring” as a numbered stage *after* “Runtime Operation” would silently reintroduce that same periodic assumption. The two are combined into a single stage that begins at deployment and continues, concurrently, until retirement. This is also where the Lifecycle View (this chapter) hands off to the Control View (Chapter 7).

Risk Classification is retained as a discrete stage, but it is re-triggered elsewhere and is not a one-time event. This reconciles with Chapter 4’s decision (D-12) to treat Risk as a cross-cutting property rather than a standing dimension. The lifecycle needs one identifiable point where a tier is first assigned, but Section 5.6 (Change Classification) and Section 5.5 (Assurance) both define explicit triggers that re-open Risk Classification. Risk is a property that travels with the AI System entity (per the Chapter 4 metamodel’s `RISK ||--o{ AI_SYSTEM : "attaches to"` relationship) and is re-evaluated; it is not a stage visited once and closed.

Loop-back edges are made explicit, not implied. Change (Stage 9) and the concurrent Monitoring state (Stage 8) can both re-open earlier stages depending on what is found. This is stated structurally instead of in an appendix note, because a lifecycle diagram that shows only forward arrows silently reintroduces the linear-cycle assumption Chapter 1 identified as a structural limitation.

Revised lifecycle (eleven stages, numbered 0–10):

Table 12. The Governed AI Lifecycle

Stage	Name	Core Question	Primary Core-Model Concept(s) Engaged
S0	Opportunity	Should this problem use AI at all — and if so, is it the right kind (Section 22 build/buy/fine-tune/RAG/API/SaaS/no-AI question)?	Intent, Capability
S1	Use-Case Qualification	Is there genuine business value, feasibility, and a	Capability, Outcome

Stage	Name	Core Question	Primary Core-Model Concept(s) Engaged
		plausible data path?	
S2	Risk Classification (initial)	What risk tier does this use case provisionally carry?	Risk
S3	Architecture Decision	What architecture (model, hosting, RAG, agent design, integration) will deliver this, and does it conform to reference patterns (FR-06)?	AI System, Decision
S4	Data Readiness	Is the grounding/training data governed, lineage-traceable (FR-13), and classified for sensitivity (FR-14)?	AI System, Control
S5	Build / Configure / Acquire	Construct, fine-tune, integrate, or procure — governed at the architecture boundary only (Chapter 4, Section 4.4), not the engineering detail	AI System
S6	AI Assurance (Validation)	Does the system perform as intended and as safely as its risk tier requires?	Control, Risk
S7	Deployment Authorization	Is this system authorized for production, by whom, on what evidence?	Decision, Control
S8	Runtime Operation & Continuous Monitoring	Is the system still behaving, still compliant, still valuable — on an ongoing basis?	Assurance, Outcome
S9	Change	Has something changed enough to require re-entry into an earlier stage?	Decision, Risk
S10	Retirement	Has the system been decommissioned with evidence preserved?	Decision, Control

IRGF — the Intent-to-Runtime Governance Lifecycle

Eleven governed stages, five decision gates, and an outer band in which the Control Plane compares deployed state against approved state without pause

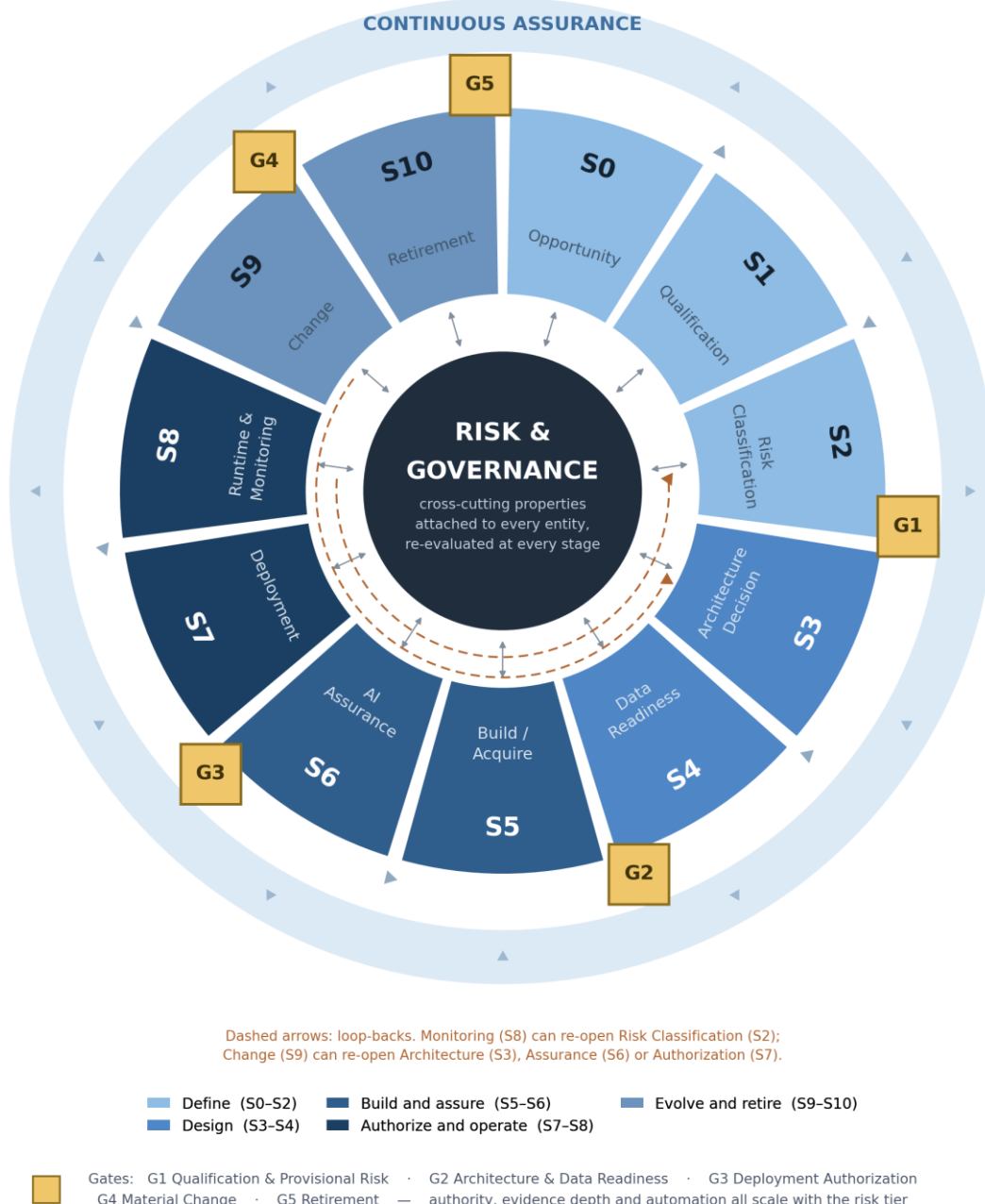


Figure 5. The IRGF lifecycle wheel: eleven stages, five gates, and Risk and Governance at the hub

Loop-back structure: S8 (Monitoring) can trigger re-entry to S2 (if observed behavior suggests the risk tier was wrong or has changed), S6 (if assurance thresholds are breached), or S9 (Change, if the trigger is a planned modification instead of an observed drift). S9 itself routes to S2, S3, S6, or S7 depending on the change classification defined in Section 5.6 — a minor change may re-enter no earlier stage at all, while a major change re-enters as though from S2.

5.2 Governance by AI Build Type

The eleven-stage lifecycle applies to every AI build type, but the *weight* of each stage differs substantially by type. Applying identical rigor to a vendor-embedded AI feature and a multi-agent system would violate FR-15 (risk-proportional governance) as surely as under-governing the latter would.

Table 13. Governance by AI Build Type

Build Type	S3 (Architecture) Weight	S4 (Data Readiness) Weight	S5 (Build) Governance Focus	S6 (Assurance) Emphasis	Distinguishing Governance Concern
Traditional ML / predictive models	Moderate — model selection and hosting decision	High — training data governance is central	Feature engineering and training pipeline conformance to data governance	Accuracy, bias, explainability	Well-precedented by existing model-risk-management practice (Chapter 2)
Externally consumed AI APIs (foundation model APIs)	High — vendor and data-flow decision is the architecture decision	Moderate — governs what leaves the enterprise boundary, not training data	Integration and prompt-boundary conformance	Output quality, data leakage, vendor dependency	Vendor concentration and model-update risk (Chapter 1, “porous system boundary”)
Generative AI (general purpose)	High	Moderate	Prompt/context architecture, guardrail configuration	Hallucination, toxicity, bias, adversarial resilience	Non-determinism (Chapter 1, Section 2)
RAG-augmented GenAI	High — grounding architecture is the core decision	High — FR-13 lineage is central	Retrieval/grounding pipeline conformance	Faithfulness to grounding source, stale-knowledge detection	AI-to-data lineage (FR-13) is the dominant concern
Fine-tuned models	High — training-data and IP implications	High	Training-data governance and change classification (fine-tuning is treated as a major change, Section 5.6)	Regression testing against base-model behavior	Overlaps model-risk-management practice most closely
AI copilots (human-paired assistance)	Moderate	Moderate	Human-in-the-loop interface as an explicit architecture element (FR-11)	Usability of oversight, not just technical accuracy	Human oversight point placement (FR-11)
AI agents (single, tool-using)	High — authority/permission boundary is the core decision	Moderate–High	Agent identity, tool access list, action boundary (FR-12)	Adversarial resilience, containment testing	Autonomy and delegation (FR-12) dominate
Multi-agent systems	Highest — inter-agent delegation and composition	High	Orchestration boundary and delegation chains (FR-12 extended)	Emergent-behavior and interaction testing between agents	Compounding autonomy risk; weakest existing standards coverage (Chapter 2)
Embedded vendor AI (AI features bundled inside purchased SaaS)	Low — architecture decision is largely procurement, not design	Low–Moderate — governs data shared with the vendor feature	Configuration and entitlement review, not construction	Confirm the feature’s risk tier based on its actual use, not the vendor’s marketing description	Risk of being governed as “just a feature toggle” when it materially affects decisions — a known blind spot named in Chapter 1, Section 4

The embedded-vendor-AI row deserves a specific caution. Because S3–S5 are lightweight for this type, there is a real risk that S2 (Risk Classification) is skipped entirely on the assumption that

“it’s just a vendor feature.” The framework treats this as a governance failure mode to design against, not a corner case — S2 applies to every build type without exception, however light S3–S5 are.

5.3 AI Risk Classification

The research brief lists twelve candidate risk dimensions (autonomy, decision impact, reversibility, data sensitivity, financial impact, safety, legal exposure, stakeholder impact, model uncertainty, external exposure, scale, systemic impact) and warns against copying a regulatory tier scheme directly. Several of these twelve correlate strongly enough in practice that scoring them independently would create false precision without added discriminating power. They are consolidated into five composite dimensions, each scored 1 (low) to 4 (high):

Table 14. AI Risk Classification

Composite Dimension	Subsumes	Scoring Anchor (1 → 4)
D1 — Decision Consequence Severity	Financial impact, safety, legal exposure, stakeholder impact	1 = negligible if wrong; 4 = severe/irreversible harm to people, safety, or major financial/legal exposure if wrong
D2 — Autonomy	Autonomy	1 = advisory only, human makes every decision; 4 = system acts without human review before effect
D3 — Reversibility Deficit	Reversibility	1 = trivially and immediately reversible; 4 = effectively irreversible or reversible only at severe cost
D4 — Exposure & Scale	External exposure, scale, systemic impact	1 = internal, single-user, isolated; 4 = external-facing, large population, or cross-system cascading potential
D5 — Data Sensitivity & Model Uncertainty	Data sensitivity, model uncertainty, explainability need	1 = non-sensitive data, well-understood/interpretable behavior; 4 = highly sensitive data or poorly-understood/uninterpretable behavior

Why five, not twelve: scoring each of the twelve original dimensions independently would require assessors to make fine distinctions (e.g., between “financial impact” and “legal exposure”) that are rarely independent in practice. An AI decision severe enough to cause major financial harm is very often also the one carrying major legal exposure. Collapsing correlated dimensions into composites is a considered precision-versus-usability tradeoff, consistent with the anti-bureaucracy principle (Section 35); it should be tested empirically in Chapter 13 for whether it loses meaningful discriminating power in edge cases (logged as an open question below).

Scoring methodology — a two-axis matrix, not a weighted sum

A straight sum or average across five dimensions would allow a single catastrophic factor to be diluted by four low scores — for example, a system with D1=4 (severe consequence if wrong) but low scores elsewhere could average out to a misleadingly moderate score. Risk-matrix practice (probability × impact) is adapted here, with **Impact** and **Control Deficit** as the two axes instead of probability and impact. For AI-specific risk, the more decision-relevant question is not “how likely is failure” but “how much human control exists over whether a failure becomes consequential.”

Impact Score = average(D1, D4, D5), rounded up. **Control Deficit Score** = average(D2, D3), rounded up. High autonomy combined with low reversibility is the combination that most sharply

reduces an enterprise’s ability to intervene before harm occurs, so this pairing is scored together instead of being folded into the same average as D1/D4/D5.

Table 15. Scoring methodology — a two-axis matrix, not a weighted sum

Impact ↓ / Control Deficit →	1	2	3	4
1	Tier 1	Tier 1	Tier 2	Tier 2
2	Tier 1	Tier 2	Tier 2	Tier 3
3	Tier 2	Tier 2	Tier 3	Tier 4
4	Tier 3	Tier 3	Tier 4	Tier 4

Safety-override rule: regardless of the matrix outcome, any system with D1 = 4 (catastrophic decision consequence) *or* the specific combination D2 = 4 and D3 = 4 (fully autonomous and effectively irreversible) is floored at Tier 3 minimum. This exists precisely to prevent averaging from masking one severe factor, a known weakness of naive composite scoring, and it responds directly to the “insufficient autonomy controls” gap named in Chapter 1.

On tier count: four tiers are retained, matching the research brief’s illustrative Tier 1 (Minimal), Tier 2 (Limited), Tier 3 (High), Tier 4 (Critical/Autonomous) labels — but this is a reasoned choice, not an assumption. Four tiers is judged, on the same practical-usability logic used to consolidate twelve dimensions to five, to be roughly the finest granularity assessors can apply consistently without significant inter-rater disagreement. That claim is not empirically tested here and is logged as an assumption requiring validation (Section 5.3 does not have access to inter-rater reliability data; Chapter 13’s expert validation is the right place to test it).

Relationship to NIST AI RMF (NIST, 2023) / ISO 23894 (ISO & IEC, 2023b) (per Chapter 2’s Integration Map): this scoring methodology is the framework’s own architectural risk-tiering mechanism. It does not replace NIST AI RMF’s trustworthiness characteristics or ISO 23894’s risk process, which remain the reference methodology an organization’s broader AI risk management should follow. The relationship is this: NIST/ISO answer “how do we think about AI risk generally,” and this Tier 1–4 output is what the *architecture governance gates* (Section 5.4) key off of, cross-referenced back to NIST/ISO risk documentation as evidence (FR-16).

5.4 Governance Gates

The research brief’s illustrative G0–G10 (eleven gates, Section 20) was tested against the eleven-stage lifecycle and found to over-specify. Several proposed gates (e.g., a separate gate for “AI Necessity” distinct from “Strategic Alignment”) do not correspond to a point where a distinct accountable decision actually needs to be recorded; they can be folded into an adjacent gate’s entry criteria. Five gates are retained as decision-critical points, each with risk-tier-dependent intensity rather than a fixed procedure. This applies FR-15 and the adaptive-governance requirement (Section 20) directly.

G1 — Qualification & Provisional Risk Gate (after S1–S2)

Purpose: Confirm that a business justification exists and assign a provisional risk tier before architecture work begins. *Entry criteria:* Use-case description, expected value, D1–D5 scoring inputs available at this early stage (may be provisional). *Required evidence:* Use-case canvas,

provisional risk score. *Decision authority*: Tier 1–2: capability/product owner, self-service. Tier 3–4: AI Governance Lead review required. *Possible outcomes*: Approve / Approve with Conditions / Remediate / Reject. *Automation potential*: High for Tier 1–2 (rule-based provisional scoring can auto-approve); low for Tier 3–4. *Risk-tier applicability*: Applies to all tiers; intensity varies sharply.

G2 — Architecture & Data Readiness Gate (after S3–S4)

Purpose: Confirm the proposed architecture (model/hosting/RAG/agent design) and data sources are governed, lineage-traceable, and conform to reference patterns or justify departure from them. *Entry criteria*: Architecture decision record (FR-07), data lineage documentation (FR-13), updated risk score reflecting now-known architecture detail. *Required evidence*: ADR, data classification record, reference-pattern conformance check or documented exception. *Decision authority*: Tier 1–2: Architect self-certification against pattern library. Tier 3–4: Architecture Review Board / AI Governance body joint sign-off. *Possible outcomes*: Approve / Approve with Conditions / Remediate / Escalate / Reject. *Automation potential*: Moderate. Pattern conformance is a strong policy-as-code candidate (FR-28) for Tier 1–2; Tier 3–4 requires human judgment on novel architecture. *Risk-tier applicability*: Applies to all tiers; Tier 1 may pass through largely automated if a pre-approved reference pattern (FR-06) is used unmodified.

G3 — Deployment Authorization Gate (after S6)

Purpose: Authorize production deployment based on assurance evidence. *Entry criteria*: Completed AI Assurance results (Section 5.5), control mapping (FR-16), confirmed risk tier. *Required evidence*: Assurance test results, control-to-risk mapping, named accountable owner (FR-25). *Decision authority*: Tier 1: system owner. Tier 2: system owner plus AI Governance Lead notification. Tier 3: AI Governance Council/body approval. Tier 4: senior executive or board-level approval, consistent with the accountability the risk tier implies. *Possible outcomes*: Approve / Approve with Conditions / Remediate / Escalate / Reject. *Automation potential*: Low for Tier 3–4 (deliberately human); moderate for Tier 1–2 where automated evidence checks can pre-clear routine cases for a human owner's final sign-off. *Risk-tier applicability*: The single most risk-differentiated gate — this is where governance intensity should diverge most sharply between tiers.

G4 — Material Change Gate (triggered by S9)

Purpose: Re-authorize a system whose change (Section 5.6) is classified material or major. *Entry criteria*: Change classification record, delta assurance evidence (not necessarily a full re-run of every S6 test, scoped to what the change affects). *Required evidence*: Change record, affected-control re-verification, updated risk score if the change touches D1–D5. *Decision authority*: Scales with the *post-change* risk tier, using the same authority table as G3. *Possible outcomes*: Approve / Approve with Conditions / Remediate / Escalate / Reject / Retire (if the change reveals the system should be decommissioned instead). *Automation potential*: High for minor changes (may not require this gate at all, per Section 5.6); low for major changes. *Risk-tier applicability*: Only triggered when Section 5.6 classifies a change as material or major; minor changes bypass G4 entirely by design.

G5 — Retirement Gate (S10)

Purpose: Confirm orderly decommissioning. *Entry criteria:* Decommissioning plan covering data retention, access removal, and knowledge preservation. *Required evidence:* Retention/deletion confirmation, access-revocation confirmation, archived audit evidence sufficient to answer future questions about a system no longer running. *Decision authority:* System owner plus Data Owner sign-off; Tier 3–4 additionally requires AI Governance Lead confirmation that no open risk or compliance obligations remain. *Possible outcomes:* Approve / Approve with Conditions / Escalate. *Automation potential:* Moderate. Access-revocation and deletion confirmation are strong automation candidates; the “knowledge preservation” judgment is not. *Risk-tier applicability:* Applies to all tiers; evidentiary depth scales with tier.

This is a reduction from the research brief’s eleven illustrative gates to five, each mapped to an actual decision point instead of every lifecycle-stage transition. It answers Chapter 3’s anti-bureaucracy requirement (FR-15) and the research brief’s instruction not to accept G0–G10 automatically.

5.5 AI Assurance Model

Five distinct assurance types are defined, each answering a different question and owned by a different accountable party. This responds directly to Chapter 4’s open question about whether the Core Model’s single “Assurance” concept bundles several logically separable ideas. It does, and this section is the decomposition Chapter 4 deferred to this point.

Table 16. AI Assurance Model

Assurance Type	Question It Answers	Owner	Primary Evidence	Feeds Into
Technical AI Evaluation	Does the system perform correctly and safely at the model/output level (accuracy, robustness, hallucination, bias, toxicity, adversarial resilience)?	AI/ML engineering, with EA visibility only (per Chapter 4, Section 4.4 scope boundary)	Evaluation test results against defined benchmarks/thresholds	G3, G4
Architecture Assurance	Does the deployed configuration still match the approved architecture (FR-21)?	Architecture function, increasingly automated (Chapter 7)	Conformance-check output, drift alerts	S8 monitoring, G4 trigger
Risk Assurance	Is the assigned risk tier still accurate, and are the mapped controls still effective (FR-16, FR-17)?	Risk & Compliance function	Control-effectiveness testing, tier re-scoring	G4 trigger, escalation (FR-17)
Compliance Assurance	Does the system still conform to applicable policy and regulation?	Risk & Compliance function, referencing NIST AI RMF / ISO 23894 / ISO 42001 (ISO & IEC, 2023a) evidence (Chapter 2 Integration Map)	Compliance checklist, external audit evidence where applicable	G3, G4, regulatory reporting
Business Outcome Assurance	Is the system still delivering the value it was authorized to deliver (FR-03)?	Business/Capability Owner	Outcome metrics against the baseline set at S1	Benefits realization (Chapter 6), retirement decision (S10)

Interaction: the five types are not independent silos feeding a single dashboard; they interact through the Decision entity. A Technical AI Evaluation failure (e.g., a hallucination-rate threshold breach detected during Monitoring) triggers Risk Assurance re-scoring, which may trigger the FR-17 escalation path, which may re-open G4. A Business Outcome Assurance failure (the system is compliant and technically sound but not delivering value) does not trigger a risk escalation at all — it is a legitimate, lower-urgency path toward S9 Change or S10 Retirement, and treating it identically to a safety failure would misallocate governance urgency. Safety and compliance failures escalate through Risk Assurance; value failures route through ordinary Change or Retirement. This differential routing keeps the highest-urgency escalation path reserved for genuine risk events, consistent with the anti-over-governance principle.

5.6 AI Change Classification

Table 17.

Change Type	Default Classification	Rationale	Gate Re-Triggered	Special Rule
Model version update (same provider/family)	Minor (Tier 1–2) / Material (Tier 3–4)	Low-tier systems tolerate routine vendor updates; high-tier systems need re-validation because behavior can shift unpredictably (Chapter 1, “probabilism”)	None (Tier 1–2) / G4 (Tier 3–4)	—
Provider or model-family change	Major	Materially different architecture, vendor dependency, and behavior profile	G2 + G4	Treated as a new Architecture Decision, not merely a change
Prompt/system-instruction change (substantive)	Material	Can materially alter behavior even with the same model	G4 (scoped re-test of affected assurance areas)	—
Prompt/formatting change (cosmetic)	Minor	No behavioral risk implication	None	—
RAG source addition/change	Material if new source sensitivity $\geq$ current classification; Minor otherwise	Directly affects FR-13 lineage and FR-14 sensitivity linkage	G4 (Material case only)	Always re-runs the lineage check regardless of classification
Training data change (fine-tuning)	Major	Materially changes model behavior; closest analogue to model-risk-management’s revalidation trigger (Chapter 2)	G2 + G4	Full re-validation, not scoped
Autonomy increase (e.g., human-in-loop $\rightarrow$ human-on-loop, or new unsupervised action)	Major, always	Directly changes D2 (Autonomy), which can shift the risk tier independent of any other factor	S2 re-entry + G2 + G4	Never eligible for automated/self-service classification, regardless of prior tier
Tool/permission addition for an agent	Material to Major, scaled by the tool’s blast radius	Directly engages FR-12	G4 (Material) or S2 + G2 + G4 (Major, e.g. financial transaction authority)	A tool granting irreversible or financial actions is always Major
User population change (e.g., internal pilot $\rightarrow$	Major	Directly changes D4 (Exposure & Scale), likely	S2 re-entry + G3	Treated as re-entering Deployment

Change Type	Default Classification	Rationale	Gate Re-Triggered	Special Rule
external release)		shifting risk tier		Authorization, not just Change
Business purpose change (repurposing)	Major — treated as a new Opportunity	The system is, in governance terms, a new use case wearing old infrastructure	S0 re-entry	Existing technical assurance evidence may be reused where genuinely still applicable, but the qualification and risk classification are redone from S1–S2

The pattern across the table: **classification tracks which composite risk dimension (D1–D5) the change plausibly moves, not the technical size of the change.** A one-line prompt edit that increases autonomy is Major; a large-looking model version bump within an already-low-tier system can be Minor. Keeping the change model risk-driven rather than effort-driven is a design choice, since effort-based change classification is what allows governance-relevant changes to slip through as “just a small update,” a failure mode named in Chapter 1.

5.7 AI Lifecycle RACI (Provisional)

Roles are provisional placeholders pending Chapter 8’s full operating-model design. They are used here only to test whether the lifecycle and gate model produce a workable accountability pattern, not to finalize role definitions.

Table 18. AI Lifecycle RACI (Provisional)

Stage/Gate	Business Sponsor	AI/Enterprise Architect	Data Owner	AI System Owner	Risk & Compliance Lead	Security Architect	Accountable Governance Body (tier-dependent)
S0 Opportunity	A	C	I	—	I	—	I
S1 Qualification	R	C	C	—	I	—	I (Tier 3–4)
S2 Risk Classification	C	C	C	—	A/R	C	I (Tier 3–4)
G1	A	C	I	—	R (Tier 3–4)	—	C (Tier 3–4)
S3 Architecture	I	A/R	C	C	I	C	C (Tier 3–4)
S4 Data Readiness	I	C	A/R	C	I	C	—
G2	I	R	R	C	C	C	A (Tier 3–4)
S5 Build/Configure/Acquire	I	C	I	A/R	—	C	—
S6 Assurance	I	C	I	R	A	C	—
G3	I	C	I	R	A	C	A (Tier 3–4, supersedes System Owner)
S7 Deployment Authorization	I	I	I	R	C	C	A
S8 Runtime & Monitoring	I	I	I	R	A	C	I
S9 Change	I	C	C	R	C	C	I (Major changes)
G4	I	C	C	R	A	C	A (Material/Major,

Stage/Gate	Business Sponsor	AI/Enterprise Architect	Data Owner	AI System Owner	Risk & Compliance Lead	Security Architect	Accountable Governance Body (tier-dependent)
							Tier 3–4)
S10 Retirement	I	I	A/R	R	C	I	C (Tier 3–4)
G5	I	I	A	R	C	I	C (Tier 3–4)

R = Responsible, A = Accountable, C = Consulted, I = Informed. Where a cell shows a tier qualifier, the role's involvement is contingent on risk tier per Sections 5.3–5.4; blank cells indicate no defined involvement at that stage for that role.

5.8 Minimum Required Evidence

Table 19.

Gate	Minimum Required Evidence
G1	Use-case canvas; provisional D1–D5 scores; named business sponsor
G2	Architecture Decision Record (FR-07); data lineage and sensitivity classification (FR-13/FR-14); reference-pattern conformance result or documented exception
G3	Technical AI Evaluation results; control-to-risk mapping (FR-16); named accountable AI System Owner (FR-25); human oversight point confirmation for Tier 2+ (FR-11)
G4	Change classification record; scoped re-assurance evidence for affected areas; updated risk score where D1–D5 are affected
G5	Data retention/deletion confirmation; access-revocation confirmation; archived audit trail sufficient for post-retirement inquiry

This is a minimum, not an exhaustive documentation set, consistent with the project's instruction to avoid documentation for its own sake (Section 19). The full artifact templates (owner, update frequency, related artifacts) are Chapter 9's job.

Chapter 5 in Summary

AI Governance Lifecycle

Eleven stages (S0–S10, Section 5.1), reconciling the research brief's two proposed stage lists, with Runtime Operation and Monitoring merged into a single concurrent state and explicit loop-back edges from Monitoring and Change into earlier stages.

AI Risk Model

Five composite dimensions (D1–D5, consolidated from the project's twelve) scored 1–4, combined via an Impact × Control-Deficit matrix (not a weighted sum) into four risk tiers, with a safety-override floor rule preventing a single catastrophic factor from being averaged away. Cross-referenced to NIST AI RMF and ISO 23894, and not a replacement for them.

Governance Gate Model

Five gates (G1–G5, Section 5.4), reduced from the project’s eleven illustrative gates, each with risk-tier-scaled decision authority and automation potential, and each mapped to an accountable decision point rather than every stage transition.

AI Change Model

Ten change types classified by which composite risk dimension they plausibly move (Section 5.6), driving which gate(s) are re-triggered; the classification is risk-driven, not effort-driven.

AI Assurance Model

Five distinct assurance types (Technical, Architecture, Risk, Compliance, Business Outcome; Section 5.5), each with a distinct owner and evidence base, interacting through the Decision entity with differentiated escalation urgency (safety/compliance escalate fast; value shortfalls route through ordinary change/retirement).

AI Lifecycle RACI

Provisional seven-role RACI across all stages and gates (Section 5.7), to be finalized in Chapter 8.

Minimum Required Evidence

Per-gate minimum evidence sets (Section 5.8), deliberately non-exhaustive.

Chapter 6. Digital Transformation Governance Model

This chapter details the Transformation dimension previewed in Chapter 4 as shallow-governance and constrained by FR-20, which grants the framework visibility of, but not ownership over, transformation and workforce change. Its central task is not to add transformation-management content. It is to define where the framework’s legitimate architectural interest ends and change-management and portfolio-office ownership begins, and then to hold that line even where a fuller transformation methodology would be tempting.

6.1 Transformation Traceability Chain

The research brief proposes a twelve-link chain: Purpose → Strategic Objective → Business Capability → Value Stream → Transformation Theme → Portfolio → Initiative → Product/Platform → Architecture → AI/Technology → Outcome → Benefit. Mapped against the Chapter 4 metamodel, ten of the twelve links already correspond to existing entities: Strategy/Objective, Capability, Value Stream, Initiative (already defined as “an instance of Transformation”), Product, Application/AI System, and Outcome. Two new entities are required. **Portfolio** is a grouping construct over Transformation Themes and Initiatives; **Benefit** is a realized-value record derived from Outcome and Metric at a specific point in time. Both are added to the metamodel as extensions, not replacements, consistent with the extension approach established in Chapter 4.

The more important finding from testing this chain against FR-20 is that **it is not uniformly governed**. Presenting all twelve links as equally “governed by this framework” would silently re-expand EA’s scope into transformation delivery, exactly what FR-20 was written to prevent. Each link is instead classified by governance depth:

Table 20. Transformation Traceability Chain

Link	Entity/Concept	Governance Depth	Owning Function	Framework’s Role
1	Purpose	Reference only	Corporate strategy / board	Consumed as context; not modeled as a distinct entity (folded into Strategy)
2	Strategic Objective	Reference only	Corporate strategy	Consumed, not set, by the framework
3	Business Capability	Deep — co-owned	Business + Enterprise Architecture jointly	Maintains the capability map (FR-01/FR-02, Mandatory)
4	Value Stream	Deep — co-owned	Business Architecture	Maintained as a Business Architecture artifact (Chapter 4, Section 4.3)
5	Transformation Theme	Shallow — visibility only	Transformation/Strategy Office	Framework records the theme’s existence and links capabilities to it; does not define theme methodology
6	Portfolio	Shallow, with a defined input right	Portfolio/PMO function	Architecture contributes a structured input package (Section 6.3); does not own prioritization
7	Initiative	Shallow ownership; deep only at architecture/AI touchpoints	Transformation/delivery function	Framework requires answers to Section 6.4’s qualification questions before an initiative reaches Portfolio, but does not manage initiative delivery

Link	Entity/Concept	Governance Depth	Owning Function	Framework's Role
8	Product/Platform	Deep	Enterprise/Solution Architecture	Application/Platform Architecture domains (Chapter 4)
9	Architecture	Deep	Enterprise/Solution Architecture	The framework's core domain
10	AI/Technology	Deep	Enterprise/AI Architecture, AI Governance	AI Architecture domain (Chapter 4) + AI Build Governance Lifecycle (Chapter 5)
11	Outcome	Deep	Business/Capability Owner, with framework tracking	Core Model concept (FR-03)
12	Benefit	Shared	Transformation/PMO function + framework	Benefit Record (Section 6.5) references, not duplicates, Outcome data owned at the AI System or Application level

This differentiated depth is the section's central design decision (logged as D-22 below). The framework is architecturally central at links 3–4 and 8–11, has a defined-but-limited input role at links 6–7 and 12, and is a consumer, not an owner, at links 1–2 and 5. A reviewer challenging “is this really an Enterprise Architecture framework or a transformation-management framework in disguise” should be pointed to this table directly.

6.2 Capability-Based Governance

Each Business Capability entity (Chapter 4 metamodel) is governed through a **Capability Record** containing: baseline maturity, target maturity, the resulting transformation gap, architecture dependencies, investment dependencies, and an AI-enablement assessment.

Maturity scale (1–5): 1 = Ad hoc (capability exists informally, no standard process or supporting system); 2 = Emerging (defined but inconsistently applied, partial system support); 3 = Defined (standardized process, dedicated system support); 4 = Managed (measured, systematically improved, integrated across the enterprise); 5 = Optimized (continuously improved using data/AI-driven feedback, industry-leading).

Transformation gap = target maturity – baseline maturity, expressed per capability, not in the aggregate. An enterprise-wide average gap score is close to meaningless for prioritization purposes, and is not used as a headline metric here.

Architecture dependency: which Application, AI System, Data, and Platform components (Chapter 4 primary domains) currently enable the capability at its baseline maturity, and which would need to exist or change to reach target maturity. This is where the Capability Record connects directly to the Architecture Repository (Chapter 4/9).

Investment dependency: which Initiative(s), and by extension which Portfolio entries, are funded to close the gap. A capability with an identified gap and no funded initiative is a visible, reportable state — not a silent gap. That applies the “no orphaned state” principle used for AI systems in FR-25, extended here to capability gaps.

AI enablement assessment: at the point a capability gap is identified, a judgment is recorded on whether closing it plausibly requires or benefits from AI. This is positioned as the true entry point to AI governance: **a capability gap judged AI-relevant is what spawns a Stage S0**

(Opportunity) assessment in the Chapter 5 AI Build Governance Lifecycle, not the other way around. That closes a traceability question Chapter 1 and Chapter 3 (FR-01) left implicit. AI use cases should not originate as free-floating ideas disconnected from a named capability gap, and the Capability Record is where the rule is enforced.

6.3 Portfolio Governance

Consistent with Section 6.1's classification of Portfolio as shallow-with-input-right, the framework does not own initiative prioritization or investment sequencing. That remains Portfolio/PMO practice, drawing on digital-transformation methodology Chapter 2 characterized as fragmented, practitioner-level, and not something this framework attempts to standardize (Chapter 2, Framework Boundary). What the framework defines is a structured **Architecture Input Package** that Portfolio governance must consume, without being required to defer to it.

Table 21. Portfolio Governance

Portfolio Decision	Architecture's Contribution	What Architecture Does NOT Own
Initiative prioritization	Capability-gap severity and architecture-readiness signal (from Section 6.2's Capability Record)	The business-value/strategic-fit judgment itself
Investment sequencing	Dependency graph showing which initiatives share underlying platform, data, or AI components (FR-18)	Funding allocation and timing decisions
Dependency identification	Explicit flags where two initiatives would build conflicting or duplicative architecture if sequenced independently	Resolution of the conflict — surfaced to Portfolio governance as a required decision, not silently resolved by architecture
Reuse	Identification of existing reference patterns (FR-06) or platform capabilities an initiative could reuse instead of building new	Whether the initiative accepts the reuse recommendation
Platform strategy	Assessment of whether a proposed initiative should build on an existing shared platform or justifies a new one	The platform investment decision itself, which remains a Portfolio/technology-investment call
Technical-debt decisions	A visible technical-debt register entry showing which initiatives increase versus reduce debt (Chapter 4 Application/Technology domains)	Trade-off decisions between new capability delivery and debt reduction, which are business priority calls

The pattern is consistent: architecture surfaces structured, visible information at each portfolio decision point; it does not make the decision. This is a testable design choice — Chapter 11's Chief Enterprise Architect reviewer persona should assess whether an input-only role is strong enough to prevent poor architecture outcomes at the portfolio level, or whether it is too weak to matter in practice. That tension is real and is not resolved here; it is logged as an open question below.

6.4 Transformation Governance Questions

The research brief's nine transformation questions are retained largely as stated, formalized into a mandatory checklist, the **Initiative Qualification Checkpoint (TG0)**, that must be answered before an initiative is admitted to the Portfolio (Section 6.3). Consistent with the shallow-governance classification of the Initiative link (Section 6.1), **TG0 is owned by the Transformation/Portfolio function, not by Enterprise Architecture**. Architecture's role is to

require that questions 4–7 below cannot be answered “unknown” without an architecture consultation; it does not chair the checkpoint.

Table 22. Transformation Governance Questions

#	Question	Answered By	Architecture’s Required Input
1	Why are we transforming?	Business Sponsor	None required
2	Which capability changes?	Business Sponsor, referencing Section 6.2’s Capability Record	Confirms the Capability Record exists and is current
3	What target state is required?	Business Sponsor + Capability Owner	Confirms target maturity is architecturally plausible
4	Which technology enables it?	Initiative lead	Mandatory — architecture must identify at least a provisional domain (Application/Data/AI/Platform/Technology)
5	Is AI necessary?	Initiative lead, informed by Section 6.2’s AI-enablement assessment	Mandatory — if yes, this is the trigger into Chapter 5 Stage S0
6	What dependencies exist?	Architecture	Mandatory — Section 6.3’s dependency-identification contribution
7	What organizational change is required?	Change-management function	Visibility only (FR-20) — architecture is informed, not consulted
8	What risk changes?	Risk & Compliance, informed by architecture where AI is involved	Mandatory where question 5 is answered “yes” — otherwise general enterprise risk practice applies, outside this framework’s AI-specific risk model (Chapter 5)
9	What measurable benefit should result?	Business Sponsor	Confirms a Benefit Record (Section 6.5) is opened

A known enforcement gap: because TG0 is owned by Portfolio/PMO and not by Enterprise Architecture, nothing in this section, by itself, prevents an initiative from bypassing TG0 entirely. This follows from honoring FR-20’s scope narrowing and is not a design oversight, but it is a real weakness. Chapter 7’s continuous assurance mechanism is the more credible answer: it retroactively detects architecture or AI System entries in the repository that have no corresponding TG0 record. Chapter 8’s operating model must then define what happens when that detection fires. The gap is carried forward, not assumed away.

6.5 Benefits Realization

Benefit Record fields: expected outcome (linked to the Outcome Core Model entity), KPI/Metric, baseline value, target value, owner, expected realization date, realized value (updated over time), and, per FR-19, explicit references to the specific Architecture, AI System, or Application records the benefit depends on. The record is not maintained as an independent transformation-office artifact disconnected from the systems that actually produce the outcome.

Relationship to Business Outcome Assurance (Chapter 5, Section 5.5): these are the same underlying Outcome/Metric Core Model concepts applied at two different levels of granularity, not two competing tracking mechanisms. Business Outcome Assurance is the ongoing, AI-System-level check that a specific deployed system is still delivering the value it was authorized for. The Benefit Record is the Initiative-level roll-up, which may aggregate several Architecture/AI System outcomes toward a single business KPI. A Benefit Record’s realized value is computed *from* the Metric data Business Outcome Assurance already collects; it is not re-

measured independently. This is the specific mechanism by which FR-19's non-duplication requirement is satisfied, and it is logged as D-25 below because it is easy for an implementation to violate silently, by standing up a separate benefits-tracking spreadsheet disconnected from system-level metrics, which is precisely the fragmentation Chapter 1 identified as a current-state problem.

Transformation Governance Lifecycle (T0–T7) and Its Relationship to the AI Build Lifecycle (S0–S10)

The Transformation Governance Lifecycle operates at the Initiative/Portfolio level and is distinct from, but connects to, the AI Build Governance Lifecycle defined in Chapter 5, which operates at the individual AI System level. Presenting them as a single merged lifecycle would misrepresent their actual relationship. Many transformation initiatives involve no AI at all, and a single AI system's lifecycle (S0–S10) can be triggered independently of, or nested inside, a transformation initiative's lifecycle.

