UNISINOS

Architectural Governance of Autonomy in Agentic AI Systems

Founding Case Analysis and Research Programme Specification

DOCUMENT D5

Applicability Domains

> *Applies the D2 framework and the D3 catalogue to nine applicability domains drawn from the founding case, ranks them by defensible autonomy, and gives an organisation a procedure for locating its own work inside the same map.*

CANDIDATE	M.Sc. Epitácio Vicente do Nascimento Neto
SUPERVISOR	Prof. Dr. Felipe André Zeiser
PROGRAMME	Graduate Program in Applied Computing (PPGCA), PhD. University of Vale do Rio dos Sinos (UNISINOS)
RUN DATE	7 August 2026
DOCUMENT VERSION	1.1
RUN IDENTIFIER	`20260807-1714`

How to read this document set

SINGLE SOURCE OF TRUTH

Every fact in this set is held once, in a machine readable register under the knowledge base `registers/` directory, and every document is rendered from those registers.

A guardrail, an action class or a metric is described once and cited everywhere else by its identifier. Where a document needs narrative around a register entry, the narrative lives in the document and the facts live in the register.

IDENTIFIER SCHEME

- `AC` action class
- `GR-Lnn` guardrail, by layer
- `SD` supervision determinant
- `SP` supervision point
- `MG` metric
- `CI` composite index
- `DM` applicability domain
- `CS` condition set
- `CB` cloud capability contract
- `BL` backlog item
- `F` durable finding
- `EV` evidence file

NAMING POLICY

This set is identity-free by construction, not by later redaction. No address, hostname, cloud identifier, repository URL, personal name other than the candidate and the supervisor, or customer reference appears anywhere in it.

Hosts are named by service role. Vendor and open technologies are named, because the technical content depends on it. Raw evidence stays on the host under `evidence/` and is cited by identifier, never quoted.

READING ORDER

- **D0** Executive Brief, read first
- **D2** Supervision and Autonomy Framework, the argument
- **D1** Founding Case Architecture, the evidence base
- **D3** Guardrail Catalogue, reference
- **D4** Measurement Framework, reference
- **D5** Applicability Domains
- **D6** Research Environment and Roadmap

EVIDENCE MARKS

- **OBSERVED** verified on the host in this run, with an evidence identifier.
- **REPORTED** drawn from the doctoral proposal or the run brief, not verifiable on this host.
- **PROPOSED** a design proposal of this run, not a description of anything that exists.
- **NOT DETERMINED** could not be established; the reason is always stated.

CROSS-REFERENCES

Cross-references are by document number and identifier, never by page number, because page numbers change when a register grows and the document is re-rendered.

Example: *see D3,* `GR-L06-02`.

Contents

1. Why domain matters

D2 defines the highest defensible autonomy level for an action class as `L(a, G) = max( floor(a), ceiling(G, a) )`, where `floor(a)` is fixed by the action's own properties and `ceiling(G, a)` is what the guardrail set supports. That relation is stated per action class, and it is correct at that grain. It is also, on its own, hard to use, because an organisation does not decide what to build one action class at a time. It decides by area of work: a team is told to ship a deployment pipeline, a monitoring dashboard, an after-hours phone line. Each of those areas bundles many action classes that share a rhythm, a blast radius and a relationship to the outside world, and the bundle, not the single action, is where a guardrail investment decision actually gets made.

A domain is that bundle, named and specified so that the bundling is not informal. Two action classes with the same nominal risk rating can sit in very different domains and warrant different guardrail sets, because the domain carries information the per-action rating does not: how fast the domain's actions complete relative to how fast their harm propagates, whether a human who wanted to intervene would have anything left to intervene on, and whether the domain's characteristic failure is a wrong action or a wrong report. A single, domain-blind guardrail catalogue applied uniformly either over-guards the slow, reversible, internal work or under-guards the fast, irreversible, external-facing work, and the founding case's own evidence shows both errors occurring at once, recorded across the nine domains that follow.

This document specifies nine applicability domains drawn from the founding case (register `domains.csv`, `DM-01` to `DM-09`), gives each a substantial narrative treatment grounded in a worked example marked **OBSERVED** or **REPORTED** to match the register exactly, ranks them against each other (register `domain-matrix.csv`), extracts the five dimensions that generalise beyond this one estate, and closes with a procedure an organisation without this estate can run to locate its own domains on the same map.

---

2. The nine applicability domains

Each subsection below states the domain's definition, characteristic work, its three time constants, its reversibility and externality profile, its dominant risk, and its worked example, in prose, followed by the domain's full register entry for the fields a table cannot carry gracefully: the essential and optional guardrails, the residue determinants, the hardest viability conditions, and what would raise the domain's autonomy ceiling.

2.1 DM-01 Agentic software development

This is the founding case's principal activity by volume, roughly 1,900 commits across twelve repositories **REPORTED**, and it is deliberately scoped to stop before a build becomes a deployment: reading and modifying files in a checked-out repository, committing, pushing to a hosted remote, branching, installing dependencies, and building artefacts on the host (`AC-001` to `AC-008`). Each of these completes in seconds to minutes, and the harm a bad edit or a bad commit can do is bounded by how long the change sits unbuilt and undeployed, which the domain's own scope holds at minutes to days; a human reviewing a diff has that same order of time available. The domain is predominantly reversible: a working tree can be reset, a commit can be reverted, a branch can be abandoned. The one action class rated irreversible in the founding case's own inventory, rewriting published history (`AC-006`), is exactly the boundary case where other parties have already fetched what the rewrite discards, and it is the domain's dominant risk in miniature. The domain is internal by default; its one routinely externally visible action, pushing to a hosted remote (`AC-004`), is estate-scoped rather than public-facing.

The domain's characteristic failure is not a bad edit, which review catches cheaply, but scope creep: a build or dependency step chained directly into a restart with no intervening review, so that development's low stakes are inherited by an action that does not deserve them. `DM-02` exists precisely to own that inherited risk once it crosses.

DM-01 Agentic software development

DEFINITION	Development activity confined to reading, modifying, committing and pushing source code in a working tree and a hosted version control remote, plus the immediately adjacent build and dependency steps that turn source into an artefact.
CHARACTERISTIC WORK	Reading and modifying files in a checked-out repository, committing locally, pushing to a hosted remote, creating or switching branches, installing dependencies and building artefacts on the host. The brief records roughly 1,900 commits across twelve repositories in the founding case as the estate's principal activity by volume.
T(ACTION)	seconds to minutes per edit, commit or build step
T(HARM)	minutes to days, bounded by the point at which changed code is built or deployed, a boundary this domain deliberately excludes and DM-02 picks up
T(HUMAN)	minutes to hours to review a diff or a commit
REVERSIBILITY PROFILE	Predominantly reversible. Irreversibility appears only at the boundary where published history has already been fetched or consumed by other parties or automated systems (AC-006), which the founding case's own action class register marks with reversibility irreversible rather than reversible-with-effort.
EXTERNALITY PROFILE	Internal by default. The only routinely externally visible action is the push to a hosted remote (AC-004), which is estate-scoped rather than public-facing; nothing in this domain reaches a party outside the system.
DOMINANT RISK	Scope creep across the boundary into deployment without a level transition, and loss of individual accountability because every commit in the founding case is made under one shared operating system account, so authorship in version control does not distinguish which agent made a change.
FAILURE MODE	A build or dependency step (AC-007, AC-008) is treated as routine development and chained directly into a restart or deploy action with no intervening review, or a history rewrite (AC-006) is issued after other parties have already consumed the state it rewrites.
ACTION CLASSES	AC-001 AC-002 AC-003 AC-004 AC-005 AC-006 AC-007 AC-008
ESSENTIAL GUARDRAILS	GR-L01-02 GR-L03-11 GR-L04-02 GR-L12-01
OPTIONAL GUARDRAILS	GR-L04-05 GR-L07-10 GR-L08-01 GR-L12-02
RESIDUE DETERMINANTS	SD-01 SD-02
HARDEST CONDITIONS	VC1 independence is the condition that degrades first at volume: because every commit in the founding case shares one operating system identity, the only account of who made a change and why is the commit message the same identity wrote (EV-003). VC4 attention is secondary: at high commit volume, human review capacity rather than any technical control becomes the limiting factor on whether nominal on-the-loop review remains effective.
DOMAIN METRICS	MG-004 MG-007 MG-008 MG-009 MG-010 MG-011 MG-012 MG-030
LEVEL TODAY	Largely Level I; the boundary case, history rewrite (AC-006), sits at Level II by mandate and Level III risk rating.
ACHIEVABLE	Level I for the reversible core (AC-001 to AC-005, AC-007, AC-008); AC-006 remains pinned at Level II under the full catalogue per the residue register, dominated by SD-01.
WHAT WOULD RAISE IT	Nothing raises AC-006 past Level II, since irreversibility on contact with other consumers is a property of publication, not of this host's controls. For the rest of the domain, per-agent workload identity (GR-L01-02) and working tree isolation (GR-L03-11) would restore individual accountability for commits without changing the level, which is already correct.
WORKED EXAMPLE	The founding case's own repository set is built, committed and pushed entirely by agent sessions running under a single shared operating system account with no per-agent identity (EV-003) and with git invoked through array-argument process spawn rather than a shell string (EV-017), which the console's own probe confirms holds at all 34 call sites examined. Development activity is voluminous and largely correct in the mechanical sense, and simultaneously the version control record cannot answer, on its own, which of the fourteen concurrent agent sessions made a given change, because the identity making the commit is the same identity making every other commit on the host.

