

Closer Protocol Evidence Governed Continuation

Version 4 — research and engineering review draft

Proposed architecture and evaluation protocol. Selected mechanisms have reported local pilot evidence; the integrated architecture, comparative benefit, and commercial value remain unvalidated. Intended for researchers, engineers, architects, and practitioners. All revised wording awaits human review; the contribution record [SR3] follows the references.

Abstract

An agent can resume a process without knowing whether its next action remains justified. Closer Protocol retains an obligation [SR4] across episodes, connects new evidence to that obligation, and checks proposed action against current user [SR1] authority. Message-addressable actors [SR6] perform bounded work using declared capabilities. Separately attributable inquiry assesses evidence and constraints; a reference monitor [SR13] mediates effects; authoritative readback [SR14] tests outcomes. Semantic adjudication [SR12] determines which conclusions become accepted and whether an obligation closes. Open obligations retain conditions for reentry [SR15] while the system works elsewhere. User-owned judgments arrive in a briefcase [SR17].

The protocol composes established actor, durable-execution, truth-maintenance, provenance, security, and deliberation concepts. Its research question is whether this composition improves justified continuation and reduces user effort compared with simpler, equally resourced workflows. Three-role superiority, transferable learning, and economic return are separate hypotheses. This document defines the proposed boundaries, payloads, failure behavior, and tests; it does not establish novelty or production assurance.

1 Problem and scope

A pricing page can be genuine yet fail to establish switching costs. A model can compare options correctly yet lack the user's [SR1] preference. A process can load a checkpoint without using its contents. Closer Protocol distinguishes these failures through an obligation [SR4]: a retained question or requirement with an origin, scope, resolution conditions, and authority boundary.

The target is consequential work, where an incorrect conclusion or effect can materially change resources, privacy, commitments, operations, or later decisions. Consequence follows the operation's effects and deployment policy, not a model-selected label. Reading a page and changing an account remain distinct operations.

Research question: When something changes, can the system identify what needs reconsideration and select the next justified step under current governing state?

1.1 Binding events to obligations

The matcher authenticates and deduplicates an event, then compares its entity, scope, change kind, and evidence references with accepted obligation [SR4] dependencies and registered reentry [SR15] predicates. It records the matched obligation IDs, rule version, evaluated state version, and reasons for match, non-match, or uncertainty. Only satisfied predicates make work eligible; a match does not authorize its eventual effect. Ambiguous matches create bounded classification work or an explicit trigger gap. An unmatched event remains recorded and does not manufacture a new duty: creating an obligation requires a declaration or accepted requirement and semantic admission. Relevant invalidations enter the urgent restrictive path (§7.2), not the ordinary research queue.

1.2 Why separate these responsibilities

Separation	Failure it prevents
Observation / inference	Treating a real citation as support for any conclusion
<u>Capability</u> [SR8] / authority	Treating ability or confidence as permission
Execution / <u>readback</u> [SR14]	Treating a successful call as its intended effect
<u>Episode</u> [SR5] stop / <u>semantic closure</u> [SR18]	Losing unfinished work when an <u>actor</u> [SR6] finishes
Event history / accepted state	Treating a transcript as current knowledge

These are requirements of this proposal, not proof that every system needs identical services. A fixed worker and deterministic validator may suffice. Three participants and dynamic coalitions are optional configurations, not mandatory overhead. The proposed contribution is the integration; agreement, persistence, and catalog size are not evidence of truth, learning, or value.

2 Terminology

Underlined terms carry stable semantic references **SR1–SR23**. These are internal definitions, distinct from bibliographic references **R1–R23**. Established foundations are identified below; the remaining definitions specify this proposal.

ID	Term	Meaning
SR1	<u>User</u>	Person or authorized organization whose goals and authority govern the deployment
SR2	<u>Governing semantic state</u>	Versioned accepted goals, constraints, grants, scoped knowledge, and obligation status; facts and normative rules remain distinct

ID	Term	Meaning
SR3	<u>Record</u>	Persistent events, state versions, evidence, proposals, decisions, and receipts; persistence does not establish truth [R4]
SR4	<u>Obligation</u>	Retained requirement or question with origin, scope, resolution conditions, and reconsideration conditions
SR5	<u>Episode</u>	Bounded work on named obligations
SR6	<u>Actor</u>	Message-addressable computational entity with behavior and local state; not a claim of personality or consciousness [R1, R12]
SR7	<u>Role</u>	Assessment responsibility, independent of implementing model or process
SR8	<u>Capability</u>	Declared ability with input/output contracts, permissions, and operating requirements
SR9	<u>Cognitive workflow</u>	Reusable reasoning or coordination procedure; an open catalog category
SR10	<u>Skill</u>	Instructions or executable package implementing a capability. SKILL.md is a Markdown instruction-file convention used by some agent products, not a universal computer-science primitive or authority grant.
SR11	<u>Delta</u>	Proposed or committed change relative to an identified state version
SR12	<u>Adjudication</u>	Evidence and permission assessment under an explicit rule
SR13	<u>Reference monitor</u>	Protected mechanism mediating an operation at the action boundary [R7]
SR14	<u>Readback</u>	Independent observation of authoritative target state after an attempt
SR15	<u>Reentry</u>	Reconsideration or resumed work when a retained obligation's eligible condition holds
SR16	<u>Handoff</u>	Accepted transfer of bounded responsibility and adequate context
SR17	<u>Briefcase</u>	User portfolio of Leads, Investigations, Briefs, and The Bench; the latter holds user judgments
SR18	<u>Semantic closure</u>	Committed, adjudicated resolution, supersession, or legitimate withdrawal of an obligation

ID	Term	Meaning
SR19	<u>Selective disclosure</u>	Purpose- and authority-bounded projection of the record for a recipient; a digest does not establish completeness
SR20	<u>Confession</u>	Attributable discrepancy or stop record: attempt, omissions, limits, cost, consequence, and reconsideration conditions; not an inference about intent
SR21	<u>Participant record</u>	Attributable inputs, assessments, and revisions held for one participant
SR22	<u>Shared case record</u>	Append-only disclosed positions, challenges, evidence, and commitments for an episode
SR23	<u>Structured evidence inquiry</u>	Bounded dialogue that tests scoped claims, supplies evidence, and revises positions before a declared decision rule [R17]

3 Prior art and reuse

3.1 Actors and durable execution

Actors [SR6] send messages, create actors, and specify subsequent behavior [R1]. Actor research also addresses persistence, sponsorship, and provenance [R12]; these are not missing inventions supplied here. Closer Protocol assigns responsibilities to actors without claiming the remainder, investigator, and regulator are actor primitives. A model call needs runtime-supplied identity, mailbox, recovery, and state lifetime to serve as a durable actor.

LangGraph persistence and CrewAI Flows are reuse candidates for stateful execution [R2–R3]. Event sourcing preserves transitions and derives current projections [R16]; W3C PROV describes entities, activities, and agents [R4]. These can carry the protocol. They do not establish that inherited knowledge actually influenced a fresh episode [SR5]. That requires a causal test (§10).

3.2 Intention and truth maintenance

Bratman's account of intentions as plans and Rao and Georgeff's belief–desire–intention architecture provide foundational distinctions between beliefs, objectives, and commitments to action [R22–R23]. Jason supplies a concrete suspension mechanism [R5]. Truth-maintenance systems track dependencies from assumptions to conclusions [R6]. Closer Protocol combines retained obligations [SR4] with scoped evidence and invalidation dependencies. Consistency relative to represented premises is not independent truth about the world.

3.3 Security organizations and interoperability

A reference monitor [SR13] checks effects using a trusted operation registry [R7]. Calling an action “research” cannot authorize a purchase. Evidence may revise a claim; it cannot enlarge a spending grant.

Adaptive organizations and coalition formation have prior art [R9]. Fission–fusion is a useful optional analogy for changing membership [R8], not a uniquely necessary substrate. MCP and A2A offer interoperability [R10–R11]; ownership, contract versions, expiry, and endpoint verification remain deployment responsibilities. A responding endpoint is not necessarily the implementation previously approved.

3.4 Memory and method catalogs

Recent *MemTX* and *Beyond Memory* preprints describe closely related transactional beliefs and continuity controls [R13–R14]. Their comparison here remains abstract-level; implementations and reported results have not been independently validated. Full related-work examination is required before novelty claims.

A cognitive workflow [SR9] catalog can link, for example, constraint solving, search, state estimation, planning, and structured inquiry to usable skills [SR10]. This is an extensible many-to-many index, not a numbered universal taxonomy. One skill may combine several methods. A missing match establishes a catalog or capability [SR8] gap, not a new mathematical category.