Table 23. Transformation Governance Lifecycle (T0–T7) and Its Relationship to the AI Build Lifecycle (S0–S10)

Stage	Name	Corresponds To
T0	Purpose & Strategic Alignment	Chain links 1–2 (Section 6.1)
T1	Capability Gap Identification	Section 6.2
T2	Initiative Definition & Qualification (TG0)	Section 6.4
T3	Portfolio Prioritization	Section 6.3
T4	Architecture & AI Enablement	Hands off to Chapter 4 architecture work; where question 5 (Section 6.4) is “yes,” spawns one or more Chapter 5 S0 entries
T5	Delivery & Organizational Change	Visibility-only (FR-20); owned by delivery and change-management functions
T6	Benefits Realization & Review	Section 6.5
T7	Capability Re-Baseline	Closes the loop into T1 with updated baseline maturity

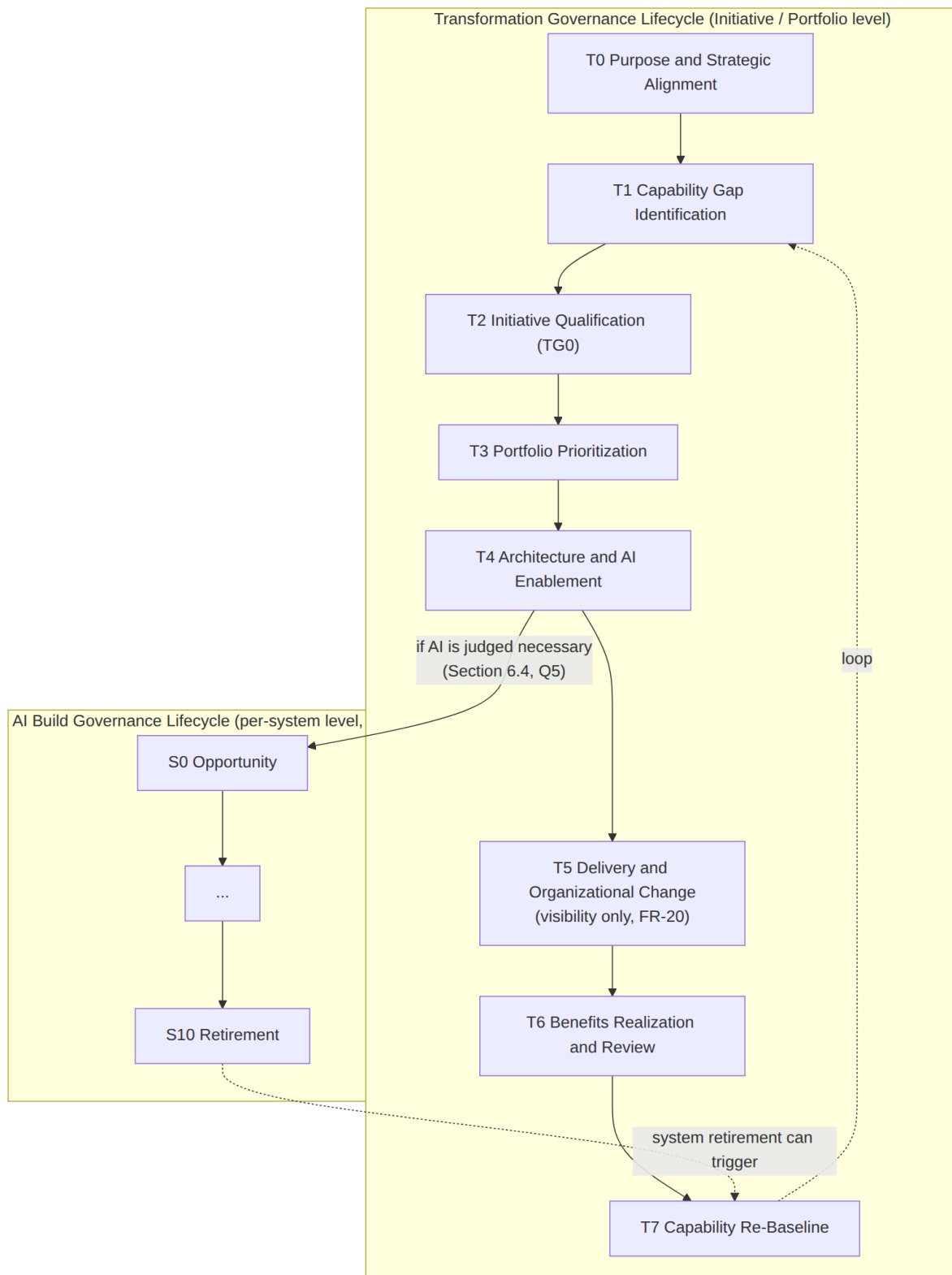


Figure 6. Transformation governance lifecycle (T0–T7) and its relationship to the AI build lifecycle

This structure answers Chapter 4’s open question about whether “Transformation” is adequately represented by a shallow dimension. It is representable as a full lifecycle in its own right, but one

whose *governance depth*, not its *structural completeness*, is limited at the stages FR-20 marked as out of scope (T5 specifically).

Transformation Scorecard

Distinct from Chapter 5’s system-level assurance metrics, the Transformation Scorecard operates at the portfolio/enterprise level, aggregating across initiatives instead of reporting per-system.

Table 24. Transformation Scorecard

Metric Category	Example Metrics	Primary Data Source
Capability Health	Number of capabilities with an open gap and no funded initiative (Section 6.2); average time-to-close for funded gaps	Capability Records
Portfolio Health	Share of active initiatives with a complete TG0 record; count of initiatives proceeding without one (the enforcement-gap signal named in Section 6.4)	TG0 records, cross-checked against the architecture repository
Architecture Contribution	Reuse rate (initiatives adopting an existing reference pattern vs. building new, FR-06); dependency conflicts surfaced vs. resolved (Section 6.3)	Architecture Decision Records
AI Enablement	Share of capability gaps assessed as AI-relevant that actually proceed through the Chapter 5 lifecycle vs. stall at S0/S1	Capability Records, AI lifecycle records
Benefits Realization	Percentage of Benefit Records with realized value at or above target by the expected realization date; realized-to-expected benefit ratio in aggregate	Benefit Records
Technical Debt Trend	Net technical-debt register movement attributable to transformation initiatives (Section 6.3)	Technical debt register

The scorecard does not include a single composite “transformation health” index. The research brief’s own critical-thinking instruction (Section 46) applies here directly: a single blended score across capability, portfolio, architecture, AI, benefit, and debt metrics would obscure exactly the kind of trade-off information (e.g., high reuse but low benefit realization) that this scorecard exists to surface. The rejection of that tempting simplification is logged below.

Chapter 6 in Summary

Transformation Governance Lifecycle

T0–T7 (above), distinct from but connected to the Chapter 5 AI Build Governance Lifecycle at T4, with the connection shown in the diagram above and not merged into a single lifecycle.

Capability-to-Initiative Model

The Capability Record (Section 6.2: baseline/target maturity, transformation gap, architecture and investment dependencies, AI-enablement assessment) as the connective structure between Business Capability and Initiative, with capability gaps judged AI-relevant serving as the defined entry point into the Chapter 5 lifecycle.

Portfolio Governance Model

The Architecture Input Package (Section 6.3): six defined contributions (prioritization signal, dependency graph, conflict flags, reuse recommendations, platform assessment, technical-debt visibility) that Portfolio governance must consume without being required to defer to.

Transformation Architecture Model

The governance-depth-differentiated traceability chain (Section 6.1) showing exactly where architecture is deep, shared, or a consumer across the twelve-link Purpose-to-Benefit chain.

Benefits Traceability Model

The Benefit Record (Section 6.5), deriving its realized-value data from Chapter 5's Business Outcome Assurance metrics instead of duplicating measurement: the mechanism satisfying FR-19.

Transformation Scorecard

Six metric categories (above), not collapsed into a single composite index.

Chapter 7. Continuous Architecture Governance

This chapter designs the Control View previewed in Chapter 4 and required by FR-21 to FR-23: the mechanism that keeps deployed architecture and AI behavior verifiably consistent with what was approved, on a continuous basis rather than a point-in-time one. It is at once the framework’s strongest candidate for real novelty and its highest risk of turning out to be existing DevOps and security posture-management practice under new vocabulary. That tension is confronted here, not deferred to Chapter 12.

7.1 Architecture Telemetry

Per FR-23, the framework specifies *what* must be observable, not *which* tool observes it. Each telemetry source is mapped to the Chapter 4 metamodel entity or relationship it populates with an “actual state” value. Section 7.3’s Compare function then has something concrete to set against the “approved state” recorded in Decision and Architecture Decision Record entries.

Table 25. Architecture Telemetry

Telemetry Source	EA-Relevant Evidence It Can Produce	Metamodel Entity/Relationship Populated (actual state)
Cloud environments (resource inventories, IaC state)	Actual deployed infrastructure configuration	Platform, Technology Architecture attributes
APIs / service meshes / gateways	Actual integration and routing patterns in effect	API, Service, and the Integration cross-cutting discipline
IAM / identity providers	Actual granted permissions for human and machine identities	Agent → constrained by → Policy (actual vs. approved authority boundary, FR-12)
Application portfolios / CMDBs	Actual application inventory and dependency graph	Application entity existence and Section 6.3 dependency map
AI platforms / model registries	Actual deployed model family and version	AI System → uses → Model (actual vs. approved)
Data catalogs	Actual grounding sources and data classification in use	AI System → grounded by → Data Product (actual vs. FR-13 approved lineage)
CI/CD systems	Deployment events, with timestamps	Decision → authorizes → AI System (used to test whether a deployment was preceded by a valid Decision)
Security tools (SIEM, scanners, prompt-injection/exfiltration detectors)	Actual security posture and incident signals	Control → mitigates → Risk (actual effectiveness)
Policy engines (e.g., OPA-class tooling)	Policy evaluation pass/fail results	Direct input to Compare (7.3) and Evidence-as-Code (7.6)
FinOps systems	Actual spend against approved budget	Cost Drift signal (7.2); often a leading indicator of unauthorized scale change (Risk dimension D4, Chapter 5)

Two sources named in the research brief, CMDBs and application portfolios, are listed together because in most enterprises they are the same system in practice. Separating them would imply a distinction the underlying tooling rarely honors.

7.2 Architecture Drift Taxonomy

The research brief supplies two overlapping drift lists that do not fully align. The first is the original nine-item list (Section 14: architecture, cloud, security, AI model, data, regulatory, capability, dependency, cost drift). The second is this chapter’s ten-item list (Section 7.2: structural, technology, configuration, integration, security, data, AI-model, policy, cost, strategic drift). Presenting both without reconciliation would leave four items (regulatory/policy, architecture/structural, cloud/technology/configuration, dependency/integration) as ambiguous near-duplicates. They are consolidated into eight categories, with two items from the original lists excluded from the *telemetry-detectable* taxonomy and relocated to where they actually belong.

Table 26. Architecture Drift Taxonomy

#	Category	Definition	Detection Mechanism
1	Structural Drift	Actual application/data/AI/technology inventory diverges from the architecture repository	Telemetry comparison (CMDB, cloud inventory vs. repository)
2	Configuration Drift	An approved component’s configuration changes without a corresponding Decision record	Telemetry comparison (IaC state vs. last-approved configuration)
3	Integration & Dependency Drift	Actual integration patterns or cross-initiative dependencies diverge from what was approved or from the Section 6.3 dependency map	Telemetry comparison (API/service mesh state)
4	Security Drift	Actual security posture (permissions, exposed surfaces) diverges from the approved Security Architecture baseline	Telemetry comparison (IAM, security tooling)
5	Data Lineage/Governance Drift	Grounding sources or data classification for an AI system change without going through the FR-13/FR-14 process	Telemetry comparison (data catalog vs. approved lineage record)
6	AI-Model Behavioral Drift	The model’s actual output behavior changes over time — via vendor-side model updates or genuine distribution shift	Technical AI Evaluation signals (Chapter 5, Section 5.5), not infrastructure telemetry
7	Policy & Regulatory Drift	The applicable policy or regulation has changed since a Decision was made, independent of whether the deployed system itself changed at all	Regulatory/policy monitoring — structurally different from the other categories, detailed below
8	Cost Drift	Actual spend diverges from approved budget, often a proxy for unauthorized scale change	Telemetry comparison (FinOps systems)

Disambiguation: “data drift” is used ambiguously across the AI-governance literature to mean two different things. The first is statistical distribution shift in a model’s input data, a model-performance concern owned by Technical AI Evaluation per Chapter 5, Section 5.5. The second is ungoverned change to what data sources ground an AI system, an architecture/lineage concern, category 5 above. Conflating them would mean an architecture-level drift alert fires for what is actually a model-quality issue, or vice versa. They are kept as two distinct signals feeding two distinct owners.

Two items relocated out of the telemetry-detectable taxonomy: “capability drift” (the actual maturity of a business capability diverging from its recorded state, Section 14) is not something an API call can observe. Capability maturity is an assessed property, not an instrumented one. It is handled by Chapter 6’s T7 periodic re-baseline cadence, not by this Control Plane. “Strategic drift” (the enterprise’s strategy itself changing, such that previously-approved architecture is no longer aligned to current intent) is likewise not telemetry-detectable; it requires human strategic judgment, not a comparison operation. It is handled by periodic Enterprise Intent-layer review — an operating-model cadence question for Chapter 8, not a Control Plane function. Claiming either

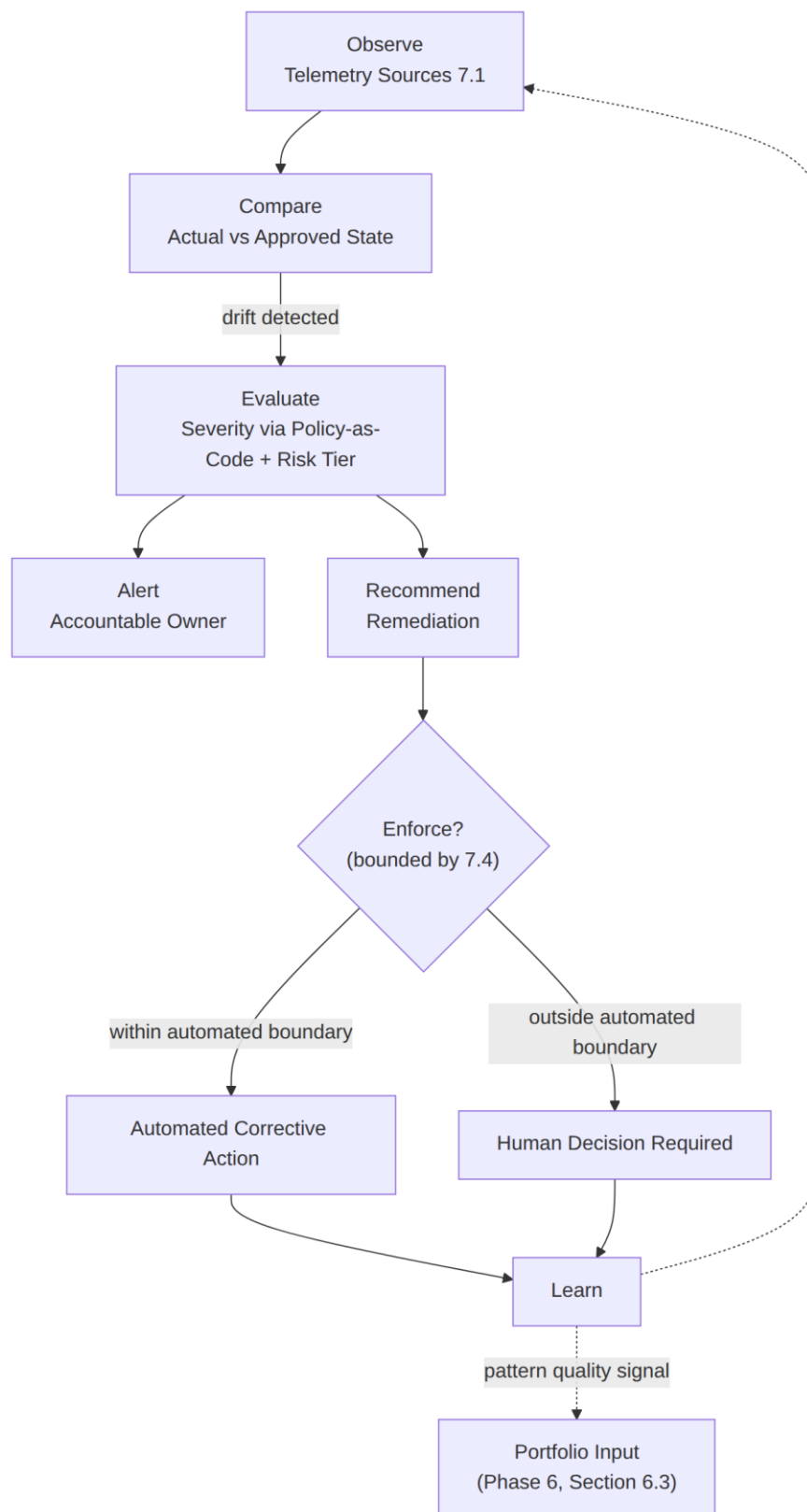
of these as automatically detectable would overstate what telemetry can actually do, which is precisely the kind of overclaim the project’s critical-thinking rule warns against.

Policy & Regulatory Drift is structurally different from categories 1–6 and 8. The others detect *actual state diverging from approved state*: something changed on the ground. Policy/regulatory drift detects the opposite, *approved state no longer meeting an external requirement that itself changed*, with nothing on the ground having moved at all. Chapter 1 and Chapter 2 already surfaced a live example. The EU AI Act (European Parliament and Council of the European Union, 2024)’s 2026 Digital Omnibus deferral of certain high-risk obligations changes what “compliant” means for an already-deployed system without that system’s architecture changing at all. Detecting this category requires a regulatory-monitoring feed as an input to Evaluate (7.3), not a telemetry comparison — logged as a distinct mechanism, not folded into the same detection pipeline as the other seven categories.

7.3 Enterprise Architecture Control Plane

The Control Plane is defined as seven functions operating as a continuous loop, not a linear pipeline. Evaluate and Alert outcomes feed back into what Observe collects next: a confirmed drift pattern warrants closer observation of similar components elsewhere in the estate.

Observe: collects Architecture Telemetry (7.1) from the defined sources, populating actual-state values on metamodel entities. **Compare:** checks actual state against the approved state recorded in Decision and Architecture Decision Record entries, producing a drift signal classified per the Section 7.2 taxonomy. **Evaluate:** assesses a drift signal’s severity using Policy-as-Code rules (7.5) and the affected AI System’s current risk tier (Chapter 5, Section 5.3), determining urgency and required response. **Alert:** notifies the accountable owner identified in the Chapter 5 RACI (Section 5.7), scaled by severity — a Tier 1 configuration drift might generate a low-urgency digest entry, while a Tier 4 security drift generates immediate, direct notification. **Recommend:** proposes a remediation, such as reverting to approved configuration or opening a Change record (Chapter 5, Section 5.6), without necessarily executing it. **Enforce:** executes a corrective action automatically, but only within the boundaries defined in Section 7.4 below. This is the function most constrained by human-authority requirements. **Learn:** feeds observed patterns back into architecture decision-making. A reference pattern that repeatedly generates drift alerts across multiple initiatives, for instance, is flagged back into Chapter 6’s Section 6.3 portfolio input (reuse recommendation quality) and into Chapter 9’s pattern library maintenance. This closes a loop between runtime observation and upstream architecture decisions that conventional EA governance, operating on periodic review cycles, cannot close in anything close to real time.

**Figure 7.** Enterprise architecture control plane

7.4 Control Plane Boundaries — the Human vs. Automated Decision Boundary

Four control types, adapted from standard audit/control-theory language, structure what can and cannot be automated.

Advisory controls inform a human without taking action (e.g., “this configuration pattern hasn’t been used elsewhere — consider review”). Delivery of an advisory is always safe to automate fully; it carries no action risk by definition.

Detective controls identify that a drift or violation has already occurred (the Compare function itself). These too are always safe to automate fully: detection carries no action risk, only an information-disclosure consideration for sensitive findings.

Preventive controls block an action *before* it takes effect (e.g., a deployment pipeline rejecting a model not on the approved list). They are safe to automate only where the underlying policy rule is objectively and unambiguously defined. The question “is this model on the approved list” is automatable; “is this architecture pattern appropriate for this use case” is a judgment call and is not, regardless of risk tier. Every automated preventive control must have a defined exception path (FR-26), so that a legitimate edge case is not simply blocked with no recourse.

Corrective controls actively remediate a detected violation (e.g., auto-reverting a configuration, auto-revoking a permission). This is the highest-risk automation category and the one requiring the clearest boundary:

Default rule: automated corrective enforcement is permitted for Tier 1–2 AI systems where the corrective action is pre-approved and reversible (e.g., reverting an unauthorized configuration change back to the last approved state). It is prohibited by default for Tier 3–4 systems, for any action that would materially change an AI agent’s authority or permissions (FR-12), and for any action with a low reversibility score (Chapter 5’s D3 dimension). In those cases a human decision is required, routed through the same Decision entity and accountable-owner structure as any other gate.

The one principled exception — pre-approved kill-switch conditions. An AI agent already exhibiting behavior matching a predefined catastrophic pattern (e.g., attempting an irreversible financial transaction outside its approved authority) is the specific case where waiting for human response may itself cause the harm the control exists to prevent. Automated shutdown is permitted here, but **only** where the triggering condition was defined and pre-approved as part of that system’s G3 Deployment Authorization gate (Chapter 5, Section 5.4), never invented ad hoc by the Control Plane at runtime. This keeps the exception narrow and auditable: a kill-switch firing is always traceable to a specific, previously human-approved condition, not to an autonomous judgment made by the Control Plane itself.

Table 27. Control Plane Boundaries — the Human vs. Automated Decision Boundary

Control Type	Automation Default	Exception Path
Advisory	Always automated	None needed
Detective	Always automated	None needed
Preventive	Automated for objectively-defined rules; human judgment required otherwise	FR-26 exception process

Control Type	Automation Default	Exception Path
Corrective	Automated only for Tier 1–2, pre-approved, reversible actions	Pre-approved kill-switch conditions (Tier 3–4 exception, defined at G3)

7.5 Policy-as-Code

Policy-as-code is the mechanism that makes Preventive controls (7.4) and part of Evaluate (7.3) executable rather than merely documented, directly implementing FR-28. As with telemetry (7.1), the framework specifies the rule’s intent and the evidence it must check, not the specific policy engine.

Table 28. Policy-as-Code

Policy Domain	Example Machine-Checkable Rule	Control Type Enabled	Risk-Tier Applicability
Cloud	Resource must carry an approved architecture-decision-record reference	Preventive	All tiers
Security	No machine identity may hold a permission absent from its approved authority boundary (FR-12)	Preventive	All tiers, critical at Tier 3–4
AI model selection	Deployed model family must be on the enterprise’s approved foundation-model list	Preventive	All tiers
Data location	Data above a defined sensitivity classification may not be transmitted to a hosting endpoint outside an approved jurisdiction list	Preventive	All tiers, sovereignty-relevant systems especially
AI agent permissions	An agent’s live tool/permission set must match its G2-approved architecture decision record	Preventive + Detective	Tier 2+
Approved technologies	Application/AI System must be built on an approved platform or carry a documented exception (FR-26)	Preventive	All tiers
Architecture patterns	AI System must reference an approved reference pattern (FR-06) or carry a documented exception	Preventive	All tiers

7.6 Evidence-as-Code

Evidence-as-code is the automation mechanism for producing the “Minimum Required Evidence” defined in Chapter 5, Section 5.8 — wherever the evidence type is amenable to automatic capture as a byproduct of normal system operation, rather than manual assembly after the fact.

Table 29. Evidence-as-Code

Chapter 5 Evidence Item	Evidence-as-Code Eligible?	Automatic Capture Mechanism
Architecture Decision Record reference (G2)	Yes	Linked automatically at deployment via IaC tagging (7.1)
Data lineage/sensitivity classification (G2)	Yes	Pulled automatically from data catalog telemetry
Reference-pattern conformance result (G2)	Yes	Policy-as-code evaluation output (7.5)
Technical AI Evaluation results (G3)	Yes	Captured directly from the evaluation pipeline, linked to the AI System record
Control-to-risk mapping (G3)	Partial	The mapping itself is a human judgment (FR-16); the evidence that a mapped control is <i>currently</i> effective can be automatically captured

Chapter 5 Evidence Item	Evidence-as-Code Eligible?	Automatic Capture Mechanism
Human oversight point confirmation (G3)	No	Requires confirming an interaction design decision was actually implemented; not inferable from telemetry alone
Change classification record (G4)	Partial	The classification decision requires judgment (Section 5.6); the delta-evidence supporting it can be automatically captured
Data retention/deletion confirmation (G5)	Yes	Captured automatically from data platform deletion logs
Access-revocation confirmation (G5)	Yes	Captured automatically from IAM deprovisioning logs
“Knowledge preservation” judgment (G5)	No	Inherently a human assessment of what is worth preserving

Roughly two-thirds of Chapter 5’s minimum evidence set is evidence-as-code eligible. The remainder (human oversight confirmation, the substance of control-to-risk mapping, change classification judgment, and knowledge preservation) requires human input by nature. That is not a gap in the automation design but a boundary of the same kind Section 7.4 draws.

Confronting the Anti-Novelty Question Directly

This is the component flagged as the framework’s strongest novelty candidate since Chapter 2 (D-03) and, simultaneously, the one most likely to fail the formal anti-novelty test reserved for Chapter 12. The tension is addressed here instead of being left to ride until Chapter 12, because this is the chapter where the temptation to overclaim is greatest.

What is not novel: telemetry collection from cloud, IAM, and application inventories; drift detection via actual-vs-approved comparison; policy-as-code enforcement of preventive controls; and automated evidence capture are all mature commercial practice. Cloud security posture management (CSPM) tooling, DevOps observability platforms, and policy-as-code engines (the OPA-class tooling referenced in 7.1) already do most of what Sections 7.1, 7.2 (categories 1–4, 8), 7.4, and 7.5 describe, for infrastructure and security specifically. A framework that claimed to invent continuous infrastructure monitoring would fail the anti-novelty test immediately, and should.

What is more defensibly novel, and why: three things distinguish this Control Plane from existing CSPM/DevOps practice, and all three are narrower claims than “continuous architecture governance” as a whole. First, the comparison target is not a generic security or cost baseline but a **specific, risk-tiered, accountable-owner-attached Decision record** in the Chapter 4 metamodel. A drift alert here is not just “configuration changed”; it is “configuration changed relative to a Decision made by a named accountable owner under a specific risk tier,” which is a governance fact, not merely a technical one. Second, categories 5–7 of the drift taxonomy (data lineage/governance drift, AI-model behavioral drift, policy/regulatory drift) and the AI-specific policy-as-code rules (agent permission conformance, model selection, data-location-for-AI) have no direct CSPM equivalent. CSPM tooling predates the AI-specific metamodel entities this framework introduces (AI System, Agent, Model, grounding relationships), and was not designed around them. Third, the Learn function’s feedback into Portfolio-level reuse and pattern-quality

decisions (Section 6.3) connects a technical monitoring loop to a governance-strategy loop, which CSPM tools, scoped to technical remediation, do not attempt.

Working verdict, provisional pending Chapter 12: the Control Plane *mechanism*, observe/compare/evaluate/alert/recommend/enforce/learn as a generic pattern, is an **Extension** of existing CSPM/DevOps/policy-as-code practice, not a New Contribution; claiming otherwise would not survive scrutiny. The narrower and defensible novelty claim concerns its **application to the specific EA-AI governance metamodel**: risk-tiered, Decision-attached comparison; AI-specific lineage/agent/model drift categories; and the Learn-to-Portfolio feedback loop. Even that claim should be tested in Chapter 12 against whether any AI-specific governance or MLOps tooling has already closed this gap by the time this research concludes. It is logged as D-27 below with status “Provisional,” not “Accepted,” so that Chapter 12 is not tempted to treat it as already settled.

Chapter 7 in Summary

Continuous Governance Architecture

The seven-function Control Plane (Observe → Compare → Evaluate → Alert → Recommend → Enforce → Learn) operating as a loop, Section 7.3, with the Learn function closing back both to Observe and to Chapter 6’s portfolio-level input.

Architecture Telemetry Model

Ten telemetry source categories mapped to specific metamodel entities/relationships (Section 7.1), specified as an interface requirement (FR-23), not a tooling mandate.

Drift Taxonomy

Eight telemetry/signal-detectable categories, reconciled from the research brief’s two non-aligned lists, with data drift disambiguated into two distinct signals and two items (capability drift, strategic drift) relocated out of the Control Plane’s reach to Chapter 6 and Chapter 8 respectively, with the reasoning stated.

Enterprise Architecture Control Plane

As above (Section 7.3), including the Mermaid diagram.

Policy-as-Code Model

Seven policy domains with example machine-checkable rules (Section 7.5), directly implementing FR-28.

Evidence-as-Code Model

Cross-referenced against Chapter 5’s minimum evidence set, with roughly two-thirds judged automation-eligible and the remainder left as human judgment (Section 7.6).

Human vs. Automated Decision Boundary

Four control types (advisory, detective, preventive, corrective) with a stated default-prohibition on automated corrective action for Tier 3–4 systems, narrowed by exactly one principled, pre-approved exception (kill-switch conditions defined at G3) — Section 7.4.

DO NOT COPY

Chapter 8. Governance Operating Model

This chapter defines who decides, at what level, and with what authority. It closes, not merely restates, three questions left open earlier: whether FR-09's architecture-integrated AI gates recreate a second bureaucracy alongside the existing governance of COBIT (ISACA, 2018) (Chapter 3), whether Portfolio governance requires more than an input-only role for high-risk initiatives (Chapter 6), and how the TGO enforcement gap identified in Chapter 6 is actually closed. The roles used provisionally in Chapter 5 are finalized here.

8.1 Avoiding Committee Proliferation

The research brief names six candidate bodies: Architecture Review Board (ARB), AI Governance Council, Data Council, Security Council, Model Risk Committee, Transformation Council. Tested against the decisions this framework's requirements (Chapter 3) oblige someone to make, four of the six do not warrant standing bodies of their own.

AI Governance Council does not become a separate body. Standing up an AI-specific approval council alongside the ARB is precisely the failure mode Chapter 3's open question anticipated — a second bureaucracy parallel to architecture governance, doubling review cycles for any AI-enabled initiative. Instead, the ARB's mandate is **expanded** to include AI Architecture as one of its five primary domains (Chapter 4, Section 4.3), and AI-specific expertise joins the ARB as an expanded membership requirement, not a parallel body. This answers Chapter 3's open question directly. FR-09's architecture-integrated gates do not recreate a second bureaucracy because there is no second bureaucracy; AI governance is integrated into the existing architecture governance body's mandate. NIST CSF 2.0 (NIST, 2024) modeled the same resolution, adding a Govern function across existing functions instead of a separate governance framework (Chapter 2, Chapter 4 precedent).

Data Council and Security Council are not created by this framework. Where they already exist, they remain untouched, external bodies: an organization's DAMA-aligned data governance, or its ISO 27001 (ISO & IEC, 2022)/NIST CSF-aligned security governance. Chapter 2's Framework Boundary treats both as inherited, not owned here. This framework's only interface to them is through the Data Owner and Security Architect roles (Section 8.4). Those roles take part in ARB and AI Governance Body decisions without requiring this framework to stand up competing structures.

Model Risk Committee does not become a separate body. Its function is independent challenge and validation for the highest-consequence AI systems (Chapter 2's model-risk-management precedent). That function is absorbed into the **AI Governance Body** and activated only for Tier 3–4 systems (Section 8.3), so no standing committee reviews every AI system regardless of risk.

Transformation Council is not part of this framework's governance operating model. Consistent with FR-20 and Chapter 6's shallow-governance classification of the Transformation dimension, whatever body an organization uses to run Portfolio/PMO governance is external to this framework. The interface to it is the Architecture Input Package (Chapter 6, Section 6.3) and,

per Section 8.3 below, a narrowly defined concurrence right for a specific decision class. It is not a seat this framework claims to chair.

Net result: two standing bodies this framework requires, not six.

Table 30. Avoiding Committee Proliferation

Candidate Body (research brief)	Disposition
Architecture Review Board	Retained, mandate expanded to include AI Architecture
AI Governance Council	Not created — folded into expanded ARB membership
Data Council	Not created — external if it exists; Data Owner role is the interface
Security Council	Not created — external if it exists; Security Architect role is the interface
Model Risk Committee	Not created — folded into AI Governance Body, active only for Tier 3–4
Transformation Council	Not created; explicitly out of scope — Architecture Input Package is the interface

Both retained bodies operate as the “Direct” and “Evaluate” components of whatever COBIT-aligned enterprise IT governance structure the organization already has (Chapter 2 Integration Map). They are not a parallel governance stack alongside existing IT governance, but the specific mechanism through which that structure exercises its architecture and AI-related governance objectives.

8.2 Federated Governance Model

The research brief proposes a four-part model: Central Principles, Federated Decisions, Automated Controls, Risk-Based Escalation. It is operationalized as follows, drawing on the data-mesh federated-computational-governance precedent identified in Chapter 2’s Integration Map.

Central Principles are set jointly by the ARB and AI Governance Body and ratified at the Enterprise governance level (Section 8.3); they are few in number by design (the full principle set is developed in Chapter 10) and change infrequently.

Federated Decisions are made at the Product/Solution level by empowered owners, for Tier 1–2 systems. They operate within pre-approved reference patterns (FR-06) and policy-as-code guardrails (Chapter 7, Section 7.5) — this is the actual locus of “federation”: most day-to-day architecture and AI decisions never reach a standing body at all.

Automated Controls are what make federation safe, not merely permissive. Chapter 7’s Control Plane, comprising preventive policy-as-code checks and detective drift alerts, is the enforcement substrate underneath federated decision rights. Without it, federating Tier 1–2 decisions to product teams would be delegation without verification. With it, a product team’s self-certified G2 pass (Chapter 5) is continuously checked against the same policy rules a human reviewer would apply, closing the loop between Chapter 7 and Chapter 8.

Risk-Based Escalation routes Tier 3–4 decisions, and any Tier 1–2 decision that trips a Control Plane alert or exception request, to the two standing bodies rather than resolving them locally. The Escalation Model in Section 8.7 sets out the detail.

8.3 Governance Levels

Table 31.

Level	Authority	Primary Body/Role	Escalates To
Enterprise	Sets Central Principles; ratifies the risk-tiering methodology (Chapter 5, Section 5.3); resolves cross-portfolio conflicts the Portfolio level cannot; owns the framework's own change control	Enterprise Governance Sponsor (CIO/CTO-equivalent), meeting on a low-frequency cadence (indicatively quarterly)	— (top of the escalation chain)
Portfolio	Prioritizes and sequences initiatives (Chapter 6, Section 6.3), consuming the Architecture Input Package	Transformation/PMO function (external to this framework)	Enterprise, only for conflicts Portfolio cannot resolve using the Architecture Input Package alone
Platform	Approves shared platform investments (e.g., a shared LLM gateway) spanning multiple products	Platform Architecture discipline lead, reporting into ARB for Tier 3–4 implications	ARB
Product	Makes Tier 1–2 architecture and AI decisions within pre-approved patterns and policy guardrails	Business Sponsor + AI/System Owner	ARB (architecture questions) / AI Governance Body (Tier 3–4 AI questions)
Solution	Designs within Product-level bounds; self-certifies G2 pattern conformance for Tier 1–2 (Chapter 5)	Solution/Enterprise Architect	ARB
Model/Agent	Operates strictly within its approved authority boundary (FR-12); cannot self-modify its own permissions under any circumstance	AI System/Agent Owner	AI Governance Body (any autonomy or permission change, per Chapter 5's Change Model, is Major by default and re-enters S2)

Resolving Chapter 6's open question: does Portfolio need more than input-only authority?

Yes, for a narrow and tightly bounded case. **Tier 3–4 AI-relevant initiatives require AI Governance Body concurrence, not mere input, before Portfolio may approve funding.** For every other initiative (Tier 1–2, or non-AI), Portfolio retains full prioritization authority, consuming the Architecture Input Package as advisory information only, per Chapter 6's original shallow-governance design. This is a deliberate, bounded exception to FR-20's visibility-only framing, justified by the same risk-proportionality logic (FR-15) that governs every other tier-scaled decision in this framework. It is not a general expansion of EA's authority over transformation, and it is logged as a specific, narrow decision (D-31 below), not a silent walk-back of Chapter 6's design.

8.4 Role Catalogue

Roles used provisionally in Chapter 5's RACI (Section 5.7) are finalized here with full specification.

Table 32. Role Catalogue

Role	Purpose	Accountability	Authority	Key Decisions	Artifacts Owned	Escalates To
------	---------	----------------	-----------	---------------	-----------------	--------------

Role	Purpose	Accountability	Authority	Key Decisions	Artifacts Owned	Escalates To
Business Sponsor / Capability Owner	Justify and fund capability investment	Business value delivered	Approve/reject use-case qualification (S1)	Business case approval	Use-case canvas, Capability Record (Chapter 6)	ARB / Portfolio
AI/Enterprise Architect	Design and certify architecture conformance	Architecture quality and pattern conformance	Self-certify Tier 1–2 G2; author ADRs	Architecture pattern selection, reference-pattern conformance	ADRs, reference pattern library (Chapter 9)	ARB
Data Owner	Govern the data a system depends on	Data classification and lineage accuracy	Approve/reject data readiness (S4)	Data sensitivity classification, lineage confirmation	Data lineage records (FR-13)	ARB (via Architecture questions)
AI/System Owner	Operate the deployed system day to day	System behavior, ongoing conformance, retirement	Responsible for evidence at every gate; requests changes	Day-to-day operational decisions within approved authority	System configuration records, Change records	AI Governance Body (Tier 3–4) / ARB (architecture)
Risk & Compliance Lead (AI Governance Lead)	Maintain and apply the risk-tiering methodology and control mapping	Risk scoring accuracy, control effectiveness evidence	Assign/re-assign risk tier (S2); accountable for G3 evidence completeness	Risk tier assignment, control-to-risk mapping (FR-16)	Risk scoring records, control mapping register	AI Governance Body
Security Architect	Apply cross-cutting security discipline to architecture and AI	Security posture of governed components	Consulted on all gates; can block on unresolved security findings	Security control design for AI-specific risks (prompt injection, exfiltration)	Security control records	ARB / external Security Council (if one exists)
Architecture Review Board (ARB)	Approve architecture decisions across all five primary domains, including AI	Architecture conformance and pattern quality across the enterprise	Approves G2 for Tier 3–4; ratifies reference patterns; resolves dependency conflicts (Chapter 6, Section 6.3) not resolvable at Solution level	G2 decisions (Tier 3–4), pattern library approval	Approved reference pattern library	Enterprise Governance Sponsor
AI Governance Body	Authorize deployment and material change for the highest-risk AI systems; absorbs the Model Risk Committee function	Deployment authorization quality for Tier 3–4 systems	Approves G3/G4 for Tier 3–4; holds concurrence right over Tier 3–4 Portfolio funding (Section 8.3)	Deployment authorization, material-change approval, residual-risk acceptance	Deployment authorization records	Enterprise Governance Sponsor
Enterprise Governance Sponsor	Hold ultimate accountability for the framework's operation	Framework integrity and cross-portfolio conflict resolution	Ratifies Central Principles; resolves conflicts ARB/AI Governance Body/Portfolio cannot	Principle ratification, top-level conflict resolution	Central Principles document (Chapter 10)	—
Transformation/PMO	Prioritize and sequence the	Portfolio delivery	Full authority for Tier 1–2/non-AI	Prioritization, sequencing,	Portfolio roadmap	Enterprise (only where

Role	Purpose	Accountability	Authority	Key Decisions	Artifacts Owned	Escalates To
Function (external)	initiative portfolio	outcomes	initiatives; requires AI Governance Body concurrence for Tier 3–4 AI-relevant initiatives	funding	(external artifact, referenced not owned by this framework)	concurrence is contested)

Design check against over-concentration: the Enterprise Governance Sponsor is Accountable only for Enterprise-level decisions (principle-setting, cross-body conflict resolution). The Sponsor is never named Accountable for an individual Tier 1–2 gate decision anywhere in this catalogue or the RACIs below, a deliberate check against the “every senior leader accountable for everything” anti-pattern the research brief warns against (Section 8.6).

8.5 Decision Rights Matrix

Table 33.

Decision	Tier 1	Tier 2	Tier 3	Tier 4
Approve architecture (G2)	AI/Enterprise Architect (self-certify)	AI/Enterprise Architect (self-certify) + ARB notified	ARB	ARB, with Enterprise Governance Sponsor briefed
Approve deployment of high-risk AI (G3)	AI/System Owner	AI/System Owner + AI Governance Lead notified	AI Governance Body	AI Governance Body + Enterprise Governance Sponsor sign-off
Approve new foundation model for the enterprise-approved list	AI/Enterprise Architect proposes; ARB ratifies (applies once, enterprise-wide, not per-tier)			
Change an AI system’s risk tier	Risk & Compliance Lead, self-service if the change is downward or lateral	Risk & Compliance Lead; AI Governance Body notified if the change is upward	AI Governance Body	AI Governance Body + Enterprise Governance Sponsor briefed
Authorize an autonomous agent (grant/increase autonomy, FR-12)	Not applicable at Tier 1 by definition (an autonomy increase re-triggers risk classification, Chapter 5 Section 5.6, and may move the system out of Tier 1)	AI Governance Body (autonomy increases are Major changes, always)	AI Governance Body	AI Governance Body + Enterprise Governance Sponsor sign-off
Approve an exception (FR-26)	AI/System Owner, time-boxed ≤ 90 days	AI Governance Lead, time-boxed ≤ 90 days	AI Governance Body, time-boxed ≤ 90 days, renewal requires re-justification	AI Governance Body + Enterprise Governance Sponsor, time-boxed ≤ 90 days, no auto-renewal ever
Retire an AI system (G5)	AI/System Owner + Data Owner	AI/System Owner + Data Owner	AI/System Owner + Data Owner + AI Governance Lead	AI/System Owner + Data Owner + AI Governance Body
Accept residual risk	AI/System Owner	AI Governance Lead	AI Governance Body	AI Governance Body + Enterprise

Decision	Tier 1	Tier 2	Tier 3	Tier 4
Governance Sponsor				
The “approve new foundation model” row is not tier-differentiated — a model either joins the enterprise-approved list or it does not. That decision is made once, centrally, by ARB ratification of an Architect’s proposal, rather than separately by every Tier 1–4 use case that might want to use it. The non-duplication logic used throughout applies here: repeating a single enterprise-wide decision across four tiers would be redundant process, not risk-proportional governance.				