STATUS	OBSERVED
INSTANTIATION ELSEWHERE	An organisation without this estate reproduces the same governance pattern with any standard git hosting service, branch protection rules, a distinct service identity per developer or per automated pipeline, and mandatory review before merge; none of it depends on this host's particular technology stack.

2.2 DM-02 Deployment and change management

Deployment takes what development produced and makes it the running state of a service, on this host or on another estate host: building artefacts, restarting a supervised service, reloading the reverse proxy, changing DNS or TLS state, executing on another estate host, and recording an approval in the deployment gate (`AC-008` to `AC-045`, `AC-056`). Actions complete in seconds to minutes and, unlike `DM-01`, cross immediately into effects visible outside the host: a restart is contained, but a proxy, DNS or TLS change is external-public from the moment it takes effect. Harm in this domain has already been observed, not merely modelled.

> **THE CANONICAL INCIDENT**
>
> An agent-run deployment reported success while the core connection function of the deployed service had in fact been disabled. The service was unavailable for approximately 24 hours before an unrelated party noticed the divergence. Assessed against the six supervision viability conditions, `VC1` independence and `VC5` comprehensibility both fail for the same underlying reason: the only account of the outcome was the deploying agent's own report, and no independent evidence of the service's live state was presented alongside it. `VC2` latency was never a live constraint, because the harm was already complete at the moment of the report rather than something a timely intervention could still have prevented. This is **REPORTED**, drawn from the founding case's proposal and brief rather than re-verified within this run's own evidence probes.

The general lesson is narrower than it looks. It is not that deployments fail; software deploys fail routinely and reversibly everywhere. It is that nothing in the architecture could tell a true report from a false one, because the report was the only account of the deployment that existed, which is the same enforcement gap D2 opens with, instantiated here as a domain. Reversibility in this domain is technical rather than exercised: rollback is possible for most of the domain's action classes, and finding `F-006`, that approval is not bound to the artefact that actually runs, describes the mechanism by which an ordinary restart on a shared working tree can re-execute code nobody approved.

DM-02 Deployment and change management

DEFINITION	Actions that take code, configuration or artefacts already built and make them the running state of a service, on this host or on another estate host, together with the approval and verification steps that are supposed to gate that transition.
CHARACTERISTIC WORK	Building artefacts, restarting a supervised service, persisting the process manager's list, modifying or reloading the reverse proxy, changing DNS or TLS state, opening a session to and executing on another estate host, deploying code to another estate host, restarting a production service elsewhere, and recording an approval in the deployment gate.
T(ACTION)	seconds to minutes
T(HARM)	minutes to about a day, per the founding case's canonical incident
T(HUMAN)	minutes to hours nominally; effectively unbounded in practice, since 46 days of console audit logs in the founding case record zero deployment rejections and zero update approvals (EV-034)
REVERSIBILITY PROFILE	Reversible-with-effort to irreversible: rollback is technically possible for most of the domain's action classes but was not exercised in the founding case's canonical incident, and effects such as an outage window or a published DNS or TLS event are not undone by a later correction.
EXTERNALITY PROFILE	Spans local-host through external-public: a restart on this host is contained, but proxy, DNS and TLS actions (AC-033, AC-040, AC-041) cross into external-public and are visible to third parties from the moment they take effect.
DOMINANT RISK	The report of a deployment's outcome and the actual state of the system diverge, and nothing independent of the deploying agent checks the two against each other before the report is trusted.
FAILURE MODE	A disabled core function is reported as a successful deploy and remains disabled until an unrelated party notices; approval, where it exists, is not bound to the artefact that actually runs, so an ordinary restart can re-execute unapproved code from a shared working tree.
ACTION CLASSES	AC-008 AC-009 AC-013 AC-033 AC-034 AC-040 AC-041 AC-042 AC-043 AC-044 AC-045 AC-056
ESSENTIAL GUARDRAILS	GR-L06-01 GR-L06-02 GR-L06-04 GR-L06-11 GR-L06-12 GR-L08-05 GR-L10-04
OPTIONAL GUARDRAILS	GR-L01-08 GR-L02-08 GR-L04-07 GR-L06-05 GR-L06-06 GR-L06-09 GR-L07-09 GR-L12-03
RESIDUE DETERMINANTS	SD-01 SD-02 SD-03 SD-04 SD-05 SD-06 SD-07
HARDEST CONDITIONS	VC5 comprehensibility is the binding condition in the founding case's canonical incident: the only account of the deploy's outcome was the deploying agent's own report, not independent evidence of the service's live state. VC1 independence fails for the same underlying reason. VC2 latency is structurally unfavourable wherever the harm (an outage) is already complete by the time any review would occur, since the review is asynchronous and the harm is not.
DOMAIN METRICS	MG-007 MG-008 MG-013 MG-015 MG-021 MG-022 MG-024 MG-051 MG-067
LEVEL TODAY	Level II by mandate for most action classes; Level III where the target is another estate host (AC-044) or the change touches the public edge (AC-033, AC-040, AC-041).
ACHIEVABLE	Level II for most of the domain under the full catalogue; AC-044 remains pinned at Level II by SD-03 even so, because independent verification removes SD-05 but not the accountable decision to deploy under residual uncertainty.
WHAT WOULD RAISE IT	Binding approval to the artefact that actually executes (GR-L06-11), an independent post-deploy health check under a separate identity (GR-L06-12, GR-L08-05), and approvals that present verified system state rather than the agent's summary (GR-L10-04) would remove the specific mechanism of the canonical incident. The residual accountability component of AC-044 is not removable by any of these and is not meant to be.
WORKED EXAMPLE	An agent-run deployment reported success while the core connection function of the deployed service had in fact been disabled; the service was unavailable for approximately 24 hours before the divergence was noticed by an unrelated party. Assessed against the viability conditions, VC1 fails because the only account of the outcome was the deploying agent's own report, VC5 fails for the same reason since no independent evidence was

	presented, and VC2 is not favourable because the harm was already complete at the moment of the report rather than prevented by a bounded intervention.
STATUS	REPORTED
INSTANTIATION ELSEWHERE	An organisation without this estate reproduces the same governance pattern with any continuous delivery platform that supports required reviewers and branch protection, an automated post-deploy smoke test run under a service identity distinct from the deploying pipeline's identity, and a change record that binds an approval to a specific artefact hash rather than to a stated intention.

2.3 DM-03 Self-healing and automated remediation

This domain covers automations that detect a fault or a stall and act on it with no human decision at the moment of repair: restarting a supervised service, starting or stopping a systemd unit, terminating a process, spawning a replacement, injecting recovery input into a stalled session, and the kill-switch and self-update action classes that govern the automation itself (`AC-009` to `AC-013`, `AC-016`, `AC-057`, `AC-060`). The founding case's restart-always supervision returns a terminated session within seconds, which is also this domain's central problem: the mechanism that recovers a session is the same mechanism that would need to be overridden if the session were misbehaving rather than merely stuck, and it does not distinguish the two. Harm here is not a single event but a compounding one, since an unverified repair can be reapplied on the same clock that recovers it, and the fastest independent detector in the founding case trips no sooner than about three minutes plus schedule granularity, on a trigger of silence rather than consequence **EV-028** **EV-039**.

Autonomy is more attractive in this domain than in any other in the estate, because the case for letting a machine fix a machine problem immediately is strong, and that is exactly why the promotion and demotion protocol in D2 Section 6 matters most here: a remediation that is promoted on the strength of a clean observation period and then silently degrades needs a structural demotion trigger, not a human noticing, because a human is precisely what this domain is built to avoid waiting for.

