Network Working Group
Internet-Draft
Intended status: Informational
Expires: 19 November 2026

B. Morrison
Alter Meridian Pty Ltd

18 May 2026

Policy Provision and Governance Inheritance from an Organisational Identity Substrate

draft-morrison-org-alter-policy-provision-01

Abstract

This memo specifies how an artificial-intelligence agent runtime, bound at instantiation to a principal identity handle, resolves at session initialisation a target organisational identity substrate from a manifest source bound to the runtime's working context and retrieves from that substrate a typed policy stack comprising a handbook artefact, a standard-operating-procedure registry pointer, an enforcement-gate specification, and an audit-signal ingestion endpoint. The policy stack is then applied as runtime constraints on subsequent tool invocations, with audit signals emitted back to the same substrate. Policy provision occurs in the same act of session initialisation as principal identification, rather than as a separate ceremony against a side-channel governance plane. A principal concurrently bound to multiple organisational substrates operates the runtime under a deterministic composition of the several policy stacks, with cross-organisational residual conflicts routed to the peer-protocol Identity Accord ceremony [IDACCORD] rather than to a meta-federation authority. The memo is Informational. The wire surface relies on the DNS-based discovery of [MCPDNS] and the handle namespace of [IDPRONOUNS]; no new transport is introduced.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 19 November 2026.

Table of Contents

1. Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

2. Introduction

Artificial-intelligence agent runtimes operated by a human principal require, at the moment they begin acting on the principal’s behalf, a corpus of policy artefacts that constrain their behaviour: permitted and refused actions, vocabulary and tone rules, escalation procedures, audit destinations, and the standard operating procedures the principal’s organisation has adopted. In current practice these artefacts are supplied to the agent runtime by a governance plane architecturally separate from the principal’s identity infrastructure. The agent runtime authenticates to one substrate (an identity provider) and receives policy from another (a governance platform, an orchestration framework’s configuration plane, a per-tool policy console). The two substrates are joined by out-of-band integration work specific to each deployment.

This memo articulates a different arrangement and specifies the wire surface that supports it. An organisational identity substrate, addressable by the same identity handle that authenticates the principal as a member of the organisation, exposes typed surfaces over the Model Context Protocol [MCP] that carry the policy artefacts the agent runtime requires. The agent runtime resolves the substrate at session initialisation, fetches the typed surfaces, applies them as runtime constraints, and emits audit signals to the same substrate. Policy provision is a byproduct of principal identification rather than a separate ceremony.

The arrangement composes directly with the discovery mechanism of [MCPDNS], the handle namespace of [IDPRONOUNS], the attribution grammar of [IDCOMMITS], the cross-session coordination posture of [SUBSTRATE], and the cross-organisational ceremony of [IDACCORD]. No new transport, no new handle category, and no new attribution slot is introduced. The contribution of this memo is the specification of

the typed surface set, the session-initialisation flow that retrieves them, the runtime application of the retrieved enforcement-gate specification, the audit-signal flow back to the substrate, the live-update propagation, the multi-organisational composition rule, and the compliance-state inheritance posture.

3. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 RFC2119 [RFC8174] when, and only when, they appear in all capitals, as shown here.

The following terms are defined for the purposes of this document. Terms previously defined by the referenced Morrison-family memos retain their established meaning and are reproduced here only when operative for the present specification.

~handle A principal identity handle as defined by [IDPRONOUNS]. A Sovereign-tier handle is human-controlled (e.g. ~alice); an Instrument-tier handle is agent-runtime-vendor-controlled and conventionally prefixed ~cc- (e.g. ~cc-example-model). A handle's trust tier is a property of the handle, not a property of any session it appears in.

Organisational identity substrate A network-addressable system that authoritatively recognises a set of ~handles as members of an organisation, maintains the organisation's policy artefacts, and exposes typed surfaces by which authenticated agent runtimes of recognised members may retrieve those artefacts and submit audit signals back. The substrate is itself addressable by a handle, conventionally domain-qualified (e.g. ~example.com).

Policy artefact A datum retrieved from the organisational identity substrate that constrains an agent runtime's subsequent behaviour. The load-bearing policy artefacts specified by this memo are the handbook, the standard-operating-procedure registry, the enforcement-gate specification, and the audit-signal ingestion endpoint.