8.6 RACI — Both Lifecycles

Two RACIs are produced, one per lifecycle defined in this framework (Chapter 5’s AI Build Lifecycle and Chapter 6’s Transformation Governance Lifecycle), since Chapter 6 did not produce one and this chapter’s output requires lifecycle coverage. Placeholder roles from Chapter 5, Section 5.7 are replaced with the finalized role names from Section 8.4. The structure and tier-conditionality are otherwise unchanged from Chapter 5’s design.

AI Build Lifecycle RACI (finalized)

Table 34.

Stage/Gate	Business Sponsor	Architect	Data Owner	AI/System Owner	Risk & Compliance Lead	Security Architect	ARB	AI Governance Body
S0–S1 Opportunity/Qualification	R	C	C	—	I	—	—	I (Tier 3–4)
S2 Risk Classification	C	C	C	—	A/R	C	—	I (Tier 3–4)
G1	A	C	I	—	R (Tier 3–4)	—	—	C (Tier 3–4)
S3 Architecture	I	A/R	C	C	I	C	C (Tier 3–4)	—
S4 Data Readiness	I	C	A/R	C	I	C	—	—
G2	I	R	R	C	C	C	A (Tier 3–4)	—
S5 Build/Configure/Acquire	I	C	I	A/R	—	C	—	—
S6 Assurance	I	C	I	R	A	C	—	—
G3	I	C	I	R (Tier 1–2 A)	A (Tier 3–4)	C	—	A (Tier 3–4)
S7 Deployment Authorization	I	I	I	R	C	C	—	A (Tier 3–4)
S8 Runtime & Monitoring	I	I	I	R	A	C	I	I (Tier 3–4)
S9 Change	I	C	C	R	C	C	—	I (Major, Tier 3–4)
G4	I	C	C	R	A (Tier 1–2)	C	C (Tier 3–4)	A (Tier 3–4)
S10 Retirement	I	I	A/R	R	C	I	—	C (Tier 3–4)

Stage/Gate	Business Sponsor	Architect	Data Owner	AI/System Owner	Risk & Compliance Lead	Security Architect	ARB	AI Governance Body
G5	I	I	A	R	C	I	—	C (Tier 3–4)

Transformation Governance Lifecycle RACI (new)

Table 35.

Stage	Business Sponsor	Architect	Portfolio/PMO (external)	Risk & Compliance Lead	ARB	AI Governance Body	Enterprise Governance Sponsor
T0 Purpose & Strategic Alignment	A	I	I	—	—	—	I
T1 Capability Gap Identification	R	C	I	I	—	—	—
T2 Initiative Qualification (TG0)	R	C (mandatory input, Section 6.4)	A	C (if AI-relevant)	—	—	—
T3 Portfolio Prioritization	I	I	A/R	—	C (dependency conflicts)	C — concurrence required for Tier 3–4 AI-relevant initiatives (Section 8.3)	I (contested conflicts only)
T4 Architecture & AI Enablement	I	A/R	I	C (if AI-relevant, spawns S0)	C	I	—
T5 Delivery & Organizational Change	I	I	A/R	—	—	—	—
T6 Benefits Realization & Review	A	I	R	—	—	—	—
T7 Capability Re-Baseline	R	C	I	—	—	—	—

Design principle applied consistently across both RACIs: no single role is Accountable at more than roughly a third of all stages/gates. The Enterprise Governance Sponsor is Accountable nowhere in either table, appearing only as Informed or, at T3, consulted for contested cross-body conflicts. This is the concrete evidence that the “avoid making every senior leader accountable for everything” instruction was applied.

8.7 Escalation Model

Table 36.

Trigger	Escalation Path	Timeliness Expectation
Risk tier crosses upward (Section 5.3 re-scoring)	S2 → AI Governance Body if resulting tier is 3–4	Within the current change cycle (Chapter 5, FR-17)
Control Plane Tier 3–4 alert (Chapter 7, Section 7.3)	Alert → AI/System Owner → AI Governance Body if unresolved within a risk-tier-scaled window	Tier 4: same business day; Tier 3: within 5 business days

Trigger	Escalation Path	Timeliness Expectation
G-gate rejection or dispute	Product/Solution level → ARB (architecture disputes) or AI Governance Body (deployment disputes)	Within one standing-body meeting cycle
Exception request beyond delegated authority (Section 8.5)	Requesting role → next authority level per the Decision Rights Matrix	Time-boxed per Section 8.5; no indefinite pending state permitted
TG0 bypass detected retroactively (Chapter 6's named enforcement gap, closed here)	Chapter 7 Control Plane detects an architecture/AI System repository entry with no corresponding TG0 record → routed to ARB , which has cross-functional standing neither Portfolio nor AI Governance Body alone holds → ARB refers to Enterprise Governance Sponsor if the initiative's sponsor disputes the finding	Within 10 business days of detection
Cross-body conflict (e.g., ARB and AI Governance Body disagree on a Tier 3–4 case)	Either body → Enterprise Governance Sponsor	Within one Enterprise-level cycle

The TG0 escalation path closes the gap Chapter 6 named explicitly and left unresolved (“nothing in Section 6.4, by itself, prevents an initiative from bypassing TG0”). Detection remains Chapter 7's job — retroactive and telemetry-based, per Section 7.2's Structural Drift category extended to cover repository entries without a matching TG0 reference. **Response authority is now assigned to ARB**, which settles the decision-rights question Chapter 6 deferred to this chapter.

8.8 Exception Governance

Building on FR-26 and the exception paths referenced throughout Chapters 5 and 7, an exception request requires: the specific requirement being excepted; a stated justification; a compensating control, where one exists, reducing the risk the excepted requirement would otherwise have addressed; and a hard expiry date. Per the Decision Rights Matrix (Section 8.5), every exception is time-boxed to a maximum of 90 days regardless of risk tier. Tier 4 exceptions carry no automatic renewal path at all — a Tier 4 exception reaching expiry must return through full G3 re-authorization, not a renewed exception. All exceptions are logged in a single register. That register is visible in the Transformation Scorecard's Portfolio Health metric (Chapter 6) as “initiatives operating under exception.” Exception volume therefore becomes a visible governance signal instead of an invisible accumulation of workarounds, addressing the risk named in Chapter 3 (FR-26's rationale) that rigid frameworks without a visible exception path accumulate silent, unrecorded non-compliance.

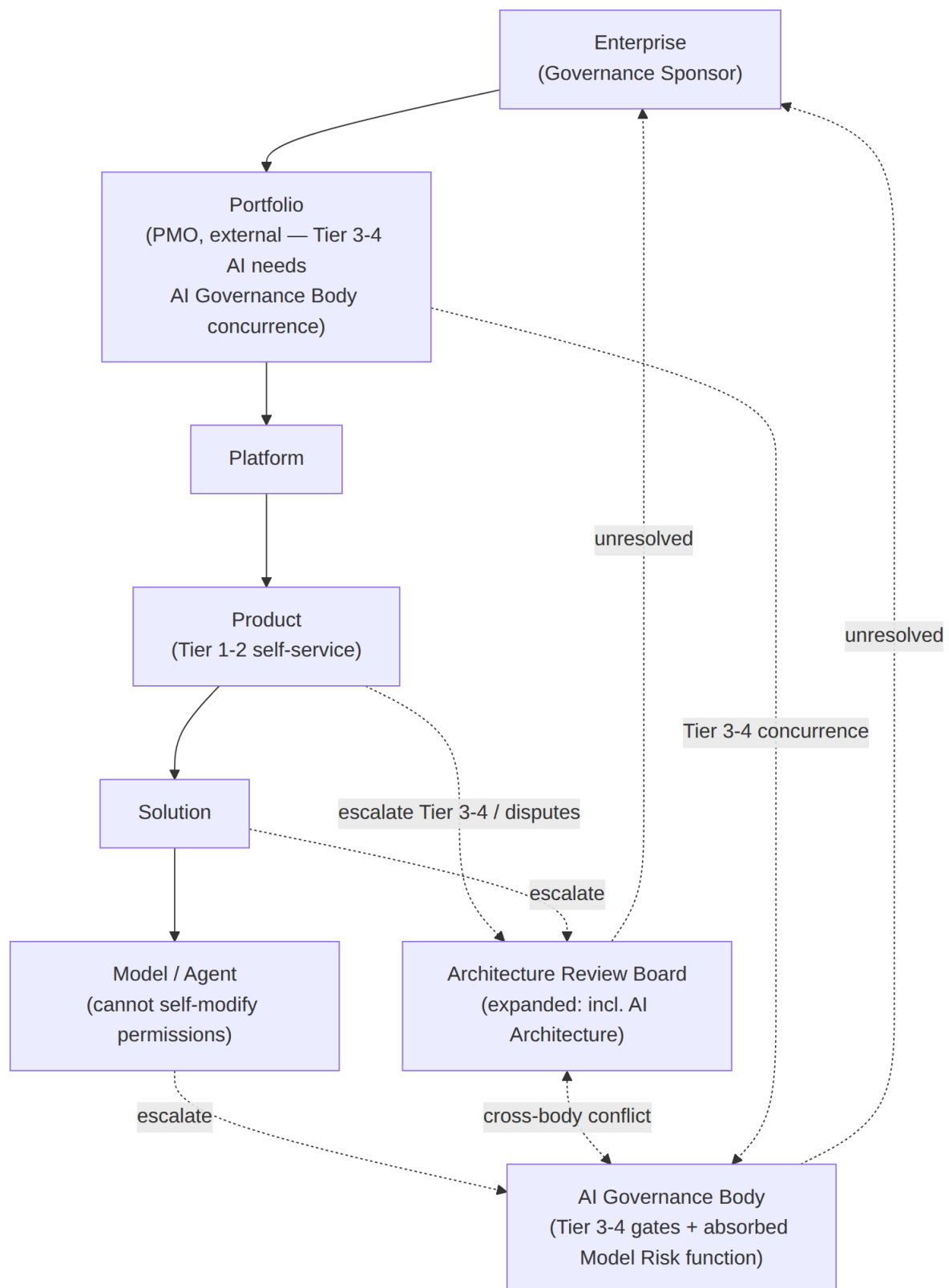


Figure 8. Exception governance flow

Chapter 8 in Summary

Governance Operating Model

Two standing bodies (expanded ARB, tier-gated AI Governance Body) replacing six candidate committees, operating within the organization's existing COBIT-aligned IT governance structure and not as a parallel stack; federated decision rights for Tier 1–2, automated-control-backed (Chapter 7), with risk-based escalation for Tier 3–4 and exceptions.

Governance Levels

Six levels (Enterprise, Portfolio, Platform, Product, Solution, Model/Agent), each with defined authority (Section 8.3), including the resolved Tier 3–4 Portfolio concurrence right.

Role Catalogue

Ten finalized roles (Section 8.4), each with purpose, accountability, authority, key decisions, owned artifacts, and escalation path, checked against the “not every senior leader is accountable for everything” discipline.

Decision-Rights Matrix

Eight decision types × four risk tiers (Section 8.5), including the tier-flat “approve new foundation model” decision.

RACI

Two lifecycle RACIs, AI Build (finalized from Chapter 5) and Transformation Governance (new), in Section 8.6.

Escalation Model

Six trigger types with defined paths and timeliness expectations (Section 8.7), closing Chapter 6's named TG0 enforcement gap.

Exception Governance

Time-boxed (≤ 90 days), no-auto-renewal-at-Tier-4 exception process with a visible register feeding the Transformation Scorecard (Section 8.8).

Chapter 9. Framework Artifacts and Templates

This chapter consolidates every record type referenced across Chapters 5 to 8 (use-case canvases, architecture decision records, risk classifications, agent cards, exception records, and benefit records among them) into one rationalized artifact set. Its central move is subtraction, not addition: several candidate artifacts prove, on inspection, to be views over data that already exists elsewhere, not new records to maintain. Rejections are recorded with their reasoning.

9.1 Rationalization Test, Applied

Every artifact below, the research brief's fifteen candidates plus everything already specified in Chapters 5–8, was tested against one question: **what decision would become impossible or materially weaker without this artifact?** An artifact that fails this test is either rejected outright or reclassified as a derived view compiled from artifacts that do pass it.

9.2 Candidate Artifact Disposition

Table 37.

Candidate (the research terms of reference.2)	Disposition	Reasoning
Enterprise Transformation Map	Rejected	Chapter 8's role catalogue already names the Portfolio roadmap as an external artifact owned by Transformation/PMO, not this framework (FR-20). A separately maintained EA version would duplicate it with no decision depending on the duplicate.
Business Capability Map	Retained — reclassified as a Derived View	No decision requires it to be independently maintained; it is the aggregate roll-up of Capability Records (Chapter 6, Section 6.2), which are the actual source of truth.
AI Capability Map	Rejected	Fully subsumed as a filtered view of the Business Capability Map (filter: AI-enablement = yes, per the Capability Record's existing field). No decision requires a separate artifact.
AI Use-Case Canvas	Retained — Primary Record	G1 (Chapter 5) has no evidence base without it.
AI System Card	Retained — reclassified as a Derived View	Genuinely valuable as a single-pane-of-glass summary, but every field it would contain already exists in another primary record (ADR, Risk Classification, Control Matrix, Deployment Authorization, Change history) — maintaining it independently would create exactly the duplicate-entry risk Section 9.4 exists to prevent.
Model Card	Rejected as a framework-owned artifact	Per Chapter 4's EA-level/engineering-level boundary (Section 4.4), a model's internal training/evaluation documentation is ML-engineering scope. Only a <i>reference</i> to an existing model card is EA-relevant, captured as one field within the AI System Card, not a framework-owned template.
Agent Card	Retained — Primary Record	FR-12's authority-boundary content (tools, actions, escalation thresholds) is explicitly EA-level per Chapter 4, and Chapter 7's agent-permission policy-as-code checks (Section 7.5) have nothing to check against

Candidate (the research terms of reference.2)	Disposition	Reasoning
AI Architecture Blueprint	Retained — Primary Record, mapped onto the existing Reference Pattern Library	without it. This is not a new artifact; it is the formal name for the reference pattern library already required by FR-06 and used throughout Chapter 5’s automation design. Introducing a separate “Blueprint” would duplicate it.
Architecture Decision Record (ADR)	Retained — Primary Record	FR-07’s foundational artifact; G2 has no evidence base without it.
Risk Classification	Retained — Primary Record	The single most load-bearing artifact in the framework — every tier-dependent decision in Chapter 8’s Decision Rights Matrix depends on it existing and being current.
Control Matrix	Retained — Primary Record	FR-16’s control-to-risk mapping; G3 evidence is incomplete without it.
Architecture Exception	Retained — Primary Record	FR-26 and Chapter 8’s exception governance (Section 8.8) have no evidentiary basis without it.
AI Change Record	Retained — Primary Record	Chapter 5, Section 5.6’s entire change-classification system depends on it; G4 cannot function without it.
Benefit Map	Retained — reclassified as a Derived View	The aggregate, portfolio-level roll-up of Benefit Records (Chapter 6, Section 6.5), which are the actual source of truth.
Assurance Dashboard	Retained — reclassified as a Derived View	Compiled from AI Assurance Summaries, Control Plane drift/alert records (Chapter 7), and Control Matrix status — no field in it is independently authored.

Four additional artifacts, not named in the project’s fifteen candidates but already load-bearing in Chapters 5–8, are added here for completeness: Capability Record (Chapter 6, Section 6.2; distinct from the Business Capability Map, which is its aggregate view), TG0 Initiative Qualification Record (Chapter 6, Section 6.4), Deployment Authorization Record (the G3 Decision entity — distinct from the ADR, which records architecture rationale, not deployment authorization), and Drift/Alert Record (Chapter 7’s Control Plane output). Each passes the rationalization test independently, as Section 9.3 sets out. Their omission from the project’s illustrative list is treated as an oversight to correct, not a signal to exclude them.

9.3 Artifact Templates

Essential Artifact Set — Primary Records

Fifteen primary records survive rationalization. Full ten-field templates are provided for five representative artifacts spanning the different record types: use-case qualification, architecture, risk, agent-specific, and a derived view for contrast. The remaining ten are specified in condensed form. This asymmetric depth follows the scoping approach of Chapter 2, which profiled four frameworks fully and condensed the rest; the reasoning is recorded here as a decision rather than left unexplained.

AI Use-Case Canvas

Table 38.

Field	Value
Purpose	Establish business justification and provisional risk signal before architecture work begins

Field	Value
Owner	Business Sponsor / Capability Owner
Trigger	A Capability Record (Section 6.2) identifies a gap judged AI-relevant, or a business need is proposed directly
Mandatory Fields	Linked Capability ID; problem statement; expected value; provisional D1–D5 risk indicators (Chapter 5, Section 5.3); named sponsor
Optional Fields	Competing alternatives considered; estimated cost range
Approval	G1 (Chapter 5, Section 5.4) — self-service for Tier 1–2, AI Governance Lead review for Tier 3–4
Update Frequency	Once at creation; superseded (not edited) if the use case materially changes before qualification completes
Retention	Retained for the life of the resulting AI System plus 3 years post-retirement, for audit traceability
Related Artifacts	Capability Record (parent); Risk Classification Record (child, created at S2)
Example	<i>“Reduce claims-processing cycle time (Capability: Claims Handling, baseline maturity 2, target 4) via a GenAI triage assistant. Expected value: 30% cycle-time reduction. Provisional D2 (Autonomy)=2 — advisory only at launch.”</i>

Architecture Decision Record (AI-extended)

Table 39.

Field	Value
Purpose	Record an architecture decision and its rationale, extended per FR-07 to capture AI-specific criteria
Owner	AI/Enterprise Architect
Trigger	S3 (Architecture stage, Chapter 5) for any new or materially changed AI System, Application, or Platform component
Mandatory Fields	Decision statement; alternatives considered; rationale; reference-pattern conformance result (Section 6.3/FR-06) or documented deviation; for AI-specific decisions — model/vendor choice, risk tier at time of decision, autonomy level, data sensitivity touched
Optional Fields	Diagrams; cost comparison across alternatives
Approval	G2 (Chapter 5, Section 5.4) — self-certified for Tier 1–2 against the pattern library, ARB for Tier 3–4
Update Frequency	New ADR per material architecture decision; superseded, not edited, when a later decision changes an earlier one (preserving history)
Retention	Retained for the life of the component plus 3 years
Related Artifacts	Reference Pattern Library (checked against); Risk Classification Record (referenced); Data Lineage Record (referenced where applicable)
Example	<i>“ADR-0142: Use enterprise RAG reference pattern v2 with [Approved Model List] Model X, grounded on [Data Product: Policy Documents, Restricted]. Deviation from pattern: none. Risk tier at decision: Tier 2.”</i>

Risk Classification Record

Table 40.

Field	Value
Purpose	Record the D1–D5 composite scoring (Chapter 5, Section 5.3) and resulting risk tier for an AI System

Field	Value
Owner	Risk & Compliance Lead
Trigger	S2 (initial); any Change Record classified Material/Major (Section 5.6); any Control Plane risk-relevant drift alert (Chapter 7)
Mandatory Fields	D1–D5 scores with justification for each; computed Impact and Control-Deficit scores; resulting tier; safety-override flag (yes/no) if triggered
Optional Fields	Comparison to prior scoring, where a re-score has occurred
Approval	Self-service for downward/lateral changes at Tier 1–2; AI Governance Body for any upward change at Tier 3–4 (Chapter 8, Section 8.5)
Update Frequency	Re-scored on every Major change trigger; reviewed at minimum annually for Tier 3–4 systems even absent a trigger
Retention	Retained for the life of the system plus 3 years; full history preserved, not overwritten
Related Artifacts	AI Use-Case Canvas (initial input); AI Change Record (re-scoring trigger); Control Matrix (downstream)
Example	<i>“D1=3, D2=2, D3=2, D4=3, D5=2 → Impact=2.67→3, Control-Deficit=2 → Tier 3 (matrix), no safety-override triggered.”</i>

Agent Card

Table 41.

Field	Value
Purpose	Record an AI agent’s identity, authorized tools, action boundaries, and escalation thresholds — the enforceable content behind FR-12
Owner	AI/System Owner, co-authored with Security Architect
Trigger	S3 for any AI System including agentic components; any autonomy-increase Change Record (always Major, Section 5.6)
Mandatory Fields	Registered machine identity; authorized tool list; action boundaries (e.g., transaction limits); reversibility of each authorized action; escalation thresholds; pre-approved kill-switch conditions (Chapter 7, Section 7.4), required at Tier 3–4
Optional Fields	Multi-agent delegation chain, where applicable
Approval	G2/G3 jointly — Tier 3–4 requires explicit AI Governance Body sign-off on kill-switch conditions specifically, not just general approval
Update Frequency	Any tool/permission change re-triggers this record and, per Section 5.6, is Material-to-Major by default
Retention	Retained for the life of the agent plus 3 years
Related Artifacts	ADR (parent architecture decision); Risk Classification (autonomy dimension D2); Policy-as-Code rules (Chapter 7, Section 7.5) enforcing the recorded boundary at runtime
Example	<i>“Agent: Claims-Triage-Agent-03. Tools: [read claims DB, draft recommendation]. NOT authorized: [approve payout]. Reversibility: all actions reversible (draft-only). Kill-switch: N/A at Tier 2 — no autonomous irreversible action authorized.”</i>

AI System Card (Derived View)

Table 42.

Field	Value
Purpose	Single-pane-of-glass summary of an AI System's governance status, compiled — not independently authored — from other primary records
Owner	No independent owner — automatically compiled; the AI/System Owner is accountable for the accuracy of the underlying records it draws from
Trigger	Regenerated on demand and on any material update to a source record
Mandatory Fields (all pulled by reference, never re-entered)	Current ADR reference; current Risk Classification tier; Control Matrix status; Agent Card reference (if applicable); most recent Deployment Authorization; open Change Records; open Exceptions; latest Assurance status
Optional Fields	Historical trend view across prior Risk Classifications
Approval	Not applicable — a compiled view has no independent approval step
Update Frequency	Real-time or near-real-time, sourced from Evidence-as-Code capture (Chapter 7, Section 7.6) where eligible
Retention	Follows the retention of its underlying source records; the compiled view itself is not separately retained beyond the current state
Related Artifacts	Compiles: ADR, Risk Classification, Control Matrix, Agent Card, Deployment Authorization, AI Change Record, Architecture Exception, Assurance Dashboard
Example	A generated report, not a manually written document — this is the artifact type Section 9.4's non-duplication principle exists to protect.

Remaining Primary Records (condensed)

Table 43.

Artifact	Purpose	Owner	Trigger	Approval	Update Frequency
Capability Record	Baseline/target maturity, gap, dependencies, AI-enablement (Chapter 6, §6.2)	Business Sponsor + Architect (joint)	Capability identified or reviewed	Business Sponsor	At T1, re-baselined at T7
TG0 Initiative Qualification Record	Evidence for the nine transformation questions (Chapter 6, §6.4)	Transformation/PMO (external), architecture input mandatory on Q4-6/8	T2	Portfolio/PMO, with mandatory architecture fields	Once per initiative, amended if scope changes materially
Reference Pattern Library entry ("AI Architecture Blueprint")	Pre-approved architecture pattern (e.g., enterprise RAG, LLM gateway) enabling FR-06 self-certification	AI/Enterprise Architect	New pattern proposed or existing pattern revised	ARB	Reviewed at minimum annually or on repeated drift signals (Chapter 7's Learn function)
Data Lineage & Sensitivity Record	FR-13/FR-14 grounding lineage and classification	Data Owner	S4	Data Owner, ARB for Tier 3-4	On any RAG/grounding source change (Material or Minor per §5.6)
Control Matrix	FR-16 control-to-risk-to-standard mapping	Risk & Compliance Lead	S6, or on control-effectiveness re-test	Risk & Compliance Lead, AI Governance Body for Tier 3-4	On Risk Classification change or scheduled control testing

Artifact	Purpose	Owner	Trigger	Approval	Update Frequency
AI Assurance Summary	Pass/fail evidence reference from Technical AI Evaluation (Chapter 5, §5.5); not the underlying evaluation methodology itself	AI/System Owner, evidence from AI/ML engineering	S6, G3	G3 authority per tier (Chapter 8, §8.5)	Every deployment and Material/Major change
Deployment Authorization Record	The G3 Decision entity — authorization to run in production	AI/System Owner (Tier 1–2) / AI Governance Body (Tier 3–4)	S7	Per Chapter 8, §8.5	Once per deployment; superseded on re-authorization (G4)
AI Change Record	Change classification (minor/material/major) and evidence	AI/System Owner	S9	Per Chapter 8, §8.5, scaled by resulting classification	Every change event
Architecture Exception	FR-26 time-boxed exception	Requesting role, approval per Chapter 8, §8.5	Any point a requirement cannot be met	Per Decision Rights Matrix, always ≤90 days	At creation; cannot be silently renewed
Drift/Alert Record	Control Plane output (Chapter 7, §7.3) — machine-generated evidentiary record	System-generated; accountable owner per Chapter 5 RACI	Any Compare function detection	N/A (detective, not a decision)	Real-time, machine-generated

Derived Views (compiled, never independently authored — full set)

Table 44.

View	Compiled From
Business Capability Map	All Capability Records (aggregate)
AI System Card	ADR, Risk Classification, Control Matrix, Agent Card, Deployment Authorization, AI Change Record, Architecture Exception, Assurance status
Benefit Map	All Benefit Records (Chapter 6, §6.5), aggregate/portfolio-level
Assurance Dashboard	AI Assurance Summaries + Drift/Alert Records + Control Matrix status, aggregated across the portfolio

Optional Artifact Set

Three artifacts improve decision quality, but no Mandatory FR or gate fails without them. They are classed as optional.

Table 45. Optional Artifact Set

Artifact	Purpose	Why Optional
Technical Debt Register Entry	Feeds Chapter 6, Section 6.3's portfolio input on debt trade-offs	Portfolio decisions can proceed using the other five Architecture Input Package elements alone; this one improves quality but is not blocking
Pattern Retrospective / Lessons-Learned Note	Feeds Chapter 7's Learn function's input to pattern-library quality	The Learn function can operate on drift-frequency data alone (Section 7.3); a written retrospective adds qualitative context but is not required for the loop to function
Stakeholder Map	Useful for complex, multi-stakeholder transformation initiatives	No gate or Decision Rights Matrix entry requires it; it is a T2/T4 planning aid, not a governance record

9.4 Artifact Integration — Relationship Map

The core integration principle: **every primary record has exactly one authoritative owner and one location; every derived view is compiled by reference, never re-entered.** This mechanism satisfies FR-19's non-duplication requirement, already applied once in Chapter 6 to Benefit Records specifically, and generalizes it to every artifact in the framework.

DO NOT COPY

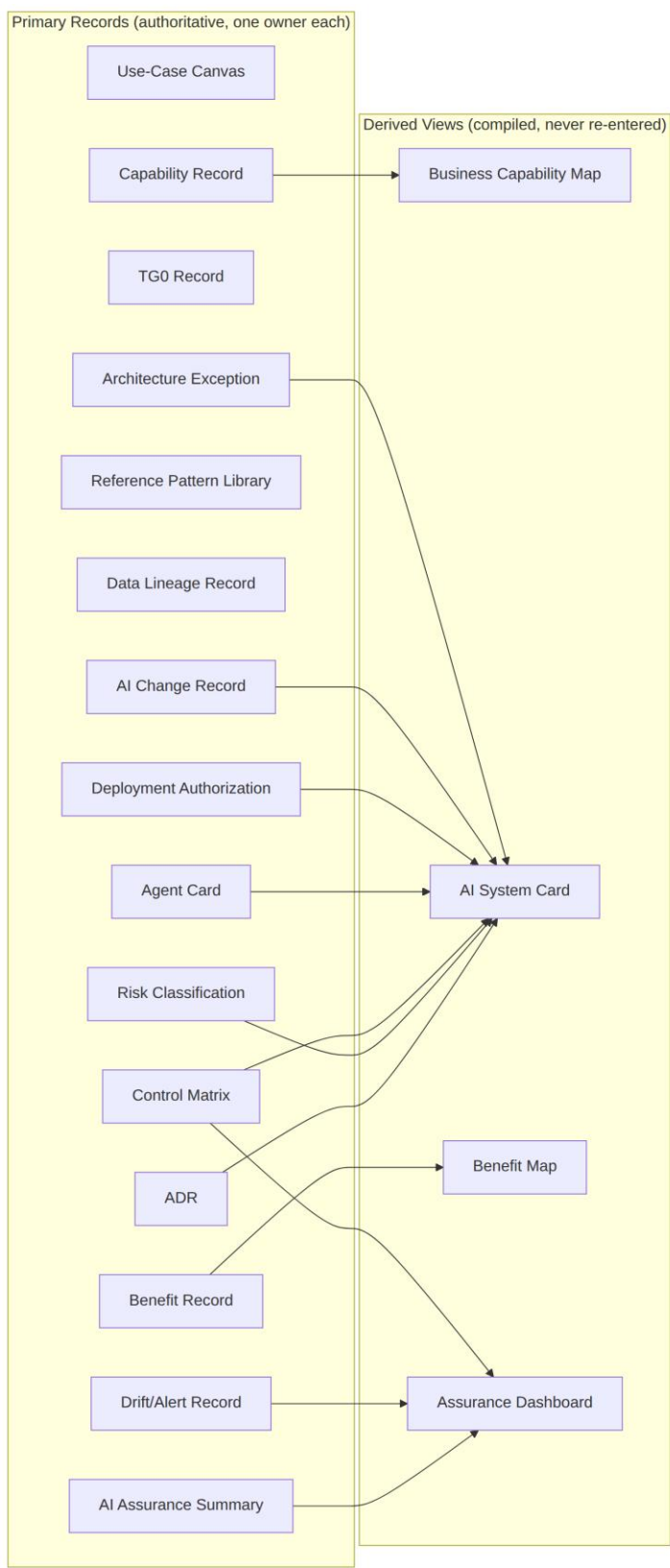


Figure 9. Artifact relationship map

Minimum Documentation Standard

Four rules, applied consistently across the full artifact set. First, an artifact exists only if it passed the Section 9.1 test, not by default inclusion in an illustrative list. That is why three of the research brief's fifteen candidates were rejected outright, and four more were reclassified as views instead of accepted as independent records. Second, no data is entered twice: if a fact belongs in a primary record, every other artifact that needs it references that record instead of repeating the value, and derived views are always compiled, never manually maintained duplicates. Third, documentation depth scales with risk tier, consistent with FR-15. The Agent Card's kill-switch field is mandatory only at Tier 3–4. The AI Assurance Summary's evidentiary depth for Tier 1 is a pass/fail reference, while Tier 4 requires the full underlying evaluation reference chain. Tier 1–2 ADRs are architect-self-certified with minimal narrative, while Tier 3–4 ADRs carry full ARB-reviewed rationale. Fourth, every artifact has a named accountable owner, with no permitted “unowned” state. This extends FR-25 from AI Systems specifically to every governance record the framework defines.

Chapter 9 in Summary

Essential Artifact Set

Fifteen primary records (Section 9.3) plus four derived views, fully specified for five representative artifacts and condensed for the remainder.

Optional Artifact Set

Three artifacts (Technical Debt Register Entry, Pattern Retrospective, Stakeholder Map) that improve decision quality without being load-bearing for any gate or Mandatory FR.

Artifact Relationship Map

The primary-record-to-derived-view compilation structure (Section 9.4 diagram), the mechanism generalizing FR-19's non-duplication principle across the entire artifact set.

Complete Templates

Five full ten-field templates (AI Use-Case Canvas, ADR, Risk Classification Record, Agent Card, AI System Card) demonstrating the pattern across primary records, agent-specific content, and a derived view; ten further primary records specified in condensed form.

Minimum Documentation Standard

Four rules (above): test-gated existence, no duplicate entry, tier-scaled depth, and universal named ownership.

Chapter 10. Framework Visualization and Identity

This chapter converts the preceding analysis into a set of explainable visual models and a provisional framework name. The naming exercise in Section 10.5 produced an unexpectedly consequential finding: of eighteen candidate names checked against products and frameworks current as of August 2026, the overwhelming majority collide with something already in the market.

10.1 Executive View

Single diagram, six nodes, for a CIO/CTO/board/transformation-leader audience. Domains, gates, tiers, and roles are omitted by design; they belong to the Practitioner View.

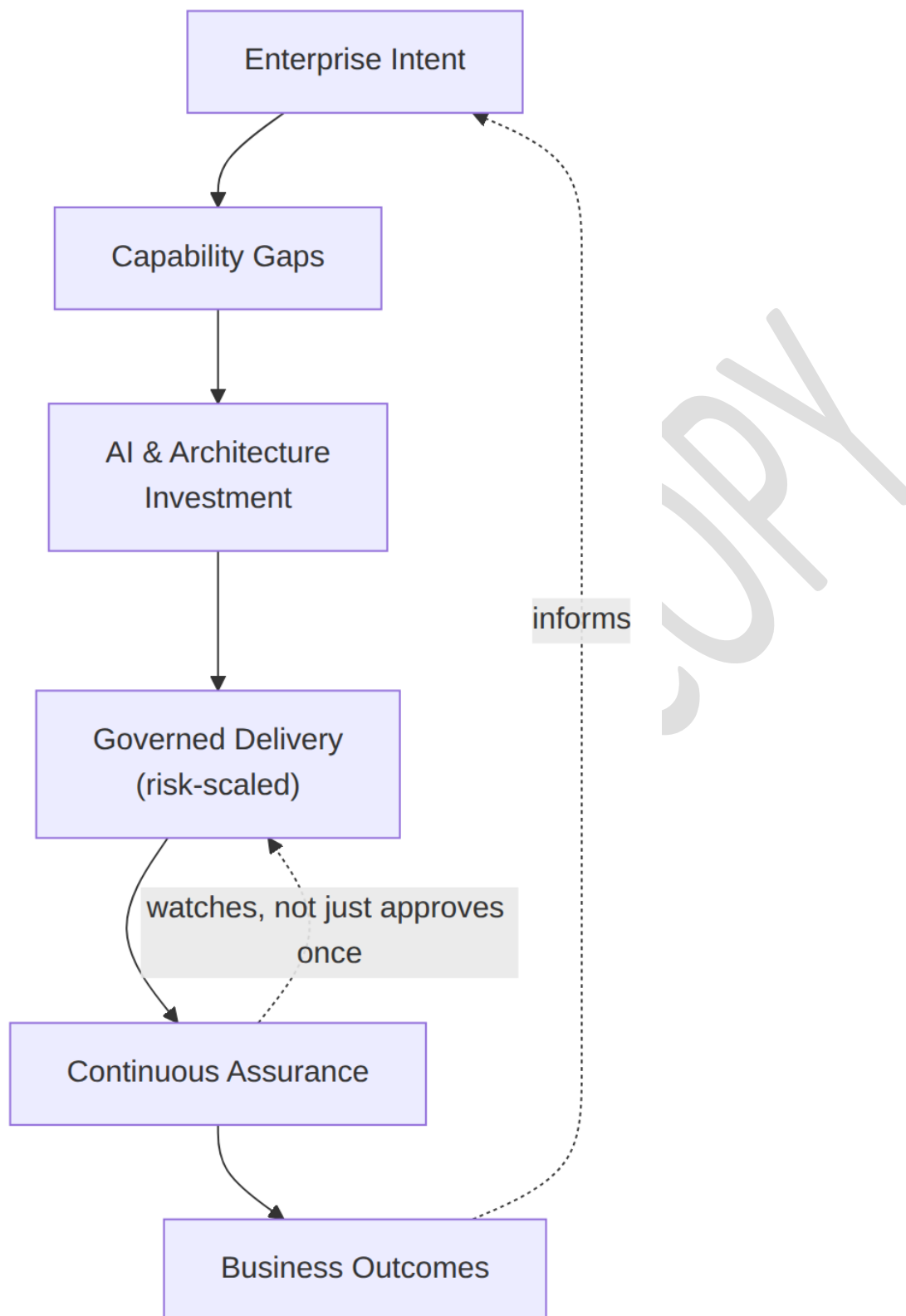


Figure 10. Framework executive view

The one idea this diagram is built to convey to an executive audience is that governance does not end at approval (the ASSURE→GOV loop). That is the framework’s central departure from conventional, document-oriented EA, put visually as well as in prose.

10.2 Practitioner View

One level down: the mechanics an architect, risk lead, or delivery lead needs to see. Lifecycle stages, gates, governance bodies, risk tiering, and the Control Plane wrap, without full metamodel detail.

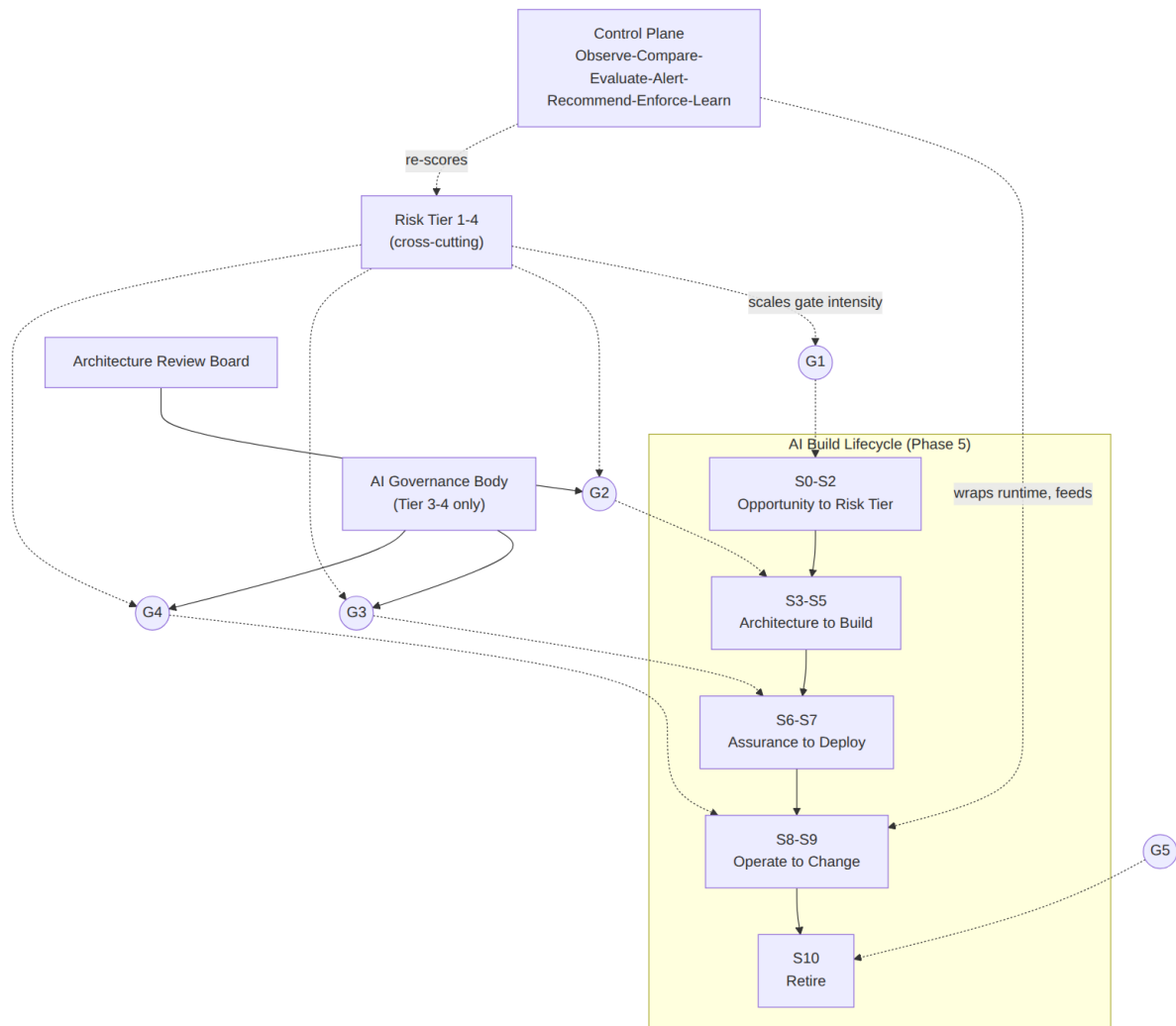


Figure 11. Framework practitioner view

10.3 Detailed Reference Model

The Chapter 4 metamodel (Diagram C), extended with the two entities Chapter 6 added (Portfolio, Benefit) that were never re-rendered visually. This is the current, complete state of the metamodel, not a repeat of Chapter 4's original.