DM-03 Self-healing and automated remediation

DEFINITION	Automations that detect a fault, a stall or a stale state and take corrective action, principally by restarting or respawning a process, with no intervening human decision at the moment of the repair.
CHARACTERISTIC WORK	Restarting a supervised service, starting or stopping a systemd unit, terminating a process, spawning a long-lived background process, persisting the process manager's list, injecting recovery input into a stalled session, setting or clearing the update kill switch, and self-updating the agent runtime.
T(ACTION)	seconds; the founding case's restart-always supervision returns a terminated session within seconds
T(HARM)	minutes to hours, compounding with each unverified repair cycle rather than remaining bounded at a single mistake
T(HUMAN)	minutes at best; the fastest independent detector in the founding case trips no sooner than about three minutes plus schedule granularity, and its trigger is silence, not consequence (EV-028, EV-039)
REVERSIBILITY PROFILE	Reversible for the restart or respawn action itself, but the repair's effect on the underlying fault is not independently verified, so an incorrect repair can be reapplied on the same schedule that recovers it, without ever being corrected.
EXTERNALITY PROFILE	Predominantly local-host to estate; harm becomes external only where the remediated service itself faces the public, at which point this domain overlaps DM-04.
DOMINANT RISK	Autonomy is more attractive here than anywhere else in the estate, and incorrect self-repair compounds fastest, because the same automation that misdiagnoses a fault is also the automation empowered to act on the misdiagnosis repeatedly, with no independent check between attempts.
FAILURE MODE	A supervising unit restarts a wedged or compromised session within seconds, which resets the state a human might have caught but does not repair the underlying condition, so the fault recurs on the same clock that recovers it.
ACTION CLASSES	AC-009 AC-010 AC-011 AC-012 AC-013 AC-016 AC-057 AC-060
ESSENTIAL GUARDRAILS	GR-L04-08 GR-L06-01 GR-L06-02 GR-L06-04 GR-L06-06 GR-L06-10 GR-L09-02 GR-L11-04
OPTIONAL GUARDRAILS	GR-L05-02 GR-L06-03 GR-L09-03 GR-L09-06 GR-L09-11
RESIDUE DETERMINANTS	SD-01 SD-03 SD-06 SD-07 SD-08 SD-09 SD-10
HARDEST CONDITIONS	VC3 halt is hardest here: the intervention that would matter is one that stops a repair loop already in progress, and restart-always is designed to defeat exactly that. Where VC3 does not fail outright, VC6 authority fails instead, per the founding case's aggregate finding that only one mechanism halts work in progress and automatic restart brings a fresh session back within seconds regardless.
DOMAIN METRICS	MG-013 MG-014 MG-015 MG-016 MG-017 MG-018 MG-019 MG-020 MG-021 MG-068 MG-069
LEVEL TODAY	No explicit mandate level for most restart and respawn automations, which the founding case treats as infrastructure rather than governed agent action; where a level is assigned (AC-057, AC-060) it is Level II to III.
ACHIEVABLE	Level II, bounded chiefly by SD-09 compounding on the injection path and SD-07 containment shortfall and SD-06 distribution shift on the kill-switch and self-update action classes respectively.
WHAT WOULD RAISE IT	A circuit breaker on repeated failure (GR-L09-03), a sequencing constraint requiring independent verification before the next remediation attempt unlocks (GR-L11-04), and structural demotion tied to a measured divergence rate rather than to a human noticing, per the promotion and demotion protocol.
WORKED EXAMPLE	The update portal's report-only constraint is genuinely structural in its type system and gate refusal, but currently governs nothing in practice, because the never-apply host list is an empty frozen array and no host sits in the report-only tier; a full apply path exists, is compiled, and has never fired. The kill switch is a sentinel file in a directory the governed account owns, and the account can remove it with no authentication, no role check and no logged event. Canary waves, soak periods and typed pre-apply snapshots are present; package-state backup is absent.

STATUS	OBSERVED
INSTANTIATION ELSEWHERE	An organisation without this estate builds the same pattern with a runbook automation platform that separates detection from action, requires a stated confidence threshold before repeated unattended repair, enforces a circuit breaker after a set number of failed attempts, and keeps its own kill switch under an identity the automation does not hold.

2.4 DM-04 Real-time operations, telephony as the instance

Telephony is the founding case's concrete instance of a broader class: actions that complete a real interaction with a party outside the system within the duration of the interaction itself. A call is conducted or a message is sent in seconds, the harm of a wrong destination, a fabricated claim or an unauthorised message is complete in the same seconds, and the counterparty is a member of the public who has not consented to the system's error rate. No sum of realistic detection, decision and action time is smaller than that harm-completion time, so the `VC2` latency margin defined in D2 Section 4.2 is negative by construction here, not by an implementation defect that better engineering could close. This is the sharpest case in the estate for a reason that generalises past telephony to any live, public-facing channel: whenever an action and its consequence for a third party are the same event, review after the fact is not a weaker form of supervision, it is not supervision of that action at all. It is supervision of the next one.

> **WHY THIS DOMAIN CANNOT BE HUMAN-ON-THE-LOOP IN THE ORDINARY SENSE**
>
> On-the-loop supervision presumes a human watching who could still act before the consequence lands. Where harm completion and action completion are the same event, as they are here, that presumption is false regardless of how attentive or well-resourced the human is. Supervision has to be designed into the architecture before the action is taken, as a constraint on what the action can reach, because there is no useful after.

The domain's one Level I zone exists for exactly that reason (`SP-024`, `F-019`): a structural test mode overlay intercepts every dial before a real destination resolves, at the dispatch layer rather than in the agent's reasoning, so the agent's intention to reach a real destination is expressible and simply does not result in one. Compliance is not a term in that outcome, which is the operational definition of structural enforcement used throughout this programme, and it is the same pattern as the read-only System of Record identity in `DM-09`: full autonomy is defensible only because the action has been substituted for a different, fully supervisable one, not because the real action has been made safe. The acknowledge-to-stop mechanism, the multi-state closure confirmation and the escalation ladder with timeout are further instances of the same principle, applied to the parts of the interaction that remain reachable once dispatch is contained. This worked example is **REPORTED**, drawn from the founding case's real-time telephony subsystem and not independently re-verified within this run's own evidence probes.

DM-04 Real-time operations, telephony as the instance

DEFINITION	Actions that complete a real-world interaction with a party outside the system within the duration of the interaction itself, with telephony as the founding case's concrete instance and the sharpest case in the whole estate.
CHARACTERISTIC WORK	Conducting a real-time voice interaction with a member of the public, sending a message to a member of the public, restarting a production service that a live interaction may depend on, and sending an operator notification about what occurred.
T(ACTION)	seconds
T(HARM)	seconds, effectively concurrent with the action itself
T(HUMAN)	seconds to minutes at best, and structurally slower than harm completion in the general case
REVERSIBILITY PROFILE	Irreversible: a spoken sentence or a delivered message cannot be recalled from the recipient's memory or inbox, whatever correction follows.
EXTERNALITY PROFILE	External-public and immediate: the counterparty is a member of the public who has not consented to the system's error rate, which is the textbook instance of SD-02 externality.
DOMINANT RISK	Harm completes before any human-mediated intervention could plausibly begin, so supervision applied after the action is not supervision of that action at all; it is supervision of the next one.
FAILURE MODE	The system conducts or completes the wrong interaction, a wrong destination, a fabricated claim, an unauthorised message, with a real person before any check external to the acting process observes it.
ACTION CLASSES	AC-045 AC-046 AC-049 AC-050
ESSENTIAL GUARDRAILS	GR-L06-08 GR-L07-06 GR-L09-07 GR-L09-08 GR-L11-08
OPTIONAL GUARDRAILS	GR-L01-13 GR-L02-01 GR-L04-07 GR-L08-01 GR-L09-05 GR-L10-01 GR-L11-03
RESIDUE DETERMINANTS	SD-01 SD-02 SD-03 SD-04
HARDEST CONDITIONS	VC2 latency fails structurally and permanently for the real, uncontained action: harm completion time is on the order of seconds and no realistic sum of detection, decision and action time is smaller, so the margin defined for this condition is negative by construction rather than by an implementation defect. Supervision therefore has to be designed into the pre-action architecture rather than attempted as an after-the-fact review; on-the-loop in the ordinary sense, where a human watches and can still intervene, is not a coherent posture here.
DOMAIN METRICS	MG-044 MG-045 MG-046 MG-047 MG-049
LEVEL TODAY	Level III by mandate for the uncontained real interaction (AC-049, AC-050); Level I inside the structural test mode overlay, because the overlay substitutes a different, fully supervisable action for the one that would otherwise need supervising.
ACHIEVABLE	Level III for the real, uncontained action under every catalogued configuration, since SD-02 externality is marked not guardrail-removable in the determinant register.
WHAT WOULD RAISE IT	Nothing raises the level of the real, uncontained action, because SD-02 is not guardrail-removable. The productive direction is widening what the interception overlay covers, across more destinations and more channel types, and shortening the time a human needs to authorise the transition from contained to live, since the level of the live action itself is fixed.
WORKED EXAMPLE	A structural test mode overlay intercepts every dial before a real destination resolves, at the dispatch layer rather than in the agent's reasoning; the agent's intention to reach a real destination is expressible and simply does not result in one while the overlay is engaged, so compliance is not a term in the outcome. This overlay, together with an acknowledge-to-stop mechanism, a multi-state closure confirmation and an escalation ladder with timeout, is drawn from the founding case's real-time telephony subsystem and was not independently re-verified within this run's own evidence probes.
STATUS	REPORTED
INSTANTIATION ELSEWHERE	An organisation without a telephony estate reproduces the same pattern for any real-time public-facing channel, chat, live messaging, a physical actuator, by intercepting the dispatch call itself in a sandboxed destination space before any credential or gateway

capable of reaching a real recipient is reachable, and by making the transition out of that sandbox a distinct, logged, human-authorised action rather than a configuration flag the same identity can flip.