Enforcement gate A single rule within the enforcement-gate specification comprising a trigger predicate evaluated against tool name and arguments, an action selected from a defined action set, an applicability scope, and an explanation string. Enforcement gates are policy retrieved from the substrate; they are not hardcoded behaviour of the agent runtime.

Audit signal An append-only record submitted by the agent runtime to the organisational identity substrate's ingestion endpoint following a runtime event that meets a substrate-specified significance predicate.

Session-bind The discrete act, at agent runtime instantiation, of authenticating the bound principal handle to the resolved organisational identity substrate and retrieving the policy artefacts that will govern the session.

Manifest source A configuration surface bound to the agent runtime's working context (project-resident anchor file, DNS TXT record under the _alter. scheme of [MCPDNS], environment variable, or handle- scoped fallback) that names the target organisational identity substrate for the session.

Accord The peer-protocol cross-organisational ceremony defined by [IDACCORD]. Referenced here as the terminator of unresolvable multi-organisational policy-composition residuals.

4. Architecture

The arrangement specified by this memo comprises four operative surfaces and three flow stages.

4.1. Operative Surfaces

The organisational identity substrate SHALL expose at minimum the following four typed surfaces over the Model Context Protocol [MCP] to authenticated agent runtimes of recognised members. Each surface is addressable as a tool invocation against the substrate.

org_alter_handbook Returns the organisational handbook artefact. The handbook comprises the body of prose policy that an organisation customarily supplies to a contractor at the commencement of an engagement: voice and tone rules, vocabulary constraints, positioning rules, decision-routing rules, and any further prose policy the organisation considers operative. The surface SHALL support both whole-handbook retrieval and section-scoped retrieval by section identifier.

org_alter_sop_registry Returns the registry of standard operating procedures maintained by the organisational identity substrate. Each registry entry carries a stable identifier, a title, a status (live, draft, deprecated), a body, and an invocation verb under which the agent runtime may execute the procedure. The surface SHALL support both registry listing and individual-procedure retrieval.

org_alter_enforcement_gates Returns the specification of enforcement gates the agent runtime is to apply to subsequent tool invocations. The grammar of an enforcement gate is defined in Section 5.

org_alter_ingest Accepts audit signals submitted by the agent runtime per Section 6. The surface is append-only; admitted signals are written to the organisational identity substrate's append-only event log and are not retractable or amendable.

Additional surfaces (a roster surface, a decisions surface, a compliance surface) MAY be exposed by the organisational identity substrate; agent runtimes consulting such surfaces operate beyond the load-bearing minimum specified here.

4.2. Flow Stages

The session-bind flow comprises three stages, executed in order:

1. *Resolve.* The agent runtime determines the target organisational identity substrate by consulting the manifest source, as specified in Section 4.

2. *Retrieve.* The agent runtime authenticates to the resolved substrate using the bound principal handle's session credential and retrieves the four load-bearing policy artefacts via the surfaces of Section 3.1.

3. *Apply.* The agent runtime translates the retrieved enforcement-gate specification into runtime hooks, registers the audit-signal endpoint as the destination for subsequent significant-event emissions, and surfaces the handbook and SOP registry to the bound principal as in-context advisory material.

The three stages constitute session-bind. All three SHALL complete before the agent runtime acts on the principal's first prompt of the session. If any stage fails, session-bind SHALL fail; partial inheritance of policy is not permitted (Section 9).

5. Discovery and Resolution

The agent runtime SHALL resolve the target organisational identity substrate from a manifest source bound to the runtime's working context. Manifest sources are evaluated in the priority order below. The first source that yields a handle is operative; later sources are not consulted.

1. *Project-resident anchor.* A file at an implementation- defined path within the working directory tree (a recommended path is .alter/org-alter.toml or an [org-alter] block within pyproject.toml, package.json, or Cargo.toml) names the target substrate by handle.

2. *DNS TXT record under the _alter. scheme of [MCPDNS].* The agent runtime resolves the working directory's source- control remote (where present) to a domain name and queries _alter.<domain> per [MCPDNS]. The TXT record's org_alter field, when present, names the target substrate.

3. *Environment variable.* An implementation-defined environment variable (a recommended name is ALTER_ORG_HANDLE) carries the target substrate handle.

4. *Handle-scoped fallback.* If sources (1) through (3) do not resolve, the runtime falls back to the principal's own handle-scoped substrate, which exposes the same typed surfaces as an organisational identity substrate but is scoped to the principal alone and does not participate in multi-organisational composition (Section 8).