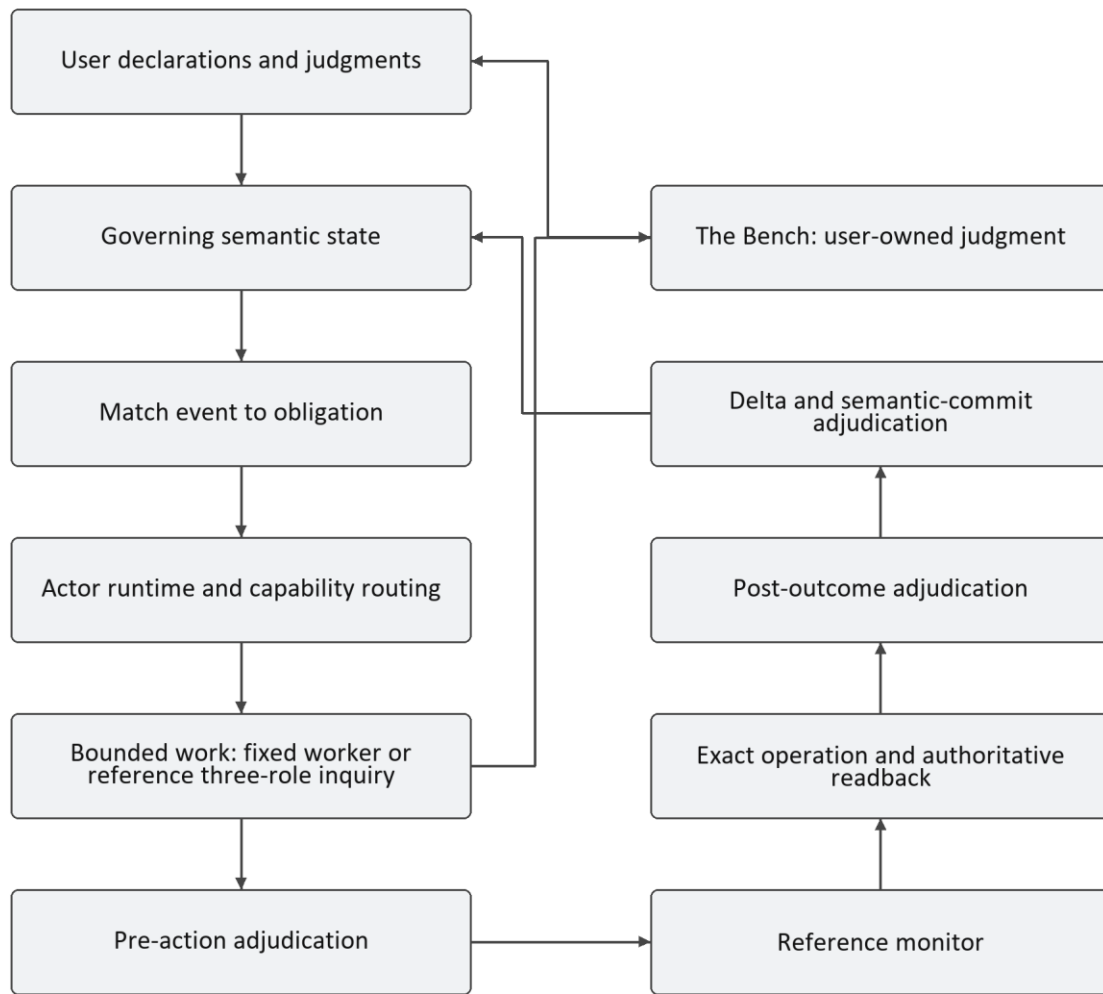
3.5 Inquiry is not majority truth

Prior art	Relevant distinction
Dialogue models [R17, R21]	Private information, public commitments, challenges, and retractions support attributable inquiry
Delphi [R18]	Independent positions and controlled feedback are useful; three named roles are not an anonymous Delphi panel
Blackboard systems [R19]	Multiple knowledge sources share hypotheses; the workspace alone supplies no authority rule
Raft [R20]	Agreement about a replicated log does not prove the log's factual claims

Closer Protocol adapts structured dialogue for collaborative inquiry. Tests must include both a correct minority and correlated false agreement (§10), rather than assume three records establish correctness.

4 Architecture and trust boundaries

Figure 1. Lifecycle: work returns evidence to accepted state; three-role inquiry is a reference configuration.

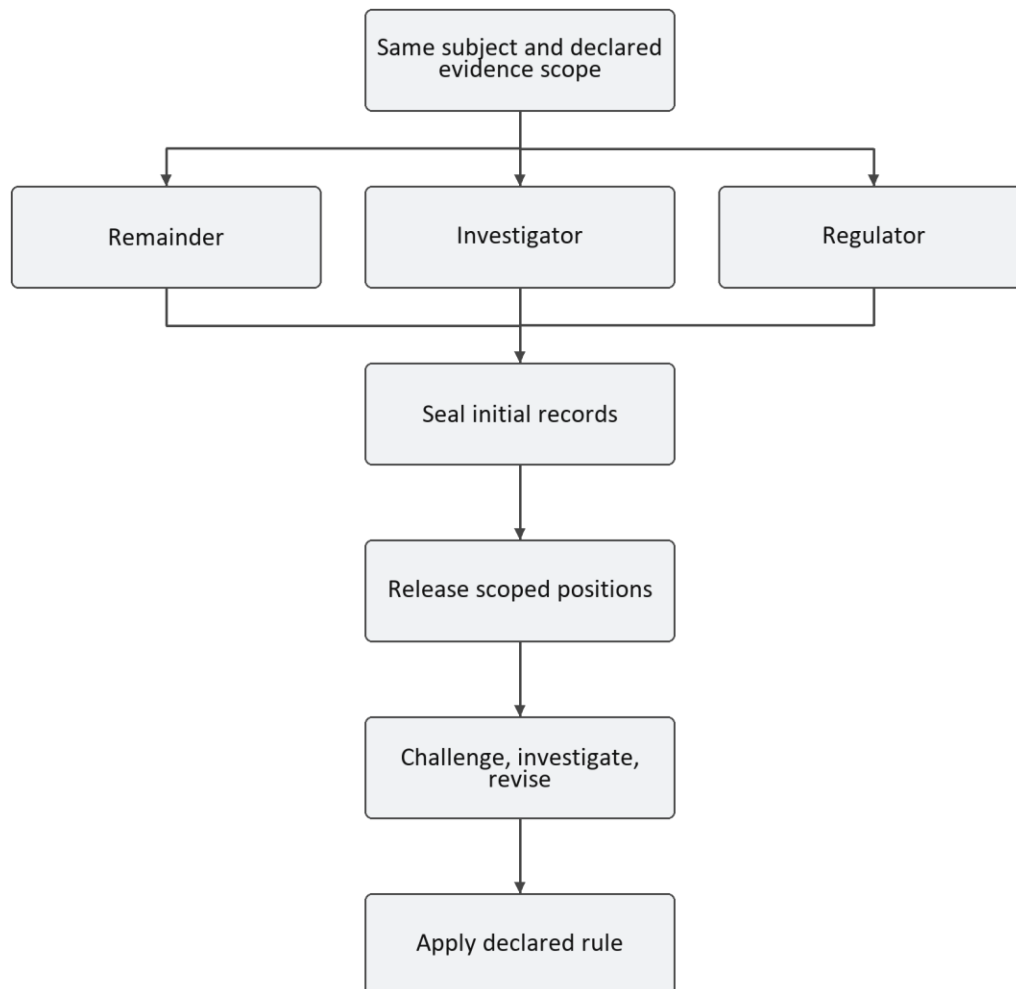


Boxes identify responsibilities, not necessarily separate services. The reference configuration separates participant contexts and uses transactional storage with a narrow dispatcher. Participants cannot redefine that dispatcher's permissions.

4.1 Three participants their records and the conference

The **remainder** explores alternatives and omissions; the **investigator** tests evidence and inference; the **regulator** checks goals, constraints, and authority. All perform work and may share or specialize skills. Their participant records [SR21] preserve disclosed inputs, assessments, and revisions. The shared case record [SR22] contains only the positions and evidence actually released to the group. Neither requires private model chain-of-thought.

Figure 2. Initial positions are sealed before release; challenges bind an exact proposition or action.



Independence is an infrastructure obligation. Before release, access controls prevent participants from reading other initial positions; the coordinator records the snapshot, participant set, and sealed receipt digests. On timeout, seal the available set and record [SR3] absence. A late participant exposed to released positions is labeled informed, not independent. A leak is a recorded protocol violation: suspend the independence claim, preserve exposure evidence, and restart with unexposed contexts if the experiment requires independence. Rebase after a state change preserves history, names the new basis, and records prior exposure; it cannot restore statistical independence by renaming the round.