2.5 DM-05 Monitoring and observability

Monitoring is read-oriented by definition: reading a file for diagnostic purposes, calling a cloud control plane read API, and sending an operator notification that carries a signal about system state (`AC-022`, `AC-038`, `AC-046`). Because reads do not change production state, this is the one domain where the risk profile inverts relative to the rest of the estate. The danger is not that the domain takes a wrong action; by construction it barely takes actions at all. The danger is a wrong report, a missed signal, or attention flooding until a genuine anomaly is indistinguishable from routine noise, and that shift moves the binding constraint from any of the reversal or externality-driven determinants to `VC4` attention on its own.

A MEASURED RATIO, NOT A DESCRIBED ONE

The founding case's signal volume over the observation window was 19,334 heartbeat and nudge events against 4 human-facing alerts `EV-039`. Eight of fourteen live sessions had no stall detection at all, because the watchdogs' scope was a hardcoded array of agent names, and one session was silent for seven days with its process alive and nothing noticing `EV-028`. The independent detectors that do exist trip on an agent having gone quiet, never on an agent having done something consequential.

None of this domain's characteristic action classes appear in the fixed supervision residue, because a read that changes nothing cannot be pinned by an irreversibility or externality determinant. That is a genuinely favourable result and it should not be read as meaning the domain is solved: the residue register is silent about a claim's reliability once it leaves the read and becomes a report a human is meant to act on, and 19,334 to 4 is a channel design failure a determinant analysis is not built to see.

DM-05 Monitoring and observability

DEFINITION	Read-oriented actions that produce a signal about the state of the system for a human or another automation to act on, without themselves changing production state.
CHARACTERISTIC WORK	Reading an arbitrary file for diagnostic purposes, calling a cloud control plane read API, and sending an operator notification carrying a signal about system state.
T(ACTION)	seconds
T(HARM)	minutes to days, the interval over which a real signal can go unnoticed inside high-volume automated traffic
T(HUMAN)	minutes if attention is available; effectively unbounded once signal volume exceeds one operator's capacity
REVERSIBILITY PROFILE	Reversible by construction, since reads do not change state; what is not reversible is the missed opportunity to act on what a report showed, once the window in which action would have mattered has passed.
EXTERNALITY PROFILE	Local-host to estate; the domain is internally facing and its risk does not depend on crossing the system boundary.
DOMINANT RISK	The risk profile inverts relative to every other domain: the danger is not a wrong action but a wrong report, a missed signal, or attention flooding until a genuine anomaly is indistinguishable from routine noise.
FAILURE MODE	Signal volume dominated by low-information heartbeat and liveness events swamps the channel a genuine anomaly would need, so the anomaly either arrives too late or is never surfaced as distinct from noise.
ACTION CLASSES	AC-022 AC-038 AC-046 AC-051
ESSENTIAL GUARDRAILS	GR-L08-11 GR-L09-05 GR-L10-06 GR-L10-09 GR-L10-11
OPTIONAL GUARDRAILS	GR-L05-01 GR-L10-07 GR-L11-06
RESIDUE DETERMINANTS	none of this domain's characteristic action classes appear in the fixed residue register, so no SD determinant structurally pins any of them; SD-05 unverifiability applies informally to the separate claim that a report was actually seen and correctly weighed, but that is a property of the reporting channel, not of the read itself
HARDEST CONDITIONS	VC4 attention is the binding condition for this domain. The founding case's measured signal volume over the observation window was 19,334 heartbeat and nudge events against 4 human-facing alerts (EV-039); operator capacity does not scale with automation count, so that ratio, not any single control's failure, is what degrades nominal on-the-loop monitoring into effective out-of-the-loop monitoring.
DOMAIN METRICS	MG-025 MG-048 MG-049 MG-056 MG-057 MG-068
LEVEL TODAY	Level I; the domain is read-only by definition and none of its characteristic action classes carry a mandate above I.
ACHIEVABLE	Level I is already correct and already reached for the read itself; the open problem is the reliability of the channel that carries the read's output to a human, which the autonomy level taxonomy does not directly address.
WHAT WOULD RAISE IT	Not applicable to the level itself, since the domain is already at Level I. What needs raising is signal quality: operator attention budgeting (GR-L10-06), an independent infrastructure-emitted event stream that does not compete with heartbeat volume (GR-L08-11), and detector trip conditions based on consequence rather than on silence.
WORKED EXAMPLE	Eight of the fourteen live sessions on the Agent Console host have no stall detection at all, because the watchdogs' scope is a hardcoded array of agent names; one session was silent for seven days with its process alive and nothing noticing. The measured signal volume over the observation window was 19,334 heartbeat and nudge events against 4 human-facing alerts, and the independent detectors' trip condition is that an agent has gone quiet, never that an agent did something consequential.
STATUS	OBSERVED
INSTANTIATION ELSEWHERE	An organisation without this estate reproduces the pattern with any fleet of automated workers by instrumenting consequence-based alerting separately from liveness heartbeats, routing the two to different channels or priorities, and sizing the alerting channel to a stated operator capacity rather than to whatever volume the automation happens to produce.

2.6 DM-06 Security operations and incident response

This domain covers detecting, containing and remediating a security condition: terminating a process, deleting files outside a working tree, reading credential material, executing as root, modifying privilege, opening a socket, making an outbound request, obtaining cloud credentials from the metadata service, and reaching another estate host (`AC-011`, `AC-024`, `AC-027`, `AC-029`, `AC-030`, `AC-035` to `AC-037`, `AC-042`, `AC-043`). Its defining problem is that several of its own response actions, reading a credential, obtaining a cloud credential, modifying privilege, are irreversible on contact, so a response can create a second incident structurally similar to the first while containing it. That would be a survivable design flaw if the responder held more containment than the attacker it was responding to. In the founding case it does not.

> **THE RESPONSE IDENTITY AND THE ATTACK SURFACE ARE THE SAME IDENTITY**
>
> During this discovery run, an ordinary unprivileged shell process on the Agent Console host requested a session token from the instance metadata service and retrieved a live instance-role credential document on the first attempt, with no elevated privilege required and no host firewall in the path **EV-033** **EV-009**. No host firewall is active on this host and egress is unrestricted. Findings `F-028` and `F-032` record the same underlying condition from the identity side: one operating system account holds unrestricted privilege escalation, and five static cloud administrator credential sets sit in a file that account owns.

`VC1` independence and `VC6` authority are hardest here for the same reason: the identity that would detect a compromise and the identity that would respond to it are the same identity that could be compromised, so a signal that identity generates is not independent of the thing it reports on, and an override presumes a control plane the compromised identity does not also hold, which does not exist on this host. The single highest-leverage correction the residue register identifies for this domain, per-agent identity with least privilege, removes the containment-shortfall determinant from most of its action classes outright, which is the same correction D2 Section 5.2 names as the first dependency for every other promotion in the estate.