The resolved handle is translated to a substrate endpoint via the DNS-based resolution mechanism of [MCPDNS]. The agent runtime opens a Model Context Protocol session against the endpoint, authenticating with the bound principal handle's session credential obtained from the implementation-defined session manifest.

A substrate that does not recognise the authenticating handle as a member SHALL refuse the session; an unrecognised handle MUST NOT receive policy artefacts. The substrate MAY further refuse on trust-tier grounds: an Instrument-tier handle SHALL be admitted only when the substrate's policy explicitly admits Instrument-tier sessions from the corresponding Sovereign-tier handle's delegation.

6. Enforcement Gate Grammar

The org_alter_enforcement_gates surface (Section 3.1) returns an enforcement-gate specification. An enforcement-gate specification is a list of enforcement gates. Each enforcement gate is an object with the following fields.

id (string, REQUIRED) A stable identifier for the gate, unique within the specification. Identifiers are used as the addressing target for audit signals (Section 6) and for policy-update propagation (Section 7).

trigger (object, REQUIRED) The trigger predicate evaluated against each prospective tool invocation. The object's keys are predicate operators; the values are operator-specific patterns. Minimum operator set:

* tool_name_match (string): regular expression matched against the tool name.

* path_glob (string): glob pattern matched against any argument resolvable as a filesystem path.

* command_substring (string): substring matched against any argument carrying a command string.

* arg_arity (object): minimum and maximum bounds on argument list length.

A trigger object matches when every operator present in the object matches. Additional operators MAY be defined by the substrate and SHOULD be ignored by agent runtimes that do not understand them.

action (enum, REQUIRED) One of:

* block: the tool invocation is refused. The runtime returns the gate's explanation string to the agent reasoning loop as a synthetic error and emits a policy.violation audit signal.

* prompt-for-confirmation: the tool invocation is paused and a confirmation prompt is rendered to the Sovereign-tier principal. The invocation proceeds only on principal confirmation. A gate.confirmation-requested audit signal is emitted on prompt; a gate.confirmation-granted or gate.confirmation-denied signal is emitted on outcome.

* allow-with-audit: the tool invocation proceeds, and a gate.allowed-with-audit audit signal is emitted.

scope (object, OPTIONAL) An applicability scope restricting the gate's effect. Recognised keys:

* trust_tiers (array of strings): the trust tiers (Sovereign, Instrument, Bot) to which the gate applies. Omission indicates all tiers.

* working_context_glob (string): a glob matched against the agent runtime's working directory path. Omission indicates all contexts.

explanation (string, REQUIRED) A human-readable explanation of the gate, returned to the agent runtime on action execution. The explanation SHOULD be sufficient for the reasoning loop to surface to the principal without further substrate round-trip.

audit_emit_on (array of strings, OPTIONAL) A list of event types for which audit signals are emitted on this gate's evaluation, beyond the action-implicit signals enumerated above. Substrate-significance predicates (Section 6) may select event types not directly tied to a gate; this field carries the per-gate overrides.

When two or more gates trigger on a single prospective tool invocation (after applicability-scope filtering), the gate whose action is most restrictive prevails. Order of restrictiveness, from most to least, is block, prompt-for-confirmation, allow-with-audit.

The agent runtime SHALL NOT maintain enforcement gates outside the specification retrieved from the substrate. Gates are policy, sourced from the substrate; an agent runtime that hardcodes a gate operates outside the surface of this memo.

7. Audit Signal Flow

The agent runtime emits audit signals to the substrate's org_alter_ingest surface for runtime events that meet a substrate-specified significance predicate. The significance predicate is itself policy retrieved from the substrate; the substrate determines which events are significant, not the runtime.

An audit signal is an object with the following minimum fields:

type (string, REQUIRED) The event type. The minimum-set of event types a conformant runtime SHALL emit when triggered comprises:

* session.start at session bind, carrying the bound principal handle, trust tier, resolved substrate handle, and manifest source used for resolution.

* session.end at session termination, carrying the bound handle and a structured summary of session activity.

* tool.invoke per tool invocation that meets the substrate-specified significance predicate, carrying the tool name, a redacted argument summary, the gate evaluation outcome, and the result classification.

* policy.violation when a block gate action fires, carrying the gate identifier and the offending invocation.

* policy.update on receipt of a live-substrate policy update (Section 7), acknowledging the new policy epoch.