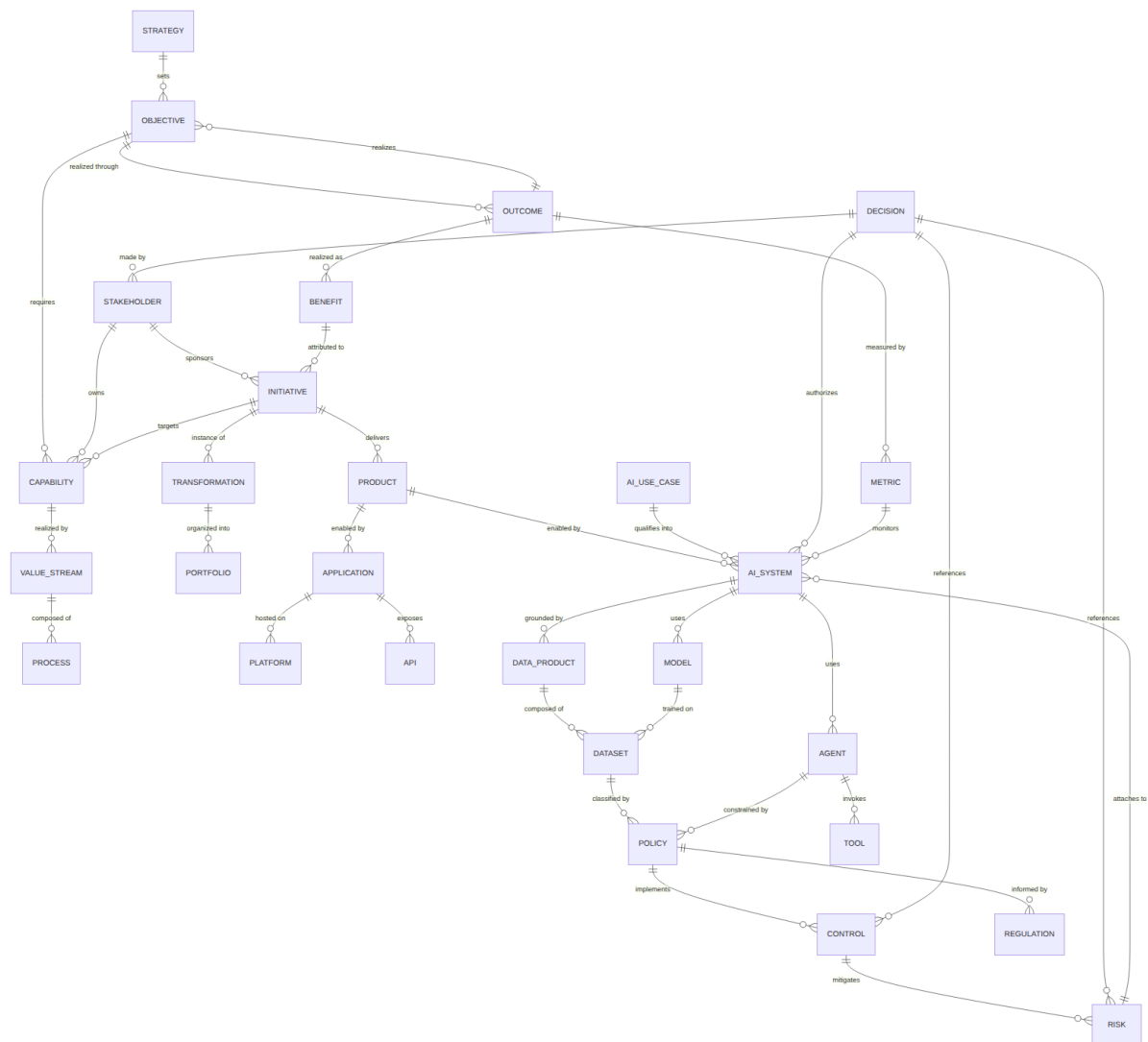


Figure 12. Detailed reference model

Framework Lifecycle Diagram

The Transformation Governance Lifecycle (T0–T7, Chapter 6) and AI Build Governance Lifecycle (S0–S10, Chapter 5) appear here as two connected but distinct cycles. This refines Chapter 6’s original: the S-lifecycle’s internal loop-backs (Chapter 5, Section 5.1) are now shown, not abstracted away.

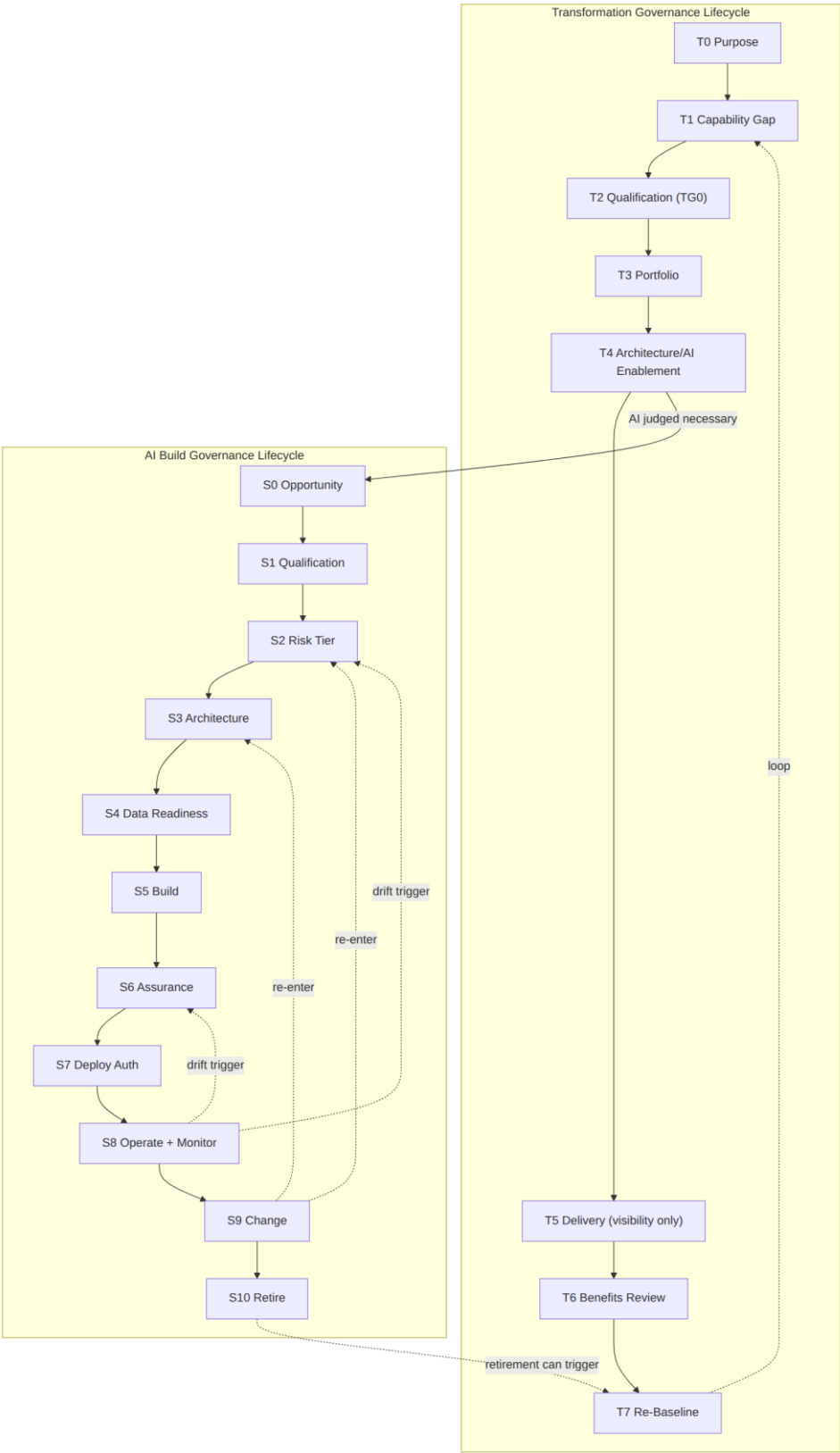


Figure 13. Framework lifecycle: transformation and AI build cycles

Governance Model Diagram

Carried forward from Chapter 8 (Section 8, governance levels and escalation), and included here so that the visual set is complete and self-contained without a cross-reference to an earlier document.

DO NOT COPY

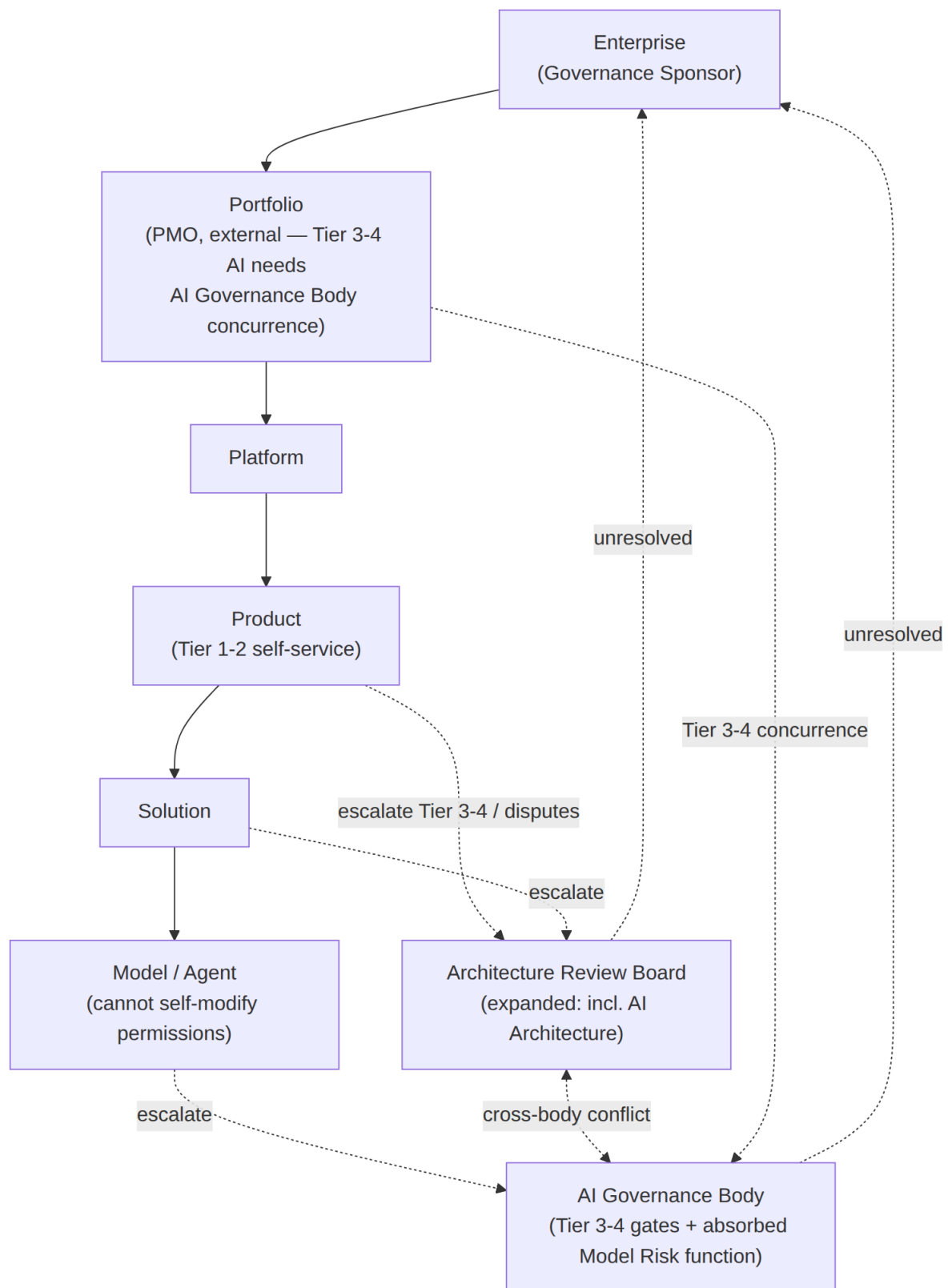


Figure 14. Governance operating model

Control Plane Diagram

Carried forward from Chapter 7 (Section 7.3), included here for completeness.

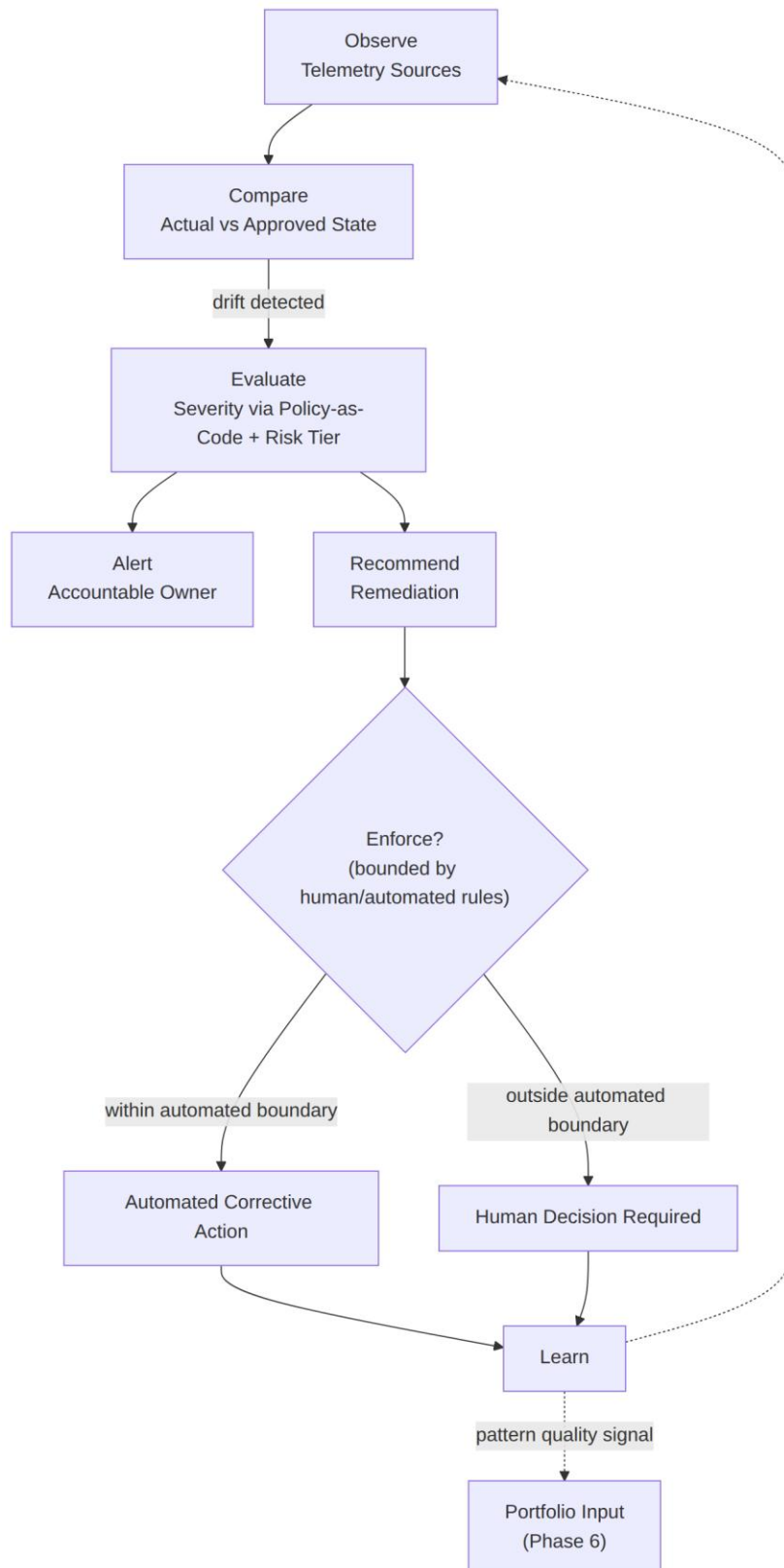


Figure 15. Continuous architecture control plane (consolidated view)

10.4 Signature Framework Diagram

Six candidate visual metaphors were evaluated against what the framework is, not against what looks most striking.

A **wheel** (equal spokes around a hub) was rejected. It implies that the framework's components are peers of equal weight, which contradicts Chapter 4's primary/cross-cutting domain distinction (Section 4.3) and the core-model/supporting-component split (Section 4.6). A **pure layered model** was rejected as the *signature* image specifically. Chapter 4 already uses layering for the Framework Layers diagram. Making a layered stack the single defining image would visually reproduce the static, top-down EA picture Chapter 1 identified as the discipline's own structural limitation, undermining the framework's central argument before a reader gets past the cover diagram. A **matrix** was rejected for the same reason it was rejected as the fundamental structure in Chapter 4 (Section 4.1): dense, static, and poor at showing the continuous feedback that distinguishes this framework's Control Plane.

An **infinity loop** was seriously considered. The T0–T7/S0–S10 dual-lifecycle structure (Chapter 6) does cross at a shared point and would render elegantly as a figure-eight. But it represents only one part of the framework (the two lifecycles), and it is better preserved as the dedicated Framework Lifecycle Diagram above than stretched to carry the whole framework's meaning. A **control-loop model alone** (just Chapter 7's Observe–Compare–Evaluate loop) was rejected as the signature image because it would overweight the Control Plane relative to the Structural and Lifecycle views Chapter 4 gave equal standing. A **cube** (domain $\times$ lifecycle-stage $\times$ risk-tier as three orthogonal axes) was the closest competitor — every gate decision in Chapter 8's Decision Rights Matrix is in fact positioned at exactly such a three-axis coordinate. It was set aside because Mermaid and standard diagramming tools cannot render a true 3D cube without misleading distortion, and because forcing an axonometric illustration risked exactly the “chosen for aesthetics” outcome this research seeks to avoid.

A layered-dependency structure was selected because the framework's own architecture is one. Chapter 4 established that eight concepts, the Core Model, are those nothing else in the framework can function without, and that everything else depends on them instead of sitting beside them. Chapter 7 established that assurance is not a stage but a continuous outer loop running over everything already in place. A concentric structure is therefore not a shape chosen in this chapter for visual effect; it is the dependency ordering the preceding chapters had already produced, drawn.

The mapping is exact, not approximate. The **Core Model** at the center is Chapter 4's eight concepts. The first ring holds the two **lifecycles**, AI build (Chapter 5) and transformation (Chapter 6), which operate on those concepts and cannot be defined without them. The second ring holds the five primary **architecture domains** (Chapter 4, Section 4.3), with Integration, Platform and Security annotated as cross-cutting instead of wedged in as false peers, consistent with their actual classification. The third ring is the **governance operating model** (Chapter 8), which sets who may decide what at each point inside it. The outer, continuously rotating band is the **Control Plane** (Chapter 7), which observes everything within and never stops — the direct visual answer to Chapter 1's central complaint that conventional architecture governance is idle

between review cycles. An earlier version of this diagram was captioned as an operating-system metaphor. That framing was withdrawn following Chapter 12’s finding that it claimed more novelty than the evidence supports. The structure it depicted was sound and is unchanged; only the over-reaching label has gone.

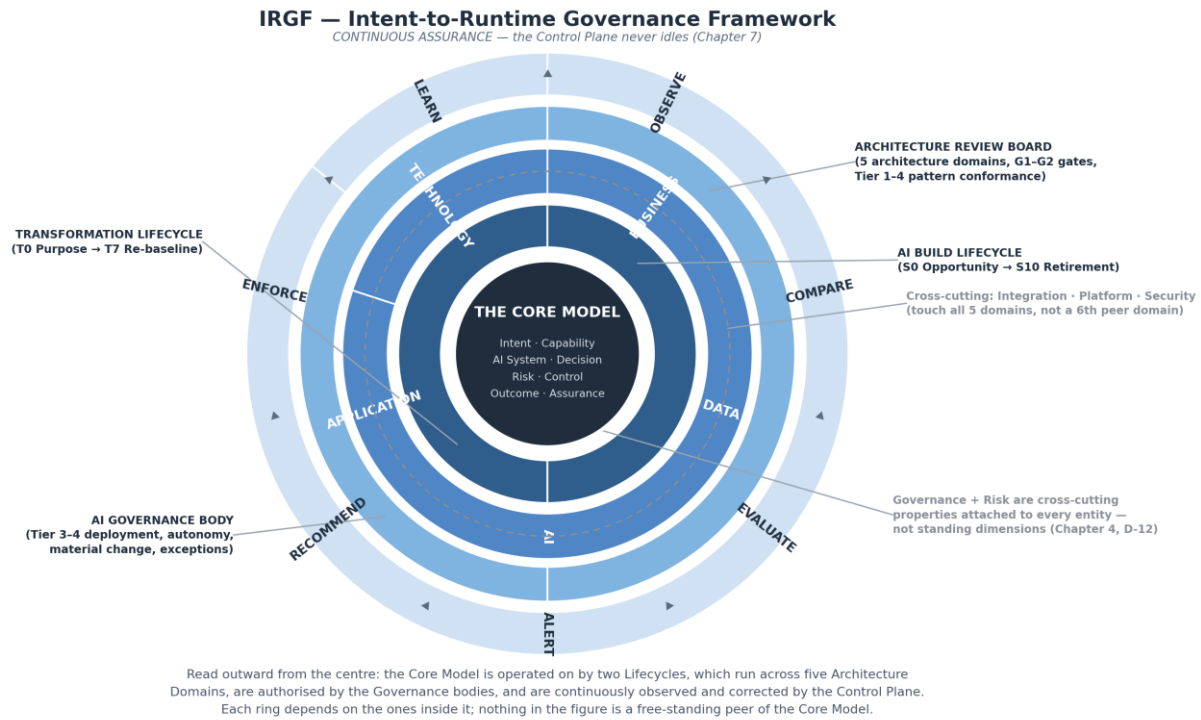


Figure 16. Signature diagram: the framework’s layered dependency structure

10.5 Name Development

Thirty-six candidate names were generated and evaluated across two rounds. Consistent with the intent not to force “AI” into the name where it would unnecessarily restrict long-term applicability, none contains “AI” as a literal token. The framework governs AI-enabled architecture today, but it is designed, per Chapter 4’s extension discipline, to generalize to whatever the next technology-driven architecture domain turns out to be.

A significant, unplanned finding emerged during evaluation: of the thirty-six candidates checked against current products, frameworks and academic work (web search, August 2026), all but three collide with something already in market or already published, several with striking closeness. An existing SSRN paper carries a near-identical title and scope to one candidate, and an active FINOS open standard occupies the architecture-as-code space of another. AGATE is already a French defense enterprise architecture framework with its own encyclopaedia entry; Keel and Trail are both live GRC and AI-governance products, the latter catalogued by the OECD; Continuous Architecture is an established published method (Erder & Pureur); Adaptive EA Governance is a Gartner-branded model with a body of research behind it; and Integrated Architecture Governance sits one word from Capgemini’s Integrated Architecture Framework. This indicates that the enterprise-architecture-plus-AI-governance

naming space is extremely crowded as of 2026, which is itself weak circumstantial evidence that the problem this research addresses is one many others are independently converging on: worth carrying into Chapter 12’s novelty assessment as context, not as proof.

Table 46. Name Development

Name	Meaning	Fit (Clarity / Originality / Academic / Enterprise)	Acronym	Collision Check (August 2026)
CALM	Continuous Architecture & Lifecycle Management	H / L / M / H	Yes	Severe — FINOS’s “Common Architecture Language Model,” an active architecture-as-code open standard
TRACE	Transformation, Risk, Architecture, Continuous Evidence	H / L / M / H	Yes	Severe — existing SSRN paper, near-identical title and scope
HELM	Steering/piloting metaphor	M / L / L / M	No	Severe — Kubernetes Helm; Stanford HELM (LLM benchmark)
LATTICE	Interconnected structure metaphor	M / L / M / M	No	Severe — Anduril Lattice (defense AI platform); AWS VPC Lattice
Continuum	Continuous governance	H / L / M / M	No	Severe — collides with TOGAF’s own “Enterprise Continuum” concept
ARC	Architecture Reference & Control	M / L / M / M	Yes	Severe — Singapore government’s ARC agentic-governance framework; ArcKit (EA governance toolkit)
AEGIS	Protection/governance shield	M / L / L / M	No	Severe — Forrester’s AEGIS Framework for agentic AI; multiple AI governance products
AXIOM	Foundational principle	M / L / M / M	No	Severe — Oteemo AXIOM (Enterprise AI Delivery Framework); Axiom Graph
FULCRUM	Pivot/balance point	M / L / L / M	No	Severe — Fulcrum Digital’s AI governance frameworks, same problem space
KEystone	Central connective piece	H / L / M / H	No	Severe — an existing published “Keystone” AI change-governance framework, near-identical scope
WARDEN	Continuous watch/protector	M / L / L / M	No	Severe — several AI agent governance products named Warden
STRATA	Layered structure	M / L / M / M	No	Severe — “STRATA,” an existing agentic-AI guardrail framework
ANCHOR	Stability/traceability roots	M / L / M / M	No	Severe — “Anchor Architecture,” existing work on AI-development traceability
NEXUS	Connection point	M / L / L / M	No	Severe — “NEXUS AI,” an existing governance-first enterprise AI framework
COMPASS	Direction/guidance	H / L / L / H	No	Severe — “Project Compass,” existing AI governance product with near-identical tagline
TESSERA	Composable tile/mosaic	M / M / L / M	No	Moderate — an enterprise AI company uses the name, different focus (not governance-framework specific)
MERIDIAN	Alignment reference line	M / M / M / M	No	Moderate — several “Meridian AI” companies, not framework-specific
VANTAGE	Elevated viewpoint	H / M / M / H	No	Moderate — several unrelated products named Vantage, none framework-specific
Archipelago	Federated islands, connected by defined bridges	M / H / M / M	No	Moderate — adjacent-domain products (insurance risk data; AEC industry), not direct governance-framework overlap

Name	Meaning	Fit (Clarity / Originality / Academic / Enterprise)	Acronym	Collision Check (August 2026)
PULSE	Continuous heartbeat/telemetry	H / L / L / M	No	Moderate — “Pulse.ai,” an enterprise AI agent platform, different focus
CADENCE	Governance rhythm/tempo	H / M / M / M	No	Low-Moderate — mostly generic-word usage; Cadence Design Systems is a large but unrelated-industry (chip design) brand
SINEW	Connective tissue — the framework’s own description of its contribution	M / H / M / M	No	Low — no usage found in AI governance, EA, GRC or assurance; only a small IT-services firm and an open-source IDE
AGATE	Architecture Governance for AI and Transformation in the Enterprise	H / L / H / M	Yes	Severe — an existing French DGA defense enterprise architecture framework
KEEL	Structural spine metaphor	M / L / L / M	No	Severe — Keel GRC, a compliance-framework platform in the same problem space
TRAIL	Audit-trail / traceability metaphor	H / M / M / H	No	Severe — trail-ml, an AI governance product listed in the OECD AI tools catalogue
Continuous Architecture	Continuous governance of architecture	H / L / M / M	No	Severe — an established published method (Erder & Pureur, 2015; 2021)
Adaptive Arch. Governance	Risk-proportionate governance	H / L / M / H	No	Moderate — Gartner’s “Adaptive EA Governance” is a branded model with a multi-report body of work
Integrated Arch. Governance	Integration as the contribution	H / L / H / M	Yes	Moderate — one word from Capgemini’s Integrated Architecture Framework (IAF)
ATRIA	Architecture, Traceability, Risk, Intent, Assurance	M / M / M / M	Yes	Moderate — an AI product marketed to architects uses the name
Throughline	The connected line from strategy to delivery	H / M / M / M	No	Moderate — a strategy consultancy trades on “strategy that stays connected”
IRGF	Intent-to-Runtime Governance Framework — names the traceability chain the framework exists to provide	H / H / H / H	Yes	None found — no product, framework or paper uses “Intent-to-Runtime” as a named construct

Only three candidates survived collision checking: IRGF (Intent-to-Runtime Governance Framework), SINEW, and the descriptive phrase *Strategy-to-Runtime*. Every other candidate was found attached to an existing product, framework, standard or published method.

Recommended name: IRGF, the Intent-to-Runtime Governance Framework. Two considerations decided it. The first is collision risk, and it is not a minor one: this exercise checked thirty-six names and found all but three already taken, several by products solving adjacent problems. A name that survives scrutiny is therefore a genuine constraint, not a preference. The second consideration, and the more important, is accuracy. “Intent” and “Assurance” are the first and last entities of the Core Model established in Chapter 4. The traceable chain running between them, from enterprise intent through architecture, risk classification and deployment authorization to runtime assurance, is precisely what Chapter 12’s novelty assessment identifies as the framework’s defensible contribution. The name therefore states the framework’s *scope*, which the evidence supports, rather than its *category of innovation*, which the evidence does not. It also corrects an earlier working name: *Enterprise Architecture Operating System* was withdrawn after the red-team review in Chapter 11 found the operating-system framing was doing rhetorical work beyond what the research could justify. The choice is

consistent, too, with the naming convention of every peer in the Chapter 2 landscape; TOGAF, COBIT, DAMA-DMBOK, NIST AI RMF and ISO/IEC 42001 are all descriptive, not metaphorical. **Formal trademark and domain-availability clearance, beyond the web searches performed here, remains outstanding and must occur before external publication or registration.**

Framework Definitions

50-word definition: IRGF extends Enterprise Architecture with AI as a first-class domain, connecting strategy, architecture, and AI-build governance through one traceable metamodel. It replaces static, periodic documentation with continuously observed, risk-tiered assurance: comparing deployed AI and architecture against what was actually approved, not merely what was once designed.

150-word definition: IRGF is a governance operating model that extends, rather than replaces, Enterprise Architecture practice (principally TOGAF) to address a gap existing standards do not close individually: no widely adopted framework connects enterprise architecture, AI-build governance, and digital-transformation governance into one continuously assured, traceable structure. IRGF adds AI as a fifth primary architecture domain, defines a risk-tiered AI Build Governance Lifecycle and a connected Transformation Governance Lifecycle, and introduces a Continuous Assurance Control Plane that observes deployed systems against approved architecture on an ongoing basis rather than at periodic review checkpoints. It deliberately does not reinvent AI risk taxonomy (NIST AI RMF (NIST, 2023), ISO/IEC 23894 (ISO & IEC, 2023b)), AI management-system requirements (ISO/IEC 42001 (ISO & IEC, 2023a)), or general IT governance (COBIT (ISACA, 2018)); it references and integrates them. Its genuine contribution is narrower and more testable: architecture-level AI governance, cross-domain traceability, and continuous conformance assurance, each still subject to formal novelty verification in Chapter 12.

Chapter 10 in Summary

Executive Framework Diagram

Section 10.1: six nodes, one central idea (assurance is continuous, not a one-time gate).

Practitioner Diagram

Section 10.2: lifecycle, gates, governance bodies, risk tier, and Control Plane wrap in one view.

Detailed Reference Model

Section 10.3: the Chapter 4 metamodel extended with Portfolio and Benefit entities.

Framework Lifecycle Diagram

Above: the dual T0–T7/S0–S10 lifecycle structure with internal loop-backs shown.

Governance Model Diagram

Above: carried forward from Chapter 8 for a complete, self-contained visual set.

Control Plane Diagram

Above: carried forward from Chapter 7 for the same reason.

Provisional Framework Name

IRGF (Intent-to-Runtime Governance Framework) — collision-checked against thirty-six alternatives, of which all but three were found to be already in use; formal trademark clearance still required before external publication.

Framework Definition in 50 Words / 150 Words

Above.

Chapter 11. Validation and Stress-Testing

This chapter attempts to break the framework, not to confirm it. It stress-tests the design against eight enterprise scenarios, subjects it to structured critique from seven reviewer perspectives, and applies an adversarial red-team pass against the framework's core legitimacy claims. The resulting revisions are consolidated as Version 0.2.

Note on method. Sections 11.1, 11.4, and 11.5 were each delegated to independent AI reviewer instances with no access to the prior design reasoning. Each reviewer received only a condensed framework briefing, with source chapters available on request, and was instructed to find failure, not to confirm success. Nine independent reviewers were used in total: four scenario reviewers (11.1), four persona reviewers (11.4), and one red-team reviewer (11.5). The purpose was to limit self-confirmation bias, since reasoning through all personas and scenarios within a single continuous authoring context risks unconsciously defending prior design decisions. Sections 11.2 and 11.3 were scoped down. Instead of running a separate six-by-four organization-size and AI-maturity grid as its own exercise, those sub-sections are synthesized from the size and maturity signals that surfaced organically within the eight scenario reviews, which spanned a regional bank, a mid-sized hospital, an engineering organization of fifty to one hundred people, a small government agency, and a forty-country multinational. This is a deliberate scope reduction, not a silent substitution; a dedicated grid exercise remains an open research question in Appendix F.

A note on the limits of this evidence. The reviews reported in this chapter are simulated adversarial evaluations, not observations of real organizational deployment. They can surface internal inconsistency, unworkable authority structures, and gameable mechanisms, but they cannot establish that a mechanism succeeds in practice. Findings should be read accordingly.

Scenario Findings (11.1)

Cross-scenario summary table

Table 47.

Scenario	Correctly Tiered?	Governance Burden	Sharpest Failure
A. Banking AI Lending	Yes — D1=4 safety-override floors Tier ≥ 3	Heavy, recurring every retraining cycle	AI Governance Body absorbing Model Risk Committee violates SR 11-7 independence expectation (approver and challenger are the same body)
B. Healthcare AI Diagnostic	Yes — D1=4 floors Tier ≥ 3	Heavy; mismatched to hospital's actual staff capacity	Zero reference anywhere in the framework to FDA/SaMD or HIPAA; D2 (Autonomy) scored as "advisory" undercounts real clinician automation-bias reliance
C. GenAI Customer Service	Ambiguous — D2/D3 judgment call swings Tier 2 vs Tier 3 for the identical system	Scales with the ambiguity above	No test anywhere for "did the system correctly recognize it should escalate," the single most safety-relevant property of this architecture
D. AI Coding Agent (auto-merge)	Yes, decisively — D2=4 is unambiguous	Disproportionately heavy for low per-action stakes (lint fixes, dependency bumps)	Autonomy-increase rule is real on paper but silently bypassable if the original G3 permission boundary wasn't captured machine-readably enough for the Control Plane to diff against
E. Autonomous	Should be, but	Heavy, no fast-path pattern	No stated minimum oversight cadence —

Scenario	Correctly Tiered?	Governance Burden	Sharpest Failure
Agent (financial)	gameable — D3 (reversibility) scoring judgment call can dodge the safety-override	exists for this class	“periodic audit” alone is nowhere prohibited as the sole human check on a Tier 3–4 standing-authority agent
F. AI SaaS Procurement	Correctly captured at S2 in principle	Backwards-shaped: heaviest scrutiny on low-volume in-house builds, lightest on highest-volume vendor risk	No Procurement role, no vendor due-diligence artifact, no Control Plane telemetry source for SaaS bought outside IT — the framework’s own named motivating problem (Chapter 1) is its thinnest-governed path
G. Government AI (small agency)	Yes — D1=4 floors Tier ≥3	Framework’s heaviest track, applied to its least-resourced scenario	No due-process/contestability mechanism for citizens; no FOIA treatment of governance artifacts themselves becoming discoverable
H. Global Transformation (40 countries)	N/A — tiering is per-system, not per-program	Undefined at scale — no inheritance/delta mechanism across 40 country instantiations	No Region/Legal-Entity governance level; the single ARB + single AI Governance Body model has no federation axis for geography, only for risk tier

Themes that recurred across four or more scenarios

1. **Sector-specific regulatory regimes are structurally invisible.** SR 11-7 (banking), FDA/SaMD and HIPAA (healthcare), FOIA/administrative law (government), and jurisdiction-specific AI regulation (EU AI Act (European Parliament and Council of the European Union, 2024) vs. none, global) all fall outside the framework’s named reference set (NIST AI RMF (NIST, 2023), ISO 23894 (ISO & IEC, 2023b), ISO 42001 (ISO & IEC, 2023a), COBIT (ISACA, 2018), ISO 27001 (ISO & IEC, 2022), DAMA-DMBOK2 (DAMA International, 2017): all cross-industry, none sector-specific). This is not an edge case; it hit 4 of 8 scenarios directly.
2. **Risk-tier self-scoring is gameable in the realistic case, not just the extreme case.** The safety-override floor (D1=4, or D2=4∧4=0) placeholder
3. **Standing-authority / continuous-effect AI systems are under-specified.** The framework’s mental model is decision-by-decision. Agents with continuous tool access (Scenario D, E) expose that “continuous conformance assurance” (state vs. approved state) is not the same claim as “continuous supervision of live actions,” and the framework nowhere distinguishes the two clearly.
4. **Two standing bodies do not scale along either the volume axis (Scenario F: vendor AI is high-volume, low-visibility) or the geography axis (Scenario H: no regional structure).**

Organization Size Test — synthesized from embedded scenario signals (11.2)

- **Small/resource-constrained (Scenario G small agency, Scenario B mid-sized hospital):** the framework’s risk-proportional intensity is calibrated to *system* risk, not *implementer capacity* — a small, high-stakes deployer is routed to the heaviest track with none of the assumed governance bench strength. Realistic failure modes: score-gaming to escape Tier 3, or governance theater (the same 2–3 people constituting “the board”).
- **Mid-sized, fast-moving (Scenario C/D, 50–100 person engineering org):** ARB and AI Governance Body collapse into the same 2–3 individuals, defeating the separation-of-duties the RACI assumes. The cross-body escalation path to an Enterprise Governance

Sponsor is close to meaningless when no one on it is distinct from the people already sitting on both bodies.

- **Large/regulated (Scenario A banking):** the framework’s assumptions fit best here, but even here it collides with a pre-existing, regulator-mandated structure (Model Risk Management) it doesn’t cleanly reconcile with.
- **Multinational (Scenario H):** breaks cleanly; no governance level below Enterprise exists to federate by geography.

AI Maturity Test — synthesized from embedded scenario signals (11.3)

- **No/early AI governance maturity (Scenario G, most of Scenario B):** the framework assumes a Control Plane telemetry backbone (cloud/IAM/model registry/CI-CD/policy-engine/FinOps integration) and a standing two-body governance structure already exist. There is no credible on-ramp from zero — a chicken-and-egg gap the Chief Enterprise Architect persona also flagged independently.
- **Early experimentation (Scenario F, most SaaS-procurement cases):** the framework’s entry point (S0/S1) assumes self-identification as “doing an AI build,” which a business unit buying a SaaS tool has no reason to do. The framework is therefore least likely to be invoked exactly where AI maturity is lowest and shadow-AI risk highest.
- **Scaled AI (Scenario A, D):** best fit; the gates, tiering, and change classification machinery add real value once an organization already has enough AI throughput to need routing logic.
- **Highly mature engineering (Scenario D, coding agent):** the framework’s rigidity (autonomy-increase = Major, always, with no fast-path) creates the most friction here: mature organizations iterate agent permissions fastest and will feel the tax hardest.

Reviewer Critiques (11.4)

Table 48.