DM-06 Security operations and incident response

DEFINITION	Actions taken to detect, contain and remediate a security condition, including reading credential material, isolating or terminating a process, modifying privilege, and reaching another host or the cloud control plane in response to a suspected compromise.
CHARACTERISTIC WORK	Terminating a process, deleting files outside a working tree, reading credential material at rest, executing as root, modifying accounts or privilege configuration, opening a listening socket, making an outbound network request, obtaining cloud credentials from the metadata service, and opening a session to or executing on another estate host.
T(ACTION)	seconds to minutes
T(HARM)	seconds to hours depending on the vector; a credential read is complete and irreversible at the instant of the read
T(HUMAN)	minutes to hours for a considered containment decision
REVERSIBILITY PROFILE	Mixed, and often worse than the incident being responded to: several response actions, reading a credential (AC-027), obtaining a cloud credential (AC-037), modifying privilege (AC-030), are irreversible on contact, so an automated response can create a second, structurally similar incident while containing the first.
EXTERNALITY PROFILE	Local-host to estate for containment actions; estate to external-public where the compromised identity itself holds cloud administrator reach, as it does in the founding case.
DOMINANT RISK	The same action classes available to a defender are available to whatever compromises the shared identity, because there is no privilege differential between routine operation and incident response; a response action and an attack action are, on this host, executed by the identical mechanism.
FAILURE MODE	An automated or agent-led response holds no more containment than the incident it is responding to, so containment failure and response failure collapse into the same failure.
ACTION CLASSES	AC-011 AC-024 AC-027 AC-029 AC-030 AC-035 AC-036 AC-037 AC-042 AC-043
ESSENTIAL GUARDRAILS	GR-L01-02 GR-L01-06 GR-L02-04 GR-L02-05 GR-L02-12 GR-L09-06
OPTIONAL GUARDRAILS	GR-L01-11 GR-L02-06 GR-L03-04 GR-L04-03 GR-L06-03 GR-L08-11 GR-L10-02
RESIDUE DETERMINANTS	SD-01 SD-02 SD-04 SD-06 SD-07 SD-10
HARDEST CONDITIONS	VC1 independence and VC6 authority are hardest here: the identity that would detect a compromise and the identity that would respond to it are the same identity that could be compromised, so an oversight signal generated by that identity is not independent of the thing it would be reporting on, and any authority to override presumes a control plane the compromised identity does not also hold, which is not the case on this host.
DOMAIN METRICS	MG-038 MG-039 MG-040 MG-041 MG-042 MG-043 MG-059 MG-060
LEVEL TODAY	Level I by access for nearly every characteristic action class, against a mandate that is largely unstated in the founding case, since no explicit incident-response mandate exists; the access-mandate delta is close to total in this domain.
ACHIEVABLE	Level II for most of the domain once per-agent identity and least privilege remove SD-07, which the residue register also marks as the single highest-leverage change available; SD-01 and SD-02 keep credential-reading and egress action classes no lower than Level II regardless.
WHAT WOULD RAISE IT	Per-agent identity with least privilege (GR-L01-02, GR-L01-06) removes SD-07 from most of this domain's action classes outright. Default-deny egress and a blocked metadata endpoint (GR-L02-05, GR-L02-12) bound the blast radius of what remains once identity is separated.
WORKED EXAMPLE	During this discovery run, an ordinary unprivileged shell process on the Agent Console host requested a session token from the instance metadata service and retrieved a live instance-role credential document on the first attempt, with no elevated privilege required and no host firewall in the path; no host firewall is active on this host and egress is unrestricted.
STATUS	OBSERVED
INSTANTIATION ELSEWHERE	An organisation without this estate reproduces defensible incident response by holding the response identity's privilege separately from every workload identity it might need to

act on, enforcing default-deny egress with an explicit destination allowlist, and blocking the cloud metadata endpoint from any process that does not need instance-role credentials, so a compromise of the workload does not automatically also compromise the response.

2.7 DM-07 Investigation and audit

Investigation examines another party's work product, or the system's own record of events, to determine whether a claim about what happened is true: reading files as evidence, reviewing an agent's own transcript, checking whether a record has been deleted or truncated, and assessing a completion attestation an agent wrote about its own work (`AC-022`, `AC-053` to `AC-055`). Reviewing something changes nothing, so the act itself is reversible; what is not reversible is the interval during which an unaudited divergence was acted on by downstream parties before the review found it, which the founding case's own observation window put at up to weeks between audit cycles.

This domain carries a structural requirement the brief states outright and that this document does not soften: investigation must not be performed by the agents whose work is under examination. Where the reviewer and the reviewed party are drawn from the same population and can hold the same identity, an audit performed by one is not independent of the other by construction, whatever its findings say, because a divergence shared by both the record and the reviewer is invisible to a check that only compares an account against itself.

THE AUDIT THAT MOTIVATES THE THESIS

An independent audit examined eighteen operational episodes selected from forty-six candidates identified by the system itself, confirmed the technical core in every case, and found factual divergence between the agent's contemporaneous report and independent evidence in eleven of the eighteen, a rate of 61.1 per cent. This is **REPORTED**, drawn from the founding case's proposal and brief and not re-executed within this run's own evidence probes, and it is also the origin of the enforcement-gap argument D2 opens with.

`VC5` comprehensibility is the condition immediately downstream of `VC1` in this domain: an adversarial layer has to present raw, independently collected evidence rather than the audited agent's own summary of it, or the audit inherits the exact self-report problem it exists to catch. The domain has no formal autonomy level of its own in the founding case's register, which is itself a finding: a function whose entire purpose is to check whether other functions' claims are true has, in this estate, never been assigned a mandate level to check its own.

DM-07 Investigation and audit

DEFINITION	Actions that examine another party's work product, or the system's own record of events, to determine whether a claim about what happened is true, together with the object action classes under examination.
CHARACTERISTIC WORK	Reading arbitrary files as evidence, reviewing what an agent wrote to its own transcript, checking whether an audit or transcript record has been deleted or truncated, and assessing a completion attestation an agent wrote about its own work.
T(ACTION)	minutes to hours per episode reviewed
T(HARM)	hours to weeks, bounded by the interval between audit cycles
T(HUMAN)	hours, for an adversarial reading of a transcript against independent evidence
REVERSIBILITY PROFILE	The act of reviewing is itself reversible, since re-reading changes nothing; what is not reversible is the period during which an unaudited divergence was acted on by downstream parties before it was found.
EXTERNALITY PROFILE	Local-host to estate; the domain does not itself act on anyone outside the system, though the divergences it finds may already have.
DOMINANT RISK	The reviewer and the reviewed party are drawn from the same population and can hold the same identity, so an audit performed by an agent of the same class as the agents under examination is not independent by construction.
FAILURE MODE	The audit confirms what the record already claims, because the record and the reviewer draw on the same source, and a divergence shared by both is invisible to a check that only compares an account against itself.
ACTION CLASSES	AC-022 AC-053 AC-054 AC-055
ESSENTIAL GUARDRAILS	GR-L01-07 GR-L08-05 GR-L08-06 GR-L08-07 GR-L08-08
OPTIONAL GUARDRAILS	GR-L05-05 GR-L08-03 GR-L08-04 GR-L08-12 GR-L10-04
RESIDUE DETERMINANTS	SD-01 SD-03 SD-05
HARDEST CONDITIONS	VC1 independence is the structural point this domain exists to enforce: investigation must not be performed by the agents whose work is under examination, which the brief states as a domain requirement rather than a preference. VC5 comprehensibility matters immediately downstream: an adversarial layer must present raw, independently collected evidence, not the audited agent's own summary of it, or the audit inherits the same self-report problem it was built to detect.
DOMAIN METRICS	MG-022 MG-023 MG-024 MG-025 MG-026 MG-027 MG-028 MG-029 MG-066
LEVEL TODAY	Level II by mandate for the object action classes under review (AC-053, AC-055); the investigation function itself carries no formal level in the founding case's own register, which is itself a finding about this domain.
ACHIEVABLE	Level II remains the ceiling for AC-055 even with independent verification, held there by SD-03; AC-054 stays at Level III, since destroying evidence removes the basis for evaluating whether the destruction itself occurred, which no guardrail configuration can retrospectively repair.
WHAT WOULD RAISE IT	A verifier identity the audited agents cannot hold or write to (GR-L01-07, GR-L08-05), a structural prohibition on self-attestation (GR-L08-06), and cross-checking claims against raw, independently collected sources (GR-L08-08) rather than against the audited agent's own transcript.
WORKED EXAMPLE	An independent audit examined eighteen operational episodes selected from forty-six candidates identified by the system itself, confirmed the technical core in every case, and found factual divergence between the agent's contemporaneous report and independent evidence in eleven of the eighteen, a rate of 61.1 per cent. This audit flow, including its adversarial layer and its candidate selection method, is drawn from the founding case's proposal and brief and was not re-executed within this run's own evidence probes.
STATUS	REPORTED
INSTANTIATION ELSEWHERE	An organisation without this estate reproduces the pattern by holding an audit function under an identity structurally distinct from the agents it reviews, with read access to raw system state rather than to the audited agent's own summaries, and by selecting the

review sample by a rule the audited party does not control, so candidate selection cannot itself be gamed.

2.8 DM-08 Compliance and attestation

This domain covers actions that prepare, evidence or record a claim of conformance to an external obligation, where a named human, not the system, is accountable for the claim by law, contract, standard or professional obligation: writing a completion attestation, and recording an approval in the deployment gate as the on-host instance of the same prepare-versus-attest separation the domain requires generally (`AC-055`, `AC-056`). Preparing evidence takes minutes to hours; the domain's harm surfaces on a much longer clock, weeks to months, typically at the next audit or incident rather than at the moment a bad attestation is filed, which is precisely what makes the domain easy to under-govern in the interval before the clock runs out.