The work contract declares rounds, calls, time, subject IDs, evidence scope, and allowed moves: assert, support, challenge, narrow, retract, request evidence, retain with reason, assent, and defer. Every move binds actor [SR6], role [SR7], local receipt, shared snapshot, exact subject digest, governing basis, reply reference, and evidence. A materially new subject starts new bounded work. A reconciler cannot invent another participant's assent.

A missing authority grant cannot be outvoted. Factual objections are resolved under a declared sufficiency rule, not by counting voices. Two differently worded descriptions are not dissent unless they concern the same proposition or exact action. Agreement that support is insufficient is a valid result;

stopping the conference does not close the obligation [SR4]. Record models, exposures, tools, and ordering because role names do not establish independent error.

4.2 Three adjudication moments

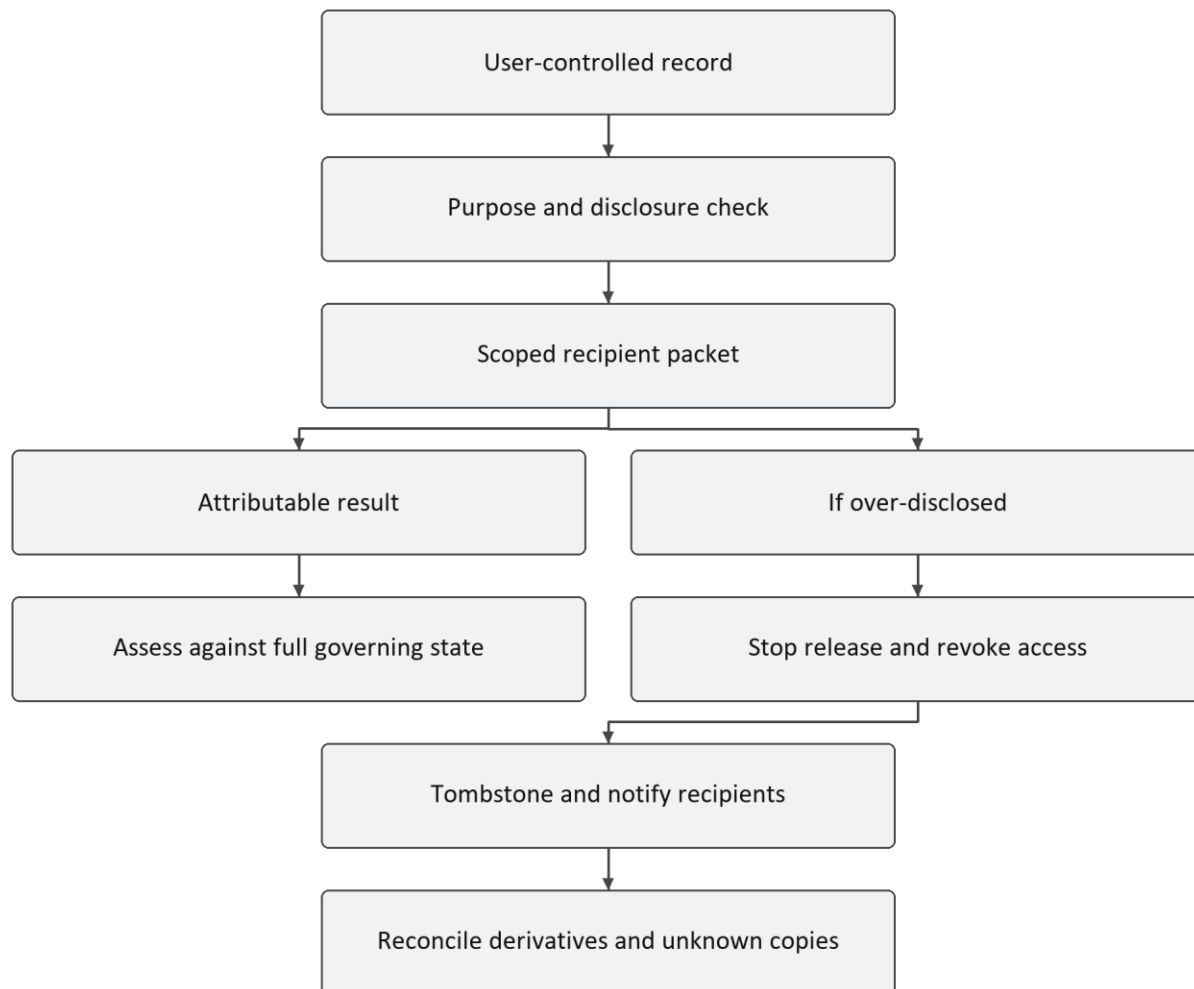
Moment	Question	Bound subject
Pre-action	Is this proposed effect justified and permitted?	Exact action and current evidence
Post-outcome	What did the attempted operation establish?	Authoritative <u>readback</u> [SR14]
Semantic commit	Which proposed changes may become accepted state?	Immutable <u>delta</u> [SR11] digest

Dispatch still checks current authority, target, parameters, ownership, versions, and budget. Earlier assent cannot authorize a changed action or endorse a new claim after execution. The urgent restrictive exception (§7.2) has its own mandatory receipt and later review; it does not collapse these subjects.

4.3 Selective disclosure and remediation

Selective disclosure [SR19] begins with the user's [SR1] governing record [SR3]. A trusted projection releases the purpose-relevant subset and declares policy coverage, withheld categories, recipient, and missing context. An investigator may need a feature requirement without account history; a regulator needs applicable authority boundaries. Both know the limits of their view. Missing coverage is a gap, not permission.

Figure 3. Disclosure is authorized outward; a breach requires containment and downstream reconciliation.



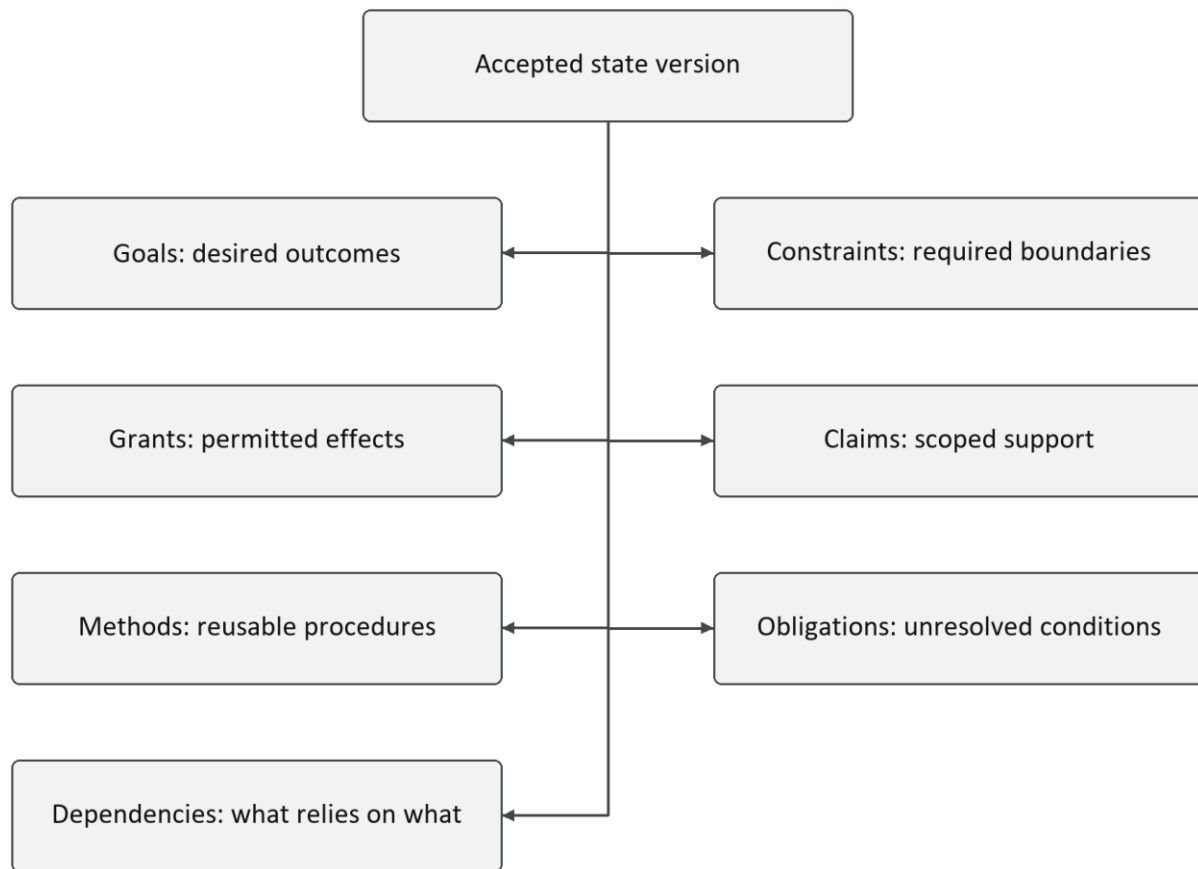
The monitor checks the authoritative state, not only the recipient's slice. Receipt possession grants no right to redistribute its contents. Hashes bind bytes, not sufficiency or completeness.