Persona	Strongest Aspect	Biggest Weakness	Missing Concept	Unnecessary Concept	Adoption Barrier
EA Academic	Honest, provisional (not accepted) novelty admission on the Control Plane (D-27)	The Core Model’s 8 entities are asserted, not derived — no minimality test	“Traceability” (the headline claim, FR-01–03) has no operational/measurable definition — currently unfalsifiable	Governance and Risk as a <i>second</i> , orthogonal “cross-cutting property” layer on top of an already-instantiated Risk Core Model entity is a parsimony violation	Every numeric design choice (4×4 matrix, 5 gates, 90-day cap, Tier-3 floor) is asserted with zero empirical/Delphi grounding
Chief Enterprise Architect	Cutting 11 candidate gates to 5, with tier-scaled automation, is the single most credible practitioner-	ARB now carries a 5th (AI) domain with no added capacity — a bottleneck by construction, only acknowledged,	Nothing addresses AI that never enters the formal lifecycle at all (shadow SaaS, personal API keys) — the Control	Full T0–T7 lifecycle for territory (T5) the framework has deliberately declined to own	No credible on-ramp from zero to even Tier 1–2 self-service without the full telemetry backbone and two standing bodies

Persona	Strongest Aspect	Biggest Weakness	Missing Concept	Unnecessary Concept	Adoption Barrier
	facing decision in the framework	never mitigated	Plane can only see what it's wired to		already in place
CIO	Real anti-bureaucracy discipline: 6→2 bodies, Tier 1→2 self-service, 11→5 gates	No cost model anywhere — FTE, cadence, tooling spend all unstated while “minimize bureaucracy” is asserted	No per-gate cycle-time SLA — can't be put in front of a business sponsor without a number	The 7-function Control Plane loop, if the org already runs CSPM/policy-as-code, reads as a second control plane being funded twice	AI Governance Body's funding-concurrence right (D-31) is a new veto on the biggest, most impatient initiatives, with no timebox or fast appeal path
AI Governance Officer	Correct scope discipline: defers to NIST AI RMF/ISO 23894/ISO 42001 rather than re-deriving them	D5 fuses Data Sensitivity and Model Uncertainty into one number — a sensitive-but-well-characterized system and an insensitive-but-uncertain system are indistinguishable in the record	No bias/fairness methodology, no subgroup testing, no HITL/HOTL taxonomy by tier, no regulatory crosswalk (e.g., EU AI Act category mapping)	D1–D5 as a <i>freestanding</i> taxonomy rather than an explicit NIST AI RMF Map/Measure crosswalk duplicates vocabulary the framework claims not to want to reinvent	Tier 1–2 systems (where most real-world GenAI usage concentrates) get zero independent technical assurance — self-certification only
Chief Risk Officer	The Tier 3–4 funding-concurrence gate is the one place the framework has real teeth tied to a lever (capital) that changes behavior	D1–D5 has no stated provenance, no back-testing against loss data, no crosswalk to the enterprise's existing risk taxonomy	No defined roll-up from Tier/D-score into the enterprise risk register or board risk reporting	A second, standing, unvalidated risk taxonomy running in parallel to the one the enterprise already trusts	The framework's own admission that scoring is “gameable under delivery pressure” is treated as an open risk to monitor, not a near-fatal flaw requiring an independent-challenge mechanism
AI Engineer	Tier-scaled gate authority and concurrent (not sequential) S8 monitoring reflect real engineering practice	“Autonomy increase = Major, always” doesn't match how agent permissions actually get iterated (multiple times a day) — will get quietly under-reported	No governance model for continuous/silent foundation-model-provider-side drift (the model changing under the engineer without any action on their part)	G2 forcing Architecture Decision and Data Readiness to finalize jointly imposes waterfall sequencing on work that's naturally iterative	A Tier 3–4 agent's routine tool-scope iteration becomes a queue behind a body that's also carrying the ARB's 5-domain load
Digital Transformation Executive	The Benefit Record deriving value from Chapter 5's own Business Outcome Assurance metrics (not a parallel spreadsheet) is genuinely well-integrated	The Tier 3–4 funding-concurrence right hands architecture a funding veto with no equal-weight business-side counterpart to contest it	No standing business-side body / Transformation Council to argue back against a AIGB funding block — only a slow, quarterly Enterprise Governance Sponsor escalation	Numbering T5 (Delivery & Organizational Change) as a peer lifecycle stage while making it explicitly visibility-only manufactures an appearance of coverage that isn't real	The framework reserves intervention rights only where AI risk is high while disclaiming ownership of the other 90%+ of transformation work — undercuts the “digital transformation governance” name itself

Failure Points

Ranked by how directly each threatens the framework's core claims (traceability, proportional risk-tiering, continuous assurance), not by scenario count:

1. **Risk-tier gaming survives the safety-override.** The floor rule ($D1=4$, or $D2=4 \wedge D3=4$) only catches extremes. Every reviewer that touched risk scoring converged on the same finding, independently: three of four scenario reviewers, the CRO, the EA Academic, and the red-team agent. Mid-range scoring choices, plausible under pressure, can keep a Tier 3-worthy system at Tier 2 without anyone acting in bad faith, and there is currently no independent verification, challenge right, or audit-sampling mechanism.
2. **Sector-specific regulatory regimes are absent from the framework's landscape review, not merely deferred.** Chapter 2's Integration Map names only cross-industry standards. SR 11-7, FDA/SaMD, HIPAA, and FOIA do not appear anywhere in Chapters 1 to 10. This is a gap in the research base the framework was built on, not a scoping choice that was made and logged.
3. **Standing-authority AI systems (continuous tool access, not single decisions) are not clearly governed.** "Continuous conformance assurance" (state vs. approved state) and "continuous supervision of live autonomous actions" are conflated. Nothing prohibits "periodic audit" as the sole human check on a Tier 3–4 agent with standing financial or system authority.
4. **The Control Plane's own self-admitted lack of novelty (D-27, Provisional) is confirmed, not refuted, by independent review.** The red-team agent's Q5 verdict and the Chief EA persona both independently concluded that the narrower novelty claim (application to a risk-tiered, Decision-attached metamodel) is plausible but unproven. Both recommended demoting the Control Plane from "central contribution" to "recommended implementation pattern" pending Chapter 12.
5. **The two-standing-body governance model does not scale on either the volume axis (vendor/SaaS AI, the framework's own named motivating problem) or the geography axis (multinational).** Both are structural, not incidental — no fast lane exists for high-volume, low-risk vendor procurement, and no governance level exists below Enterprise for federating by jurisdiction.
6. **The SR 11-7 independence conflict is a genuine regulatory-compliance risk, not a design nuance.** Folding the Model Risk Committee function into the same AI Governance Body that authorizes deployment (Chapter 8, D-32) combines "approve" and "independently challenge" in one body — precisely the structure model-risk regulation exists to prevent, in the framework's own flagship regulated-industry scenario.

Complexity Issues

1. **Gate consolidation (11→5) is mostly bundling, not burden reduction.** G1 bundles two prior approvals, G2 bundles two more, and G3 compresses five distinct assurance types into one checkpoint. The volume of judgment calls and evidence required is roughly

unchanged; what changed is the number of labeled synchronization events. The framework's own Chapter 5/8 Open Research Questions already flagged this as unresolved. Independent review confirms it should be marketed as *consolidation*, not *burden reduction*.

2. **The six-level governance hierarchy (Enterprise→Model/Agent) is oversized for small/single-product organizations** (Scenario B, G) and undersized for multinational ones (Scenario H, no regional level) — it fits best only the mid-to-large single-jurisdiction enterprise the framework implicitly assumes throughout.
3. **T0–T7's full seven-stage numbering for a domain the framework explicitly declines to own across three of eight stages (Theme, Portfolio, T5 Org Change)** creates the appearance of governance depth the framework doesn't actually provide. Both the Chief EA and the Digital Transformation Executive personas flagged this independently.
4. **Recurring re-governance cost is invisible in the "5 gates" framing.** Because "autonomy increase" and "model/provider swap" are Major-always regardless of prior tier, and clinical/financial models retrain or get patched routinely, the *effective* cost is 5-gate-equivalent weight repeated every cycle, not a one-time cost — this compounding effect is not stated anywhere in the framework's own burden claims.
5. **No cost, staffing, or SLA model exists anywhere**, despite "minimize governance bureaucracy" being a stated design principle from the outset. Three independent reviewers (CIO, Chief EA, red-team Q9) converged on this same gap without prompting each other.

Missing Components

Consolidated, deduplicated across all nine reviewers:

- **Sector-specific regulatory overlay:** no mechanism naming or reconciling with SR 11-7, FDA/SaMD (including Predetermined Change Control Plans), HIPAA, FOIA, or jurisdiction-specific AI law (EU AI Act vs. none).
- **Procurement as a governed path and a named role:** the framework's own motivating problem (Chapter 1: AI SaaS proliferation outside IT) is its thinnest-governed path, yet Procurement appears in no role catalogue, artifact, or gate.
- **Independent verification of self-reported risk scores:** no second-party sign-off, audit-sampling, or consequence mechanism for tier misclassification.
- **Regional/Legal-Entity governance level:** no federation axis for jurisdiction, only for risk tier.
- **Minimum oversight-cadence floor for standing-authority agents:** "periodic audit alone" is nowhere explicitly disallowed for Tier 3–4 continuous-authority systems.
- **Escalation-path-integrity testing:** no assurance test anywhere for "does this system correctly recognize when it should hand off to a human," despite this being named by

reviewers as the single most safety-relevant property of GenAI-assistant and agent architectures alike.

- **D3 (Reversibility) scoring anchor ambiguity:** no guidance distinguishing mechanism-reversibility (“can we `git revert`”) from consequence-reversibility (“can we undo what already happened downstream”) — a concrete, exploitable scoring gap.
- **Machine-readable permission-boundary requirement at G3** for agentic systems: without it, the autonomy-increase detection rule has nothing to diff against.
- **Enterprise risk register crosswalk:** no defined roll-up from Tier/D-score to the categories the enterprise’s existing risk function already reports against.
- **Cost / capacity / SLA model:** no FTE, cadence, or turnaround-time commitment anywhere, despite proportionality being a stated design goal.
- **Individual due-process/contestability mechanism** for citizen- or consumer-facing automated decisions (adverse-action explainability, appeal rights): currently entirely outside the framework’s five assurance types, all of which are system-level, not individual-decision-level.
- **Independent technical assurance for Tier 1–2 systems:** currently zero, since Technical AI Evaluation is wholly owned by ML engineering with EA/governance “visibility only,” and the AI Governance Body activates only at Tier 3–4.

Recommended Revisions

Each revision below states what changes, why, and what it does *not* fully resolve, consistent with the framework’s standing rule against overclaiming.

Table 49. Recommended Revisions

#	Revision	Responds to	What it does NOT resolve
R1	Independent scoring sign-off + audit sampling. D1–D5 scoring may no longer be sole-authored by the Business Sponsor/AI Owner; the Risk & Compliance Lead must countersign at S2, and the Control Plane’s Learn function must periodically re-sample a percentage of Tier 1–2 self-certifications retroactively (cadence to be calibrated, not invented here).	Failure Point 1, CRO, EA Academic, red-team Q6	Does not eliminate gaming of <i>mid-range</i> dimension scores by a compliant-but-motivated pair of sign-offs; raises the cost of gaming, doesn’t remove the incentive.
R2	D3 scoring anchor clarified: consequence-reversibility, not mechanism-reversibility. Explicit guidance added that D3 must be scored on whether the downstream <i>effect</i> of an action can be undone, not whether the technical artifact can be reverted.	Scenario D (git-revert gaming vector)	Still a judgment call at the margins; doesn’t remove D3 ambiguity entirely, only the specific loophole found.
R3	D5 reporting split (without changing the tiering math). The Risk Classification Record must report Data Sensitivity and Model Uncertainty as two distinct sub-scores alongside the existing composite D5, so audit/regulatory readers can distinguish “sensitive-but-well-characterized” from “low-sensitivity-but-uncertain.” The 5-dimension tiering formula itself is unchanged, to avoid dimension-	AI Governance Officer	Does not add a 6th tiering dimension; this is a disclosure fix, not a scoring-methodology fix — the underlying averaging critique (CRO) remains open.

#	Revision	Responds to	What it does NOT resolve
	count creep the EA Academic warned against.		
R4	Minimum oversight-cadence and mandatory Preventive control for standing-authority agents. Any Tier 3–4 system with standing (non-per-transaction) authority now requires (a) a policy-as-code Preventive control (pre-action boundary/cap) as mandatory, not optional, at G3, and (b) an explicit maximum audit-latency commitment recorded in the Agent Card — “periodic, unspecified-interval audit” alone no longer satisfies G3 for this system class.	Failure Point 3, Scenario E	Does not specify the actual cadence number (left for calibration); does not solve the cumulative/structuring-exposure gap identified in Scenario E, which is logged below as still open.
R5	Machine-readable permission-boundary requirement at G3 for agentic systems , so the Control Plane’s drift detection has a baseline specific enough to diff against — narrative-only permission descriptions no longer satisfy G3 for agent deployments.	Scenario D, Failure Point 3	Implementation-dependent; doesn’t guarantee organizations actually build the tooling to encode/diff boundaries — logged as an adoption risk, not eliminated.
R6	Narrowed trigger for “autonomy increase = Major, always.” The rule now applies only to changes that <i>widen</i> a previously-approved, machine-readable action boundary (new consequence class, new side-effect capability). Changes that stay strictly within an already-approved boundary (e.g., a second read-only call within an approved read-only scope) may be classified Minor/self-service.	AI Engineer, Chief EA (friction complaints)	Explicitly flagged as a genuine open tension, not a resolved one: this narrowing reopens a gaming vector (teams arguing a change was “within boundary” when it wasn’t) that R1’s audit-sampling is meant to catch, but the interaction between R6 and R1 has not been tested. Logged as a new Open Research Question below rather than asserted as safe.
R7	Regulatory Overlay Reference — new artifact (16th primary record). A lightweight, Risk & Compliance-maintained mapping naming the sector-controlling regimes a given AI System is subject to (SR 11-7, FDA SaMD/PCCP, HIPAA, FOIA, EU AI Act, etc.) and stating how IRGF’s generic mechanisms (Change Classification, Compliance Assurance, gates) must be read alongside each. For regulated-industry Tier 3–4 systems specifically, this overlay may require an <i>external/independent</i> validation function distinct from the AI Governance Body — resolving the SR 11-7 conflict by exception pattern rather than by pretending the two-body model is sufficient in every regulated context.	Failure Point 2 & 6, Scenario A/B/G	Does not embed sector-specific rules into IRGF itself (correctly, per the framework’s standing “don’t reinvent domain regulation” principle) — the overlay is a pointer/reconciliation mechanism, not a compliance engine. Coverage is only as good as what’s populated into it.
R8	Procurement named as a role; ADR mandatory fields extended (not a new artifact) to require vendor AI due-diligence evidence (certifications, available model documentation, incident history, contractual data-use/training-reuse terms) before G1/G2 for procured AI. Explicit resolution of the evidence-stalemate case: if a vendor cannot/will not provide S6-equivalent evidence, the system is capped at Tier 2 regardless of D5 score, or the AI Governance Body must explicitly accept and record the evidence gap as residual risk at G3 — silence is no longer a valid outcome.	Scenario F, Failure Point 2, red-team Q9	Deliberately did not create a 16th-plus artifact for this (extended the existing ADR instead) to avoid the artifact-count creep the EA Academic and Chief EA both warned against. Does not solve shadow-AI <i>entry</i> (see R9).
R9	Control Plane Observe function gains a named (recommended, not mandatory) SaaS-discovery/expense/contract-system telemetry source category , specifically to catch AI SaaS procured outside IT.	Chief EA, Scenario F	Honestly partial: this is detection-after-the-fact at best, like the already-acknowledged TG0-bypass mitigation it’s modeled on. Shadow AI entering through personal accounts/expensed subscriptions with no corporate telemetry footprint remains undetectable by design; the framework cannot solve what it cannot observe.

#	Revision	Responds to	What it does NOT resolve
R10	New governance level: Region/Legal Entity , positioned between Enterprise and Portfolio, activated conditionally (only where jurisdiction-specific regulatory exposure or country-level Tier 3–4 volume justifies it — not mandatory for single-jurisdiction organizations, preserving proportionality for the common case). Regional instances of ARB/AI Governance Body may be stood up under this level with a defined escalation/reconciliation path to the single enterprise-level bodies. A cross-border AI System is defined as one Risk Classification Record with N jurisdiction-specific Compliance Assurance addenda (resolving the one-record-vs-N-records ambiguity).	Scenario H, Failure Point 5	Does not solve the bottleneck at the enterprise level for organizations that don't stand up regional bodies; does not specify how conflicts between a regional and enterprise ADR are adjudicated beyond "escalation," which is logged as still needing detail.
R11	Tier 1–2 systems gain a lightweight, sampling-based independent technical check (not a full Tier 3–4-style gate) — the AI Governance Body or a delegate periodically samples Tier 1–2 systems for basic technical assurance, rather than zero independent check ever.	AI Governance Officer	Sampling rate and delegation model are unspecified — deliberately left uncalibrated rather than inventing an unfounded number.
R12	T5 relabeled from a governed lifecycle stage to an "External Input Interface" in framework language and diagrams — still numbered/tracked for traceability continuity, but no longer presented as a peer stage implying ownership it doesn't have.	Chief EA, Digital Transformation Executive	Naming/framing fix only — does not add any actual governance capability over organizational change, which remains a deliberate non-goal.
R13	G2 internal sequencing loosened: Architecture Decision and Data Readiness sub-components may be worked iteratively/in parallel; both must still be resolved before G2 exit, but the gate no longer mandates a strict waterfall order internally.	AI Engineer	Does not change G2's exit criteria or evidence requirements — this is a process-flexibility fix, not a rigor reduction.
R14	Enterprise risk register crosswalk requirement — Tier/D-score outputs must map to a named crosswalk table against the enterprise's existing risk categories (financial/operational/regulatory/reputational), owned by the existing Risk & Compliance Lead role (no new role created).	CRO	The crosswalk's actual content/calibration is organization-specific and left undefined here — logged as an implementation requirement, not a solved methodology.

Explicitly NOT revised (carried forward as open, not silently dropped)

- **ARB/AI Governance Body bottleneck at true scale.** R10 (regional levels) partially addresses the geography dimension; the volume dimension (many simultaneous Tier 3–4 initiatives in a single jurisdiction) remains unmitigated. No sub-panel/delegated-review-tier mechanism has been designed.
- **Individual due-process/contestability mechanism.** After deliberation, this is logged as a **new explicit non-goal**, consistent with how the framework already treats workforce/organizational change (visibility, not ownership). Per-decision administrative and consumer-protection due process is a legal-domain concern (adverse-action notice content, appeal procedure design) that EA governance should no more own than it owns fine-tuning standards or incident-response runbooks (Chapter 3's existing exclusions). The framework will name this as an explicit interface point in the Regulatory Overlay Reference (R7) — "this AI System is subject to a due-process/appeal obligation, see [external process]" — without attempting to design that process itself. This is a scope clarification, not a capability gap left silently unaddressed.

- **Full cost/FTE/SLA model.** Not fabricated with invented numbers; carried forward as an Open Research Question requiring real calibration data the project does not have.
 - **Gameability of “cosmetic vs. substantive” change classification (§5.6).** No bright line is asserted without evidence; carried forward as an Open Research Question.
 - **R6’s reopened gaming vector.** Logged above as unresolved by design, per the project’s standing rule.
 - **Automation-bias blind spot for formally-advisory systems (Scenario B).** Addressed only as an assessor guidance note (“score D2 on formal authority, but flag effective-reliance risk narratively in the Risk Classification Record”), not as a new dimension, to avoid dimension-count creep from a single scenario’s finding.
-

FRAMEWORK VERSION 0.2

Status: Revised following adversarial stress-testing (Chapter 11). This is a targeted revision, not a rewrite. The Core Model, the S0–S10/T0–T7 lifecycles, the D1–D5 risk model’s five dimensions, the 4×4 tiering matrix, the five gates, and the two-standing-body governance model are all **retained** with their core logic intact. Fourteen revisions (R1–R14) are incorporated; six findings are carried forward as unresolved rather than papered over.

What changed from v0.1

Risk & Assurance layer - D1–D5 scoring at S2 now requires Risk & Compliance Lead countersignature, not sole authorship by the Business Sponsor/AI Owner (R1). - D3 (Reversibility Deficit) scoring anchored to consequence-reversibility, not mechanism-reversibility (R2). - D5 (Data Sensitivity & Model Uncertainty) reported as two distinct sub-scores in the Risk Classification Record alongside its existing composite value; tiering math unchanged (R3). - Control Plane’s Learn function gains a periodic retroactive audit-sampling responsibility over Tier 1–2 self-certifications (R1). - Tier 3–4 standing-authority (continuous, non-per-transaction) agents now require a mandatory Preventive policy-as-code boundary control and a stated maximum audit-latency commitment in the Agent Card; “periodic audit” alone is no longer sufficient (R4). - Tier 1–2 systems gain a lightweight, sampling-based independent technical check, replacing the prior zero-independent-check state (R11). - New Enterprise Risk Register crosswalk requirement, owned by the existing Risk & Compliance Lead (R14).

Change Classification - “Autonomy increase = Major, always” is narrowed to apply when a change *widens* a previously-approved, machine-readable action boundary; changes remaining strictly within an approved boundary may be self-service (R6) — **logged as an explicitly open tension with R1’s audit-sampling control, not a fully resolved trade-off**. - G3 Deployment Authorization Records for agentic systems must now encode machine-readable permission/action boundaries, not narrative-only descriptions, so the above rule is actually enforceable by Control Plane drift detection (R5).

Artifacts - ADR mandatory fields extended to require vendor AI due-diligence evidence for procured/integrated AI, with an explicit resolution path when a vendor cannot provide it (Tier-2 cap, or AI Governance Body documented risk acceptance) (R8). - **New 16th primary artifact: Regulatory Overlay Reference.** A Risk & Compliance-maintained mapping of sector-specific controlling regimes (SR 11-7, FDA SaMD/PCCP, HIPAA, FOIA, EU AI Act, etc.) per AI System, stating how generic IRGF mechanisms interact with each. For regulated Tier 3–4 systems, it may require an external/independent validation function distinct from the AI Governance Body (R7).

Roles - **Procurement** added to the Role Catalogue, consulted at S1/S2 for any procure/integrate/SaaS-sourced AI system (R8).

Governance Levels - **New conditional level: Region/Legal Entity**, positioned between Enterprise and Portfolio, activated only where jurisdiction-specific regulatory exposure or country-level Tier 3–4 volume justifies it. A cross-border AI System is one Risk Classification Record with N jurisdiction-specific Compliance Assurance addenda (R10).

Control Plane - Observe function gains a recommended (not mandatory) SaaS-discovery/expense/contract-system telemetry source, framed as partial, detection-after-the-fact mitigation for shadow AI (R9).

Transformation Governance - T5 (Delivery & Organizational Change) relabeled in framework language/diagrams from a governed lifecycle stage to an **External Input Interface**; numbering and traceability role retained, ownership framing corrected (R12). - G2 internal sequencing loosened to allow iterative/parallel resolution of Architecture Decision and Data Readiness sub-components before joint exit sign-off (R13).

Novelty posture (carried from Chapter 7, reinforced by independent Chapter 11 review) - The Control Plane's mechanism remains marked **non-novel** (extension of CSPM/DevOps/policy-as-code practice); its status stays **Provisional**, not Accepted, per independent red-team confirmation. Pending Chapter 12's formal novelty assessment, Framework v0.2 recommends positioning the Control Plane as a *recommended implementation pattern*, not a headline contribution. The narrower, more defensible novelty claim rests on the 8-category drift taxonomy's Data-Lineage/AI-Model-Behavioral separation and the Tier 3–4 default-prohibition on automated corrective enforcement.

What did NOT change, and why

The Core Model's eight entities, the six functional dimensions, the five architecture domains, the eleven S-stages, the eight T-stages, the five gates, and the two-standing-body model all survive Chapter 11 review intact. Independent reviewers criticized their *sufficiency* and *scaling behavior*, not their *basic validity*. No reviewer, scenario, or red-team question produced evidence that the core metamodel is wrong, only that specific mechanisms built on top of it (scoring self-certification, sector regulatory integration, procurement pathway, geographic federation, standing-authority oversight) were underspecified. Revising the Core Model itself in response to implementation-layer gaps would violate the project's own discipline against churning terminology and major decisions without evidence requiring it.

Explicitly unresolved, carried into Chapter 12+

1. ARB/AI Governance Body volume bottleneck within a single jurisdiction (only the geography dimension was addressed, via R10).
2. Individual due-process/contestability mechanisms: reclassified as an explicit non-goal (interface point only, via the Regulatory Overlay Reference), not a capability gap to be closed by IRGF itself.
3. Full cost/FTE/SLA model: no invented numbers; requires real calibration data.
4. Bright-line test for “cosmetic vs. substantive” change classification.
5. The R6/R1 interaction (narrowed autonomy-increase trigger vs. audit-sampling as its backstop): untested.
6. Cumulative/structuring exposure tracking for standing-authority agents (sub-threshold actions accumulating between audit cycles); no taxonomy category currently covers this.

Major Findings

- The framework's core metamodel (Core Model, six dimensions, five domains, two lifecycles) survived nine independent adversarial reviews without a single finding that invalidated it structurally. All confirmed weaknesses were in mechanisms layered on top of it (scoring integrity, sector regulatory integration, procurement pathway, geographic scale, standing-authority oversight), not in the metamodel itself.
- Risk-tier self-scoring gameability, previously an acknowledged-but-untested open risk (Chapter 3, Chapter 8), is now **confirmed as a live, exploitable gap** by three independent lines of evidence (three scenario reviewers, the CRO persona, and the red-team agent, all converging without shared context).
- Sector-specific regulatory regimes (SR 11-7, FDA/SaMD, HIPAA, FOIA) are absent not just from the framework's mechanisms but from Chapter 2's landscape review itself: a research-base gap, not merely a scoping choice.
- The framework's own self-assessment of the Control Plane's non-novelty (Chapter 7, D-27) is independently confirmed, not refuted: a rare and creditable instance of a stress test validating a framework's own humility instead of exposing hidden overclaiming.
- The two-standing-body governance model, while a real simplification versus six candidates, does not scale along either the volume axis (vendor/SaaS AI) or the geography axis (multinational). Both gaps are structural, not incidental.

Implications for the Framework

- Framework v0.2 is measurably more defensible against regulated-industry and multinational adoption objections than v0.1, at the cost of one new artifact, one new role, and one new conditional governance level — a small, targeted bureaucracy increase in exchange for closing the most consequential gaps found.
- The Control Plane's positioning should shift in future chapters (especially Chapter 12's novelty assessment and Chapter 10's visualization materials, if revisited) from "central mechanism" toward "recommended implementation pattern," with novelty argued narrowly.
- Chapter 12 (Novelty & Contribution Assessment) now has sharper, evidence-backed material to work with. The red-team agent's Q1–Q10 verdicts, taken together, argue for a narrower novelty claim than the framework has implicitly carried through Chapters 1–10; Chapter 12 should treat this as its starting hypothesis to test, not assume the broader claim.

Issues Requiring Validation

- Whether R6 (narrowed autonomy-increase trigger) and R1 (audit-sampling backstop) actually interact safely, or whether R6 reopens more risk than R1 closes; this needs real pilot data, not further desk review.

- Whether the Regulatory Overlay Reference (R7) is adopted with real content by regulated organizations, or becomes an unmaintained placeholder like the risk Chapter 9 already flagged for Derived Views.
- Whether the conditional Region/Legal Entity governance level (R10) is actually activated appropriately, or becomes either over-used (unnecessary regional bureaucracy) or under-used (multinational orgs stay on the single-body model that Scenario H showed doesn't fit).
- Whether independent scoring sign-off (R1) meaningfully reduces gaming in practice or merely shifts the gaming target to convincing the countersigning Risk & Compliance Lead.

Chapter 12. Novelty and Contribution Assessment

This chapter compares the framework against existing approaches to establish where genuine contribution exists. Every component is classified as Existing, Extension, Integration, or New Contribution. Each surviving novelty claim is then put to an anti-novelty test: could TOGAF (The Open Group, 2022), COBIT (ISACA, 2018), NIST AI RMF (NIST, 2023), ISO/IEC 42001 (ISO & IEC, 2023a), and existing practice already accomplish it? Where the answer is yes, the component is repositioned or withdrawn.

Note on method. This chapter is a synthesis-and-judgment exercise applied to evidence already established: the landscape review in Chapter 2, the self-assessed non-novelty of the Control Plane recorded in Chapter 7 (D-27, status provisional), and, most consequentially, the nine independent adversarial reviews reported in Chapter 11, whose findings did substantial novelty-testing work as a byproduct of testing for legitimacy. The task here is to turn that evidence into a component-by-component ruling, not to re-litigate it. Where the evidence warrants, this chapter downgrades prior claims instead of confirming them, and it does so in several places.

1. Method

For every component of IRGF carried into v0.2, two tests are applied in sequence:

1. **Classification.** Existing (a restatement of established practice/standard) / Extension (a direct, incremental addition to an established practice, staying within its own logic) / Integration (a genuine connective mechanism binding two or more existing practices that were not previously bound) / New Contribution (a mechanism with no clear prior art, existing/extension/integration classifications exhausted).
2. **Anti-novelty test.** *Could TOGAF + COBIT + NIST AI RMF + ISO 23894 (ISO & IEC, 2023b) + ISO 42001 + existing architecture/MLOps/GRC practice already accomplish this, without IRGF?* A “yes” does not automatically disqualify a component; many “yes” components are still worth keeping for their **Integration** value, since the connective tissue itself is the point. It does disqualify a component from being marketed as a **New Contribution**. And where a component adds process weight without adding anything a practitioner couldn’t already assemble from existing standards, it is flagged for removal or repositioning.

Validation status from Chapter 11, where a component was actually stress-tested, is carried into the table below. A component surviving as “New Contribution” but *failing* Chapter 11’s stress test is marked accordingly. Novelty and correctness are separate questions, and this chapter does not conflate them.

2. Component-by-Component Novelty Classification

Table 50.

Component	Classification	Nearest Prior Art	Chapter 11 Validation Status
-----------	----------------	-------------------	------------------------------

Component	Classification	Nearest Prior Art	Chapter 11 Validation Status
AI as a fifth, first-class EA architecture domain (Business/Data/AI/Application/Technology)	Extension	TOGAF's content framework is explicitly extensible; informal "AI as an EA domain" proposals already circulate in practitioner literature (Gartner-style EA guidance)	Red-team Q2: real but narrower than advertised — governance/taxonomy teeth, no change to who owns AI's technical substance
S0–S10 AI Build Governance Lifecycle bound to EA gates (G1–G5)	Integration	MLOps/LLMOps lifecycle stages are Existing industry practice (and reflected in NIST AI RMF's Govern-Map-Measure-Manage structure); EA-style architecture gates are Existing (ADM phases). Binding the two was not previously standard.	Not separately stress-tested as a whole; sub-components below were
D1–D5 risk dimensions (as a taxonomy)	Extension	Decision Consequence Severity and Data Sensitivity are standard categories in essentially every risk framework; Autonomy and Reversibility are less commonly formalized as explicit scoring axes but are consistent with, not distinct from, NIST AI RMF/ISO 23894/EU AI Act (European Parliament and Council of the European Union, 2024)'s underlying concerns	Red-team Q4: "Extension/Integration, not New." Confirmed.
Safety-override floor (D1=4, or D2=4∧D3=4, forces Tier ≥3 regardless of matrix result) + binding of Tier to gate-automation level	New Contribution (narrow)	No reviewed standard prescribes an equivalent floor-plus-automation-binding mechanism; NIST AI RMF discusses risk tiering conceptually but not as a mechanical floor rule tied to governance automation	Confirmed real in Scenarios A/B/G (correctly prevented under-tiering); confirmed gameable at the margins in Scenarios C/D/E (Chapter 11 Failure Point 1) — novel but not yet safe
Agent Card / Identity→Permission→Tool→Policy→Action→Verification→Audit chain	Integration	Agent governance concepts (least privilege, machine identity, sandboxing) are Existing in security practice generally; binding them into a single EA-owned artifact tied to AI-specific change classification is not standard as of this research	Scenario D/E: mechanism sound in concept, enforcement gap found (baseline-granularity requirement added as R5 in v0.2)
"Autonomy increase = Major, always" change-classification rule	New Contribution (narrow, unvalidated)	No reviewed AI governance standard specifies an equivalent mechanical binding between an autonomy change and mandatory re-governance	AI Engineer/Chief EA personas: real friction; found gameable and later narrowed (D-46) with the narrowing itself flagged as an open, untested trade-off — novel but currently unstable
Control Plane (Observe→Compare→Evaluate→Alert→Recommend→Enforce→Learn)	Existing	CSPM, DevOps observability, and policy-as-code practice already implement this loop	Self-admitted non-novel at Chapter 7 (D-27); independently confirmed by red-team Q5 and Chief EA persona, who both recommend demotion to "implementation pattern"
8-category drift taxonomy, specifically the deliberate separation of Data-Lineage/Governance drift from AI-Model-Behavioral drift	New Contribution (narrow)	DAMA-DMBOK2 (DAMA International, 2017) covers data lineage generically; MLOps observability covers model drift generically; the explicit, EA-governance-embedded separation of the two as distinct tracked categories is not found named this way elsewhere in the reviewed landscape	Not directly stress-tested in Chapter 11's scenarios; flagged in Chapter 12 as needing empirical testing before being trusted, not merely asserted
Preventive/Detective/Corrective control types	Existing	Standard COBIT/ISO 27001 (ISO & IEC, 2022) control taxonomy	N/A
Tier 3–4 default-prohibition on automated	Integration	Synthesizes existing change-management (ITIL (AXELOS, 2019)	Not separately stress-tested; logically sound, unvalidated

Component	Classification	Nearest Prior Art	Chapter 11 Validation Status
corrective enforcement, with a single pre-approved kill-switch exception defined at G3		emergency change), security (break-glass procedure), and AI-safety (kill-switch) norms — no reviewed standard binds all three into one governance rule specifically for AI	in practice
Two-standing-body governance model (ARB expanded + AI Governance Body absorbing Model Risk Committee function)	Extension	Committee consolidation is standard organizational-design practice, not EA-specific theory	Red-team Q8: “org-chart simplification... not new governance theory,” and independently found not to scale on volume (Scenario F/E) or geography (Scenario H) axes — a component that is neither novel nor, as specified, fully sound
Core Model (8 entities: Intent, Capability, AI System, Decision, Risk, Control, Outcome, Assurance) + “one metamodel, three views”	Extension	Metamodeling itself is standard EA practice (ArchiMate (The Open Group, 2026), Zachman); this is a specific instantiation, not a new modeling paradigm	EA Academic (Chapter 11): “asserted, not derived — no minimality test.” Cannot be classified higher than Extension without that derivation.
Purpose→Benefit 12-link traceability chain	Existing	Strategy-to-outcome traceability is standard EA/balanced-scorecard/OKR practice	N/A
Per-link traceability- depth classification (deep/shared/shallow-with-input/reference-only)	New Contribution (narrow, modest)	Traceability modeling tools (ArchiMate relationships) model <i>that</i> links exist, not their governance <i>depth</i> ; this distinction is not found named elsewhere in the reviewed landscape	Not stress-tested directly; low-risk, low-ceremony addition
Benefit Record deriving value from Business Outcome Assurance metrics (transformation value traceability)	Integration	Balanced-scorecard-style benefit tracking is Existing; connecting it specifically to AI lifecycle assurance metrics rather than a parallel spreadsheet is the actual contribution	Digital Transformation Executive (Chapter 11): confirmed as “the one place transformation governance is earned honestly”
Tier-scaled gate automation (“adaptive risk-based architecture governance”)	Integration	Risk-based/proportional control intensity is Existing in IT audit and COBIT practice generally; applying it specifically to bind AI risk tier to EA gate automation is the connective contribution	Red-team Q7: real at the scheduling/UX level, but gate-count reduction itself ruled largely cosmetic at the evidentiary level — the <i>binding logic</i> survives, the <i>burden-reduction claim</i> does not
Policy-as-code / architecture-as-code framing	Existing	Policy-as-code (OPA, Sentinel-style tooling) is mature, general industry practice by the time of this research	N/A
Regulatory Overlay Reference (v0.2 addition, D-47)	Integration	GRC compliance-mapping tooling already does “map generic controls to specific regulations” generically; embedding this as a named, owned EA artifact reconciling sector regimes with IRGF’s own generic mechanisms is not found in the reviewed EA/AI-governance standards themselves	New in v0.2; unvalidated — Issue flagged in Chapter 11 (does it get populated and maintained, or decay like Derived Views)
Conditional Region/Legal Entity governance level (v0.2 addition, D-50)	Existing	Federated regional architecture governance is already standard practice in most multinational EA functions; this addition fills a gap the framework should have had from v0.1 , rather than introducing new theory	Closes Scenario H’s finding; not itself a novelty claim
Independent scoring sign-off + audit sampling (v0.2, D-41)	Extension	Second-line risk challenge (three-lines-of-defense model) is Existing, standard risk-governance practice; applying it specifically to AI risk self-certification	Responds directly to CRO persona’s explicit request for exactly this pattern

Component	Classification	Nearest Prior Art	Chapter 11 Validation Status
		is a direct, unsurprising extension	

3. Anti-Novelty Test Applied to the Surviving “New Contribution” Claims

Six components survived initial classification as New Contribution. Each is now tested against the specific question: *could TOGAF + COBIT + NIST AI RMF + ISO 23894 + ISO 42001, used competently together by an existing EA/risk function, already produce this without IRGF?*

1. **Safety-override floor + Tier-to-gate-automation binding.** No. None of the five reference standards specifies a mechanical floor rule, and none binds a risk outcome to a specific governance-automation level. A skilled practitioner *could* invent this independently, and plausibly some already have, informally; but it is not documented, named, or specified in any of the reviewed standards. **Verdict: survives as New Contribution, with the caveat that Chapter 11 found it exploitable at the margins — it is a novel proposal, not yet a novel proven safeguard.**
2. **Autonomy-increase change-classification rule.** No standard specifies this. However, Chapter 11 found it operationally unstable (gamed by omission, later narrowed at cost of reopening a different gap). **Verdict: survives as New Contribution on originality grounds, but must be presented as an unvalidated proposal requiring pilot testing, not as a proven mechanism.**
3. **Drift taxonomy’s Data-Lineage/AI-Model-Behavioral separation.** No. DAMA-DMBOK2 and MLOps observability practice each cover one half; neither, nor NIST AI RMF, names the distinction as such. **Verdict: survives as New Contribution.** This is arguably the cleanest, lowest-risk novelty claim in the framework — it costs nothing to implement, being a taxonomy label and not a new control, and no Chapter 11 finding contradicts it.
4. **Tier 3–4 corrective-enforcement default-prohibition + kill-switch pre-approval rule.** A sufficiently senior architect *could* assemble this from ITIL emergency-change process + security break-glass procedure + general AI-safety kill-switch discourse. Each piece exists. **Verdict: this is Integration, not New Contribution, reclassified down one level from its Chapter 7/10 framing.** The value is real, since nobody had written this specific synthesis down as an EA governance rule before. But by the anti-novelty test’s own standard, “an informed practitioner could already do this by combining known parts” disqualifies a New Contribution claim. It qualifies for Integration.
5. **Traceability-depth classification.** No standard or common EA tool names this distinction. **Verdict: survives as New Contribution,** but modestly so: a documentation-quality improvement, not a governance mechanism, and it should not be oversold as framework-defining.
6. **Regulatory Overlay Reference.** GRC compliance-mapping tools already do the generic version of this. **Verdict: this is Integration, not New Contribution,** reclassified down from its initial v0.2 framing. It is valuable specifically *because* it sits inside the EA

artifact set instead of a separate GRC tool most IRGF adopters won't already own. That is an integration argument, not an originality argument.

Net result of the anti-novelty test: of six initially-claimed New Contributions, **four survive** (safety-override/tier-automation binding, autonomy-increase rule, drift-taxonomy separation, traceability-depth classification) and **two are reclassified down to Integration** (corrective-enforcement rule, Regulatory Overlay Reference). None of the four survivors is validated as *correct*, only as *original* — Chapter 11 already found real cracks in two of the four: the safety-override floor's marginal gameability, and the autonomy-increase rule's enforcement gap and unstable narrowing.

4. Components Formally Repositioned

Consistent with the requirement to remove or reposition overclaimed components instead of retaining them:

- **Control Plane mechanism:** repositioned from “central novel mechanism” (its Chapter 4/10 framing) to **recommended implementation pattern**, formalizing what Chapter 7 proposed provisionally and Chapter 11 confirmed independently. It remains in the framework, since it is good practice, but it is no longer part of the novelty case.
- **Two-body governance model:** repositioned from “novel governance operating model” to **practical consolidation guidance**. It is a legitimate, evidence-informed recommendation: six candidates reduced to two for good organizational-design reasons. But it is organizational design practice, not a research contribution, and Chapter 11 additionally found it doesn't scale as specified — a further reason not to rest any novelty claim on it.
- **Core Model:** repositioned from “the framework's foundational theoretical contribution” to **a working ontology, proposed but not yet derived or validated**, pending the kind of expert Delphi review Chapter 33 (Research Validation Strategy) will need to actually perform. Continuing to use it as the organizing structure for the framework is reasonable; claiming it as settled theory is not.
- **D1–D5 taxonomy:** repositioned from “the framework's own risk model” (implying a competing taxonomy) to **a NIST AI RMF / ISO 23894-consistent implementation schema**. IRGF's dimensions should be presented in Chapter 14 as an operationalization of NIST's Map/Measure functions for architecture-governance purposes, not as an alternative risk taxonomy competing for the same intellectual space. A formal crosswalk table (D1–D5 ↔ NIST AI RMF categories ↔ ISO 23894 categories) is recommended as required content for the Chapter 14 final specification; it is not fabricated here without the source material to do it properly.
- **Regulatory Overlay Reference and corrective-enforcement rule:** repositioned from New Contribution to Integration, per Section 3 above.

Nothing is recommended for outright **removal**. Every component tested either earns Integration-or-higher status on its own connective merits or, in the case of the two-body model and Control Plane, remains valuable as practical guidance once its novelty claim is withdrawn. **The anti-**

novelty test did not surface any component that is pure, valueless duplication — the harshest realistic finding is “this is good synthesis, not new theory,” never “this is unnecessary.”

5. Revised Gap Matrix (updated from Chapter 2)

Table 51.