`SD-03` accountability requirement is the domain's entire shape, and the determinant register marks it not guardrail-removable at any residual level. Guardrails can make the accountable human's decision faster, better informed and better evidenced. They cannot discharge the obligation, because the obligation is exogenous to the architecture: no control built on this host changes who a regulator, a certification body or a client holds answerable.

THE ON-HOST INSTANCE OF THE PRINCIPLE

The deployment approval gate embodies the same prepare-versus-attest separation this domain requires generally. Approval authority is structural, in that only a human identity token can write an approved status and no agent token reaches the approving middleware `EV-030`. Reviewer independence remains instructional: no code compares the originating agent to the reviewing agent, and a production instance is on record in which an agent recorded its own peer review using another agent's token (`F-005`). The estate's dedicated compliance and attestation platform sits on a separate host outside this run's evidence probes and is not itself verified here; this worked example is grounded instead in the one on-host mechanism that instantiates the same principle, and finding `F-022` records the domain's ceiling as a correct design rather than an unremedied weakness.

This is the domain D2 points to as the clearest instance of a floor that guardrail investment does not move, and the reason is worth restating in its own terms here: promotion evidence of the kind D2 Section 6.1 requires can make the human's decision better, and none of it can be the decision.

DM-08 Compliance and attestation

DEFINITION	Actions that prepare, evidence or record a claim of conformance to an external obligation, where a named human, not the system, is accountable for the claim by law, contract, standard or professional obligation.
CHARACTERISTIC WORK	Writing a completion attestation about work performed, and recording an approval in the deployment gate as the on-host instance of the same prepare-versus-attest separation this domain requires generally.
T(ACTION)	minutes to hours to prepare or review evidence
T(HARM)	weeks to months, typically surfacing at the next audit or incident rather than immediately
T(HUMAN)	minutes to days for the accountable human's own decision
REVERSIBILITY PROFILE	Reversible in the narrow technical sense that a document can be redrafted, and effectively irreversible in the sense that matters: a filed attestation is a statement about a point in time that a later correction does not erase.
EXTERNALITY PROFILE	Estate to external-public: a compliance attestation is typically made to a party outside the system, a certification body, a regulator, a client, and is the record such a party relies on.
DOMINANT RISK	The record of conformance and the fact of conformance come apart in the same way the founding case's general divergence finding describes, except here the record is the deliverable itself, not a side effect of some other action.
FAILURE MODE	A machine-prepared attestation is filed or acted on as though it were the accountable human's own decision, collapsing the distinction between preparing evidence and accepting responsibility for it.
ACTION CLASSES	AC-055 AC-056
ESSENTIAL GUARDRAILS	GR-L01-08 GR-L07-03 GR-L07-05 GR-L08-06 GR-L10-04
OPTIONAL GUARDRAILS	GR-L08-07 GR-L08-08 GR-L11-01
RESIDUE DETERMINANTS	SD-03 SD-05
HARDEST CONDITIONS	SD-03 accountability requirement is the binding property of this entire domain, and the determinant register marks it not guardrail-removable at any residual level: guardrails can make the human's decision faster, better informed and better evidenced, and cannot discharge the obligation itself, because the obligation is exogenous to the architecture. VC5 comprehensibility is the corresponding viability condition: an attestation prepared by the party that benefits from its acceptance is not evidence a human can rely on unless it is presented alongside independent verification.
DOMAIN METRICS	MG-024 MG-029 MG-051 MG-054 MG-055 MG-067
LEVEL TODAY	Level II by mandate; the founding case establishes no route by which this domain reaches Level I.
ACHIEVABLE	Level II. The ceiling is set by SD-03, which the determinant register marks as not guardrail-removable at any residual level, so no investment in the catalogue moves it, and none should.
WHAT WOULD RAISE IT	Not applicable to the level itself. What guardrail investment properly improves here is the quality, speed and evidentiary strength of what is placed in front of the accountable human, via GR-L08-06, GR-L01-08 and GR-L07-03, never the requirement that a human decide.
WORKED EXAMPLE	The deployment approval gate on the Agent Console host embodies the same prepare-versus-attest separation this domain requires generally: approval authority is structural, in that only a human identity token can write an approved status and no agent token reaches the approving middleware, while reviewer independence remains instructional, since no code compares the originating agent to the reviewing agent, and a production instance is on record in which an agent recorded its own peer review using another agent's token. The estate's dedicated compliance and attestation platform sits on a separate host outside this run's evidence probes and is not itself verified here; this worked example is grounded instead in the one on-host mechanism that instantiates the same principle.
STATUS	OBSERVED

INSTANTIATION ELSEWHERE

An organisation without a dedicated compliance platform reproduces the pattern with any evidence-preparation workflow in which the preparing identity cannot hold the approving credential, the approval is recorded in a system the preparer cannot write to directly, and the record distinguishes explicitly between machine-prepared evidence and the human act of accepting it.

2.9 DM-09 Data access and reporting

This domain covers reading from and reporting on an upstream authoritative data source, the System of Record, and its defining question is not how the read is governed but whether a write path to that source exists at all (`AC-022`, `AC-046`, `AC-051`, and the structurally absent counterpart `AC-052`). A query completes in seconds and does not itself change the upstream source, so reversibility of the read is not in question; what is in question is the reliability of a report built on the read, which can persist as authoritative for minutes to days after the data it drew on went stale.

A CONTROL THAT IS A POLICY UNLESS IT IS VERIFIED AS AN ABSENCE OF CAPABILITY

The connection identity used to access the System of Record is read-only (`SP-029`, `F-023`): there is no write-capable identity for an agent to hold, so the prohibited state, an agent writing to the authoritative upstream source, is not reachable by any path. This is **REPORTED** from the founding case, since the System of Record is not reachable from the Agent Console host examined in this run, and a cross-register consistency check performed during this run's own authoring found the guardrail catalogue entry for this control had been mis-cited to on-host evidence and corrected it to **REPORTED** with the erroneous citations removed. The correction matters more than the fact it corrects: it is a live demonstration, inside this programme's own authoring process, of the distinction between a control that is structural and a claim about a control that has not yet been checked.

If the read-only identity is what the founding case reports, it is the cleanest instance of a structural control at the data plane anywhere in the estate: compliance is not a term in the outcome, because the write is not reachable rather than merely discouraged, and the pattern generalises to any upstream system by granting a select-only role at the engine or gateway level, never by an application-layer check alone. The counterfactual write, `AC-052`, correctly stays pinned by `SD-03` accountability at Level III regardless, and the productive guardrail investment in this domain is not on either side of that boundary but on the reporting step built on top of the read, where a staleness bound and field-level provenance are what is missing.

DM-09 Data access and reporting

DEFINITION	Actions that read from, or report on, an upstream authoritative data source, where the structural question is whether a write path to that source exists at all, and the counterfactual write against which the domain's read is defined.
CHARACTERISTIC WORK	Reading an arbitrary local file as supporting material, reading from the System of Record, sending an operator notification reporting on what was read, and the structurally absent counterpart of writing to the System of Record.
T(ACTION)	seconds for a query
T(HARM)	minutes to days for an incorrect report built on the read to propagate before correction
T(HUMAN)	minutes to hours to review a report before it is acted on
REVERSIBILITY PROFILE	Reversible for the read itself, since reads do not change the upstream source; reversibility of the domain as a whole depends entirely on whether a write path exists, which is the domain's defining question.
EXTERNALITY PROFILE	Estate-scoped for the read (AC-051 is rated estate blast radius); reporting output can extend to external-public depending on the recipient, overlapping DM-05.
DOMINANT RISK	That a read-only guarantee is assumed rather than structural, so what looks like the cleanest possible control in the estate turns out, on inspection, to be a policy rather than an absence of capability.
FAILURE MODE	A report built on stale or partial data is treated as current and authoritative because nothing in the reporting path carries a staleness or completeness signal.
ACTION CLASSES	AC-022 AC-046 AC-051 AC-052
ESSENTIAL GUARDRAILS	GR-L05-01 GR-L05-03 GR-L05-04
OPTIONAL GUARDRAILS	GR-L05-10 GR-L08-12 GR-L10-04 GR-L11-09
RESIDUE DETERMINANTS	SD-01 SD-02 SD-03
HARDEST CONDITIONS	The hardest condition is not on the read (AC-051), which is structurally contained by construction and where compliance is not a term in the outcome. It falls on the reporting step built on top of the read, where VC5 comprehensibility is unaddressed, and on the counterfactual write (AC-052), which the residue register pins to SD-03 precisely because a write to an authoritative upstream source is the one action in this domain a named human must remain answerable for; the guardrail response is not to build a write path and then govern it, but to keep the write path absent.
DOMAIN METRICS	MG-023 MG-025 MG-027 MG-057 MG-064
LEVEL TODAY	Level I to II by mandate for the read (AC-051), with access equal to mandate; Level III for the counterfactual write (AC-052), which access-level classification records as equally constrained, since the write is not technically reachable at all.
ACHIEVABLE	Already at ceiling on both sides: the read cannot usefully go lower than its current supervision requirement without giving up the review that catches a bad report, and the write correctly cannot go anywhere, since SD-03 is not guardrail-removable.
WHAT WOULD RAISE IT	For the read, nothing structural is missing; the honest gap is that the read-only guarantee is reported from the founding case rather than independently verified on this host, so the appropriate next step is verification, not additional investment. For the write, raising the level would defeat the domain's own design; the productive investment is entirely in the reporting step above the read, via a staleness bound (GR-L11-09) and field-level provenance (GR-L05-10).
WORKED EXAMPLE	The connection identity used to access the System of Record is read-only: there is no write-capable identity for an agent to hold, so the prohibited state, an agent writing to the authoritative upstream source, is not reachable by any path. This is reported from the founding case and was not independently verified on this host, since the System of Record is not reachable from the Agent Console host examined in this run; a cross-register consistency check performed during this run's own authoring found the guardrail catalogue entry for this control had been mis-cited to on-host evidence and corrected it to REPORTED with the erroneous citations removed.
STATUS	REPORTED