After over-disclosure: record affected items, recipients, policy version, and attempts; immediately stop further release and revoke revocable access. Issue an append-only withdrawal/tombstone and authenticated downstream notice under the deployment's incident policy. Trace dependent artifacts, suspend affected use, request deletion or quarantine where supported, and record acknowledgments, failures, and unknown copies. Report the incident and unresolved exposure to the authorized user/incident owner. Deletion of already received data cannot be guaranteed; a tombstone does not undo a disclosure. Notification content must itself remain scoped. Incident reconciliation is mandatory and retains historical evidence under the custody policy.

5 Accepted state and invariants

Governing semantic state [SR2] answers seven practical questions. Raw observations and historical assertions remain in the record [SR3], separately from what is currently accepted.

Figure 4. Accepted state separates mission, authority, knowledge, and unfinished work.



A delta [SR11] proposes changes against a named parent. Semantic admission is atomic: stale parents require rejection or reassessment. Evidence can revise a descriptive claim without revising a normative grant. Accepted methods retain preconditions, success/failure criteria, exceptions, and dependencies.

Obligations [SR4] may be active, awaiting an event, awaiting a user [SR1] judgment, resolved, superseded, or withdrawn. Only semantic adjudication [SR12] and commit establish the last three. An episode's [SR5] budget stop does not.

Invariants

1. Accepted conclusions identify assessed evidence and applicability conditions.
2. Policy and grant changes require authority independently of factual support.
3. Assessment, authorization, dispatch, and readback [SR14] bind the same logical action.
4. A successful call alone does not establish its intended effect.
5. Reentry [SR15] uses current relevant state, including invalidated premises.
6. Semantic closure [SR18] has a committed adjudication explaining the result.
7. A retained obligation has a reconsideration condition or an explicit trigger gap.
8. User silence creates no permission.

9. Budgets have one authoritative accounting path across processes.
10. Failures and unknown outcomes remain visible in evaluation accounting.

6 Payload protocol

The envelope carries lineage and governing context; a typed body carries work. References must resolve, and operative context must be materialized within disclosure bounds. Appendix C defines design-level contracts, not executable schemas.

6.1 Envelope and authentication

This illustrative envelope contains a partial body; placeholders are not valid production values.

JSON payload

```
{
  "protocol_version": "closer-draft-4",
  "payload_id": "pl-work-1",
  "type": "WORK",
  "created_at": "2026-01-01T12:00:00Z",
  "producer": {"actor_id": "coordinator-1", "role": "coordinator"},
  "parents": ["route-1"],
  "obligation_ids": ["o-12"],
  "episode_id": "ep-3",
  "semantic_basis": {"version": 14, "digest": "sha256:..."},
  "disclosure": {"id": "disclosure-3", "digest": "sha256:..."},
  "budget_ref": "budget-5",
  "causation_event_id": "ev-9",
  "body": {"work_id": "work-1", "role": "investigator"},
  "content_digest": "sha256:..."
}
```

DisclosureRef resolves to the immutable projection receipt defined in C1; it is not inline projection content. The envelope digest covers all envelope fields except itself under a documented encoding such as JCS [R15]. Operational identity is authenticated through a declared transport/session binding or signature verified against a trusted identity registry. A caller-supplied actor [SR6] name or unsigned digest is insufficient. Signatures are optional only where another specified authentication mechanism supplies the required assurance; authority remains checked separately. Key rotation, revocation, and signing scope must be specified when signatures are used.

6.2 Message map

Payload	Producer → consumer	Carries
EVENT	Observer → matcher	What changed, where, when, and with what evidence
<u>OBLIGATION</u> [SR4]	Authorized planner → semantic service/scheduler	Gap and resolution conditions

Payload	Producer → consumer	Carries
ROUTE	Router → runtime	<u>Actors</u> [SR6], capabilities, versions, disclosure, allocation
WORK	Coordinator → participant	Question, context, output contract, stop condition
RESULT	Participant → adjudicator	Claims, exact proposals, evidence relations, objections
ACTION	Action planner → <u>reference monitor</u> [SR13]	Typed operation, target, authority, postcondition
EVIDENCE	Observer/ <u>readback</u> [SR14] → store/assessor	Target-bound observation and limitations
<u>ADJUDICATION</u> [SR12]	Assessment service → planner/distiller	Phase-bound judgment and dissent
<u>DELTA</u> [SR11]	Distiller or restrictive issuer → semantic service	Immutable proposed changes
<u>REENTRY</u> [SR15]	Scheduler/adjudicator → matcher	Typed trigger and eligible work
<u>HANDOFF</u> [SR16]	Coordinator → recipient/ownership service	Responsibility, context, lease, fencing
<u>BRIEFCASE</u> [SR17]	Coordinator → <u>user</u> [SR1] decision view	Exact judgment, alternatives, uncertainty, waiting policy

6.3 Scope and evidence

An obligation [SR4] includes `closure_authority` alongside its resolution conditions (C3). Evidence is assessed **per proposition–evidence pair**, not as one global Boolean. An unrelated item neither supports nor vetoes a different conclusion.

JSON payload

```
{
  "proposition_id": "required-export-preserved",
  "evidence_id": "export-probe-3",
  "reference_status": "resolved",
  "scope_match": "applicable",
  "relation": "supports",
  "valid_at": {
    "observed_at": "2026-01-01T12:01:00Z",
    "observation_sequence_ref": "service-A/candidate/export-probes",
    "observation_version": 3,
    "target_version": "release-7"
  },
  "limitations": ["Observation does not guarantee future contractual availability"]
}
```

The counter is the monotonic sequence in the named observation stream, not an unexplained global version. If the source has no native target version, record [SR3] that as unknown rather than invent one. Relations are supports, contradicts, irrelevant, or unresolved; absent assessment differs from contradiction. Sufficiency is the declared rule over relevant relations. A participant's assertion that a reference exists is not independent verification.

6.4 Logical action versus attempt

The **logical action digest** binds a canonical whitelist: operation, exact target, typed parameters, postcondition, authority reference, relevant semantic/registry versions, and idempotency key. It excludes attempt_id, ownership_epoch, and its own digest. An attempt envelope covers them all. The ownership epoch is a dispatch fencing condition, not part of logical identity; legitimate reassignment can therefore preserve an operation key. Changed content requires fresh assessment; same-key changed-content reuse is rejected. Comparison-only normalization never changes dispatch identity.

6.5 Local custody

Local-first custody can keep the accepted record [SR3] and receipts under user [SR1] control while releasing authorized packets. It does not prove synchronized folders, backups, tools, or cloud calls preserve privacy. Local Lotor inspection identified receipt-chain/action-gating reuse potential, not a general selective-disclosure implementation: whole-chain export and a separate narrow AIP assertion prototype did not establish verified end-to-end hidden-source binding. These unpublished observations motivate adapters and tests, not a privacy assurance claim. Permission to perform an operation and permission to release its inputs remain separate checks.