* gate.confirmation-requested, gate.confirmation-granted, gate.confirmation-denied on prompt-for-confirmation flow.

* gate.allowed-with-audit on the corresponding action.

payload (object, REQUIRED) Event-type-specific structured data. The substrate's significance predicate MAY constrain payload shape per event type.

attribution (object, REQUIRED) Carries the Sovereign-tier handle and any in-scope Instrument-tier handle. The grammar follows the trailer slots defined by [IDCOMMITS]; the audit-signal attribution field is the protocol-layer companion to the commit-trailer block.

timestamp (string, REQUIRED) RFC 3339 timestamp at which the runtime emitted the signal.

gate_id (string, OPTIONAL) When the signal arises from a gate evaluation, the identifier of the gate.

The substrate's audit-signal endpoint is append-only. Admitted signals SHALL NOT be retracted or amended by the emitting runtime or by the substrate operator. The structural co-location of policy and audit at the same substrate is the load-bearing property of this section; an audit channel addressable separately from the policy that governed the audited events does not satisfy this specification.

8. Live Policy Updates

The agent runtime maintains, for the duration of the session, a subscription channel against the resolved organisational identity substrate over which the substrate emits policy-update notifications. The subscription channel SHOULD be implemented as a Server-Sent Events stream [RFC8441] or equivalent unidirectional-from-substrate transport that the existing Model Context Protocol session can carry without additional authentication round-trip.

On receipt of a policy-update notification, the runtime SHALL:

1. Re-fetch the affected policy artefact via the corresponding typed surface of Section 3.1.

2. Recompute the runtime hooks of Section 5 from the updated enforcement-gate specification.

3. Atomically replace its in-memory policy state. No tool invocation issued after the atomic replacement observes a partial composition of the pre-update and post-update gate sets.

4. Emit a policy.update audit signal acknowledging the new policy epoch.

A runtime SHALL NOT require process restart to apply a policy update. An update notification that the runtime cannot apply (because the substrate returned a malformed artefact, or because the runtime's hook surface cannot represent the updated gate set) SHALL cause the runtime to emit a policy.update-failed audit signal and either retain the prior policy state and surface the condition to the principal, or terminate the session at the substrate's configured failure-mode.

9. Multi-Organisational Composition

A principal MAY be concurrently recognised by multiple organisational identity substrates. When the manifest source resolution of Section 4 returns more than one substrate handle (for example, when the project-resident anchor names a primary substrate and the session credential carries auxiliary memberships), the agent runtime composes the retrieved policy stacks under the following rules.

org_alter_handbook composition Handbooks compose by union. Where two handbooks declare conflicting sections, the substrate declared earlier in the manifest's precedence order prevails. In the absence of explicit precedence, the substrate resolved from the working- context anchor (Section 4(1) or 4(2)) prevails.

org_alter_sop_registry composition Standard-operating-procedure registries compose by union. Procedures are identified by the tuple (substrate-handle, procedure-identifier) to permit identically-named procedures across substrates without collision.

org_alter_enforcement_gates composition Enforcement gates compose by union under a strictest-applicable rule: where two gates from distinct substrates trigger on a single prospective tool invocation, the gate whose action is most restrictive prevails (order as in Section 5).

org_alter_ingest fan-out An audit signal arising from a gate whose

evaluation drew on policy from multiple substrates SHALL be emitted to the audit- signal endpoints of all participating substrates. Each substrate receives the audit trail for its share of the agent runtime's activity. Fan-out is realised by parallel append calls to each substrate's audit endpoint.

Cross-organisational residual conflicts that the composition rules above cannot resolve (for example, two substrates' handbooks declaring mutually-inconsistent positioning rules where neither is clearly subordinate under the manifest precedence) SHALL be routed to the peer-protocol Identity Accord ceremony [IDACCORD] between the participating substrates. The agent runtime emits an accord.residual audit signal to all participating substrates and suspends the conflicting action pending resolution by the substrates' principals. Resolution does not proceed via a meta- federation authority; a meta-federation authority is structurally precluded by the multi-organisational topology this section specifies.

10. Compliance-State Inheritance

At session-bind, the agent runtime inherits the organisational identity substrate's then-current compliance state as a single coherent snapshot. The snapshot comprises at minimum:

* The audit-signal endpoint URI and its current write credential.

* The enforcement-gate specification at its current epoch.

* The standard-operating-procedure registry pointer at its current revision.

* A hash of the handbook artefact at its current revision.