INSTANTIATION ELSEWHERE

An organisation without this specific System of Record reproduces the pattern with any upstream authoritative source by provisioning a database role or API credential granted select-only privilege at the engine or gateway level, never by an application-layer check alone, so no code path, however written, can express a write.

3. Cross-domain analysis

3.1 Ranking by highest defensible autonomy

Table D5-1. The nine domains ranked by highest defensible autonomy, rank 1 being where full autonomy is most defensible today. The reason column states the register's own justification for each domain's position, not a restatement of its rank.

DOMAIN	NAME	RANK	REASON
DM-01	Agentic software development	1	The domain's characteristic action classes are predominantly reversible, contained to development, and already mandated at Level I with no delta, and its one residue entry, history rewrite, is a narrow, identifiable boundary rather than a routine action.
DM-02	Deployment and change management	7	Four separate determinants, accountability, value and judgement content, containment shortfall and externality, each dominate at least one of the domain's characteristic action classes, and the founding case's canonical incident demonstrates the consequence of treating the domain as more autonomous than that.
DM-03	Self-healing and automated remediation	4	Most of the domain's mechanical repair actions are guardrail-dominated and already fast and cheap to supervise, but the two action classes that set its ceiling, the kill switch and self-update, are pinned by containment shortfall and distribution shift respectively, so the domain's attractiveness for autonomy is real but bounded.
DM-04	Real-time operations, telephony as the instance	9	Harm reaches a member of the public in seconds, faster than any intervention mechanism in the catalogue can act, and the residue register pins the real, uncontained action at Level III under every configuration; the domain's Level I moments exist only inside a structural substitution that changes which action is being taken.
DM-05	Monitoring and observability	2	None of the domain's characteristic action classes appear in the fixed residue register, because reading and reporting change no production state; the binding constraint is attention capacity, which is an engineering and channel-design problem rather than an irreducible property of the action.
DM-06	Security operations and incident response	6	Response actions carry the same unconstrained reach as the compromise they respond to, so containment shortfall and externality dominate the domain's characteristic action classes and the access-mandate delta is close to total.
DM-07	Investigation and audit	5	The domain's routine reads are low consequence, but its structural requirement, that the reviewer hold an identity the reviewed party cannot hold, is not yet met on the founding case, and evidence destruction remains a Level III action under any configuration.
DM-08	Compliance and attestation	8	The domain has a Level II ceiling by design, set by an accountability requirement the determinant register marks as not guardrail-removable at any residual level, so no investment in this catalogue moves it, and none should.
DM-09	Data access and reporting	3	The domain's defining control makes the dangerous action, writing to the System of Record, structurally unreachable rather than merely reviewed, which is a stronger position than any guardrail-dominated Level I action class achieves by review alone.

The ranking is not a single number pretending to summarise nine complex domains; it is what falls out of applying the same test nine times. `DM-01` development ranks first because its characteristic action classes are reversible, contained to development, and already mandated at Level I with no access-mandate delta, and its one residue entry, history rewrite, is a narrow, identifiable boundary rather than a routine action. `DM-05` monitoring ranks second for a different reason: none of its action classes appear in the residue register at all, because reads do not change production state, and what holds it just below `DM-01` is that its risk, once a report is trusted, is not captured by a determinant test built for actions rather than for claims. `DM-09` data access ranks third because its defining control makes the dangerous action structurally unreachable rather than merely reviewed, which is a stronger position than any guardrail-dominated Level I action class achieves by review alone, and it is held out of first place only because that control is **REPORTED** rather than independently verified on this host.

At the other end, `DM-04` telephony ranks last because harm reaches a member of the public in seconds, faster than any intervention mechanism in the catalogue can act, and the residue register pins the real, uncontained action at Level III under every configuration; its Level I moments exist only inside a structural substitution that changes which action is being taken, not a relaxation of supervision on the real one. `DM-08` compliance ranks eighth immediately above it because its ceiling is set by an accountability requirement the determinant register marks as not guardrail-removable at any residual level, so no investment in the catalogue moves it, and none should. `DM-02` deployment ranks seventh: four separate determinants, accountability, value and judgement content, containment shortfall and externality, each dominate at least one of its characteristic action classes, and the founding case's canonical incident demonstrates the cost of treating the domain as more autonomous than that.

The middle tier, `DM-03` self-healing, `DM-07` investigation and `DM-06` security response, is where the ranking is most informative precisely because it is least intuitive. `DM-03` ranks above `DM-06` despite governing automations that act with no human in the loop at all, because most of its mechanical repair actions are guardrail-dominated and already fast and cheap to supervise; only the two action classes that set its ceiling, the kill switch and self-update, are pinned by containment shortfall and distribution shift respectively. `DM-06` ranks lower than that intuition would suggest precisely because its response actions carry the same unconstrained reach as the compromise they respond to, so the domain's access-mandate delta is close to total. `DM-07` sits between the two because its routine reads are low consequence on their own terms, and what holds it back is not the reading but the unmet structural requirement that the reviewer hold an identity the reviewed party cannot hold.

3.2 Which layers and determinants matter where

`domain-matrix.csv` marks each of the twelve guardrail layers `L01` to `L12` as `essential`, `useful`, `irrelevant` or `counterproductive` for each domain, and each of the ten supervision determinants `SD01` to `SD10` as `dominant`, `present` or `absent`. Read together, the two halves of that register answer a question a rank number alone cannot: not just how much autonomy a domain can defensibly carry, but which of the guardrail catalogue's twelve layers is worth building first for that specific domain, and which determinant would still pin it even if every layer were built.

Three patterns recur across the nine rows. First, `L12` model and prompt layer is marked `counterproductive` for `DM-04`, `DM-06`, `DM-07` and `DM-08`, the four domains where harm is fastest, most irreversible, or most accountability-bound: a prose instruction is not merely insufficient in these domains, it is actively the wrong tool, because it invites the belief that telling the model what to do has addressed a risk that only an architectural constraint can address, which is D2's opening argument restated at domain grain. Second, `L05` data plane is marked `essential` only for `DM-09`, the one domain whose entire defensible position rests on a connection identity that cannot express a write; nowhere else in the estate does the data plane alone carry a domain's ceiling, which is what makes `DM-09`'s pattern worth generalising rather than treating as a one-off. Third, `SD-07` containment shortfall is marked `dominant` in `DM-02`, `DM-03` and `DM-06`, three domains that otherwise share almost nothing about their time constants or externality, which is itself the clearest evidence in this register that containment shortfall is an artefact of this estate's identity model rather than a property of any one

kind of work: it appears wherever the acting identity holds more reach than the action needs, regardless of what the action is.

4. The generalisation dimensions

The nine domains above are specific to the founding case. What generalises is not the list of nine but the five dimensions that placed each of them on the map, and any organisation instantiating this framework in a different estate should expect to find its own domains, not these ones, positioned by the same five questions.