7 Routine hierarchy and user attention

7.1 Ordinary episodes

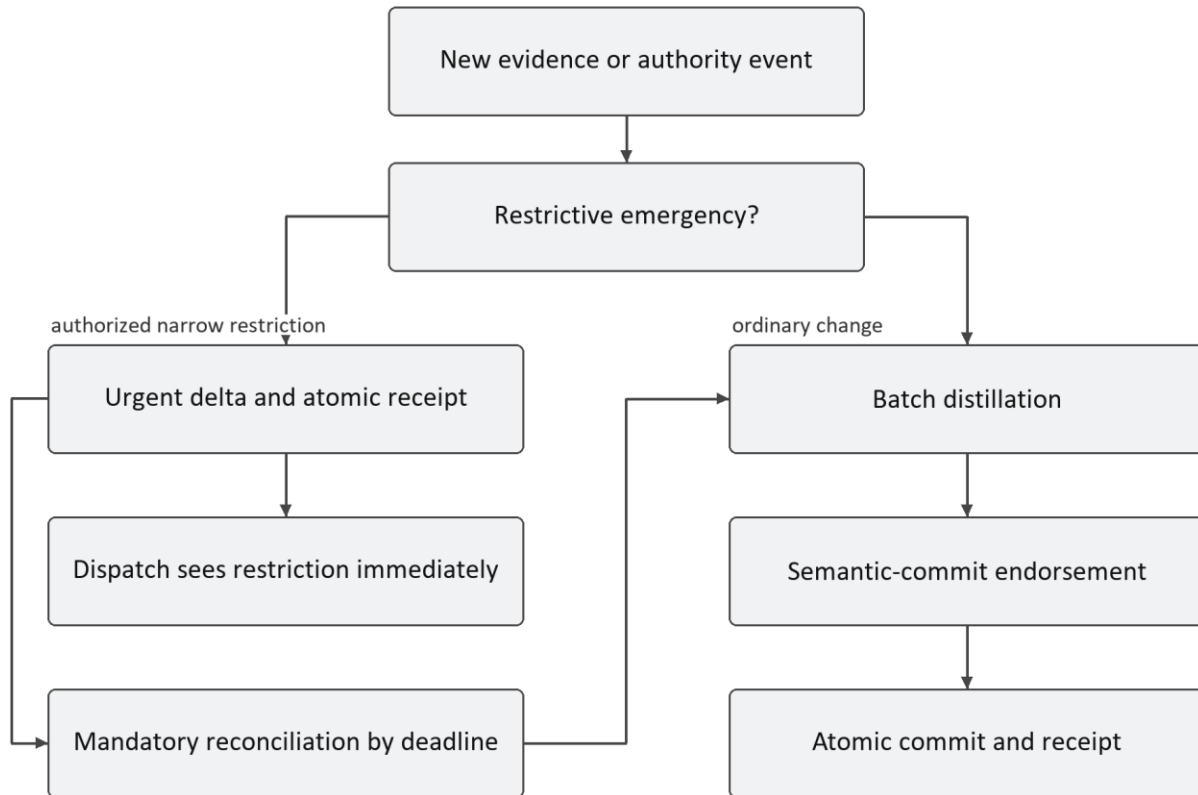
The scheduler loads current state, matches an event or due obligation [SR4], checks resources and context, then selects bounded work. Results may justify investigation, an operation proposal, a handoff

[SR16], a user [SR1] question, or a reentry [SR15] trigger. All outcomes return evidence and candidate deltas. The episode [SR5] does not close its obligation.

When one obligation cannot advance, the scheduler works on another eligible obligation. When none qualify it becomes dormant, preserving triggers rather than consuming continuous model calls. Frequency never expands authority.

7.2 Fast restrictions and slower distillation

Figure 5. Restrictions act immediately; ordinary knowledge commits require endorsement. Both leave mandatory receipts.



Ordinary distillation batches supported changes, preserves exceptions, and consolidates dependencies. It is less frequent than operational work. Safety-critical invalidation cannot wait for that cadence.

Urgent exception: a separately authorized issuer may immediately commit only a restriction—revoking a grant or suspending specified action eligibility—without prior participant endorsement. It cannot expand authority, close an obligation [SR4], promote positive knowledge, or alter its own emergency privileges. The immutable delta [SR11], authenticated issuer, restrictive authority, scope, reason/evidence, and a non-null reconciliation deadline are recorded atomically with an urgent commit receipt (C10, C14). A receipt cannot be deferred until consolidation.

Dispatch observes the restrictive version before further admission; no cooldown or novelty threshold may delay it. Post-hoc justification and review are **mandatory**. A deployment must set finite deadlines, bounded issuer scope, and per-issuer/target outstanding-case limits. Repeated identical restrictions coalesce without resetting the deadline. Exceeding a limit or missing reconciliation triggers an

independent incident owner/Bench escalation and suspension of that issuer's ability to create further broad restrictions, under predeclared policy. Existing restrictions remain fail-closed; fresh affected work goes through the independent restrictive authority. Lifting a restriction requires authorized reassessment, never timer expiry. This bounds an abuse surface without letting ordinary scheduling delay a valid revocation.

7.3 Briefcase and timer behavior

The briefcase [SR17] contains **Leads** (possibilities), **Investigations** (casework), **Briefs** (formed assessments), and **The Bench** (user [SR1] judgments). These are views, not a compulsory pipeline. An urgent incomplete decision can reach The Bench early, with its gap and consequence stated.

Condition	Required transition
Research can resolve the gap within grant and budget	Continue investigation; do not manufacture a user question
Preference, new authority, or material tradeoff belongs to user	Bench packet; preserve uncertainty and alternatives
Evidence changes	Refresh or visibly withdraw the stale packet; retain linked <u>obligations</u> [SR4]
Bench deadline passes without answer	Mark overdue; apply only an already-authorized waiting/safety policy; issue one deduplicated escalation to designated owner; retain or replace <u>reentry</u> [SR15] condition
Reentry trigger expires while obligation remains open	Disable that trigger and atomically register a replacement or explicit trigger gap with responsible owner and bounded review condition

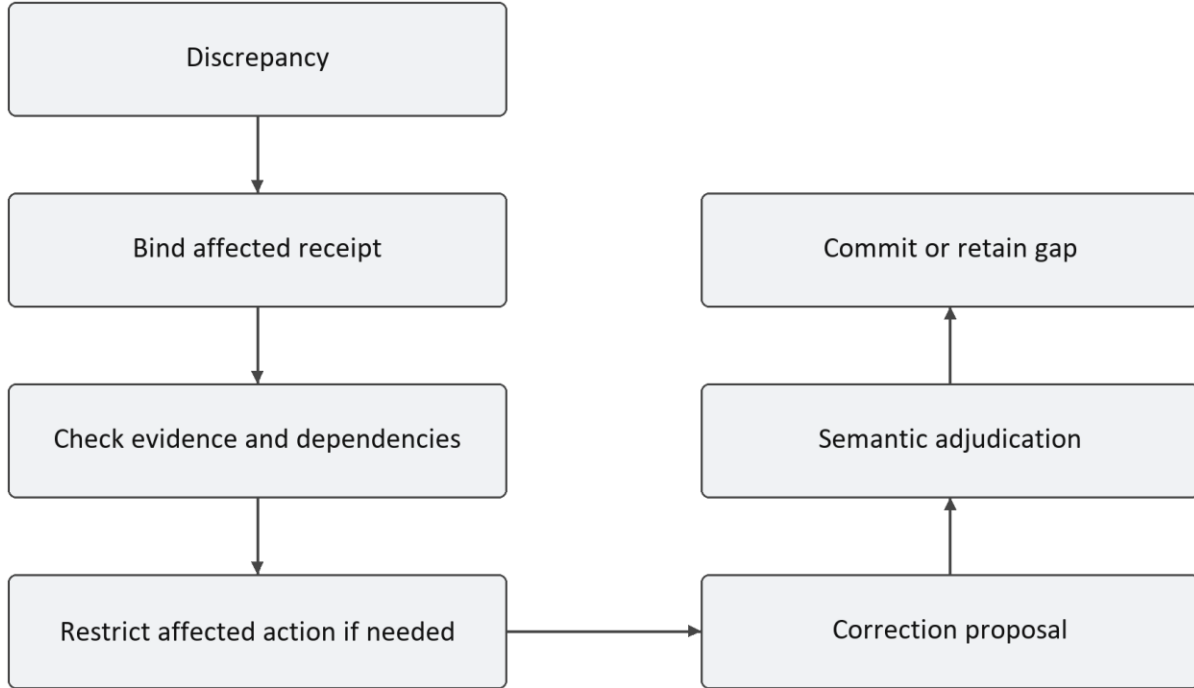
Expiry never implies approval or closure. If a safe waiting policy is unavailable, keep affected effects blocked and surface the policy gap. Related questions may share one card while preserving every obligation. Notification rate limits may group reminders but cannot erase a new material deadline or its recorded consequence.

8 Runtime correctness and failure behavior

8.1 Confessions discrepancy to correction

A confession [SR20] names the exact assertion or attempt that the record [SR3] cannot support. Example: “I reported completion after checking only the tool response; the target has not been read back.” It records affected obligation [SR4], prior receipt, omitted check or deviation, evidence, known/unknown effects, cost, current status, and reentry [SR15] condition. Historical assertions remain visible; the confession does not prove its own explanation.

Figure 6. A confession identifies a discrepancy; checked evidence determines the correction.



Automated readback [SR14], artifact comparison, and independent review are required complements: the system cannot rely on voluntary confession. Disclosures follow §4.3. Concurrent proposals bind their actual state versions; an atomic parent check rejects stale application. Across a local store and remote service, full atomicity may be unavailable. Record durable intent, use conditional/idempotent operations where supported, and reconcile the actual target. Receipts expose the residual interval between local authorization and remote effect.