Reference Standard	What It Already Covers Well	Does IRGF Duplicate It?	What IRGF Actually Adds Relative to It
TOGAF / ADM	Architecture method, domain taxonomy (BDAT), governance board structure, ADRs	Partially — IRGF’s board structure and ADR practice are direct TOGAF-lineage extensions, not replacements	AI as a named 5th domain; binding of architecture gates to AI-specific risk tiering; continuous (not periodic) conformance checking
Zachman / ArchiMate	Metamodeling notation and relationship taxonomy	No — IRGF’s Core Model is a specific ontology, not a competing notation	An AI/risk/governance-specific entity set (unvalidated minimality, per Section 4)
COBIT / ISO 38500 (ISO & IEC, 2024)	General IT governance, control types, risk-based control intensity	Partially — control-type vocabulary (Preventive/Detective/Corrective) is reused, not reinvented	Binding of control-type permissibility to AI risk tier specifically (the Tier 3–4 corrective-prohibition rule)
NIST AI RMF / ISO 23894	AI risk taxonomy and lifecycle functions (Govern/Map/Measure/Manage)	Substantially — D1–D5 is best understood as an implementation schema of these functions, not a parallel taxonomy (repositioned, §4)	Mechanical binding of risk outcome to EA governance-gate automation level; the safety-override floor rule
ISO 42001	AI management system requirements	No — IRGF explicitly defers to ISO 42001 for AIMS scaffolding	Embeds ISO 42001-referenced evidence into EA-specific gates and artifacts rather than requiring a separate AIMS audit trail
ISO 27001 / NIST CSF (NIST, 2024)	Security controls	No — deferred entirely	Security Architect role integrated into AI-specific RACI (though Chapter 11 found this integration under-weighted, still “Consulted,” never “Accountable,” a known gap not yet fixed)
DAMA-DMBOK2	Data governance and lineage methodology	No — deferred, extended only at the AI-grounding seam	Data-Lineage vs AI-Model-Behavioral drift separation (confirmed New Contribution, §3)
CSPM / DevOps / policy-as-code practice	Continuous observability, drift detection, automated policy enforcement	Yes, substantially (confirmed non-novel, §2–3)	Application to a risk-tiered, Decision-attached AI/architecture metamodel — a narrower, still-unvalidated claim
GRC compliance-mapping tooling	Mapping generic controls to specific regulatory regimes	Yes, at the generic-mechanism level	Embedding as an EA-owned artifact (Regulatory Overlay Reference) rather than requiring a separate GRC tool

6. Overall Positioning Determination

The research terms of reference required this determination to be reached through analysis, not assumed in advance. Five candidate positionings were named: (1) extension of TOGAF; (2) overlay on TOGAF; (3) complementary governance framework; (4) next-generation EA

framework; (5) fundamentally new architectural governance paradigm. The evidence from this chapter rules out four of the five:

- **Not (1) an extension of TOGAF.** IRGF adds substantial lifecycle and risk-governance content (S0–S10, T0–T7, D1–D5, the gate model) that goes beyond what “extending TOGAF” normally means; it is not merely new ADM phase content.
- **Not (2) an overlay on TOGAF.** An overlay implies sitting on top without touching TOGAF’s own structure. IRGF modifies the core domain taxonomy itself (AI as a 5th domain) and the ARB’s composition, which is more invasive than an overlay.
- **Not (4) a next-generation EA framework.** This chapter’s own findings rule this out directly. A “next generation” claim requires the core theoretical apparatus (the Core Model, the risk taxonomy, the Control Plane, the governance model) to represent a genuine advance, and §2–4 above found most of that apparatus to be Extension or Integration, with several components (Control Plane, two-body model) downgraded after independent stress-testing.
- **Not (5) a fundamentally new paradigm.** No evidence anywhere in ten chapters of design work or Chapter 11’s nine independent reviews supports this. It was never seriously in contention once the anti-novelty test was applied.
- **(3) A complementary governance framework — confirmed.** IRGF’s actual, defensible identity is an **integration and governance-orchestration layer** that binds TOGAF-style architecture practice, AI-specific risk/management standards (NIST AI RMF, ISO 23894, ISO 42001), and generic IT/security/data governance standards (COBIT, ISO 27001, DAMA-DMBOK2) into one traceable structure spanning enterprise architecture, AI build, and (more shallowly) digital transformation. It does so without replacing, competing with, or duplicating the substance of any of those standards. Its genuine research contribution is concentrated in four narrow, still-unvalidated mechanisms (§3) plus the connective/binding work itself (§5), not in a new theoretical foundation.

This conclusion is materially more modest than the framework’s original Chapter 10 branding implied: “Enterprise Architecture Operating System,” with a kernel at the center of concentric rings. Per the red-team reviewer’s independently-reached Q10 verdict in Chapter 11: *“the ‘Operating System’/Kernel framing is doing real rhetorical work to make this synthesis read as more paradigm-shifting than its component parts actually are.”* This chapter concurs, and the finding was acted on. The framework was subsequently renamed **IRGF (Intent-to-Runtime Governance Framework)** and the central entity set renamed the **Core Model**, retiring the operating-system metaphor altogether. The present name states the framework’s scope, the chain it makes traceable from enterprise intent to runtime assurance, and asserts no category of innovation the evidence in this chapter does not support.

7. Final Novelty Statement

IRGF is a complementary governance framework, not a new architectural paradigm and not a TOGAF replacement. It does not introduce a new AI risk taxonomy, a new AI management-

system standard, a new general IT governance model, or a new continuous-observability mechanism. Each of those already exists, and IRGF defers to the existing standard instead of re-deriving it. Its genuine, defensible contribution is narrower. It consists of: (a) the connective/binding work that makes AI risk tier mechanically govern architecture-gate automation level and control-enforcement permissibility; (b) a small number of specific, original mechanisms, namely the safety-override floor, the autonomy-increase change-classification rule, the Data-Lineage/AI-Model-Behavioral drift separation, and traceability-depth classification, each narrow in scope and, per Chapter 11, each still requiring empirical validation before being trusted as sound and not merely original; and (c) the single coherent structure (the Core Model and its three views) that lets an organization trace one AI system from strategic intent through architecture, risk tier, deployment authorization, and runtime assurance without switching frameworks mid-chain. This is a real and worthwhile contribution, and a bounded one. It is not the “operating system for the AI enterprise” its own branding implies, and this chapter recommends that framing be softened in the final specification.

Major Findings

- Of roughly twenty candidate framework components tested, the large majority (13 of ~19 in the table) classify as Existing, Extension, or Integration. This confirms the project’s own repeated internal warnings (Chapters 2, 7, 9, 11) that most of the framework’s value is synthesis, not invention.
- Four components survive the anti-novelty test as genuine New Contributions. Two previously-claimed New Contributions (the corrective-enforcement rule, the Regulatory Overlay Reference) are reclassified down to Integration, on the grounds that a sufficiently informed practitioner could already assemble them from existing standards.
- None of the four surviving New Contributions is validated as *correct*, only as *original*. Two of the four (the safety-override floor, the autonomy-increase rule) were independently found by Chapter 11 to have real, currently-unresolved weaknesses. Originality and soundness are kept as separate findings, per this project’s standing practice against overclaiming.
- No component tested qualifies for outright removal. The harshest finding produced by the anti-novelty test is “this is good synthesis, not new theory,” never “this is unnecessary duplication.”
- The overall positioning question (Section 3) is now resolved with evidence: **complementary governance framework**, not TOGAF extension, not overlay, not next-generation EA, not new paradigm.

Implications for the Framework

- Chapter 14’s final specification must state the novelty claim at the level established here: narrow, four-mechanism, and explicitly unvalidated. It should not revert to the broader framing implicit in Chapter 10’s visualization and naming work.

- The four surviving New Contributions should be the focus of Chapter 13’s academic research questions and hypotheses; they are the only components for which “is this actually a contribution” is a live, researchable question. The Existing/Extension/Integration components are not appropriate subjects for novel empirical research, only for adoption/implementation research.
- The D1–D5-to-NIST/ISO crosswalk table (flagged in D-59) is now a required, not optional, piece of Chapter 14 content. It was not produced in this chapter because doing so rigorously requires the source standards’ actual clause-level text, which this project has treated as external reference material throughout instead of reproducing verbatim.

Issues Requiring Validation

- Whether the four surviving New Contributions hold up under real pilot deployment, not just desk review (this chapter can only assess originality, not real-world soundness).
- Whether softening the OS/Core Model framing in Chapter 14 changes stakeholder reception in a way worth testing empirically (a Chapter 13/33-style validation question, not resolvable here).
- Whether the Core Model’s 8-entity minimality claim would survive an actual Delphi panel, as opposed to this project’s internal reasoning.

Chapter 13. Academic Research Model

This chapter develops the conceptual model, constructs, relationships, research questions, propositions, and validation methodology through which the framework’s claims could be empirically tested. It also assesses which theoretical foundations support the framework and which would amount to academic decoration.

Note on method. Consistent with the closing recommendation of Chapter 12, this chapter builds on the narrowed contribution established there: the complementary governance framework positioning, and the four surviving, still-unvalidated new contributions. Those four are the safety-override and tier-automation binding, the autonomy-increase change-classification rule, the separation of data-lineage from AI-model-behavioral drift, and traceability-depth classification. It does not re-inflate the broader claims Chapter 12 withdrew, namely Control Plane novelty, two-body governance novelty, and treatment of the Core Model as settled theory. This is a research-design exercise rather than fieldwork. The work is to convert the preceding design and stress-test evidence into a testable academic model, not to generate new evidence.

1. Revisiting the Primary Research Question

Chapter 1’s preliminary research question was broad by design, appropriate for an exploratory problem-definition stage: *how can Enterprise Architecture be extended or reconceptualized to provide an integrated governance framework for AI build, AI-enabled systems, and enterprise digital transformation?* Chapter 12 answered the identity half of that question with evidence (Complementary Governance Framework, not extension/overlay/next-generation/new paradigm), but it left the *effectiveness* half untested. It could establish what IRGF *is*, not whether its actual mechanisms *work*. That is the gap this chapter’s research model is built to close.

Revised Primary Research Question:

Under what organizational and risk conditions do explicit mechanical bindings between AI risk classification, architecture-governance intensity, and autonomy-change detection (as instantiated in IRGF’s four surviving novel mechanisms) measurably improve risk-proportional governance outcomes relative to the loosely-coupled status quo (TOGAF (The Open Group, 2022), NIST AI RMF (NIST, 2023)/ISO 42001 (ISO & IEC, 2023a), and COBIT (ISACA, 2018) applied independently, without an explicit binding layer), and under what conditions do they break down?

This is narrower and more falsifiable than the Chapter 1 version, by design. It names the specific mechanisms at stake; it names a comparison condition (loosely-coupled standards use, the realistic status quo this framework competes against, per Chapter 12’s positioning); and it asks a boundary question (“break down”) instead of assuming universal applicability. That last move responds directly to Chapter 11’s finding that organizational capacity and risk context matter more than the framework’s original design work assumed.

2. Theoretical Foundations

A central question for this chapter is which theoretical perspectives strengthen the framework and which merely decorate it. Six candidate theories were evaluated against that test; three are load-bearing, two are secondary/contextual, and three are rejected as decorative for this specific, now-narrowed contribution.

Adopted as load-bearing

IT Governance theory (Weill & Ross, 2004, decision-rights archetypes). This is the framework’s actual disciplinary home, not an exotic addition. The Decision Rights Matrix, the federated Tier 1–2/Tier 3–4 split, and the six governance levels are a direct application of the federated/centralized/decentralized decision-rights archetypes IT governance theory already formalizes. This grounding was implicit through Chapters 4–8 but never named — Chapter 14 should name it, since it is the academic home against which reviewers will evaluate the governance-operating-model content.

Cybernetics (Ashby, 1956, Law of Requisite Variety; feedback control). Directly explains *why* a continuous-assurance loop is theoretically necessary and not merely convenient: a governance mechanism can only regulate a system whose variety (rate and diversity of state change) it can match. A periodically-reviewed architecture has far lower requisite variety than a continuously-retrained, tool-using AI agent — that is the theoretical reason periodic EA governance is insufficient for AI-intensive environments, and it holds even though the Control Plane’s *implementation* mechanism was found non-novel in Chapters 7/11/12. Novelty and theoretical grounding are separate claims; this theory strengthens the latter without rescuing the former.

Socio-Technical Systems Theory (Trist & Bamforth, 1951, joint optimization). Directly explains a finding Chapter 11 surfaced empirically before this chapter had a theory for it: the Scenario B (healthcare) finding that a formally-advisory AI system (D2 scored low, “human makes every decision”) can carry much higher *effective* autonomy through clinician automation bias. The core claim of STS theory is that social and technical subsystems must be jointly optimized, and that a purely technical specification will misjudge real risk when human behavior mediates it. That is precisely what the healthcare finding illustrates. Chapter 14 should cite this theory wherever D2 (Autonomy) scoring guidance is discussed.

Adopted as secondary / contextual

Dynamic Capabilities Theory (Teece et al., 1997, sense–seize–transform). Maps cleanly onto the Control Plane’s Observe/Compare/Evaluate (sense) → Recommend/Enforce (seize) → Learn/T7 Capability Re-Baseline (transform) sequence, and gives a defensible reason to prefer continuous adaptation over periodic re-planning as the framework’s operating assumption. It frames the *transformation*-governance half of the model (T0–T7) well. But it is not central to the four surviving novel mechanisms, so it is kept secondary, not load-bearing.

Institutional Theory (DiMaggio & Powell, 1983, isomorphism and decoupling). Directly explains a specific, concerning failure mode Chapter 11 found rather than hypothesized: Scenario G’s “governance theater” pattern, where a small, resource-constrained organization stands up nominal governance bodies to satisfy process without substantive independent review. Institutional theory’s decoupling concept, formal structure adopted for legitimacy while diverging

from actual practice, is the correct academic lens here, and it motivates Secondary Research Question 5 and Proposition 5 below. Kept secondary because it explains an *adoption risk*, not a *design mechanism*.

Technology–Organization–Environment (TOE) framework (Tornatzky & Fleischer, 1990). Useful for structuring future adoption research, since it separates organizational context, technological context, and environmental/regulatory context as independent adoption determinants — these map directly onto the org-size and regulatory-intensity variance Chapter 11’s scenarios exposed as the framework’s most consistent fault line. Recommended as the organizing frame for the case-study design set out below, not as a source of new framework mechanisms.

Rejected as decorative for this contribution

Resource-Based View (RBV; Barney, 1991). A stretch: IRGF is a governance framework, not itself a firm resource in the VRIN sense, and no finding from Chapters 1–12 depends on or is illuminated by competitive-advantage reasoning. Including it would add academic vocabulary without adding explanatory power, precisely the parsimony violation the EA Academic persona (Chapter 11) flagged elsewhere in this project.

Complexity Theory (complex adaptive systems, emergence, self-organization). Tempting given the source material’s language (“autonomous systems,” “emergent AI behavior”). But the framework’s actual mechanisms (hierarchical gates, tiered risk scoring, a Core Model of fixed entities) are conventional governance-design constructs, not operationalizations of self-organization or emergence. Adopting Complexity Theory here would be citation without engagement; rejected.

Organizational Ambidexterity (exploration/exploitation; March, 1991). Relevant in spirit to the “prevent over-governance” design goal of this research, but not embodied in any specific mechanism tested in this project — worth a passing mention in Chapter 14’s discussion of governance-speed trade-offs, not worth building research constructs around.

3. Conceptual Research Model — Constructs and Relationships

The model below treats the four surviving New Contributions (Chapter 12) as the mechanisms under test. Organizational context (per TOE) enters as a moderator, and institutional decoupling (per Institutional Theory) as a rival explanation the model must rule out before crediting any observed effect to the mechanisms themselves.

Constructs

Table 52.

Construct	Definition	Type	Theoretical Grounding
Risk-Tier Fidelity (RTF)	The degree to which an assigned D1–D5/Tier classification matches a system’s actual (independently-assessed) risk profile, as opposed to a self-serving or miscalibrated score	Outcome / mediator	Cybernetics (measurement fidelity is a precondition for effective regulation)
Autonomy-Change Detection Fidelity	The degree to which autonomy-increasing changes to an AI system are actually captured and	Outcome / mediator	Cybernetics; Institutional Theory

Construct	Definition	Type	Theoretical Grounding
(ACD)	routed through governance, as opposed to silently bypassing it		(decoupling risk)
Governance-Automation Binding Strength (GAB)	The degree to which a system's actual governance intensity (gate rigor, review frequency) is coupled to its assigned risk tier in practice, rather than decoupled from it	Mediator	IT Governance theory (decision rights should track risk-bearing)
Governance-Practice Decoupling (GPD)	The gap between an organization's formally adopted governance structure (boards, gates, artifacts exist on paper) and its substantively practiced governance (independent review actually occurs)	Moderator / rival explanation	Institutional Theory
Organizational Governance Capacity (OGC)	An organization's staffing, telemetry maturity, and pre-existing committee infrastructure available to run the framework as specified	Moderator	TOE framework
Traceability-Depth Awareness (TDA)	Whether stakeholders correctly perceive which links in the Purpose→Benefit chain are deeply governed versus shallow/reference-only, as opposed to assuming uniform governance depth	Outcome	(Framework-specific; grounded loosely in IT Governance theory's transparency principle)
Drift Root-Cause Diagnosis Speed (RCD)	Time/accuracy to correctly diagnose the root cause of an AI-system incident, specifically for incidents spanning the data/model boundary	Outcome	Cybernetics (signal differentiation improves control response)
Risk Mitigation Effectiveness (RME)	The ultimate outcome construct: whether governed AI systems exhibit fewer/less severe risk-realized incidents than ungoverned or loosely-coupled-standard-governed comparators	Dependent variable	—
Perceived Governance Burden (PGB)	Practitioner-perceived friction/delay attributable to governance mechanisms	Trade-off outcome, tracked against RME	—

Conceptual Model Diagram

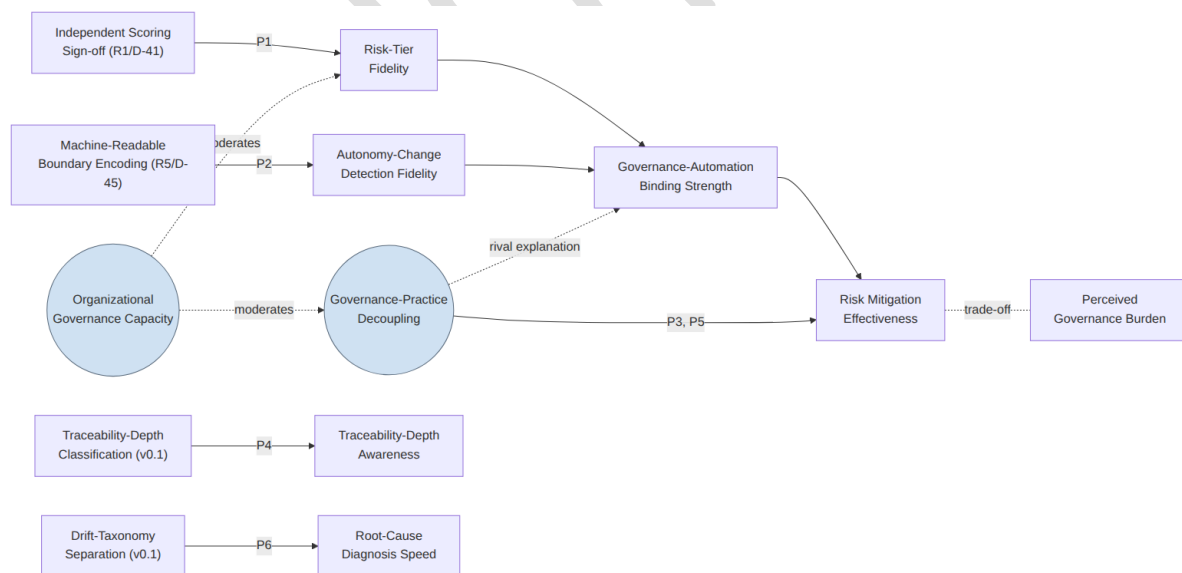


Figure 17. Conceptual research model: constructs and relationships

The two moderator/rival-explanation constructs (Organizational Governance Capacity, Governance-Practice Decoupling) are drawn as circles and shaded to mark them structurally distinct from the mechanism-outcome chain. Any empirical test of P1–P3 or P5 that does not measure and control for both risks attributing an effect to IRGF's mechanisms that actually

belongs to organizational capacity or institutional pressure. That is the single most important methodological caution this chapter can hand to future validation work, and it exists because of what Chapter 11's scenario reviewers found: the same finding, org size and capacity swamping mechanism design, recurred in five of eight scenarios.

4. Research Questions

Primary Research Question: stated in §1 above.

SRQ1 (tests R1/D-41): Does independent (second-party) sign-off on AI risk-tier self-scoring measurably improve Risk-Tier Fidelity compared to sole self-certification? And does the effect hold when the countersigning role lacks structural independence from delivery incentives (the role-collapse condition found in Chapter 11's 50–100 person engineering-org scenarios)?

SRQ2 (tests R5/D-45, and the D-46 narrowing tension): Does encoding machine-readable permission boundaries at deployment authorization improve Autonomy-Change Detection Fidelity for agentic AI systems compared to narrative-only architecture records, and at what velocity of legitimate permission iteration does the detection mechanism's latency exceed the window needed to prevent harm?

SRQ3: How does Organizational Governance Capacity moderate the relationship between assigned AI risk tier and actual Governance-Automation Binding Strength? Specifically, does low capacity at high risk tier (the Scenario B/G finding: hospital, government agency) predominantly produce score-gaming (lower RTF), governance-theater decoupling (higher GPD), or both?

SRQ4: Does explicit per-link traceability-depth classification measurably change Traceability-Depth Awareness among architecture and risk stakeholders, compared to organizations using an undifferentiated ("uniformly traceable") strategy-to-outcome chain? And does increased awareness translate into different audit behavior at shallow-classified links, or only into more accurate (but unchanged) expectations?

SRQ5 (Institutional Theory-driven): To what extent does organizations' actual adoption behavior match IRGF's stated non-competing, integration-only design intent (Chapter 12's Complementary Governance Framework positioning), versus diverging toward compliance-checkbox adoption under coercive isomorphic pressure (regulatory audit expectations) regardless of design intent?

SRQ6 (tests the drift-taxonomy separation): Does separating Data-Lineage/Governance drift from AI-Model-Behavioral drift as distinct, independently-tracked categories improve Root-Cause Diagnosis Speed for incidents spanning the data/model boundary (e.g., RAG grounding failures), compared to the merged "data drift" categorization common in current MLOps observability tooling? And is any improvement concentrated at that boundary, or general?

5. Propositions

P1: Independent second-party sign-off on risk-tier scoring will increase Risk-Tier Fidelity relative to sole self-certification, but the effect will be significantly attenuated (not eliminated) in organizations exhibiting role collapse, where the same individuals hold both the scoring and the countersigning roles.

P2: Machine-readable permission-boundary encoding will increase Autonomy-Change Detection Fidelity relative to narrative-only records, but detection latency will still exceed the harm-prevention window for high-velocity systems (e.g., auto-merging coding agents) unless paired with a mandatory Preventive (pre-action) control, not a Detective one alone.

P3: Organizational Governance Capacity will moderate — not eliminate — the risk-tier-to-binding-strength relationship. Low-capacity organizations facing high assigned risk tiers will disproportionately exhibit either downward score-gaming (lower RTF) or governance-practice decoupling (higher GPD), consistent with Institutional Theory’s decoupling prediction, rather than successfully absorbing the full governance load as specified.

P4: Traceability-depth classification will increase Traceability-Depth Awareness (stakeholders correctly identify which links are shallow) without, by itself, increasing actual audit rigor applied at those shallow links. The primary benefit is corrected expectations, not automatically stronger governance, and closing that residual gap would require a separate mechanism this framework does not yet provide.

P5: Divergence between stated design intent and actual adoption behavior will increase in proportion to external regulatory/audit pressure: organizations under heavier scrutiny will more often exhibit compliance-checkbox (“governance theater”) adoption regardless of IRGF’s own non-competing design intent, consistent with coercive isomorphism.

P6: Drift-taxonomy separation will measurably improve root-cause diagnosis speed specifically for boundary-spanning incidents (data/model), with no measurable improvement for incidents cleanly contained within one category. The benefit is concentrated exactly where the prior, merged taxonomy created ambiguity; it is not general.

Each proposition is falsifiable in the ordinary sense: a null or reversed finding is a coherent, reportable outcome, not a design failure. That is consistent with this project’s standing refusal to treat its own claims as pre-confirmed.

6. Validation Methodology

Positioning within Design Science Research Methodology (DSRM)

Design Science Research Methodology (DSRM; Peffers et al., 2007) is the most defensible overall frame for this project, and its own twelve chapters already map onto DSRM’s stages with reasonable fidelity, a correspondence the table below sets out for scrutiny rather than asserts retroactively:

Table 53. Positioning within Design Science Research Methodology (DSRM)

DSRM Stage	This Project's Corresponding Work	Honest Gap
Problem Identification & Motivation	Chapter 1	Grounded in standards/practitioner literature, not a systematic academic literature review (see below)
Define Objectives of a Solution	Chapter 3 (28 formal requirements)	—
Design & Development	Chapters 4–10	—
Demonstration	Chapter 11's eight scenarios	Simulated, not real — independent LLM agents role-playing organizational stress tests, not actual pilot deployments in real organizations. This is a materially weaker form of demonstration than DSRM normally expects, and should be named as such, not glossed over.
Evaluation	Chapter 11 (adversarial review) + Chapter 12 (novelty test)	Evaluation of internal coherence and stress-test robustness is real and independently adversarial; evaluation of <i>actual effectiveness in the field</i> has not occurred at all
Communication	Chapter 14 (pending)	—

The table is itself a finding: the project has completed the *design* half of DSRM rigorously and only a simulated version of the *demonstration/evaluation* half. Chapter 14's final specification should state this limitation, and should not present Chapter 11's agent-based stress test as equivalent to field validation. The two are not the same, and conflating them would be exactly the kind of overclaiming this project has otherwise been careful to avoid.

Recommended empirical sequence (beyond this project's own scope)

1. **Expert interviews first (not a Delphi study yet).** Semi-structured interviews with practitioners across the roles named in the framework (Enterprise/AI Architects, CIOs, AI Governance Officers, Chief Risk Officers) to (a) validate that the constructs in §3 are recognizable and meaningful to practitioners, and (b) refine P1–P6 before committing to a heavier study design. This is the lowest-cost, highest-information first step, and it should precede anything requiring a standing expert panel.
2. **A targeted Delphi study, scoped narrowly and not run as a blanket validation exercise.** Chapter 12 (D-58) flagged the Core Model's 8-entity minimality claim as asserted, not derived. A Delphi panel suits exactly this kind of question — convergent expert judgment on a bounded taxonomic problem. It should be scoped to that, and not expanded into a general “is this framework good” exercise, which Delphi handles poorly, since it converges best on bounded, structured questions and not on open-ended framework evaluation.
3. **Case studies: 3 to 5 organizations, sampled across the variance Chapter 11 found decisive.** Because P3 and P5 both hinge on Organizational Governance Capacity and regulatory-pressure context, case-study sampling should deliberately span: (a) a resource-constrained, high-risk-tier organization (replicating Scenario B/G's condition), (b) a fast-moving, low-formal-governance engineering organization (replicating Scenario C/D's condition), and (c) a regulated, higher-capacity organization (replicating Scenario A's condition). A convenience sample within a single organizational type could not test P3 or P5 at all, since both propositions are about *variance* across capacity/pressure conditions.
4. **A systematic literature review is recommended but has NOT been performed by this project and should not be assumed complete.** Chapter 2's “Framework Landscape Review” was a structured comparison against named standards and frameworks (TOGAF,

COBIT, NIST AI RMF, ISO 42001/23894, etc.). That was valuable, but it is not equivalent to a systematic review of peer-reviewed EA, AI-governance, or IT-governance academic literature, which this project has not conducted at any chapter. The gap is unresolved and methodological: **the propositions in §5 are reasoned from standards analysis and structured adversarial synthetic review (Chapter 11's agents), not benchmarked against a peer-reviewed evidence base.** Any future work treating this chapter's propositions as literature-validated would be overclaiming relative to what was actually done.

5. **Large-N survey research is NOT recommended at this stage.** The constructs in §3 are freshly defined and have no validated measurement instruments (scales, items) yet. Administering a survey before instrument development and qualitative validation (steps 1–3) would produce data with unknown construct validity. It is a *later*-stage option, not a near-term recommendation.

7. Potential Academic Contribution

Distinguishing the three forms of contribution, and consistent with Chapter 12's narrowed novelty findings:

Contribution type: primarily a **reference/integration model** (the dominant contribution, binding existing standards into one traceable structure), combined with a **governance model** (the operating model: decision rights, gates, roles) and a narrow **decision model** (the risk-tier-to-gate-automation binding specifically). It is **not** primarily a new metamodel/ontology, since the Core Model is flagged, per D-58, as unvalidated, and **not** primarily a new research methodology, since this project borrowed DSRM rather than contributing to it.

Theoretical Contribution: modest and specific: the application of cybernetic requisite-variety reasoning to justify continuous over periodic EA governance for AI-intensive environments, together with the four narrow original mechanisms named in Chapter 12 (§2 there), presented here as testable constructs and propositions rather than settled claims. This project does not claim to advance IT governance theory, cybernetics, socio-technical systems theory, institutional theory, dynamic capabilities theory, or the TOE framework themselves. It claims a coherent, non-decorative application of each to a specific, previously under-theorized problem (AI-intensive EA governance).

Methodological Contribution: the *design method* of binding risk classification to architecture-governance automation level, and of classifying traceability depth per link instead of assuming uniformity, are reusable design techniques other EA/governance-framework designers could apply elsewhere. This is a contribution to design practice, not to research methodology.

Practical Contribution: organizations gain two things. First, a structured vocabulary and artifact set for identifying governance gaps — demonstrated concretely by how effectively Chapter 11's scenario agents used the framework's own terms to find real gaps in it, which is itself evidence the vocabulary has diagnostic power even where the mechanisms it names are still immature. Second, a starting point (roles, gates, the four narrow mechanisms) for the specific integration gap this project's Chapter 1/2 research identified as real and unaddressed by any single existing

standard. This practical contribution does not depend on the theoretical or methodological contributions being validated; it stands on its own as a synthesis artifact, consistent with Chapter 12's Complementary Governance Framework finding.

Major Findings

- A defensible, falsifiable academic research model can be built directly from this project's own stress-test evidence (Chapter 11) without inventing new empirical grounding. Its constructs and propositions trace to specific, cited findings, not to freestanding academic exercises.
- Two theoretical perspectives initially plausible for a project like this (Resource-Based View, Complexity Theory) were tested and rejected as decorative rather than included for completeness, consistent with the requirement to discriminate between theories instead of merely listing them.
- This project's own phase structure maps onto DSRM with reasonable fidelity through the Design stage, but Chapter 11's "Demonstration" is simulated (agent-based), not a real organizational pilot — a genuine, previously unstated limitation that Chapter 14 must disclose rather than let stand implicitly.
- No systematic academic literature review has been performed at any chapter of this project; Chapter 2's standards/framework landscape review is a different, narrower activity and should not be conflated with it going forward.
- The two moderator constructs (Organizational Governance Capacity, Governance-Practice Decoupling) are structurally necessary controls for any future empirical test of this model. Omitting them risks crediting IRGF's mechanisms with effects that actually belong to organizational context, exactly the confound Chapter 11's scenarios kept surfacing.

Implications for the Framework

- Chapter 14's final specification must disclose the Demonstration/Evaluation gap (D-67) plainly in its Limitations section. The gap is now a formally logged finding, not an omission to carry forward quietly.
- The theoretical-foundations discrimination (D-64) gives Chapter 14 a settled, non-decorative theory set to cite, rather than the exhaustive, undifferentiated list of candidate theories considered at the outset.
- The mandatory moderator controls (D-65) should be stated as a methodological requirement in Chapter 14's "Research Opportunities" content, not left implicit in this chapter's diagram alone.

Issues Requiring Validation

- Whether the four surviving mechanisms (already flagged unvalidated in Chapter 12) would even survive the expert-interview stage (step 1 of §6) before any heavier study is warranted. Practitioners may find the constructs as defined here unrecognizable or poorly operationalized, which would require revising §3 before Delphi/case-study work proceeds.
- Whether a real (not simulated) demonstration would surface materially different findings than Chapter 11's agent-based stress test. This project has no way to know, and Chapter 14 should not imply otherwise.
- Whether the rejected theories (RBV, Complexity Theory, Organizational Ambidexterity) would be judged decorative by an actual academic review panel, or whether this chapter's own reasoning underestimates their fit. This is a live disagreement possibility, not a settled question.

Chapter 14. Final Framework Specification — IRGF Version 1.0

This chapter consolidates the preceding analysis into a single structured specification containing the framework's name, definition, purpose, problem statement, scope, principles, domains, dimensions, governance model, lifecycles, risk and autonomy models, gates, roles, decision rights, artifacts, reference architectures, metamodel, continuous assurance mechanism, metrics, maturity model, standards relationships, implementation methodology, adoption roadmap, validation methodology, limitations, and research opportunities.

Note on construction. Where a component was already fully designed in an earlier chapter (the AI Build Lifecycle in Chapter 5, or the Governance Operating Model in Chapter 8), this specification summarizes it and cites the source chapter together with its Version 0.2 revisions from Chapter 11. Re-deriving that work here would bury the components supplied for the first time. Those new components are the Design Principles, Architecture Patterns, Maturity Model, TOGAF crosswalk, NIST and ISO crosswalk, Implementation Methodology, and Adoption Roadmap. Two disclosures established in Chapters 12 and 13 are carried forward into this Version 1.0: the framework's name states its scope and makes no claim of paradigm-level novelty, and the framework has been demonstrated through simulated adversarial review rather than real organizational deployment.

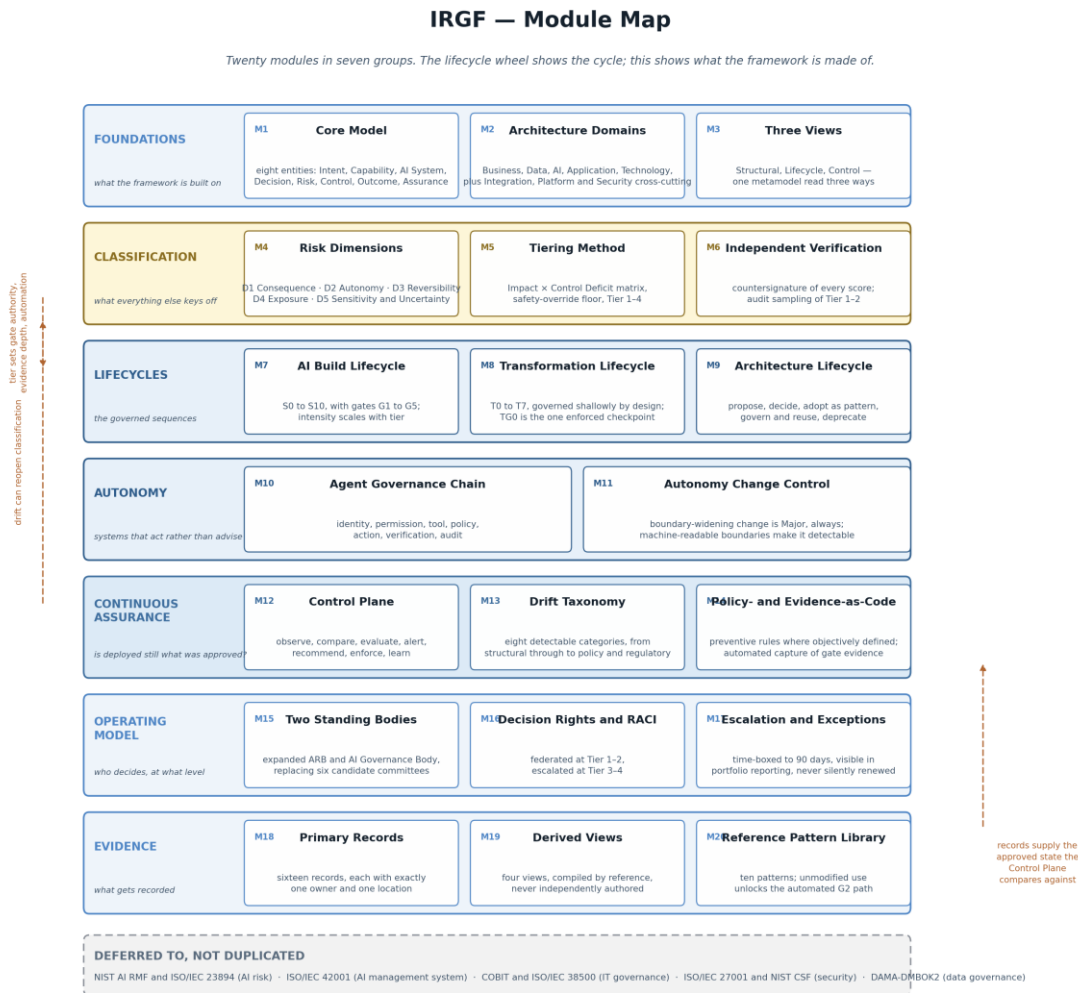


Figure 18. The IRGF module map: twenty modules in seven groups, and what each depends on

Part 0 — Identity

1. Framework Name

IRGF. Selected in Chapter 10 after collision-checking 22 candidate names against existing 2026 products, frameworks, and publications (14 of the original 22 were found to collide and were rejected).

2. Framework Definition

IRGF is a complementary governance framework. It binds Enterprise Architecture practice (grounded in TOGAF), AI-specific risk and management standards (NIST AI RMF (NIST, 2023), ISO/IEC 23894 (ISO & IEC, 2023b), ISO/IEC 42001 (ISO & IEC, 2023a)), and general IT/security/data governance standards (COBIT (ISACA, 2018), ISO/IEC 27001 (ISO & IEC, 2022), DAMA-DMBOK2 (DAMA International, 2017)) into one traceable structure spanning enterprise architecture, AI build governance, and, more shallowly by scope decision, digital transformation governance. **[Positioning finding, Chapter 12, D-55]**

A note on the name. IRGF stands for *Intent-to-Runtime Governance Framework*. The name states the framework’s scope: the chain it makes traceable, running from enterprise intent through architecture, risk classification and deployment authorization to runtime assurance. It claims nothing about the degree of innovation involved. An earlier working name, “Enterprise Architecture Operating System,” was retired because Chapter 12’s novelty assessment found most of the framework’s components to be Extensions or Integrations of existing practice, with only four narrow and still-unvalidated mechanisms qualifying as New Contributions (§13 below). A name implying an operating system asserted more than that evidence supports. The central eight-entity set is correspondingly called the **Core Model**, not a kernel. [D-62]

3. Purpose

To give an enterprise a single, continuously-verifiable answer to the seventeen governance questions posed in the originating research brief (why is this AI being used; which capability does it serve; who owns it; what data/model does it depend on; what can it decide and do; how autonomous is it; what risk does it carry; which policies/regulations apply; what mitigates that risk; who approved it; is it still compliant; is it behaving as expected; has its risk changed; what value is it generating; should it change, be restricted, or be retired), without requiring the enterprise to abandon or duplicate the standards it already uses to answer pieces of that question today.

4. Problem Addressed

Summarized from Chapter 1: existing frameworks each govern a slice competently (TOGAF: architecture method; NIST AI RMF/ISO 23894: AI risk; ISO 42001: AI management systems; COBIT: IT governance) but none connects them into one traceable, continuously-assured structure spanning EA, AI build, and transformation. Chapter 1 additionally found that “insufficient continuous architecture assessment” and “insufficient governance of third-party foundation models” were the two structural gaps least well covered by any single existing standard. Chapter 11’s adversarial review later confirmed both as live, unresolved weak points (Scenario F; the Control Plane’s own admitted non-novelty).

5. Scope

In scope: AI-as-EA-domain governance; AI build lifecycle governance (traditional ML through autonomous multi-agent systems); AI risk tiering and gating; agent autonomy governance; continuous architecture/AI conformance assurance; the architecture-facing interface to digital transformation. **Out of scope (by design decision, not oversight):** AI risk taxonomy design (defers to NIST AI RMF/ISO 23894), AI management-system requirements (defers to ISO 42001), general IT governance (defers to COBIT/ISO 38500 (ISO & IEC, 2024)), security control design (defers to ISO 27001/NIST CSF (NIST, 2024)), data governance methodology (defers to DAMA-DMBOK2), workforce/organizational change management (visibility only, Chapter 6), individual due-process/contestability mechanisms for algorithmic decisions (interface-only via the Regulatory Overlay Reference, Chapter 11 D-54), and fine-tuning standards, code-review standards, incident-response runbooks, and project scheduling (Chapter 3, FR scope boundary).

Positioning Statement

IRGF is a **complementary governance framework** — not an extension of TOGAF, not an overlay on TOGAF, not a next-generation EA framework, and not a fundamentally new architectural governance paradigm. This positioning was determined through evidence (Chapter 12), not assumed. [D-55]

Part I — Foundations

6. Design Principles

Fourteen principles, each tied to a specific mechanism and not offered as free-floating aspiration. Weak or redundant candidates from an initial list of ten were merged or dropped. Four principles absent from that initial list were added because they reflect lessons this project's own stress-testing (Chapter 11) forced onto the design.