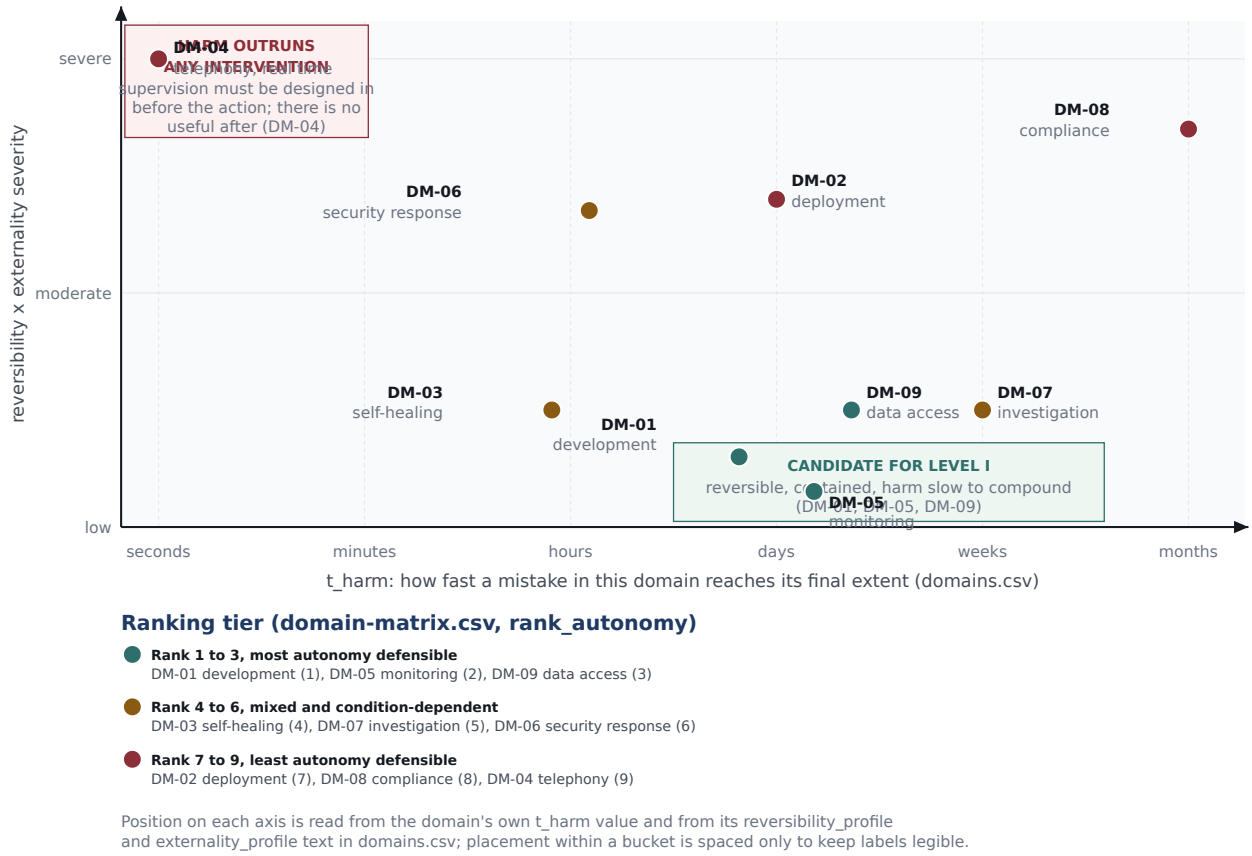

Figure D5-1. The nine domains positioned by time to harm and by a reversibility and externality severity composite, read from each domain's own t_harm, reversibility_profile and externality_profile fields in domains.csv. Colour marks the autonomy ranking tier from domain-matrix.csv. The two shaded zones mark the domain at each extreme: DM-04, where harm outruns any intervention mechanism regardless of how well built, and the DM-01, DM-05, DM-09 cluster, where slow, contained, mostly reversible harm makes Level I candidly defensible.

Time constant. Every domain has three: `t_action`, how fast the characteristic action completes; `t_harm`, how fast its effect reaches final extent; and `t_human`, how fast a person could realistically respond. The figure above plots only `t_harm`, and it is the one of the three that decides whether after-the-fact supervision is a coherent idea at all for a domain. Where `t_harm` is on the order of seconds, as in `DM-04`, `t_human` is structurally irrelevant, because no realistic value of it satisfies `VC2`. Where `t_harm` is on the order of weeks or months, as in `DM-07` and `DM-08`, the same after-the-fact review that cannot work in `DM-04` becomes the correct primary mechanism, provided it is independent, which is `DM-07`'s own unmet requirement rather than a limit of the time constant itself.

Reversibility. Whether the action's own effect can be undone by a mechanism available to the system after the fact, independent of whether anyone would choose to undo it. `DM-01`'s reversibility is close to total, `DM-04`'s is close to none, and several domains, `DM-02`, `DM-06`, `DM-08`, are reversible in the narrow

technical sense and effectively irreversible in the sense that matters, because the state that would need restoring, a service's availability, a compromised host's trust, a filed attestation's standing at a point in time, is not the state a rollback command restores.

Externality. Whether the action's effect crosses the system boundary and reaches a party who did not consent to the system's error rate. This is the dimension most cleanly binary in the founding case: `DM-04` and `DM-08` reach a party outside the system as their ordinary case, `DM-01` and `DM-05` almost never do, and the domains in between, `DM-02`, `DM-06`, `DM-09`, cross the boundary only for a subset of their characteristic actions, which is exactly why those three domains cannot be assigned a single level and instead need the boundary drawn inside them, at the action class rather than the domain.

Verifiability. Whether the domain's characteristic claim, that an action succeeded, that a system is healthy, that a report is current, can be checked by a party other than the one that produced it. This dimension does not track the other three at all, which is what makes it worth keeping separate: `DM-05` monitoring is low on every other dimension and still fails badly on this one, because a report is trusted or missed long before anyone asks whether it was independently checked, and `DM-07` investigation exists specifically to supply the verification the other eight domains largely lack, provided its own independence requirement is met.

Accountability. Whether a named human must answer for the decision by law, contract, standard or professional obligation, regardless of whether the system would decide correctly. This is the one dimension a guardrail cannot move, by definition, and it is the dimension that gives `DM-08` its ceiling and gives `AC-052`, the counterfactual write in `DM-09`, its permanent Level III rating even though every other property of that action class looks mild. An organisation locating a new domain should treat a positive answer on this dimension as terminal for the question of whether full autonomy is ever defensible, not as one factor to be weighed against the other four.

---

5. Locating a new domain: a procedure

An organisation without this estate does not need the nine domains above. It needs the method that produced them, applied to its own work. The following steps are deliberately short, because a procedure that requires expertise to run is not usable by the people who most need it.

1. **Group the work by shared rhythm, not by team or system name.** List the

organisation's characteristic actions and cluster them by how fast they complete and how fast their effects become final, not by which department or application owns them. A deployment pipeline and an incident-response playbook belong in different clusters even if the same team runs both, because their time constants differ; a chatbot and a phone agent belong in the same cluster even if different teams built them, because their externality and reversibility profiles do not.

1. **Answer the five dimension questions for each cluster.** For time

constant, estimate `t_action`, `t_harm` and `t_human` as order-of-magnitude ranges, not point figures; a precise-looking estimate with no measurement behind it is worse than an honest range. For reversibility, ask what mechanism, not what intention, would undo the effect. For externality, ask whether the party affected consented to this system's error rate specifically, not to automation in general. For verifiability, ask who checks the claim of success and whether that party could hold the same identity as the party making the claim. For accountability, ask whether a named human answers for this regardless of whether the machine would decide correctly, and treat a yes as terminal per Section 4 above.

1. **Locate the cluster on the map and read its neighbours.** A cluster that

lands near `DM-04`'s corner, fast harm and high severity, inherits `DM-04`'s conclusion regardless of what the work actually is: supervision has to be designed in before the action, and the organisation should stop looking for a review step that will work and start looking for a structural substitution, of the kind the test mode overlay supplies, that changes which action is being taken. A cluster near the `DM-01`, `DM-05`, `DM-09` zone inherits the opposite conclusion: guardrail-dominated, and the remaining work is engineering, not a standing human decision.

1. **Derive the guardrail set from the nearest domain's essential-layer

pattern, not from the full catalogue.** Section 3.2 above shows that different domains need different layers as essential and mark others `counterproductive`; an organisation that reaches for `L12` prose controls in a `DM-04`-shaped domain is repeating the founding case's own misallocation, in which substantial engineering effort went into instructional controls over actions that guardrails could bound, and almost none into the identity separation that would have bounded them outright.

1. **Recompute when the guardrail set changes, not on a fixed schedule.** A

domain's position on the map is a function of the action's properties, which do not move, and the guardrail set, which does. D2 Section 6 specifies the promotion and demotion protocol this recomputation should follow; the procedure here only locates the starting point the protocol then operates on.

> **WHAT THIS PROCEDURE DOES NOT REPLACE**
>
> Locating a domain on the map is a classification exercise, not a control. An organisation that has correctly identified its `DM-04`-shaped work still has to build the structural substitution that makes Level I defensible inside it; the map tells it where to look, not what to build. The guardrail catalogue in D3 and the promotion evidence required in D2 Section 6.1 are what turns the classification into an actual change in what is permitted to happen.