8.2 Retry ownership and unknown effects

At-least-once delivery requires deduplication. A timeout can conceal a completed remote effect, so the status remains `outcome_unknown` until authoritative reconciliation. Retry requires a repeat-safe operation contract and available grant. One stable key identifies one logical operation; attempts have distinct IDs. Ownership transfer changes the dispatch epoch, not the logical digest (§6.4/C7). Fence the former owner, reconcile any in-flight effect, then permit a new owner to resume. A stale owner cannot dispatch under an expired epoch. Neither lease expiry nor a human acknowledgment authorizes repetition.

8.3 Budgets and process control

Reserve calls, tokens, cost, time, and concurrency through one process-safe ledger. Before admission reserve bounded worst-case exposure, including applicable provider cancellation limits. Completion reconciles observed usage; failed, truncated, timed-out, and uncertain dispatches remain counted. A reconstructed row keeps provenance and is not a refund.

Mid-flight ceiling: request cancellation at the provider boundary when supported, halt new dispatches, and retain the reservation until final usage or conservative reconciliation. A local timeout is not proof of

remote cancellation. If a provider cannot enforce the required hard ceiling, do not admit that route under a hard-cap contract. Any overshoot becomes an explicit budget incident, not silently absorbed success.

No speculative overbooking by default. Reserve an entire next executable unit atomically; never hold partial multi-resource reservations while waiting for the rest. Queue fairly, reduce parallelism, or choose an already-authorized cheaper unit when capacity is insufficient. Expire only undispached reservations; uncertain in-flight exposure remains reserved. Any alternative overbooking policy must state maximum exposure and user-authorized risk. This trades throughput for an auditable cap while preventing hold-and-wait deadlock.

Terminate only recorded task-owned processes and descendants. Live state and ledgers stay outside disposable test directories. Cleanup cannot remove the evidence used to judge the run.

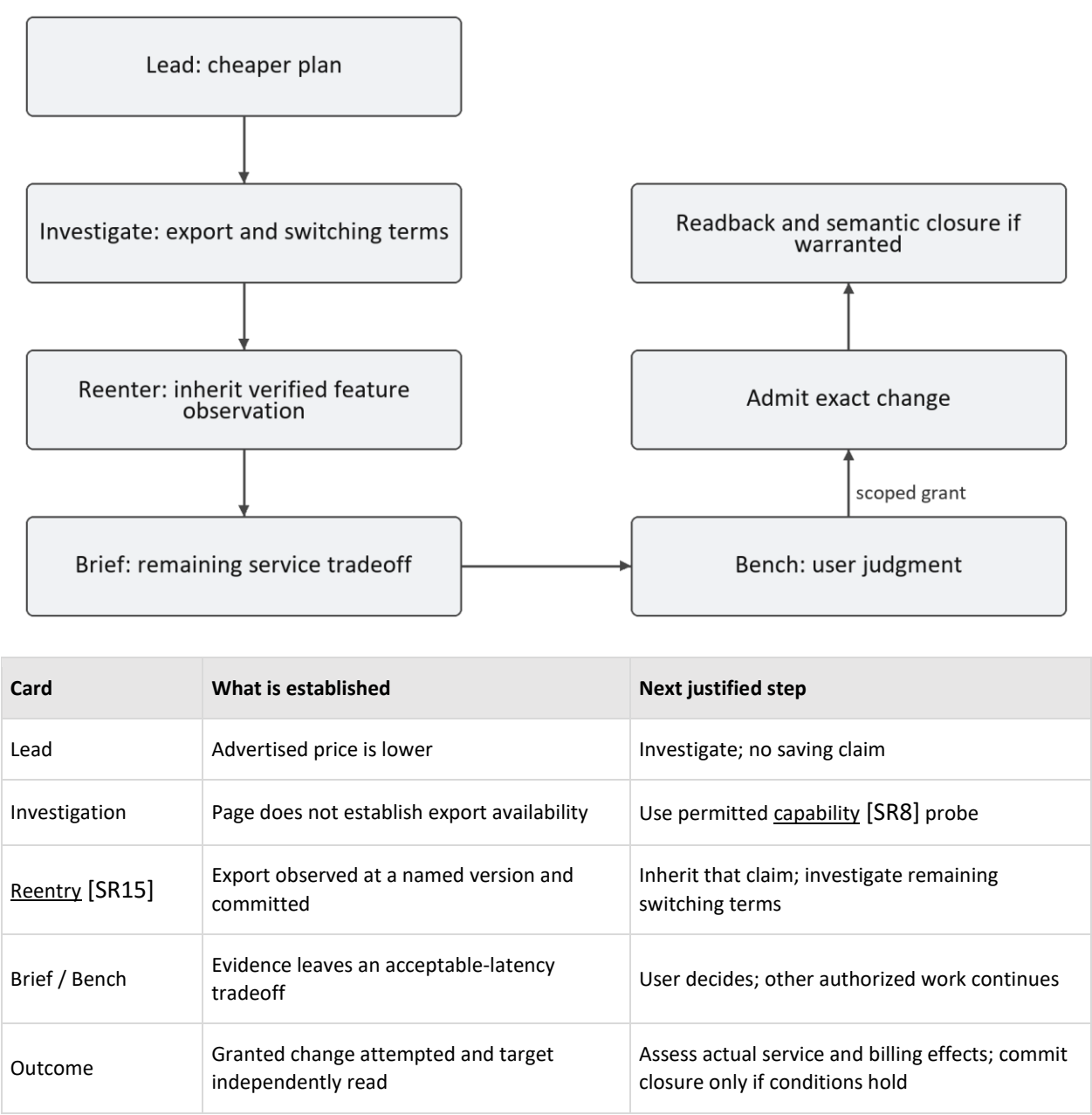
8.4 Scheduling without churn

Use explicit bounded fan-out, deduplication, and per-obligation budgets. Document thresholds and cooldowns; hidden novelty filters can suppress required reentry [SR15]. Deadline, consequence, dependency impact, information gain, and user [SR1] effort can guide priority, but their estimates are not realized value. Repeated work without new qualifying evidence becomes dormant or a capability [SR8] gap. Coalition dissolution cannot orphan ownership. Urgent restrictions remain outside ordinary cooldowns.

9 Worked example a cost candidate

The user [SR1] permits research but no account change. The mission is lower cost while preserving a required export feature.

Figure 7. The case becomes actionable as evidence and user authority arrive.

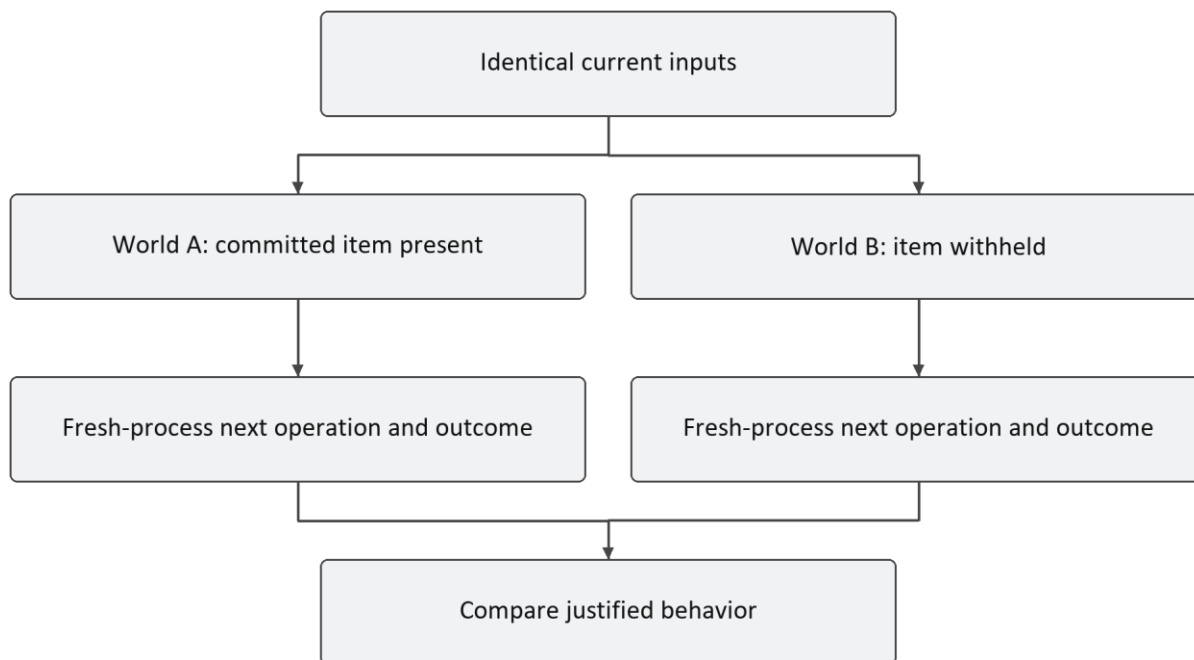


An unknown effect remains reconciliation work. A discovered possibility, reduced uncertainty, refused inference, and realized saving are distinct entries; activity cannot substitute for income.