1. **Business Before AI** — an AI system must address a named capability or stakeholder need (S0 Opportunity gate); AI is never self-justifying.
2. **Risk Proportionality** — governance intensity scales with assigned risk tier (D1–D5 → Tier 1–4 → gate automation level). *Known limitation:* Chapter 11 found this only partially realized in practice, because gate consolidation proved to be partly cosmetic bundling and not true burden reduction. The principle stands as the design target; the gap to it is tracked.
3. **Human Accountability Cannot Be Delegated** — every Decision entity has a named accountable human role, including for autonomous agent actions (Agent Owner), never the agent or model itself.
4. **Independent Verification of Risk** — self-reported D1–D5 scores require independent (Risk & Compliance Lead) countersignature, and Tier 1–2 self-certifications are periodically audit-sampled. *Added in v0.2 (R1/D-41) after Chapter 11 found unverified self-scoring to be the framework's single most confirmed, most convergent weakness.*
5. **Continuous Over Periodic Assurance** — architecture and AI system state must be continuously compared against approved state, not merely reviewed on a calendar cadence, grounded in the requisite-variety argument from cybernetics (Chapter 13).
6. **Traceable, Honestly-Graded Intent** — every AI System traces to a named capability/objective, and the *depth* of that trace (deep, shared, shallow-with-input, reference-only) is recorded rather than assumed uniform.
7. **Controlled, Detectable Autonomy** — autonomy-increasing changes require re-authorization, and deployment records must encode machine-readable permission boundaries so that increases are actually detectable, not just formally prohibited on paper. *The boundary-widening scope of this rule (R6) is itself flagged as an open, not fully resolved, design tension.*
8. **Defer to Existing Standards** — do not reinvent AI risk taxonomy, AI management-system requirements, general IT governance, security controls, or data governance

methodology; integrate and bind, don't duplicate. This is the operational expression of the framework's Complementary Governance Framework identity.

9. **Minimum Necessary Governance** — governance ceremony should be the least sufficient to manage the assessed risk. *Known limitation:* independent review found that the framework's own five-gate consolidation reduces the number of labeled checkpoints more than it reduces actual evidentiary burden. The principle is aspirational and its realization incomplete.
10. **Reversibility-Aware Design** — prefer reversible AI actions; score and govern irreversibility (D3), anchored to consequence-reversibility, not mechanism-reversibility.
11. **Explainability Proportional to Impact** — explainability requirements scale with D1 (Decision Consequence Severity) and D5 (Data Sensitivity/Model Uncertainty), not applied uniformly.
12. **Security and Data Governance Are Embedded, Not Bolted On** — Integration, Platform, and Security cut across all five architecture domains; they are not an afterthought layer.
13. **No Orphaned Systems** — every AI System and every primary artifact has exactly one accountable owner at all times; ownership gaps (e.g., for procured/vendor AI, which lacked a clear owner in v0.1) are treated as defects to close, not tolerated.
14. **Governance Claims Must Be Falsifiable** — the framework's own claims about its novelty, effectiveness, and burden are treated as testable propositions (Chapter 13), not settled truths, and are revised when contrary evidence appears, including evidence from this project's own adversarial stress-testing.

7. Architecture Domains

Five primary domains (Business, Data, AI, Application, Technology) plus three cross-cutting disciplines (Integration, Platform, Security) that touch all five rather than standing as peers. AI's status as a first-class domain gives it real governance-body representation (the ARB's expanded mandate). It does not extend to owning AI's technical substance: Technical AI Evaluation remains with ML engineering. [Chapter 4; reclassified Extension, not New Contribution, Chapter 12 §2]

8. Framework Dimensions

Six functional dimensions (Enterprise Intent, Capability & Value, Architecture, AI Lifecycle, Transformation [shallow by design], Continuous Assurance) plus two cross-cutting properties (Governance, Risk) attached to every entity rather than standing as independent dimensions. [Chapter 4, D-12]

20. Architecture Metamodel

The Core Model holds eight entities (Intent, Capability, AI System, Decision, Risk, Control, Outcome, Assurance), supported by a broader metamodel of ~23 further entities (Strategy, Objective, Value Stream, Dataset, Model, Agent, Policy, Metric, etc.). Both are expressed through three views: Structural, Lifecycle, Control. [Chapter 4, D-11/D-15]

Status note: Chapter 12 (D-58) repositioned the Core Model from “foundational theoretical contribution” to **a working ontology, asserted but not derived**. No minimality test has been performed, and an independent EA Academic reviewer flagged the point in Chapter 11. It remains the organizing structure for this specification because no superior alternative has been proposed or tested, not because its minimality has been validated.

Part II — Lifecycles & Risk

10. AI Lifecycle

S0 Opportunity → S1 Use-Case Qualification → S2 Risk Classification → S3 Architecture Decision → S4 Data Readiness → S5 Build/Configure/Acquire → S6 AI Assurance → S7 Deployment Authorization → S8 Runtime Operation & Continuous Monitoring (concurrent) → S9 Change → S10 Retirement, with loop-backs from S8/S9 into S2/S3/S6/S7. **[Chapter 5]** Version 0.2 revisions: independent scoring sign-off at S2 (R1); consequence-reversibility scoring anchor for D3 (R2); mandatory Preventive control + audit-latency ceiling for Tier 3–4 standing-authority agents (R4); machine-readable permission boundaries required at S7 for agentic systems (R5); iterative/parallel resolution of S3/S4 sub-components permitted before joint G2 exit (R13). **[Chapter 11]**

11. Transformation Lifecycle

T0 Purpose & Strategic Alignment → T1 Capability Gap Identification → T2 Initiative Qualification (TG0 checkpoint) → T3 Portfolio Prioritization → T4 Architecture & AI Enablement (spawns S0) → T5 (reabeled from “Delivery & Organizational Change” to **External Input Interface**; R12/D-52, correcting an appearance-of-ownership the framework never actually had) → T6 Benefits Realization & Review → T7 Capability Re-Baseline (loops to T1). **[Chapter 6]** Governance depth is non-uniform across this chain by design: deep at Capability/Architecture/AI/Outcome, shallow at Transformation Theme/Portfolio/T5.

12. Architecture Lifecycle

The cycle governing how an individual architecture decision becomes reusable, enterprise-standard practice: **Propose** (ADR drafted at S3/T4) → **Decide** (ARB review, risk-tier-scaled) → **Adopt as Pattern** (promoted to the Reference Pattern Library, §19, if reusable) → **Govern & Reuse** (subsequent Tier 1–2 builds may self-certify against an unmodified pattern) → **Deprecate** (pattern retired when superseded, triggering re-review of any system still depending on it). This lifecycle was implicit in Chapters 4–9 but had not been named as its own numbered lifecycle before this specification; it is presented here as an organizing device, not a new governance mechanism.

13. AI Risk Model

Five composite dimensions (D1 Decision Consequence Severity, D2 Autonomy, D3 Reversibility Deficit, D4 Exposure & Scale, D5 Data Sensitivity & Model Uncertainty), each scored 1–4. Impact = avg(D1,D4,D5); Control Deficit = avg(D2,D3); a 4×4 matrix maps to Tier 1–4; a safety-override floor (D1=4, or D2=4∧D3=4) forces Tier ≥3 regardless of matrix outcome. **[Chapter 5]**

Version 0.2 revisions: independent countersignature required (R1); D3 anchored to consequence-, not mechanism-, reversibility (R2); D5 reported as two distinct sub-scores (Data Sensitivity, Model Uncertainty) alongside its composite, without changing the tiering formula (R3). **Confirmed limitation, not resolved:** the safety-override floor catches extremes but not the mid-range, delivery-pressure-driven scoring choices that three of Chapter 11's reviewers independently identified as a live gaming vector. [Chapter 11, Failure Point 1]

Standards positioning [D-59]: D1–D5 should be read as an operational implementation schema for NIST AI RMF's Map/Measure functions and ISO 23894's risk-criteria concepts, not as a competing risk taxonomy. See §25 for a first-cut crosswalk.

14. AI Autonomy Model

Governance for AI agents runs through a defined chain, with v0.2 revisions annotated directly on the mechanism:

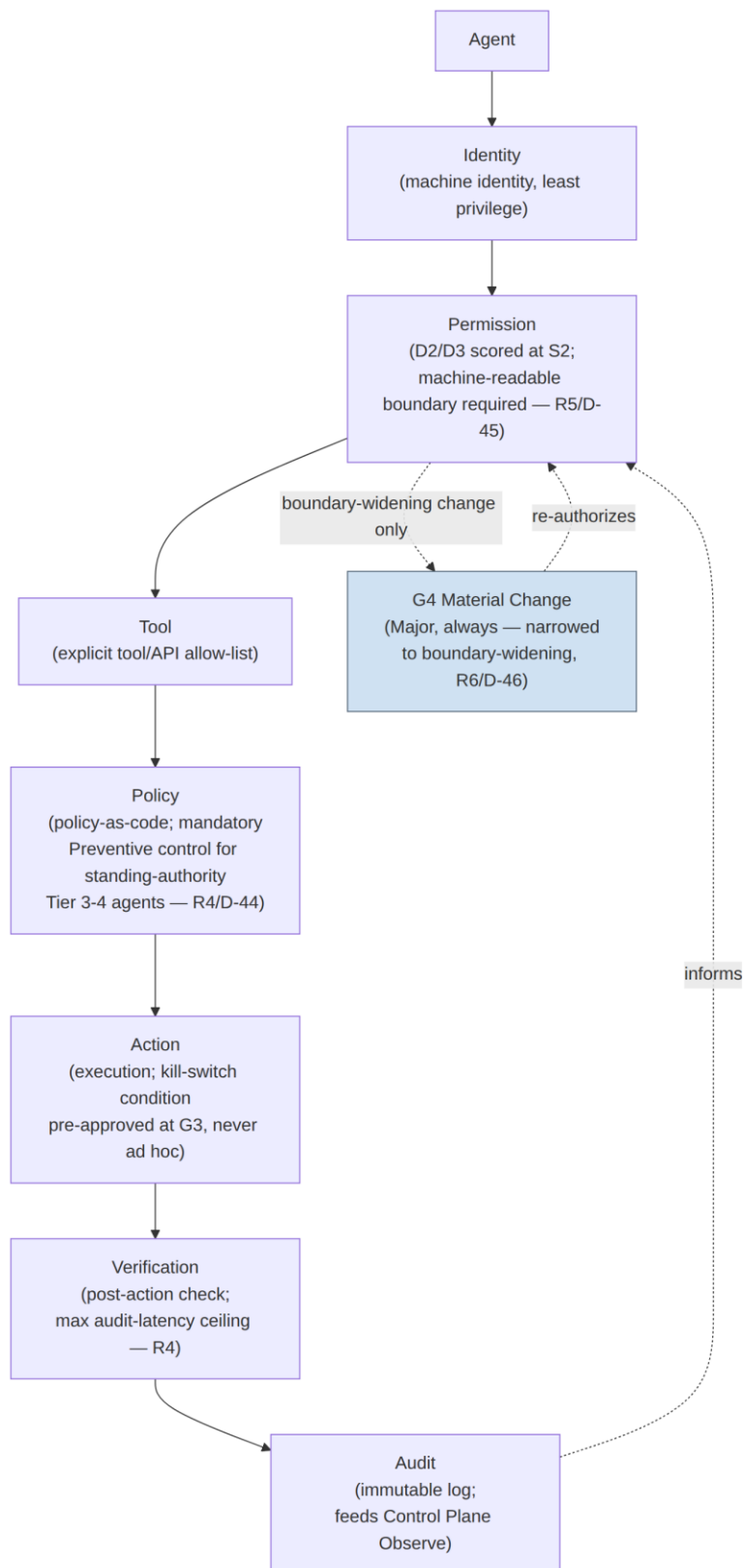


Figure 19. AI agent governance chain

Confirmed limitation: R6’s narrowing of the “autonomy increase = Major, always” rule to boundary-widening changes only was logged as an open, untested trade-off against R1’s audit-

sampling backstop (Chapter 11, R6/D-46). The diagram depicts the current best-specified version of this control, not a validated one.

15. Governance Gates

G1 Qualification & Provisional Risk, G2 Architecture & Data Readiness (internal sequencing loosened per R13), G3 Deployment Authorization, G4 Material Change, G5 Retirement, with automation and authority scaled by risk tier. [Chapter 5] At the evidentiary level, independent review found the reduction from an illustrative eleven touchpoints to five gates to be **largely bundling, not burden reduction**. Adopters should be told that this is consolidation, not a burden-reduction guarantee. [Chapter 11, Complexity Issue 1]

Part III — Governance Operating Model

9. Governance Model

Six governance levels (Enterprise, Portfolio, Platform, Product, Solution, Model/Agent), plus a **conditional seventh level, Region/Legal Entity**, added in v0.2 and activated only where jurisdiction-specific regulatory exposure or country-level Tier 3–4 volume justifies it (R10/D-50). Tier 1–2 decisions federate to Product/Solution self-service; Tier 3–4 escalate to two standing bodies. [Chapter 8]

16. Roles

Business Sponsor, AI/Enterprise Architect, Data Owner, AI/System Owner, Risk & Compliance Lead, Security Architect, Architecture Review Board (ARB, expanded to five domains), AI Governance Body (absorbs Model Risk Committee function), Enterprise Governance Sponsor, Portfolio/PMO (external), plus **Procurement**, added in v0.2 as a named role consulted at S1/S2 for procured/integrated AI (R8/D-48). [Chapter 8; v0.2 Chapter 11]

Confirmed limitation: Security Architect remains “Consulted,” never “Accountable,” across the governance RACI. Independent review found this under-weighted relative to security’s actual role in agent/coding-agent failure modes (Scenario D, E). It is logged as an unresolved gap.

17. Decision Rights

The Decision Rights Matrix (Chapter 8) assigns approval authority per decision type and tier. The one intended exception to the general “architecture advises, Portfolio decides” boundary is the AI Governance Body’s **funding concurrence right** (not mere input) over Tier 3–4 AI-relevant Portfolio decisions (D-31). Independent review found that this concentrates real power in a body with no equal-weight business-side counterpart to contest it (Chapter 11, Digital Transformation Executive persona). It is logged as an unresolved tension, not a defect requiring reversal, since removing it would reopen the funding-bypass gap it exists to close.

18. Required Artifacts

Sixteen primary records (the original fifteen from Chapter 9, plus the **Regulatory Overlay Reference** added in v0.2, R7/D-47) plus four derived views (Business Capability Map, AI System Card, Benefit Map, Assurance Dashboard). The ADR’s mandatory fields were extended in v0.2 to require vendor AI due-diligence evidence for procured/integrated systems, with a

resolution rule for evidence-stalemate cases (Tier-2 cap, or AI Governance Body documented risk acceptance). [Chapter 9; v0.2 Chapter 11]

19. Reference Architectures

Ten pre-approved patterns, condensed. A system built from an unmodified pattern may use the lightweight, largely-automated G2 path (Chapter 5, §5.4) at Tier 1–2.

Table 54. Reference Architectures

Pattern	Problem	Core Architecture	Key Controls	Primary Risk	Applicability
Enterprise RAG	Ungoverned proliferation of RAG copilots, each with its own grounding source	Shared vector DB + governed knowledge base + retrieval gateway	Data Lineage Record per source; S4 gate	Grounding-source staleness or poisoning	Tier 1–3; fast-path eligible
Secure LLM Gateway	Direct, ungoverned app-to-foundation-model calls	Centralized gateway: auth, logging, rate-limiting, PII redaction	Gateway logs feed Control Plane Observe	Gateway becomes single point of failure	All tiers; foundational infrastructure
AI Agent	Ad hoc agent tool-access grants	Agent Card-governed Identity→Permission→Tool chain (§14)	Machine-readable boundary (R5); kill-switch	Autonomy creep (confirmed, Scenario D)	Tier 2–4; never self-service at Tier 1
Human-in-the-Loop	High-consequence decisions made without a real human check	Mandatory approval step before consequential action executes	Documented human oversight point (FR-11)	Automation bias undermines the check’s substance (confirmed, Scenario B)	Tier 3–4 decision-consequence systems
AI Decision Support	Advisory AI mistaken for autonomous authority	Recommendation-only output; no direct action capability	D2 capped by architecture, not policy alone	Formal-vs-effective autonomy gap (unresolved, §13)	Tier 1–3
Private/Sovereign AI	Sensitive data cannot leave the enterprise boundary	Self-hosted/VPC-isolated model, no external egress	Data Lineage Record confirms no egress	Higher operational burden, capability lag vs. frontier models	Tier 3–4 with high D5
AI SaaS Consumption	Vendor AI features with no visibility (confirmed structural gap, Scenario F)	Vendor feature behind Procurement + extended-ADR due-diligence gate	Vendor evidence requirement (R8); Tier-2 cap if unavailable	Opacity; evidence-stalemate	All tiers; primary entry point for procured AI
Multi-Model Routing	Single-vendor lock-in, no fallback	Gateway routes across models by cost/capability/risk	Each candidate model separately ratified enterprise-wide	Inconsistent behavior across routed models complicates D1–D5 scoring	Tier 1–2 primarily
Responsible AI	Bias/fairness/toxicity testing done inconsistently	Standardized S6 evaluation suite, reusable	AI Assurance Summary artifact	Tier 1–2 systems get only sampling-based checks (R11), not full assurance	All tiers; intensity scales by tier
AI Observability	No visibility into live AI behavior post-deployment	Telemetry feeding Control Plane Observe, drift-taxonomy-tagged	8-category drift taxonomy	Underlying mechanism is non-novel (D-56); value is in the AI-specific categorization	Mandatory Tier 2–4; recommended Tier 1

Part IV — Continuous Assurance

21. Continuous Assurance Mechanism

The Control Plane: Observe → Compare → Evaluate → Alert → Recommend → Enforce → Learn, against an 8-category drift taxonomy (Structural, Configuration, Integration & Dependency, Security, Data-Lineage/Governance, AI-Model-Behavioral, Policy & Regulatory, Cost; the first two are kept distinct by design, one of the four surviving novel mechanisms). Corrective automated enforcement is default-prohibited for Tier 3–4 and any agent-authority change, with a single exception: a pre-approved kill-switch condition defined in advance at G3. [Chapter 7]

Status **note** [D-56]: the mechanism itself (observe/compare/evaluate/alert/recommend/enforce/learn) is formally repositioned from “central novel contribution” to **recommended implementation pattern**. It applies existing CSPM/DevOps/policy-as-code practice, as Chapter 11’s red-team review independently confirmed. It remains good practice; it is not this framework’s research contribution.

22. Metrics

The Transformation & Value Scorecard tracks five non-composited metric families: Business, Transformation, Architecture, AI, and Risk. These are left uncollapsed into a single index by design (Chapter 6, D-26), so that, for example, high reuse cannot mask low benefit realization. [Chapter 6] Chapter 13’s conceptual research model proposes two forward-looking outcome constructs, Risk Mitigation Effectiveness and Perceived Governance Burden, as candidate metrics for future validated instruments. These are research constructs, not yet operational metrics an adopter can measure today.

Part V — Maturity & Adoption

23. Maturity Model

Five levels, each defined by adoption of specific named mechanisms rather than by narrative vibes. That makes the model more falsifiable than the illustrative version in the original research brief, and it is critically revised at Level 5 (see note below). No organization has been observed progressing through these levels; this is asserted design, consistent with Chapter 13’s disclosure that this project’s demonstration work was simulated, not field-based.

Table 55. Maturity Model

Level	People	Process	Technology	Governance	Architecture	Data/AI	Measurement
L1 — Fragm ented	No named AI governance role	Ad hoc; no risk tiering	No shared artifacts, no telemetry	None/informal	AI not recognized as a domain	Ungoverned	None
L2 — Contro lled	Part-time AI Governance Lead	S0–S2 tiering adopted for <i>new</i> initiatives; G1–G3	Manual artifact tracking	Single informal review body	AI named as a domain, not yet in ARB composition	Risk Classification Records exist but unaudited (v0.1-	Basic, per-initiative only

Level	People	Process informal	Technology	Governance	Architecture	Data/AI	Measurement
L3 — Integrated	Full Chapter 8 role catalogue staffed	Full G1–G5	Partial Control Plane (manual + some automated Observe)	ARB expanded to 5 domains; AI Governance Body stood up; R1 independent sign-off adopted	AI as 5th domain, formally represented	equivalent — the confirmed R1 gap) 16-artifact set actively maintained	Transformation/Value Scorecard in use
L4 — Adaptive	Procurement role integrated; Security Architect elevated toward co-accountability (adoption recommendation, not yet a framework requirement)	G2 iterative/parallel sequencing (R13) in practice	Real Control Plane: automated Observe/Compare/Evaluate; machine-readable G3 boundaries (R5) enabling actual drift detection	Audit-sampling of Tier 1–2 self-certifications (R11) running; Regulatory Overlay Reference populated	Conditional Region/Legal Entity level activated if multinational (R10)	Tier/D-score crosswalked to enterprise risk register (R14)	Empirically-informed thresholds beginning to replace asserted ones
L5 — Continuously Assured (renamed — see note)	Roles shift from transaction-level review to exception-handling and D1–D5 anchor calibration	Continuous, evidence-as-code	Full telemetry integration; Enforce active for objectively-defined Preventive rules	Federated, risk-based, minimal friction for low-tier work	Patterns (§19) cover the large majority of new builds	Real incident data feeding periodic recalibration of scoring anchors	Metrics validated against real outcome data, not asserted

Critical note on Level 5. This level’s defensibility is assessed here rather than assumed. The label most commonly proposed for it is “Autonomous Governance.” That label is rejected as internally inconsistent with the framework’s own design: Chapter 7 (D-30) makes automated *corrective* enforcement for Tier 3–4 systems and agent-authority changes **default-prohibited by design, permanently**, not as a maturity gap to eventually close. A framework that reached “autonomous governance” in the literal sense would have to abandon its own human-accountability principle (§6, Principle 3). Level 5 is therefore renamed **Continuously Assured** and defined as maximal *automation of detection and evidence*, with human authority *by design, permanently* retained over consequential correction — maturity means faster, better-evidenced human decisions, not fewer of them.

26. Implementation Methodology

Minimum Viable Framework (MVF). Applying the 80/20 test against this research’s own accumulated evidence (Chapter 11): the highest-leverage 20% is not the full governance-body structure — it is **independent risk-tier verification**, because Chapter 11 found unverified self-scoring to be the single most confirmed weakness. The MVF is:

1. Run S0–S2 (Opportunity, Qualification, Risk Classification) on all *new* AI initiatives, with mandatory independent countersignature (R1). This is the cheapest, highest-leverage control identified across all thirteen chapters.

2. Five gates, but Tier 1–2 self-service by default; only Tier 3–4 requires formal review.
3. For organizations too small to credibly staff two distinct bodies, **merge ARB and AI Governance Body functions** into one reviewing group rather than pretending a separation exists when it does not. Chapter 11 found this collapse happens informally anyway in 50–150 person organizations; the MVF makes it a declared simplification instead of an unacknowledged one.
4. Minimum artifact set: Risk Classification Record, Data Lineage & Sensitivity Record, ADR. Three records, not sixteen; defer the rest to L3.
5. Agent Card only for systems with actual tool-access or autonomous-action capability; skip for advisory-only systems.
6. Regulatory Overlay Reference only if operating in a named regulated sector (financial services, healthcare, government).

27. Organizational Adoption Roadmap

Table 56.

Timeframe	Maturity Transition	Key Actions
Months 0–3	L1 → L2	Name an AI Governance Lead (part-time acceptable); adopt the MVF artifact set; run S0–S2 on all new initiatives (not retroactively)
Months 3–9	L2 → L3	Expand ARB to five domains; stand up the AI Governance Body (or the merged MVF variant); retroactively risk-classify existing production AI systems; adopt the full 16-artifact set
Months 9–18	L3 → L4	Implement machine-readable G3 boundaries for agentic systems; stand up Control Plane telemetry starting with 2–3 highest-value sources (cloud + IAM + model registry, typically), not all seven at once; populate the Regulatory Overlay Reference for regulated systems
Months 18+	L4 → L5	Full Control Plane; policy-as-code Preventive automation; recalibrate D1–D5 scoring anchors against real incident data; activate conditional regional governance if multinational

Part VI — Standards Relationship

24. TOGAF Relationship

Table 57.

ADM Phase	Status	What IRGF Adds
Preliminary Phase	Valid + Extended	Design Principles (§6) extend principle-setting with explicit AI/risk principles; establishes AI Governance Body and ARB’s 5th-domain mandate
Architecture Vision	Valid + Extended	S0: should this problem use AI at all, before architecture work begins
Business Architecture	Valid + Extended	T1: AI-enabled capability-gap identification
Information Systems Architecture	Partially Replaced	AI is carved out as a peer domain, not folded into Application Architecture as one more application type

ADM Phase	Status	What IRGF Adds
Technology Architecture	Valid + Extended	AI-specific infrastructure decisions (hosting, gateways, vector DBs) — governed at the architecture-boundary level only, engineering detail remains out of scope
Opportunities & Solutions	Valid + Extended	Explicit build-vs-fine-tune-vs-RAG-vs-API-vs-SaaS-vs-no-AI decision framework (S3)
Migration Planning	Valid + Extended	Portfolio prioritization (T3) sequenced by risk tier and dependency, via the Architecture Input Package
Implementation Governance	Partially Replaced	TOGAF's periodic Architecture Compliance review is superseded, for AI systems specifically, by the Control Plane's continuous conformance checking
Architecture Change Management	Partially Replaced	S9's risk-driven change classification (autonomy increase, model swap, data-source change) is materially more granular than generic ADM change management
Requirements Management	Valid, unchanged	This project's own 28 formal requirements (Chapter 3) are themselves an application of this ADM discipline

No phase was forced into a one-to-one mapping where none existed.

25. Integration with AI Governance Standards

Consolidated from Chapter 12's Gap Matrix, plus a first-cut D1–D5-to-standards crosswalk fulfilling the requirement Chapter 12 (D-59) logged as outstanding:

Table 58. Integration with AI Governance Standards

IRGF Dimension	Nearest NIST AI RMF Concept	Nearest ISO/IEC 23894 Concept	Caveat
D1 Decision Consequence Severity	MEASURE function: impact assessment	Risk consequence/severity criteria	Directly analogous
D2 Autonomy	GOVERN/MAP: human-AI configuration and oversight characteristics	Risk criteria relating to automation level	IRGF's explicit 1–4 scale is more operationalized than either source discusses generically
D3 Reversibility Deficit	Not explicitly named as a distinct RMF category; closest is MEASURE's discussion of consequence characteristics	Risk-treatment feasibility / control effectiveness	Weakest mapping of the five — likely IRGF's most original dimension
D4 Exposure & Scale	MAP function: context, scale, reach characteristics	Likelihood/exposure criteria	Directly analogous
D5 Data Sensitivity & Model Uncertainty	MAP function: data characteristics; MEASURE: uncertainty quantification	Risk criteria for data sensitivity and model performance uncertainty	Fused into one score in IRGF; source standards generally treat separately, supporting Chapter 11's finding (R3) that this fusion loses information

Caveat, carried from Chapter 13's disclosure (D-69): this crosswalk is this project's own reasoned mapping, built from general familiarity with the structure of these standards. It is **not a clause-by-clause verified crosswalk against current published primary-source text**, and it should be independently verified before being relied upon for a compliance claim.

For the remaining standards, IRGF's relationship is one of deferral, not integration-by-crosswalk: ISO 42001 (AI management system requirements; IRGF references, does not restate), COBIT/ISO 38500 (general IT governance; control-type vocabulary reused, not reinvented), ISO 27001/NIST CSF (security controls; Security Architect role integrated but under-weighted, a confirmed open gap), DAMA-DMBOK2 (data governance; extended only at the AI-grounding seam, via the Data-Lineage/AI-Model-Behavioral drift separation).

Part VII — Academic & Critical Material

28. Validation Methodology

Design Science Research Methodology (DSRM; Peffers et al., 2007) is the governing frame. This project's own chapters map to its stages through Design with reasonable fidelity, but **Demonstration was simulated** (independent LLM agents role-playing organizational stress tests in Chapter 11) and not conducted with real organizations, and **Evaluation** of actual field effectiveness has not occurred. Recommended empirical sequence for future work: expert interviews → a Delphi study scoped narrowly to the Core Model's minimality question → three to five case studies sampled across the organizational-capacity and regulatory-pressure variance Chapter 11 found decisive. Large-N survey research is deferred as premature, since the constructs in Chapter 13's research model have no validated measurement instruments yet. [Chapter 13, §6]

29. Limitations

Consolidated from every chapter that logged an unresolved finding, so that the "Version 1.0" label does not imply completeness:

1. **This framework has been stress-tested by simulated adversarial review, not real organizational deployment.** Chapter 11's nine reviewers were independent LLM agents, not real practitioners, CIOs, or risk officers. This is the single most important limitation and should be disclosed in any external communication of this specification.
2. **No systematic academic literature review has been performed.** Chapter 2's standards/framework landscape review is a different, narrower activity. The propositions in Chapter 13 are reasoned, not literature-benchmarked.
3. **Risk-tier self-scoring remains gameable at the margins,** even after the v0.2 independent-sign-off revision (R1): the safety-override floor catches extremes, not delivery-pressure-driven mid-range judgment calls.
4. **The ARB/AI Governance Body model does not scale on the volume axis** within a single jurisdiction; only the geography axis was addressed (the conditional Region/Legal Entity level, R10).
5. **No cost, staffing, or SLA model exists.** This specification does not fabricate FTE or turnaround-time figures it has no calibration data to support.
6. **The Core Model's minimality is asserted, not derived or empirically validated.**
7. **The Maturity Model (§23) is asserted design, not an empirically observed progression;** no organization has been tracked through these levels.
8. **The NIST/ISO crosswalk (§25) is a reasoned mapping, not a clause-verified one.**
9. **Individual due-process/contestability mechanisms are a declared non-goal,** not a gap left unaddressed by oversight (Chapter 11, D-54). The framework provides an interface point (the Regulatory Overlay Reference) to external due-process mechanisms, but does not design them.

10. **The R6 autonomy-change narrowing and R1 audit-sampling backstop have not been tested together**; their interaction is unknown.
11. **No mechanism exists for cumulative/structuring exposure** in standing-authority agents (many sub-threshold actions accumulating between audit cycles).
12. **“Cosmetic vs. substantive” change classification has no bright-line test** and remains judgment-based.

30. Research Opportunities

Consolidated and prioritized from the Open Research Questions logged across all thirteen prior chapters:

1. Does independent risk-tier countersignature (R1) measurably reduce under-tiering in practice, and does the effect survive role-collapse conditions in smaller organizations? (Chapter 13, SRQ1)
2. Would a real (not simulated) organizational demonstration surface materially different findings than Chapter 11’s agent-based stress test? (Chapter 13)
3. Would an independent Delphi panel validate the Core Model’s eight entities as minimal and complete? (Chapter 12, D-58)
4. Does machine-readable permission-boundary encoding (R5) actually improve autonomy-change detection latency enough to matter for high-velocity agentic systems? (Chapter 11, Chapter 13 SRQ2)
5. Does organizational governance capacity moderate risk-tier-to-governance-rigor as Institutional Theory predicts (score-gaming or governance-theater decoupling), and can a capacity-adjusted tiering model close the gap? (Chapter 13, SRQ3)
6. Does the Data-Lineage/AI-Model-Behavioral drift separation measurably improve root-cause diagnosis speed specifically at the data/model boundary? (Chapter 13, SRQ6)
7. What is the actual, empirically-calibrated implementation cost of running IRGF at Tier 3–4 intensity for a mid-sized regulated enterprise?
8. Does a full, independently-conducted systematic literature review surface prior art that would downgrade any of the four remaining New Contribution claims further?

Major Findings

- All thirty components required by the master specification (§38) are now present across thirteen chapters plus this one; six components (Design Principles, Maturity Model, Reference Architectures, TOGAF Crosswalk, NIST/ISO Crosswalk, Implementation Methodology + Adoption Roadmap) had no prior chapter and were built for the first time in this document.

- The conventional Level 5 maturity label (“Autonomous Governance”) was found internally inconsistent with the framework’s own Chapter 7 design decision (permanent human authority over Tier 3–4 corrective action), and the level was renamed.
- This specification is the first point in the project where every open limitation from Chapters 11–13 is gathered into one place. The result is a “Version 1.0” document that is substantially less validated than its publication-ready format might otherwise imply.

Implications for the Framework

This specification is complete relative to the thirty-component requirement set for it. Any future work extending IRGF should treat the consolidated design log at Appendix A as the authoritative decision record, and should cite decision identifiers rather than restating resolved decisions, consistent with the practice applied since Chapter 2.

Issues Requiring Validation

All items in §29 (Limitations) and §30 (Research Opportunities) remain open. None is resolved by this specification, whose contribution is to consolidate and disclose them, not to close them.

Future Work

Future work should prioritize, in order: (1) the expert-interview validation step from Chapter 13’s methodology, since every proposition in this specification currently rests on reasoned design and simulated stress-testing rather than real evidence; (2) a full systematic literature review, given this project’s own repeated disclosure that one was never performed; and (3) piloting the Minimum Viable Framework (§26) with a single real organization before attempting broader validation, since Chapter 11 found organizational capacity, not mechanism design, to be the framework’s most consistent fault line.

Appendix A — Consolidated Framework Design Log

Every design decision taken during this research is recorded here under the chapter in which it was made, with its rationale and status. Decisions are referenced by identifier (D-1 through D-70) throughout the report. Where a later chapter reversed, narrowed, or reclassified an earlier decision, both the original and the revision are retained rather than the earlier entry being overwritten, so that the reasoning remains auditable.

Chapter 2 — Framework and Standards Landscape

Decision ID	Decision	Rationale	Chapter	Status
D-01	Adopt “integration/connective structure,” not “new AI governance content,” as the framework’s contribution thesis	Landscape review shows AI risk, management-system, and IT-governance content are already mature; duplicating them would fail the non-duplication requirement	2	Accepted
D-02	Full 17-criterion profiles limited to TOGAF, NIST AI RMF, ISO 42001, COBIT; other frameworks profiled at 6 points	Manage document scope while preserving rigor where the “why not just use X” challenge is most likely to land	2	Accepted — revisit if Chapter 11 stress-testing finds an under-profiled framework was load-bearing
D-03	Continuous architecture/AI conformance assurance identified as the single strongest innovation candidate	Section B shows this capability rated Weak/Not Addressed more consistently than any other row in the matrix	2	Proposed — requires the Chapter 12 anti-novelty test against DevOps/security posture-management practice before being accepted as genuinely novel
D-04	Zero Trust Architecture adopted as the explicit precedent for AI agent identity/permission governance rather than inventing new agent-identity concepts	Zero Trust’s least-privilege/continuous-verification principles transfer directly; inventing a parallel concept would be duplicative	2	Accepted
D-05	New AI risk taxonomy, new AI management-system standard, and new general IT-governance model explicitly excluded from the framework’s innovation claims	These are adequately addressed by NIST AI RMF/ISO 23894, ISO 42001, and COBIT/ISO 38500 respectively	2	Accepted
D-06	Digital-transformation “frameworks” treated as fragmented practitioner practice rather than a peer, standards-body framework	Avoids over-crediting generic consultancy models; transformation literature is directionally useful but not authoritative in the same sense as ISO/NIST/COBIT	2	Accepted

Chapter 3 — Framework Design Requirements

Decision ID	Decision	Rationale	Chapter	Status
D-07	Adopt 28 requirements across 9 categories rather than the 15–25 initially suggested	The AI-domain and runtime-assurance categories each needed enough granularity to be independently testable; consolidating further would have hidden distinct validation needs behind a single	3	Accepted

Decision ID	Decision	Rationale	Chapter	Status
		requirement		
D-08	Explicitly reject four candidate requirements (fine-tuning standards, code review standards, incident-response runbooks, project scheduling) as out of EA scope	Applying Section 3.6 discipline; prevents scope creep into engineering/ops/PM domains	3	Accepted
D-09	Narrow workforce/change-management governance to a visibility hook (FR-20) rather than full ownership	Avoids duplicating organizational change-management discipline; consistent with Chapter 1's framing of transformation-adjacent-but-not-EA-owned concerns	3	Accepted
D-10	Treat FR-21/FR-22 (continuous conformance, drift detection) as simultaneously the top differentiating requirements and the top novelty risk	Directly carries forward Chapter 2's D-03 flag; must not be allowed to quietly become an accepted novel claim before Chapter 12's test	3	Accepted — high-priority carry-forward

Chapter 4 — Conceptual Framework Architecture

Decision ID	Decision	Rationale	Chapter	Status
D-11	Adopt “one metamodel, three views” (Structural / Lifecycle / Control) instead of a single unified diagram or three independent frameworks	No single pure structure (layered, lifecycle, matrix, control-system) covers the Mandatory FR set without distortion; a shared metamodel avoids record duplication across views	4	Accepted — test in Chapter 11 whether practitioners perceive this as coherent or as three frameworks loosely joined
D-12	Collapse Governance and Risk from standing dimensions (per the eight originally proposed) into cross-cutting properties	Every Mandatory FR referencing governance or risk attaches it to another entity; no FR requires governance or risk to exist independently	4	Accepted
D-13	Reduce the proposed 10-element architecture domain chain to 5 primary domains + 3 cross-cutting disciplines	Experience/Process already handled within Business Architecture by TOGAF; Integration/Platform/Security function as connective/hosting/protective concerns across domains rather than peer domains	4	Accepted
D-14	Define AI Architecture with an explicit EA-level vs. engineering-level split per component (Section 4.4 table)	Prevents AI Architecture from sprawling into solution-engineering and data-science territory, extending the same scope discipline applied to the FR set in Chapter 3	4	Accepted
D-15	Replace “Architecture” with “Decision” in the framework's Core Model relative to the illustrative example in the research brief	“Architecture” is not itself an entity in the metamodel and is definitionally circular as a Core Model concept; “Decision” is the entity that actually answers “who approved it” and is referenced by more Mandatory FRs	4	Accepted

Chapter 5 — AI Build Governance Model

Decision ID	Decision	Rationale	Chapter	Status
D-16	Merge Runtime Operation and Monitoring into one concurrent	Presenting them as sequential stages would reintroduce the periodic-review	5	Accepted

Decision ID	Decision	Rationale	Chapter	Status
	lifecycle state	assumption Chapter 1 identified as a structural EA limitation		
D-17	Consolidate the project's twelve risk dimensions into five composites, scored via an Impact × Control-Deficit matrix rather than a weighted sum	Prevents a single catastrophic factor from being diluted by averaging; matches how risk-matrix practice already separates likelihood-like and impact-like axes	5	Accepted — inter-rater reliability of the 5-composite scoring is untested (see Assumptions Register)
D-18	Reduce the project's eleven illustrative governance gates to five	Several proposed gates did not correspond to a distinct accountable decision point and could be folded into an adjacent gate's entry criteria	5	Accepted
D-19	Classify AI changes by which risk dimension (D1–D5) they plausibly move, not by technical change size	Effort-based classification allows risk-relevant changes to slip through as “small updates,” a failure mode named in Chapter 1	5	Accepted
D-20	Decompose the Chapter 4 core model's single “Assurance” concept into five distinct assurance types	Directly resolves the open question Chapter 4 deferred about whether Assurance bundles separable concepts	5	Accepted — the Core Model entry “Assurance” should now be understood as an umbrella over these five, not a single monolithic check

Chapter 6 — Digital Transformation Governance Model

Decision ID	Decision	Rationale	Chapter	Status
D-21	Add Portfolio and Benefit as new metamodel entities, extending (not replacing) the Chapter 4 metamodel	Ten of the project's twelve traceability-chain links already mapped to existing entities; only these two required genuine addition	6	Accepted
D-22	Classify each of the twelve traceability-chain links by governance depth (deep / shared / shallow-with-input / reference-only) rather than treating the chain as uniformly governed	Prevents the elegant-looking chain from silently re-expanding EA's scope past FR-20's boundary	6	Accepted
D-23	Define TG0 (Initiative Qualification Checkpoint) as owned by Portfolio/PMO with mandatory-but-bounded architecture input, not as an EA-owned gate	Reaffirms FR-20; avoids the framework absorbing transformation-delivery ownership the Framework Boundary in Chapter 2 excluded	6	Accepted — creates a known enforcement gap, explicitly carried forward, not hidden
D-24	Keep the Transformation Governance Lifecycle (T0–T7) and AI Build Governance Lifecycle (S0–S10) as two distinct, connected lifecycles rather than merging them	Many initiatives involve no AI; merging would misrepresent the actual relationship and force AI-specific stages onto non-AI initiatives	6	Accepted
D-25	Benefit Records derive realized-value data from Chapter 5's Business Outcome Assurance metrics rather than maintaining independent measurement	Direct mechanism satisfying FR-19's non-duplication requirement; prevents the exact fragmentation between architecture governance and transformation governance Chapter 1 identified	6	Accepted

Decision ID	Decision	Rationale	Chapter	Status
D-26	Reject a single composite “transformation health” index in the Transformation Scorecard	A blended score would obscure real trade-offs (e.g., high reuse, low benefit realization) the scorecard exists to surface	6	Accepted

Chapter 7 — Continuous Architecture Governance

Decision ID	Decision	Rationale	Chapter	Status
D-27	The Control Plane mechanism itself is an Extension of CSPM/DevOps/policy-as-code practice; its application to the risk-tiered EA-AI metamodel is the narrower, more defensible novelty claim	Directly addresses the novelty risk flagged since Chapter 2 (D-03) rather than deferring it unexamined to Chapter 12	7	Provisional — final classification reserved for Chapter 12’s formal anti-novelty test
D-28	Reconcile the research brief’s two non-aligned drift-type lists (nine items and ten items respectively) into eight telemetry-detectable categories, explicitly relocating capability drift and strategic drift out of the Control Plane	Neither is telemetry-observable; claiming otherwise overstates automated detection capability	7	Accepted
D-29	Split “data drift” into Data Lineage/Governance Drift (architecture concern) and AI-Model Behavioral Drift’s data-distribution-shift component (Technical AI Evaluation concern)	The term is used ambiguously across the literature; conflating the two would misroute alerts to the wrong owner	7	Accepted
D-30	Default-prohibit automated corrective enforcement for Tier 3–4 systems and any agent-authority change, with exactly one exception (pre-approved kill-switch conditions defined at G3)	Prevents the Control Plane from becoming an unsupervised actor on high-consequence systems while still allowing the one case where automated response genuinely reduces harm	7	Accepted

Chapter 8 — Governance Operating Model

Decision ID	Decision	Rationale	Chapter	Status
D-31	Grant AI Governance Body concurrence (not just input) over Portfolio funding decisions for Tier 3–4 AI-relevant initiatives specifically	Resolves Chapter 6’s open question; a narrow, risk-proportional exception to FR-20’s shallow-governance framing, not a general expansion	8	Accepted — narrow scope, logged explicitly to prevent silent creep
D-32	Reduce six candidate governance bodies to two standing bodies (expanded ARB, AI Governance Body absorbing Model Risk function)	Resolves Chapter 3’s open question about FR-09 recreating a second bureaucracy; both operate within existing COBIT-aligned governance, not parallel to it	8	Accepted
D-33	Assign TG0-bypass response authority to ARB, with Chapter 7’s Control Plane providing detection	Closes the enforcement gap Chapter 6 explicitly named and deferred; ARB has the cross-functional standing neither Portfolio nor AI Governance Body alone holds	8	Accepted
D-34	Cap every exception at 90 days with no automatic renewal at Tier 4	Prevents exceptions from becoming permanent unrecorded policy changes; makes exception volume a visible	8	Accepted

Decision ID	Decision	Rationale	Chapter	Status
		governance signal via the Transformation Scorecard		

Chapter 9 — Framework Artifacts and Templates

Decision ID	Decision	Rationale	Chapter	Status
D-35	Reject Enterprise Transformation Map, AI Capability Map, and full Model Card ownership from the framework’s artifact set	Each duplicates an artifact already owned elsewhere (external Portfolio roadmap, Capability Record filter, ML-engineering documentation) with no decision depending on a framework-owned duplicate	9	Accepted
D-36	Reclassify Business Capability Map, AI System Card, Benefit Map, and Assurance Dashboard as Derived Views rather than independently maintained artifacts	None of their content requires independent authorship; each is fully compiled from artifacts that already pass the rationalization test	9	Accepted
D-37	Add Capability Record, TG0 Record, Deployment Authorization Record, and Drift/Alert Record to the artifact set despite absence from the fifteen artifacts initially proposed	Each is already load-bearing in Chapters 5–8; omitting them from the consolidated inventory would be inconsistent with earlier chapters, not a genuine simplification	9	Accepted
D-38	Provide full ten-field templates for five representative artifacts and condensed specification for the remaining ten	Manages document scope while preserving rigor across every distinct artifact pattern (qualification, architecture, risk, agent-specific, derived view)	9	Accepted, mirroring the Chapter 2 (D-02) scoping precedent

Chapter 10 — Framework Visualization and Identity

Decision ID	Decision	Rationale	Chapter	Status
D-39	Select the operating-system metaphor for the Signature Diagram over wheel, layered, matrix, infinity-loop, control-loop-alone, and cube alternatives	The framework’s own vocabulary (Chapter 4’s “core model,” Chapter 7’s OS-scheduler analogy) had already converged on this metaphor before Chapter 10; a layered or wheel image was specifically rejected as visually undermining the framework’s own critique of static EA	10	Accepted
D-40	Recommend IRGF over fourteen more conventionally evocative but collision-positive alternatives	Every strongly evocative single-word candidate checked collided with existing 2026 products/frameworks, several severely (CALM/FINOS, TRACE/SSRN paper, CONTINUUM/TOGAF); IRGF was the only zero-collision candidate found and has the strongest substantive tie to content already developed	10	Provisional — formal trademark/domain clearance required before any external use

Chapter 11 — Validation and Stress-Testing

See Framework Design Log below (D-41 through D-54) for the fourteen accepted revisions (R1–R14) and their rationale, plus the explicit scope clarification reclassifying individual due-process mechanisms as a non-goal.