* The set of compliance commitments the substrate has accepted and currently asserts (for example, a refusal of a specified category of automated invocation, or a specified regulatory posture).

Inheritance SHALL be atomic. Either all snapshot elements are inherited at a single substrate epoch, or session-bind fails. A session that proceeds with a partial snapshot is non-conformant. The runtime SHALL surface session-bind failure to the principal with the substrate-returned diagnostic; it SHALL NOT silently degrade to a fallback policy stack.

Subsequent live updates (Section 7) modify the snapshot at the runtime in place but do not retroactively alter the snapshot epoch recorded at session-bind. The audit trail of a session is the sequence of policy epochs the runtime observed across its lifetime, anchored by the session-bind snapshot.

11. IANA Considerations

This memo requests no IANA action.

The four load-bearing typed surfaces named in Section 3.1 (org_alter_handbook, org_alter_sop_registry, org_alter_enforcement_gates, org_alter_ingest) are illustrative of the reference substrate operated by Alter Meridian Pty Ltd. Conforming substrates MAY name their surfaces by any convention consistent with their addressing primitive; the load-bearing contribution of this memo is the typed-surface enumeration over an organisational identity substrate, not the surface names themselves. If a future revision of this memo, or a companion specification, proposes a registry for canonical substrate-surface names, that revision will request the corresponding IANA action.

Where a substrate elects to advertise its handle in the [MCPDNS] discovery record, the org_alter field is added under the field-extension mechanism of [MCPDNS]; this memo requests no separate registry allocation. No new DNS RR types, transport identifiers, port numbers, URI schemes, or media types are introduced. The reuse of the _alter.<domain> DNS label (Section 4(2)) is per [MCPDNS] and requires no further allocation here.

The session-manifest path layout referenced by Section 4(1) and Section 4(3) is implementation-defined and is not registered.

12. Security Considerations

The arrangement specified by this memo concentrates policy, attribution, and audit on a single substrate addressable by the principal’s identity credential. The concentration is the load-bearing property; it is also the principal source of the following security considerations.

12.1. Substrate Compromise

A compromised organisational identity substrate may serve falsified policy artefacts to authenticated members, induce the runtime to emit audit signals to an attacker-controlled endpoint, or suppress update notifications to keep runtimes operating under stale gates. Mitigations:

* The substrate's policy artefacts SHOULD be served over a channel authenticated by the cryptographic identity envelope of [MCPDNS] (the _alter.<domain> Ed25519 binding) so that a consuming runtime can verify the artefact bears the substrate's declared signing key.

* Audit-signal endpoints SHOULD be pinned at session-bind time to the endpoint URI recorded in the compliance snapshot (Section 9); mid-session redirection of the endpoint SHALL require a policy.update notification carrying the new endpoint under the same signing key.

* Runtimes SHOULD treat suppressed update notifications as an observable substrate signal under the substrate-observation posture of [SUBSTRATE]; prolonged absence of update events on a substrate that asserts an active policy lifecycle is itself diagnostic.

12.2. Trust-Tier Escalation

An Instrument-tier handle that successfully presents a Sovereign-tier session credential (through credential theft, compromised session manifest, or substrate misissuance) would receive the Sovereign-tier gate set, which is by construction more permissive. Mitigations:

* The substrate SHALL bind trust tier to the handle itself, not to the session, and SHALL refuse Instrument-tier handles presenting Sovereign-tier credentials at the recognition step.

* Audit signals SHALL carry attribution per Section 6; an Instrument-tier session writing to the audit log under a Sovereign-tier attribution is detectable by post-hoc audit and by the cross-tier checks defined in [IDCOMMITS].

* Sovereign-tier confirmation prompts (Section 5's prompt-for-confirmation action) SHOULD be rendered through an out-of- band channel addressable only by the human principal, so that an Instrument-tier session in possession of the Sovereign-tier session credential cannot satisfy a confirmation on the principal's behalf.

12.3. Multi-Organisational Conflict Exploitation

A principal recognised by multiple substrates may be the vector for an exploit in which one substrate's gate is suppressed by a falsified or absent gate from a second substrate. Mitigations:

* The strictest-applicable rule of Section 8 SHALL be evaluated over the gates actually retrieved from each substrate. A substrate that fails to return its enforcement-gate specification at session-bind SHALL cause session-bind to fail for that substrate (no implicit empty-gate-set composition).