10 Evaluation

10.1 Mechanism and comparison

Figure 8. The causal test varies inherited state while keeping current inputs identical.



First test whether committed state causally changes fresh-episode behavior. Compare materialized packets before dispatch, varying only the relevant inherited item. If promotion failed, nothing exists to withhold: count the pair as invalid, not a negative causal result.

Then compare separate roles with one-agent procedure. Match model access, evidence, tools, and adjudication [SR12] rule. Declare a **resource vector per arm**: maximum calls, total input/output tokens (including reasoning and cache treatment), metered dollars, elapsed wall time, and concurrency. Report actual consumption in each currency, including coordination. For example, a six-call conference budget is compared with six solo calls, not one. If fixed dollar and token constraints cannot both be equated, identify the primary budget and report the unmatched dimensions; do not call the comparison equally resourced without qualification. Unknown coordinator usage remains unknown, not free.

Report all enrolled pairs, completed valid pairs, invalid pairs by cause, and task/commit/contract failures. Publish both the valid-pair effect estimate and end-to-end completion/failure rate over all enrolled cases. Invalid-pair exclusion must not conceal commit failure. Attribute a shared finalizer's protection to the finalizer. Coalition adaptation, catalog quality, and specialized skills are later separately varied factors.

10.2 Required cases

Family	Failure or success the test must expose
Supported continuation	Useful permitted action actually occurs
Relevant but insufficient evidence	Gap retained without invention
Real citation, wrong inference	Unsupported promotion refused
Contradicted inherited premise	Dependent belief/action reconsidered
Changed grant before dispatch	Stale authority refused
Owner-only tradeoff	<u>User</u> [SR1] choice preserved; sealed answer not guessed
Irrelevant event	No unnecessary activation
Expired registry entry	Stale <u>capability</u> [SR8] not silently invoked
Unknown external outcome	Reconciliation precedes repetition
Reopened closed <u>obligation</u> [SR4]	Explicit semantic transition preserves history
Correct minority	Supported objection survives majority pressure
Correlated false agreement	Three agreeing false claims fail independent oracle/evidence checks

Use trustworthy held-out ground truth for the final two; agreement cannot be its own oracle. An all-refusal suite cannot establish continuation, and an all-success suite cannot establish restraint. Timer expiry, disclosure incidents, urgent flooding, and ownership reassignment also require deterministic fault cases.

10.3 Measures and discipline

Measure correct operation/target, unsupported promotions, invalidation misses, attempted versus actual unauthorized effects, verified postconditions, duplicates, unnecessary probes, and retained obligations [SR4]. Score returned questions for necessity, specificity, evidence, alternatives, consequence clarity, and preserved choice.

User effort: record [SR3] review minutes, decision touches, repeated explanations, corrective interventions, and decision-packet reading load per enrolled and resolved obligation. Separate necessary owner judgments from avoidable repair work. Assess quality and missed deadlines alongside effort, so silence or hidden assumptions cannot count as improvement. Longitudinal comparison should include onboarding and review costs, not only later low-effort episodes.

Freeze fixtures, prompts, schemas, model controls, scoring, budgets, and comparison mappings before live dispatch. Test legitimate maximum shapes and meaningful mutations offline. Post-freeze repairs create a separately reported revision; preserve failures and original artifacts. A time cap limits exposure, not a target to fill.

10.4 Present evidence boundary

Local reports describe action-identity fidelity and a valid negative inheritance contrast, alongside earlier comparator, integrity, context-loading, and truncation defects. Later interface failures also showed that a structured-output request did not guarantee the requested cardinality. These reports motivate the design; they are not independently replicated benchmarks. Integrated benefit, durable learning, three-role advantage, and economic return remain open. Publication of empirical claims requires auditable artifacts and separation of corrected interpretations from observations.

11 Value hypothesis

The proposed value is a useful next action or question the user [SR1] would otherwise miss, with less repeated coordination. Revenue, verified avoided cost, preserved service, shorter cycle time, and reduced explanation effort are different outcomes. Store candidate estimates, assumptions, measured intermediates, realized effects, and operating cost separately. Unused subscription capacity is not lost cash; projected savings require billing and service verification.

Offering hypothesis	Test against a simpler alternative
Runtime adapters and conformance	Fewer lost/misbound transitions at lower integration cost
Validated domain routing procedures	Better scoped reuse and next-step accuracy than repeated search
Managed <u>obligations</u> [SR4] and <u>briefcase</u> [SR17]	Lower user effort without silent assumptions
Assurance/ <u>readback</u> [SR14] tooling	Fewer unsupported promotions and unreconciled effects

A “make money” goal grants no trading, outreach, purchase, or disclosure authority. The same structure can support research claims, incident recovery, and learning prerequisites, each with domain-specific evidence standards. No proprietary catalog is required. Commercial differentiation must exceed the cost of setup, maintenance, runtime, and reviewing the system's own records.

12 Limitations and threat model

The system can enforce only represented, applicable constraints. Models can share errors, misuse valid sources, or satisfy syntax while making invalid inferences. Distillation may erase exceptions; catalogs may become stale; dynamic groups add cost. Human review can become habitual approval. Signatures authenticate origin, not truth. None of these limits is resolved merely by adding receipts.

Adversary or failure	Required defense and residual limit
Compromised participant	Treat proposals as untrusted; authenticate producer, validate schema, restrict tools/disclosure; forged reasoning can still fool assessors
Colluding roles or common model error	Independent evidence/oracles and correlated-error tests; <u>role</u> [SR7] separation alone is insufficient
Malicious/replaced registry endpoint	Authenticate identity, pin contract/version, verify target, expire trust; health checks alone prove little
Urgent-commit flooding	Narrow issuer authority, deduplication, fixed reconciliation deadlines, independent escalation and issuer suspension (§7.2)
<u>Briefcase</u> [SR17] flooding	Group by <u>user</u> [SR1] judgment, bound producer rate, preserve material urgency and all linked <u>obligations</u> [SR4]; notification suppression is not resolution
<u>Record</u> [SR3] or dispatcher compromise	Protected authentication, storage, and action boundary are trust assumptions; a compromised monitor is outside the guarantees it enforces
Over-disclosure	Stop further release and reconcile downstream (§4.3); already copied data may be irrecoverable

Next studies must demonstrate warranted continuation and restraint in fresh processes, then test role specialization and longitudinal user effort. Equivalent simpler implementations remain valid competitors. This paper defines a testable composition, not a proof that this architecture is uniquely required.

Appendix A Deployment decisions

Publish a versioned deployment profile answering these questions before consequential use:

1. How are user [SR1], issuer, producer, endpoint, and registry identities authenticated and revoked? Which unsigned channels provide sufficient assurance?
2. Who can alter goals, policy, grants, and emergency restrictions; how are independent incident owners appointed?
3. Which event–obligation [SR4] rules and evidence predicates activate work, and how are ambiguous matches retained?
4. Which authoritative observations establish each postcondition, including an unknown remote effect?
5. How are state commits, ownership fences, stable operation keys, and reassignment synchronized?
6. What data may each recipient receive; how are over-disclosure, withdrawal, downstream notification, and unknown copies reconciled?
7. What happens at every lease, Bench, reentry [SR15], budget, and urgent-review deadline?

8. Which work remains permitted while a user judgment waits, and what triggers dormancy?
9. How are whole-unit reservations, cancellation limits, and any overbooking exposure bounded?
10. Which evidence would falsify the deployment's claimed decision or user-effort benefit?

Appendix B Validation boundary

These are **design-level contracts**, not production JSON Schema. ID is an immutable resolvable identifier; Digest names its algorithm; Timestamp is UTC. A generic immutable-artifact reference contains an ID and digest; named references follow their declared type. StateRef carries version and digest, while actor [SR6]/source identity references resolve through an authenticated registry binding. Core fields below are required unless nullable. Unknown core versions fail closed. Extensions cannot override core meaning. Bound strings/arrays and paginate with completeness metadata rather than silently dropping legitimate candidates.

Schema validation, reference resolution, evidence support, authority, and readback [SR14] are separate checks. Report each. Test maximum legitimate shape, empty valid collections, unknown IDs, wrong enums, stale versions, scope-irrelevant evidence, and operation-specific parameters. Free-text equality cannot establish action identity. Experimental normalization must remain outside dispatch hashing.