Decision ID	Decision	Rationale	Chapter	Status
D-41	Require independent (Risk & Compliance Lead) countersignature on D1–D5 scoring; add periodic retroactive audit-sampling of Tier 1–2 self-certifications	Closes confirmed, convergent gameability finding (Failure Point 1)	11	Accepted
D-42	Anchor D3 scoring explicitly to consequence-reversibility, not mechanism-reversibility	Closes specific gaming vector found in Scenario D (git-revert)	11	Accepted
D-43	Report D5 as two distinct sub-scores (Data Sensitivity, Model Uncertainty) alongside existing composite; tiering formula unchanged	Addresses AI Governance Officer critique without dimension-count creep	11	Accepted
D-44	Mandate Preventive policy-as-code boundary control + stated max audit-latency for Tier 3–4 standing-authority agents	Closes Scenario E’s core finding (periodic audit alone insufficient)	11	Accepted
D-45	Require machine-readable permission/action boundaries in G3 records for agentic systems	Makes autonomy-increase detection actually enforceable by Control Plane	11	Accepted
D-46	Narrow “autonomy increase = Major, always” to boundary-widening changes only	Responds to AI Engineer/Chief EA friction findings	11	Accepted — flagged as open tension with D-41, not fully resolved
D-47	Add Regulatory Overlay Reference as 16th primary artifact; permit external/independent validation function for regulated Tier 3–4 systems in place of AI Governance Body self-validation	Closes SR 11-7 independence conflict and sector-regime blind spot (Failure Point 2 & 6)	11	Accepted
D-48	Add Procurement role; extend ADR mandatory fields with vendor AI due-diligence evidence and an explicit evidence-gap resolution rule	Closes Scenario F’s structural gap — framework’s own motivating problem was its thinnest-governed path	11	Accepted
D-49	Add recommended (not mandatory) SaaS-discovery telemetry source to Control Plane Observe function	Partial, honestly-framed mitigation for shadow AI detection	11	Accepted
D-50	Add conditional Region/Legal Entity governance level between Enterprise and Portfolio; define cross-border AI System as one record with N jurisdictional addenda	Closes Scenario H’s structural gap — no prior federation axis for geography	11	Accepted
D-51	Add lightweight, sampling-based independent technical check for Tier 1–2 systems	Closes AI Governance Officer’s “zero independent check below Tier 3” finding	11	Accepted
D-52	Relabel T5 from governed lifecycle stage to “External Input Interface” in framework language/diagrams	Corrects appearance-of-coverage critique (Chief EA, Digital Transformation Executive)	11	Accepted
D-53	Loosen G2 internal sequencing to allow iterative/parallel Architecture Decision + Data Readiness resolution before joint exit	Responds to AI Engineer’s waterfall-mismatch finding	11	Accepted
D-54	Reclassify individual due-process/contestability mechanisms as an explicit non-goal (interface-only via Regulatory Overlay Reference), consistent with existing workforce-change non-ownership precedent	Scope discipline — legal-domain discipline, not EA governance, per Chapter 3’s existing exclusion pattern	11	Accepted (new non-goal, not a gap left silently unaddressed)

Chapter 12 — Novelty and Contribution Assessment

Decision ID	Decision	Rationale	Chapter	Status
D-55	Overall framework positioning determined as Complementary Governance Framework	Only positioning consistent with the component-level evidence in §2–5	12	Accepted
D-56	Control Plane formally repositioned from core novel mechanism to recommended implementation pattern	Confirms Chapter 7’s provisional self-assessment (D-27) and Chapter 11’s independent confirmation	12	Accepted
D-57	Two-body governance model formally repositioned from novel governance theory to practical consolidation guidance	Organizational-design synthesis, not EA research contribution; also found not to scale (Chapter 11)	12	Accepted
D-58	Core Model repositioned as a working, undervaluated ontology pending future Delphi-style expert review	EA Academic persona (Chapter 11): asserted, not derived — no minimality test performed	12	Accepted
D-59	D1–D5 reframed as a NIST AI RMF/ISO 23894-consistent implementation schema, not a competing taxonomy; formal crosswalk table required in Chapter 14	AI Governance Officer persona (Chapter 11); avoids the vocabulary-duplication critique	12	Accepted
D-60	Corrective-enforcement default-prohibition rule and Regulatory Overlay Reference reclassified from New Contribution to Integration	Anti-novelty test (§3): both are assemblable by an informed practitioner from existing standards	12	Accepted
D-61	Four components confirmed as genuine (but unvalidated) New Contributions: safety-override/tier-automation binding; autonomy-increase change-classification rule; Data-Lineage/AI-Model-Behavioral drift separation; traceability-depth classification	Survive anti-novelty test on originality grounds; explicitly flagged as unvalidated, not proven	12	Accepted
D-62	Operating-system branding retired; framework renamed IRGF (Intent-to-Runtime Governance Framework) and the central entity set renamed the Core Model	Red-team Q10 (Chapter 11): the metaphor did rhetorical work beyond what the evidence supports. A scope-stating name is consistent with the complementary-governance-framework positioning established in Chapter 12	12	Accepted

Chapter 13 — Academic Research Model

Decision ID	Decision	Rationale	Chapter	Status
D-63	Primary research question revised to explicitly name the four surviving mechanisms and an explicit comparison condition (loosely-coupled standards use), rather than restating Chapter 1’s broad version	Chapter 12’s Complementary Governance Framework finding requires a narrower, falsifiable question	13	Accepted
D-64	Six theoretical foundations classified: IT Governance theory and Cybernetics and Socio-Technical Systems Theory adopted as load-bearing; Dynamic Capabilities and Institutional Theory and TOE adopted as secondary/contextual; RBV, Complexity Theory, and Organizational Ambidexterity rejected as decorative	Section 32’s explicit instruction to discriminate rather than list every candidate theory	13	Accepted
D-65	Nine-construct conceptual model adopted, with Organizational Governance Capacity and Governance-Practice Decoupling designated as mandatory moderator/rival-	Directly responsive to Chapter 11’s recurring finding that org capacity/context swamps mechanism-level design	13	Accepted

Decision ID	Decision	Rationale	Chapter	Status
	explanation controls for any future empirical test			
D-66	Six secondary research questions and six matched propositions adopted, each traceable to a specific Chapter 11/12 finding rather than freestanding	Section 42's requirement that questions be "academically researchable rather than promotional"	13	Accepted
D-67	DSRM adopted as the overall methodological frame, with this project's own chapters explicitly mapped to its stages and the Demonstration/Evaluation stages' simulated (not field-based) nature disclosed as a limitation	Section 33's requirement to determine the most academically defensible methodology, applied honestly rather than retrofitted to look complete	13	Accepted
D-68	Empirical validation sequence set as: expert interviews → targeted (Core Model-scoped) Delphi study → 3–5 deliberately-varied case studies; large-N survey explicitly deferred as premature	Constructs are freshly defined with no validated instruments; premature survey work would produce data of unknown construct validity	13	Accepted
D-69	Systematic academic literature review formally named as an unperformed, outstanding methodological step — not assumed complete by virtue of Chapter 2's standards review	Prevents future overclaiming that this project's propositions are literature-validated	13	Accepted
D-70	Academic contribution classified as primarily reference/integration model + governance model + narrow decision model; explicitly not primarily a metamodel or a new research methodology	Consistent with Chapter 12's downgrading of the Core Model to unvalidated ontology and the project's borrowing (not contributing to) DSRM	13	Accepted

Chapter 14 — Final Framework Specification

Decision ID	Decision	Rationale	Chapter	Status
D-71	Fourteen Design Principles adopted, each tied to a named mechanism; four added beyond the illustrative list in the research brief (Independent Verification of Risk, Controlled/Detectable Autonomy, No Orphaned Systems, Falsifiable Governance Claims)	Reflects lessons Chapter 11 forced onto the design that the original illustrative principle list could not have anticipated	14	Accepted
D-72	Five-level Maturity Model adopted with Level 5 renamed from "Autonomous Governance" to "Continuously Assured"	Internal consistency with D-30's permanent human-authority requirement	14	Accepted
D-73	Ten Reference Architecture Patterns adopted in condensed form	Fulfills §36/§19, previously unaddressed by any chapter	14	Accepted
D-74	TOGAF ADM crosswalk adopted with three phases marked "Partially Replaced" rather than forcing full 1:1 correspondence	Explicit caution against forcing artificial correspondence	14	Accepted
D-75	First-cut NIST AI RMF/ISO 23894 crosswalk for D1–D5 adopted, explicitly caveated as unverified against primary source text	Fulfills Chapter 12's D-59 requirement while preserving Chapter 13's D-69 honesty discipline	14	Accepted
D-76	Minimum Viable Framework defined around independent risk-tier verification as the highest-leverage 20%, not the governance-body structure	Directly derived from Chapter 11's single most confirmed finding	14	Accepted

Appendix B — Glossary

Core Model — the eight entities (Intent, Capability, AI System, Decision, Risk, Control, Outcome, Assurance) without which the framework’s traceability claims do not function; status: unvalidated ontology (§20). **Tier** — a system’s overall risk classification (1–4), derived from D1–D5 via the Impact×Control-Deficit matrix and safety-override floor (§13). **Gate** — a governance checkpoint (G1–G5) whose authority and automation level scale with Tier (§15). **Control Plane** — the continuous observe/compare/evaluate/alert/recommend/enforce/learn loop (§21); status: recommended implementation pattern, not novel mechanism. **Drift** — divergence between deployed and approved state, across eight named categories (§21). **Agent Card** — the primary artifact recording an AI agent’s identity, permissions, tool access, and authorized action boundary (§14, §18). **Regulatory Overlay Reference** — the artifact reconciling sector-specific regulatory regimes with IRGF’s generic mechanisms (§18, added v0.2). **MVF** — Minimum Viable Framework, the ~20% subset of IRGF judged to deliver ~80% of its governance value (§26).

Appendix C — Diagram Index

#	Diagram	Source Chapter
1	Overall Framework (Executive & Practitioner views)	Chapter 10
2	Framework Layers	Chapter 10
3	AI Governance Lifecycle (S0–S10)	Chapter 5
4	EA + AI + Transformation Integration (Traceability)	Chapter 4
5	AI Risk Governance Model	Chapter 5
6	AI Agent Governance Model	Chapter 14, §14
7	Continuous Architecture Control Plane	Chapter 7
8	Framework Metamodel (erDiagram)	Chapter 4, extended Chapter 10
9	Governance Operating Model	Chapter 8
10	TOGAF-to-IRGF Mapping	Chapter 14, §24
—	Signature (concentric-ring) Diagram	Chapter 10
—	Chapter 13 Conceptual Research Model	Chapter 13

Appendix D — Design Requirements

The full 28 formal requirements (FR-01 through FR-28) are specified in Chapter 3 and are not reproduced here; every component of the specification traces to at least one requirement, consistent with the traceability discipline applied throughout this research.

DO NOT COPY

Appendix E — Version History

- **v0.1** — Chapters 1–10: initial design.
 - **v0.2** — Chapter 11: fourteen revisions (R1–R14) following adversarial stress-testing across eight scenarios, seven reviewer personas, and one red-team pass.
 - **v1.0 (the specification presented in Chapter 14)** — Chapter 12’s novelty repositioning, Chapter 13’s academic research model, and Chapter 14’s remaining components (Design Principles, Maturity Model, Reference Architectures, TOGAF and NIST/ISO crosswalks, Implementation Methodology, Adoption Roadmap) consolidated into one publication-suitable specification, with all confirmed limitations carried forward rather than resolved by omission.
-

Appendix F — Open Research Questions

Questions raised but not resolved by this research, recorded under the chapter that raised them. They are reproduced here as stated at the time rather than retrospectively edited; several were subsequently answered in later chapters, and where that occurred the answering chapter is cited in the main text.

Chapter 2 — Framework and Standards Landscape

Carried forward from Chapter 1 and updated: does TOGAF's extension mechanism, examined against the primary specification text rather than characterized from general knowledge, already provide a sanctioned way to add an AI domain — and if so, does the proposed framework become an TOGAF extension pattern rather than a freestanding framework? Is the “continuous architecture/AI conformance assurance” candidate (D-03) actually novel, or is it a re-labeling of cloud security posture management and policy-as-code practice already mature in DevOps/platform engineering — this is the single highest-priority item for the Chapter 12 anti-novelty test. Does COBIT's 2019 core governance-objective set have a formal AI extension already published by ISACA that would narrow or eliminate the AI-governance-decision-rights gap claimed here? How mature is NIST's agentic-AI-specific guidance as of the time of framework finalization, given this area is evolving quickly and the Section B “W” ratings for AI agent governance may not hold for long? Is data-mesh's federated computational governance pattern robust enough, empirically, to be relied on as a structural precedent (Section C), given it remains a practitioner concept without a formal standards body behind it?

Chapter 3 — Framework Design Requirements

Does FR-09's “architecture-integrated AI gate” concept risk recreating a second approval bureaucracy alongside COBIT's existing EDM-based IT governance gates, rather than truly integrating with them — this needs concrete testing in Chapter 8's operating-model design, not just assertion here. Can FR-12's agent-authority modeling actually be enforced technically (not just documented) using current identity and access management tooling, or does it require capability that does not yet exist in mainstream IAM platforms — relevant to FR-27's minimum-viable-implementation feasibility. Is the FR-15 risk-proportionality claim empirically achievable, or does risk-tiering itself become a new source of gaming/misclassification incentive (an organization under delivery pressure may be tempted to under-tier a system to avoid FR-09's gates) — this is a governance-design risk that Chapter 5 and Chapter 11 must address directly, not assume away.

Chapter 4 — Conceptual Framework Architecture

Does the “one metamodel, three views” structural decision (D-11) actually reduce practitioner cognitive load relative to a single unified diagram, or does presenting three views risk practitioners maintaining three mental models instead of one — this can only be tested with real users in Chapter 11's reviewer simulation, not resolved analytically here. Is collapsing Experience and Process Architecture into Business Architecture (D-13) defensible for organizations undergoing CX-led transformation specifically, where experience design may need more architectural visibility than a sub-discipline typically receives — Scenario D (retail/GenAI

customer service) in Chapter 11 should stress-test this directly. Does the AI Architecture EA-level/engineering-level boundary (Section 4.4) hold up when tested against Scenario F (coding agents generating production software), where the boundary between “architecture” and “engineering” may itself become blurred by the AI system’s own outputs? Is an 8-concept core model actually minimal, or does “Assurance” bundle several logically separable concepts (telemetry, comparison, alerting) that Chapter 7 will need to decompose — if so, the Core Model may need revision after Chapter 7 rather than being treated as settled at Version 0.1.

Chapter 5 — AI Build Governance Model

Does the five-composite risk scoring (D1–D5) lose meaningful discriminating power relative to the original twelve dimensions in genuinely ambiguous cases — for example, a system with high legal exposure but low financial impact, which D1 would currently score identically to a system with the reverse profile? This needs testing against real or simulated cases in Chapter 11/13, not assumed away. Is the Impact × Control-Deficit matrix’s specific tier boundaries (the 4×4 mapping in Section 5.3) defensible, or arbitrary — the safety-override rule was designed to catch the most obvious failure mode (averaging masks a catastrophe), but the matrix’s other boundaries have not been stress-tested against edge cases. Does reducing eleven gates to five (D-18) actually reduce governance burden in practice, or does it just concentrate the same total evidentiary requirements into fewer, heavier checkpoints that feel equally burdensome to a solution team — this is a practitioner-perception question for Chapter 11’s Chief Enterprise Architect reviewer. Is the differential escalation urgency between Risk Assurance failures and Business Outcome Assurance failures (Section 5.5) actually achievable in a single governance system, or will organizations tend to treat every assurance failure with the same urgency regardless of design intent?

Chapter 6 — Digital Transformation Governance Model

Is an input-only architecture role in Portfolio governance (Section 6.3) actually sufficient to prevent poor architecture outcomes at the portfolio level, or does it need some form of veto/escalation right for a defined subset of decisions (e.g., Tier 3–4 AI-relevant initiatives) to be credible — this is a real, unresolved tension between the shallow-governance discipline and the framework’s practical effectiveness, and should be tested directly with the Chief Enterprise Architect reviewer persona in Chapter 11. How should the TG0 enforcement gap (Section 6.4) actually be closed — is retroactive detection via Chapter 7’s continuous assurance mechanism sufficient, or does it need a harder control (e.g., architecture repository access gated on a valid TG0 reference)? That is an operating-model decision-rights question for Chapter 8, explicitly flagged here as unresolved rather than assumed. Is a single 1–5 capability maturity scale (Section 6.2) meaningful across genuinely heterogeneous capability types — a customer-facing digital capability and a back-office finance capability may not compress onto the same five anchors without losing meaning; this needs practitioner testing in Chapter 13, not analytical resolution here.

Chapter 7 — Continuous Architecture Governance

Does the D-27 “provisional” novelty verdict survive Chapter 12’s formal test, or does closer investigation of MLOps-specific observability tooling (which, unlike general CSPM, may already

model AI-specific entities like agents and RAG pipelines) narrow the novelty claim further than stated here? Is the kill-switch exception (Section 7.4) actually implementable given current agent architectures — can a “predefined catastrophic pattern” be specified precisely enough at G3 to be machine-detectable at runtime, or does this exception’s practical value collapse because most catastrophic agent behavior is not cleanly pre-specifiable? Does relocating capability drift and strategic drift out of the Control Plane (D-28) leave a real gap — is there a lighter-weight, human-assisted (not fully automated) detection mechanism worth designing for these two categories rather than leaving them purely to periodic review cadence?

Chapter 8 — Governance Operating Model

Does the narrow Tier 3–4 Portfolio concurrence right (D-31) actually get respected in practice, or does funding pressure lead organizations to under-tier AI-relevant initiatives specifically to avoid triggering it — this is the same gaming-incentive risk flagged in Chapter 3 (FR-15’s open question) now recurring at the portfolio level, and deserves specific attention in Chapter 11’s Digital Transformation Executive reviewer persona. Is a 90-day exception cap workable for genuinely long-lead-time remediations (e.g., a legacy system migration that cannot realistically complete in 90 days) — does the framework need a distinct “long-form exception with mandatory quarterly re-justification” category rather than a flat cap, and if so does that undermine the visibility benefit D-34 was designed to protect? Does the ARB’s expanded mandate (now covering five architecture domains including AI) create a capacity or expertise bottleneck in practice — this is a structural question about whether one board can credibly govern both traditional and AI architecture at scale, worth testing against Chapter 11’s organization-size scenarios (startup through multinational).

Chapter 9 — Framework Artifacts and Templates

Does the Derived View compilation mechanism (Section 9.4) actually work in practice without dedicated tooling — the design assumes an underlying system capable of live compilation across primary records, and an organization without such tooling might default back to manually maintained (and inevitably duplicated) capability maps and system cards despite this section’s design intent. Is the AI Assurance Summary’s boundary between “EA-relevant evidence reference” and “ML-engineering evaluation methodology” (per Chapter 4’s scope discipline) actually clean in practice, or will Risk & Compliance Leads find they need more of the underlying evaluation detail than a summary reference provides, especially at Tier 4 — a question for Chapter 11’s AI Engineer reviewer persona.

Chapter 10 — Framework Visualization and Identity

Does the informal web-search collision check performed here (Section 10.5) constitute adequate due diligence, or does Chapter 13/14 need a formal trademark-registry and domain-availability search before IRGF (or any alternative) can be treated as safe to publish externally — this is explicitly not resolved here and should not be treated as resolved. Is the finding that fourteen of eighteen checked names collide with existing 2026 AI-governance products itself evidence worth incorporating into Chapter 12’s novelty assessment — a crowded naming space in the exact problem area this research targets is weak but real evidence that the underlying problem is widely

recognized, which cuts toward “this is a real gap others are also chasing” rather than “this problem doesn’t exist.”

Chapter 11 — Validation and Stress-Testing

- Does R6’s narrowed autonomy-increase trigger, combined with R1’s audit-sampling backstop, net-reduce or net-increase gaming risk in practice?
- What is the actual, empirically-calibrated cost (FTE, cadence, tooling spend) of running IRGF at Tier 3–4 intensity for a mid-sized regulated enterprise?
- Does the Regulatory Overlay Reference get populated and maintained in practice, or does it join Derived Views as a tooling-dependent artifact that silently decays?
- At what actual multinational scale (number of countries, Tier 3–4 volume) does the conditional Region/Legal Entity level need to activate, and who decides?
- Is “cosmetic vs. substantive” prompt/config change classifiable by an objective test, or is it irreducibly judgment-based?
- Does a full, separately-run Organization Size × AI Maturity grid (the 11.2/11.3 exercise scoped down in this chapter) surface findings beyond what emerged organically from the eight scenarios?

Chapter 12 — Novelty and Contribution Assessment

- Does a formal NIST AI RMF/ISO 23894 crosswalk of D1–D5 reveal any dimension currently missing from IRGF’s schema, or any IRGF dimension with no counterpart worth keeping?
- Would an independent Delphi panel of EA/AI-governance experts validate the Core Model’s 8 entities as minimal and complete, add entities, or remove some as redundant with Metamodel-level (non-core model) entities?
- Of the four surviving New Contributions, which (if any) would still be judged novel by a formal, citation-backed literature search rather than this project’s own standards-landscape review — this chapter’s novelty judgments are reasoned, not literature-verified, and that distinction should be preserved going into Chapter 13.

Chapter 13 — Academic Research Model

- Do the nine constructs in §3 hold up to expert-interview scrutiny as recognizable and distinct, or do some collapse into each other in practitioners’ actual mental models (e.g., do practitioners distinguish Governance-Automation Binding Strength from Risk Mitigation Effectiveness, or treat them as the same thing)?
- Is DSRM actually the right frame, or would a different design-research tradition (e.g., Action Design Research, which more explicitly accommodates organizational intervention during the design process) fit this project’s iterative, agent-driven design process better in retrospect?

- Given that Chapter 11’s “Demonstration” was agent-simulated, how much confidence should be placed in P1–P6 even provisionally, before any real-world data exists — should Chapter 14 present them as hypotheses worth testing (this chapter’s framing) or withhold them entirely pending real demonstration?

Chapter 14 — Final Framework Specification

See §30 (Research Opportunities) — restated here as the closing research agenda for the specification as a whole.

Appendix G — Assumptions Register

Assumptions relied upon in the design that were not tested during this research. Each is recorded under the chapter in which it was made. Collectively these constitute the principal threat to the validity of the framework’s claims and are the natural starting point for the empirical validation program set out in Chapter 13.

Chapter 2 — Framework and Standards Landscape

Assumption	Origin	Status	Validation Needed
TOGAF 10’s metamodel lacks AI as a first-class domain and this is a structural, not superficial, gap	Chapter 1, reaffirmed Chapter 2	Unverified against primary text	Direct review of the TOGAF 10 content metamodel and extension guidance
The EU AI Act (European Parliament and Council of the European Union, 2024) “Digital Omnibus” has deferred certain high-risk obligations as of August 2026	Chapter 1	Based on secondary reporting, not primary legal text	Direct review of the adopted legislative text before citing exact dates/scope in later chapters
No formal COBIT AI-specific extension exists that would close the AI-governance-decision-rights gap	Chapter 2	Unverified	Search ISACA’s current publication catalogue directly
Continuous architecture/AI conformance assurance is not already substantively solved under a different name in DevOps/cloud security posture management	Chapter 2	Unverified — flagged as the highest-priority open question	Literature and product-landscape review in Chapter 12
Data mesh’s federated governance pattern is mature enough to serve as a structural precedent for the proposed framework’s operating model	Chapter 2	Practitioner-level evidence only	Case-study or empirical literature review in Chapter 13
NIST’s agentic-AI guidance maturity as characterized here (still developing, not yet closing the agent-governance gap) will not materially change before framework finalization	Chapter 2	Time-sensitive assumption	Recheck NIST publication status before Chapter 14 finalization

Chapter 3 — Framework Design Requirements

Assumption	Origin	Status	Validation Needed
TOGAF’s ADR practice can be extended to carry AI-specific fields (FR-07) without breaking compatibility with standard ADR tooling	Chapter 3	Unverified	Prototype an extended ADR template and test with existing ADR tooling in Chapter 9
Mainstream IAM/identity platforms can technically enforce the agent authority boundaries required by FR-12	Chapter 3	Unverified	Technical feasibility review in Chapter 4/5 design work
Risk-tier misclassification/gaming incentive (noted in Open Research Questions) is a real, not hypothetical, organizational risk	Chapter 3	Plausible but unvalidated	Practitioner interview or case-study evidence in Chapter 13

Chapter 4 — Conceptual Framework Architecture

Assumption	Origin	Status	Validation Needed
A shared metamodel expressed through three views will be perceived by practitioners as coherent rather than as three separate frameworks	Chapter 4	Unvalidated design bet	Chapter 11 reviewer simulation, specifically the Chief Enterprise Architect and AI Engineer personas
Experience and Process Architecture can be adequately handled as Business Architecture sub-disciplines without losing governance visibility for CX-led or	Chapter 4	Unvalidated	Stress-test against Chapter 11 Scenario D

Assumption	Origin	Status	Validation Needed
process-led transformation			
The EA-level / engineering-level boundary for AI Architecture (Section 4.4) remains stable when AI systems themselves generate architecture-adjacent artifacts (e.g., agent-generated infrastructure code)	Chapter 4	Unvalidated, flagged as a likely stress point	Stress-test against Chapter 11 Scenario F
The 8-concept core model will remain stable once the Lifecycle View (Chapter 5) and Control View (Chapter 7) are developed in full detail	Chapter 4	Provisional — “Assurance” may need decomposition	Revisit the Core Model after Chapter 7

Chapter 5 — AI Build Governance Model

Assumption	Origin	Status	Validation Needed
Four risk tiers represent roughly the finest granularity assessors can apply with consistent inter-rater agreement	Chapter 5	Unvalidated, reasoned by analogy to the dimension-consolidation logic, not tested	Chapter 13 expert validation / Delphi round should test tier agreement directly
The Impact × Control-Deficit matrix produces more defensible tier assignments than a simple weighted sum	Chapter 5	Design rationale stated, not empirically compared	Comparative testing against sample AI systems in Chapter 11
Reducing gates from eleven to five will be experienced by practitioners as reduced burden, not merely concentrated burden	Chapter 5	Unvalidated	Chapter 11 Chief Enterprise Architect reviewer persona should test this directly
Risk-driven (not effort-driven) change classification is operationally learnable by delivery teams without extensive training	Chapter 5	Unvalidated	Pilot / case-study testing in Chapter 13

Chapter 6 — Digital Transformation Governance Model

Assumption	Origin	Status	Validation Needed
Portfolio/PMO functions will meaningfully consume the Architecture Input Package (Section 6.3) even without EA holding decision authority over prioritization	Chapter 6	Unvalidated — a real adoption risk	Chapter 11 reviewer simulation (Chief Enterprise Architect, Digital Transformation Executive personas); Chapter 13 practitioner interviews
Retroactive detection (via Chapter 7) is an adequate substitute for a hard TGO enforcement control	Chapter 6	Unvalidated, flagged as a known gap rather than assumed solved	Test against Chapter 11 Scenario H (large simultaneous cloud+data+AI transformation), where bypass risk is highest
A single enterprise-wide capability maturity scale (1–5) is comparable and meaningful across heterogeneous capability types	Chapter 6	Unvalidated	Chapter 13 practitioner/expert validation

Chapter 7 — Continuous Architecture Governance

Assumption	Origin	Status	Validation Needed
The Control Plane’s application-to-metamodel novelty claim (D-27) will survive comparison against current AI-specific MLOps/observability tooling, not just general CSPM	Chapter 7	Explicitly provisional	Chapter 12 anti-novelty test, informed by a direct product-landscape review of AI observability platforms
Catastrophic agent behavior can be specified precisely enough at G3 to make the kill-switch exception (Section 7.4) practically usable, not just theoretically sound	Chapter 7	Unvalidated	Technical feasibility review against real agent architectures, informed by Chapter 11 Scenario C (autonomous agents)
Roughly two-thirds evidence-as-code eligibility (Section 7.6) holds across organizations of varying tooling maturity, not just well-instrumented enterprises	Chapter 7	Unvalidated	Chapter 13 case-study validation across organizations of different AI/DevOps maturity

Chapter 8 — Governance Operating Model

Assumption	Origin	Status	Validation Needed
An expanded ARB (five domains including AI) is organizationally capable of the review volume this implies, rather than becoming a bottleneck	Chapter 8	Unvalidated	Chapter 11 organization-size and AI-maturity scenario testing
A flat 90-day exception cap is workable across all exception types, including long-lead-time remediations	Chapter 8	Unvalidated, flagged as a likely revision point	Chapter 11/13 practitioner testing
The Tier 3–4 Portfolio concurrence right (D-31) will not itself become a gaming incentive to under-tier initiatives	Chapter 8	Unvalidated, directly analogous to a risk already flagged in Chapter 3	Chapter 11 Digital Transformation Executive persona; Chapter 13 practitioner interviews

Chapter 9 — Framework Artifacts and Templates

Assumption	Origin	Status	Validation Needed
Organizations adopting this framework will have or build tooling capable of live-compiling Derived Views rather than defaulting to manual duplication	Chapter 9	Unvalidated, a real implementation-maturity dependency	Chapter 14's Minimum Viable Framework should state what happens absent this tooling
A pass/fail evidence reference is sufficient EA-level detail for the AI Assurance Summary even at Tier 4, without needing the full evaluation methodology	Chapter 9	Unvalidated	Chapter 11 AI Engineer and Chief Risk Officer reviewer personas

Chapter 10 — Framework Visualization and Identity

Assumption	Origin	Status	Validation Needed
IRGF has no undiscovered collision beyond what an informal web search surfaced	Chapter 10	Explicitly unvalidated	Formal trademark and domain-availability search, Chapter 13/14, before any external publication
The operating-system metaphor will read as intuitive to a non-technical executive audience, not just to the technically literate	Chapter 10	Unvalidated	Chapter 11 reviewer simulation, particularly the CIO/Board-adjacent personas

Chapter 11 — Validation and Stress-Testing

- Assumed that a Risk & Compliance Lead role, as already defined, has sufficient standing and independence to meaningfully countersign D1–D5 scores (D-41) — not validated against real organizational reporting lines.
- Assumed sector-specific regulatory overlays (D-47) can be meaningfully maintained by an internal Risk & Compliance function without becoming outdated relative to fast-moving regimes like the EU AI Act.
- Assumed a “boundary-widening” test (D-46) is operationally distinguishable from a “within-boundary” change by engineers under delivery pressure — not validated.

Chapter 12 — Novelty and Contribution Assessment

- Assumed that the five reference standards used for the anti-novelty test (TOGAF, COBIT, NIST AI RMF, ISO 23894, ISO 42001) are jointly representative of “existing practice” for this test’s purposes — a narrower or wider reference set could shift some classifications, particularly for the Control Plane (where CSPM/DevOps vendor practice,

not a formal standard, is the actual prior art) and the Regulatory Overlay Reference (where GRC tooling, not a named standard, is the prior art).

- Assumed that “an informed practitioner could already assemble this” is the correct bar for disqualifying a New Contribution claim (per §41’s own phrasing) — a stricter bar (has anyone actually published this synthesis) or a looser bar (is this synthesis non-obvious) would shift some of the Integration/New Contribution boundary calls in §2–3.

Chapter 13 — Academic Research Model

- Assumed that LLM-agent-based adversarial stress testing (Chapter 11) is a legitimate, if weaker, substitute for real organizational demonstration sufficient to motivate propositions worth testing — not validated against any real case, and explicitly named as a limitation (D-67) rather than asserted as sufficient.
- Assumed that the construct definitions in §3 are operationalizable in principle (i.e., that a future researcher could design measurement instruments for them) — not tested by actually attempting to draft such instruments in this chapter.
- Assumed the six theories retained as load-bearing/secondary (§2) are the correct minimal set — a broader academic review might justify including Organizational Ambidexterity or Complexity Theory more substantively than this chapter’s reasoning allows for.

Chapter 14 — Final Framework Specification

- Assumed that consolidating limitations into one place (§29) is more useful to a future reader than leaving them distributed across thirteen separate chapter documents — not tested against actual reader behavior.
- Assumed the Minimum Viable Framework’s role-merger recommendation (§26, point 3) is a safe simplification for small organizations rather than a reintroduction of the separation-of-duties failure Chapter 11 flagged — logged as a live tension: the MVF explicitly trades away independence for feasibility at small scale, and this trade-off is asserted, not validated.

References

- Ashby, W. R. (1956). *An introduction to cybernetics*. Chapman & Hall.
- AXELOS. (2019). *ITIL foundation: ITIL 4 edition*. AXELOS.
- Barney, J. (1991). Firm resources and sustained competitive advantage. *Journal of Management*, 17(1), 99–120. <https://doi.org/10.1177/014920639101700108>
- DAMA International. (2017). *DAMA-DMBOK: Data management body of knowledge* (2nd ed.). Technics Publications.
- DiMaggio, P. J., & Powell, W. W. (1983). The iron cage revisited: Institutional isomorphism and collective rationality in organizational fields. *American Sociological Review*, 48(2), 147–160. <https://doi.org/10.2307/2095101>
- European Parliament and Council of the European Union. (2024). *Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act)*. Official Journal of the European Union. <https://eur-lex.europa.eu/eli/reg/2024/1689/oj>
- Federal Chief Information Officers Council. (2013). *Federal enterprise architecture framework, version 2*. U.S. Office of Management and Budget.
- International Organization for Standardization [ISO]. (2018). *Risk management — Guidelines* (ISO 31000:2018).
- International Organization for Standardization [ISO] & International Electrotechnical Commission [IEC]. (2022). *Information security, cybersecurity and privacy protection — Information security management systems — Requirements* (ISO/IEC 27001:2022).
- International Organization for Standardization [ISO] & International Electrotechnical Commission [IEC]. (2023a). *Information technology — Artificial intelligence — Management system* (ISO/IEC 42001:2023).
- International Organization for Standardization [ISO] & International Electrotechnical Commission [IEC]. (2023b). *Information technology — Artificial intelligence — Guidance on risk management* (ISO/IEC 23894:2023).
- International Organization for Standardization [ISO] & International Electrotechnical Commission [IEC]. (2024). *Information technology — Governance of IT for the organization* (ISO/IEC 38500:2024).
- ISACA. (2018). *COBIT 2019 framework: Introduction and methodology*. ISACA.
- March, J. G. (1991). Exploration and exploitation in organizational learning. *Organization Science*, 2(1), 71–87. <https://doi.org/10.1287/orsc.2.1.71>
- National Institute of Standards and Technology [NIST]. (2020). *Zero trust architecture* (NIST Special Publication 800-207). U.S. Department of Commerce. <https://doi.org/10.6028/NIST.SP.800-207>
- National Institute of Standards and Technology [NIST]. (2023). *Artificial intelligence risk management framework (AI RMF 1.0)* (NIST AI 100-1). U.S. Department of Commerce. <https://doi.org/10.6028/NIST.AI.100-1>

- National Institute of Standards and Technology [NIST]. (2024). *The NIST cybersecurity framework (CSF) 2.0* (NIST CSWP 29). U.S. Department of Commerce. <https://doi.org/10.6028/NIST.CSWP.29>
- Organisation for Economic Co-operation and Development. (2024). *Recommendation of the Council on Artificial Intelligence* (OECD Legal Instrument 0449, 2024 amendment). <https://legalinstruments.oecd.org/en/instruments/oecd-legal-0449>
- Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45–77. <https://doi.org/10.2753/MIS0742-1222240302>
- Teece, D. J., Pisano, G., & Shuen, A. (1997). Dynamic capabilities and strategic management. *Strategic Management Journal*, 18(7), 509–533.
- The Open Group. (2022). *The TOGAF® standard, 10th edition*. The Open Group.
- The Open Group. (2026). *ArchiMate® 4 specification*. The Open Group.
- Tornatzky, L. G., & Fleischer, M. (1990). *The processes of technological innovation*. Lexington Books.
- Trist, E. L., & Bamforth, K. W. (1951). Some social and psychological consequences of the longwall method of coal-getting. *Human Relations*, 4(1), 3–38. <https://doi.org/10.1177/001872675100400101>
- Weill, P., & Ross, J. W. (2004). *IT governance: How top performers manage IT decision rights for superior results*. Harvard Business School Press.
- Zachman, J. A. (1987). A framework for information systems architecture. *IBM Systems Journal*, 26(3), 276–292. <https://doi.org/10.1147/sj.263.0276>