* The principal's manifest precedence declarations SHOULD be authenticated against the principal's signing key per [IDPRONOUNS] so that a forged precedence claim cannot install a less-restrictive substrate as the primary.

12.4. Live-Update Replay

An attacker positioned to observe the subscription channel may attempt to replay an aged policy.update notification to roll a runtime back to an earlier policy epoch. Mitigations:

* Update notifications SHALL carry a monotonic substrate-emitted epoch identifier.

* Runtimes SHALL reject notifications carrying an epoch less than or equal to the runtime's currently-applied epoch.

* The substrate's append-only audit log retains the ordered history of issued epoch identifiers and is consultable for post-hoc replay detection.

12.5. Pseudonymous Discovery Substrates

The handle-scoped fallback of Section 4(4) operates the typed surfaces against a principal-scoped substrate that does not assert organisational membership. An agent runtime in this configuration inherits the principal's own policy stack but does not benefit from multi-organisational composition. Implementations SHOULD surface to the principal that the session is operating in the fallback configuration so that the absence of an organisational substrate is not silently consumed.

13. Privacy Considerations

The audit-signal flow of Section 6 records the agent runtime's tool-invocation activity on the substrate. The substrate operator has visibility into the principal's session activity at the granularity of the substrate-specified significance predicate. Three privacy postures arise.

13.1. Significance-Predicate Scope

The substrate determines which events are significant and therefore audited. A significance predicate covering every tool invocation yields a complete activity log; a narrower predicate audits only events the substrate considers operative. Substrate operators SHOULD publish their significance predicates as part of the handbook artefact so that authenticated members understand the scope of audit they consent to as a function of membership.

13.2. Argument Redaction

Tool-invocation arguments SHOULD be redacted before inclusion in the tool.invoke audit signal payload. Minimum redaction practice is removal of secret material (credentials, signing keys), personally-identifying information about third parties referenced in the invocation, and any field the principal has marked sensitive in a per-session redaction profile. Substrate operators SHOULD specify their argument-redaction expectations in the handbook artefact.

13.3. Cross-Substrate Audit Fan-Out

Under multi-organisational composition (Section 8), audit signals arising from gates contributed by multiple substrates are fanned out to all contributing substrates. Each substrate receives the audit trail for invocations its policy participated in evaluating; each substrate may therefore see activity the principal did not intend to surface to it. Principals SHOULD be made aware of the fan-out posture at the time their multi-organisational membership is established; substrates SHOULD declare in their handbook artefact the fan-out events they expect to receive from sessions of members concurrently bound to peer substrates.

14. Relation to Companion Memos

This memo composes with five Morrison-family Internet-Drafts.

[MCPDNS] supplies the DNS-based discovery surface from which the manifest-source resolution of Section 4(2) draws and the cryptographic identity envelope referenced in Section 11. This memo introduces no new DNS records or labels beyond those specified by [MCPDNS].

[IDPRONOUNS] supplies the handle namespace and trust-tier taxonomy referenced throughout this memo. This memo introduces no new handle category.

[IDCOMMITS] supplies the attribution grammar that the audit- signal attribution field of Section 6 mirrors at the protocol layer. An audit signal and a Acted-By: / Drafted-With: commit trailer block carry the same attribution shape, one at runtime, one at version-control commit time.

[SUBSTRATE] supplies the substrate-observation posture under which the runtime treats absence of expected update notifications as a substrate signal (Section 11). Substrate observation also supplies the cross-session coordination floor against which multiple concurrent runtimes of the same principal deconflict without exchanging coordination messages.

[IDACCORD] supplies the peer-protocol ceremony to which Section 8’s cross-organisational residuals are routed.

15. Implementation Status

A reference implementation of the agent-runtime side of this specification is operated by the present author against a production substrate that exposes the surfaces of Section 3.1. The reference deployment supplies policy artefacts to instrument-tier agent-runtime sessions of recognised members and writes audit signals to the substrate’s append-only event log.

In the spirit of [RFC7942], the present author notes that this section is intended to document implementation experience and is expected to be removed before the document advances beyond the Independent Stream. No claim of interoperability is made; the reference deployment is a single substrate operated by the specification’s author.

16. Document History

draft-morrison-org-alter-policy-provision-01 (May 2026):

* Retitles the memo from "Org-Alter-Mediated Policy Provision and Governance Inheritance for Agent Runtimes Bound to a Principal Identity" to "Policy Provision and Governance Inheritance from an Organisational Identity Substrate". The retitled framing generalises the substrate above the operator-specific org_alter_* surface naming and clarifies that the load-bearing contribution is the typed-surface enumeration over the substrate, not the surface-name convention. The abbreviated title and the body terminology are retained.

* Softens the IANA Considerations section. The previous revision requested establishment of a Model Context Protocol Tool Surface Names registry with the four org_alter_* names as initial entries. The revised section requests no IANA action; the surface names are explicitly illustrative of the reference substrate, and conforming substrates MAY name surfaces by any convention consistent with their addressing primitive. A future revision or companion specification proposing such a registry remains possible.

* Folds in the architectural framing developed in the parallel draft-morrison-alter-collective-policy-provision-00 (May 2026) on the substrate-as-load-bearing-primitive question. That parallel draft is retired in favour of this revision; the organisational-identity-substrate framing it introduced is carried forward here.

* No substantive change to the typed surface set, the session-bind flow, the enforcement-gate grammar, the audit- signal flow, the live-update mechanism, the multi-organisational composition rules, or the compliance-state inheritance posture.

draft-morrison-org-alter-policy-provision-00 (May 2026):

* Initial submission.

* Specifies the four load-bearing typed surfaces of the organisational identity substrate (org_alter_handbook, org_alter_sop_registry, org_alter_enforcement_gates, org_alter_ingest).

* Defines the session-bind flow (Resolve, Retrieve, Apply).

* Specifies the enforcement-gate grammar and the strictest-applicable composition rule.

* Specifies the audit-signal flow and the append-only ingestion endpoint.

* Specifies the live-policy-update subscription and atomic-replacement requirement.

* Specifies multi-organisational composition and the cross-organisational residual route to [IDACCORD].

* Specifies compliance-state inheritance and the atomic-snapshot requirement.

17. References

17.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8615] Nottingham, M., "Well-Known Uniform Resource Identifiers (URIs)", RFC 8615, DOI 10.17487/RFC8615, May 2019, <https://www.rfc-editor.org/info/rfc8615>.

[MCPDNS] Morrison, B., "Discovery of Model Context Protocol Servers via DNS TXT Records", 2026, <https://datatracker.ietf.org/doc/draft-morrison-mcp-dns-discovery/>.

[IDPRONOUNS] Morrison, B., "Identity Pronouns: A Reference-Axis Extension to ~handle Identity Systems", 2026, <https://datatracker.ietf.org/doc/draft-morrison-identity-pronouns/>.

[IDCOMMITS] Morrison, B., "Identity-Attributed Git Commits via Tier-Structured Trailers", 2026, <https://datatracker.ietf.org/doc/draft-morrison-identity-attributed-commits/>.

[SUBSTRATE] Morrison, B., "Substrate-Observation as an Alternative to Envelope Coordination for Concurrent Sessions", 2026, <https://datatracker.ietf.org/doc/draft-morrison-substrate-observation/>.

[IDACCORD] Morrison, B., "Identity Accord Protocol", 2026, <https://datatracker.ietf.org/doc/draft-morrison-identity-accord/>.

[MCP] Agentic AI Foundation, "Model Context Protocol Specification", 2026, <https://modelcontextprotocol.io>.

17.2. Informative References

[RFC8499] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS Terminology", RFC 8499, DOI 10.17487/RFC8499, January 2019, <https://www.rfc-editor.org/info/rfc8499>.

[RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", BCP 205, RFC 7942, DOI 10.17487/RFC7942, July 2016, <https://www.rfc-editor.org/info/rfc7942>.

[RFC8441] McManus, P., "Bootstrapping WebSockets with HTTP/2", RFC 8441, DOI 10.17487/RFC8441, September 2018, <https://www.rfc-editor.org/info/rfc8441>.

Acknowledgements

This memo grew out of internal architectural work on the question of how an agent runtime, bound to a principal at instantiation, should receive the corpus of policy artefacts a real organisation supplies a new contractor on commencement of an engagement. The realisation that the corpus is structurally co-located with the identity that names the principal as a member, and that the prevailing architectural separation between governance plane and identity plane is itself the failure mode, is the load-bearing insight behind this specification.

Author’s Address

Blake Morrison Alter Meridian Pty Ltd Email: blake@truealter.com

Author’s Address

Blake Morrison
Alter Meridian Pty Ltd
Email: blake@truealter.com