Appendix C Payload contracts

C1 Envelope and shared references

```
Envelope = protocol_version, payload_id, type, created_at, producer:ActorRef,
  parents:List<ID>, obligation_ids:List<ID>, episode_id:ID|null,
  semantic_basis:StateRef|null, disclosure:DisclosureRef,
  budget_ref:ID|null, causation_event_id:ID|null, body:TypedBody,
  content_digest:Digest
StateRef = version, digest
DisclosureRef = id, digest
DisclosureReceipt = purpose, recipient, policy_basis:StateRef,
  materialized_item_refs, withheld_categories, coverage_receipt_ref,
  missing_required_context, permitted_use, issued_at
```

An intake EVENT may have no obligation [SR4], episode [SR5], or governing basis before matching; work may not. Missing lineage is explicit, never invented. Producer authentication is mandatory (§6.1); each receipt's release is separately governed. Integrity digests cover complete objects except their own digest field.

C2 Event

```
EVENT = event_id, source:SourceRef, observed_at, received_at, entity_ref,
  change_kind, evidence_refs, deduplication_key,
  certainty:observed|reported|unknown
MATCH_RECEIPT = event_ref, state_basis, rule_ref, matched_obligation_ids,
  predicate_results, outcome:matched|unmatched|ambiguous, rationale
```

The observer emits; the matcher verifies source and duplicates before activation (§1.1). Reported events can justify investigation without factual acceptance. Restrictive authorized events enter §7.2.

C3 Obligation

```
OBLIGATION = obligation_id, origin:DeclarationOrCommitRef, scope, gap,  
  resolution_conditions:List<Condition>[1..], dependencies, status,  
  responsible_owner, authority_boundary, closure_authority,  
  reentry_refs, trigger_gap:TriggerGap|null  
Condition = id, proposition, evidence_rule_ref, assessment_status  
TriggerGap = reason, responsible_owner, review_predicate:TypedPredicate
```

Semantic admission establishes the proposed obligation [SR4]. Closure requires a typed delta [SR11] and adjudication [SR12], not a scheduler flip. Every open obligation retains an eligible reconsideration condition or explicit TriggerGap.

C4 Route

```
ROUTE = route_id, capability_gaps, selected_actors, role_assignments,  
  coalition_id, membership_version, capability_contract_refs,  
  workflow_options, disclosure_plan_ref, allocation_ref
```

The router emits; runtime verifies registry identity, contract, version, availability, and authority. Membership changes create versions. Unavailability produces an authorized alternative or retained gap, not silent endpoint substitution.

C5 Work

```
WORK = work_id, question, role, coalition_ref, membership_version,  
  context_items:List<MaterializedItem>, proposal_vocabulary_ref,  
  output_contract_ref, stop_condition, deadline:Timestamp|null
```

The coordinator sends to the participant after coverage and compatibility checks. Materialized items retain immutable references. Missing prerequisites return insufficiency. Examples are not grants. Initial-position exposure follows §4.1.

C6 Result

```
RESULT = work_id, claims:List<ScopedClaim>, action_proposals,  
  assessments:List<EvidenceRelation>, objections, unresolved, skill_receipts  
EvidenceRelation = proposition_id, evidence_id, reference_status,  
  scope_match, relation:supports|contradicts|irrelevant|unresolved,  
  valid_at, limitations
```

Empty meaningful lists are allowed; fabricated proposals or confidence are not required. Objections identify the exact subject and reason class. Unknown IDs or malformed output are counted failures. Concise rationale is sufficient; private chain-of-thought is not required.

C7 Action

```
ACTION = action_id, operation, target, parameters:OperationSpecific,  
    expected_postcondition, authority_ref, relevant_semantic_version,  
    registry_version, idempotency_key, logical_digest,  
    attempt_id, ownership_epoch  
LogicalContent = operation, target, parameters, expected_postcondition,  
    authority_ref, relevant_semantic_version, registry_version, idempotency_key  
logical_digest = H(canonical(LogicalContent))
```

action_id resolves to that logical digest. The attempt's complete envelope additionally binds epoch and attempt ID. Reassignment preserves LogicalContent; dispatch checks the current accepted epoch independently. Changed semantic authority or operation content requires reassessment and a new logical operation/key, with any previous effect reconciled first. The monitor verifies actual effects, target, grants, versions, reservation, and fencing; model labels convey no authority.

C8 Evidence and readback

```
EVIDENCE = evidence_id, source, artifact_ref, observed_at,  
    target_binding, observation:TypedValue, limitations, action_ref:ID|null  
Readback additions = attempt_id, expected_postcondition_ref,  
    readback_method_ref, outcome:matched|not_matched|unknown
```

The observer records; the assessor interprets. A wrong-account observation cannot establish the intended effect. The source artifact may resolve and still be stale, irrelevant, or inconclusive. Evidence does not grant authority.

C9 Adjudication

```
ADJUDICATION = adjudication_id,  
    phase:pre_action|post_outcome|semantic_commit,  
    subject:{id,digest,type}, evidence_snapshot:{refs,digest}, readback_refs,  
    governing_basis, role_positions, decision_rule_ref, dissent,  
    unresolved, disposition
```

Each phase binds its own subject (§4.2). Changed content or evidence requires scoped reassessment. A pre-action position cannot approve later observations; a post-outcome assessment cannot replace semantic-commit endorsement. The declared aggregation rule determines required participants and unresolved objections.

C10 Delta and admission order

```
DELTA = delta_id, parent_state, changes:List<TypedChange>,  
    support_relations, dependency_updates, required_authority_refs,  
    admission_path:ordinary|urgent_restrictive, proposal_digest
```

The immutable proposal digest covers the full proposal body except itself. Endorsements remain separate objects referencing that digest; the later commit receipt references both. This avoids circular hashing. Ordinary admission requires semantic-commit endorsement and an atomic parent check. Post-outcome assessments can support it but do not replace it. Normative changes always require their own authority.

Exception: `urgent_restrictive` permits the narrow, authenticated restriction defined in §7.2 without prior endorsement. The same atomic parent/integrity checks apply. Its receipt requires the urgent fields in C14 and mandatory later reconciliation. It cannot introduce a permissive change or semantic closure [SR18]. The accepted after-state appears only in the commit receipt.

C11 Reentry and typed predicates

```
REENTRY = reentry_id, obligation_id, predicate:TypedPredicate,
  event_sources, eligible_work, resource_posture, deduplication_key,
  expires_at:Timestamp|null, refresh_policy_ref, expiry_policy_ref
TypedPredicate =
  EvidenceAdded(scope_ref, permitted_source_refs, after_cursor)
  | DependencyVersionChanged(dependency_ref, compared_version)
  | UserDecisionReceived(decision_ref, required_decision_class)
  | CapabilityVerified(capability_ref, contract_version)
  | HandoffAccepted(handoff_ref, recipient_ref, contract_digest, ownership_epoch)
  | TimeReached(at:Timestamp)
  | All(items:List<TypedPredicate>[1..])
  | Any(items:List<TypedPredicate>[1..])
Evaluation = satisfied|not_satisfied|unknown
```

Arguments are typed identifiers/times; arbitrary executable expressions are excluded. Evaluation records predicate version, current governing state, event cursor, input receipt references, authenticated source binding, scope, and result. `HandoffAccepted` requires the ownership service's atomic acceptance receipt matching recipient, contract digest, handoff [SR16], and epoch; a model's claimed acceptance is insufficient. Unknown source, missing input, or unsupported operator yields `unknown`, never permission. `All` requires every child satisfied; `Any` requires one satisfied; `unknown` propagates when truth is otherwise undetermined. A qualifying event permits reconsideration only; current authority still governs work and effects. Expiry disables this trigger and transactionally installs a replacement or `TriggerGap` for the retained obligation [SR4] (§7.3). Trigger expiry never closes it.

C12 Handoff

```
HANDOFF = handoff_id, obligation_id, responsibility, sender, recipient,
  contract_ref, context_refs, ownership_epoch, lease_expiry:Timestamp|null,
  acceptance_condition:TypedPredicate, return_contract_ref
```

The service validates recipient consent, contract compatibility, and context coverage before the atomic ownership transfer. `acceptance_condition` verifies the completed transfer from its authenticated receipt; it is not a prerequisite requiring that receipt before the transaction. Until acceptance the sender/designated coordinator retains responsibility. Expiry fences the old epoch; uncertain effects are reconciled before repetition. An accepted recipient acts with the current epoch under unchanged user [SR1] authority. The epoch is excluded from logical operation identity (C7).

C13 Briefcase decision packet

```
BRIEFCASE = decision_id, obligation_ids, question, decision_class,  
  recommendation:Recommendation|null, recommendation_reason:String|null,  
  alternatives, evidence_refs, dissent, remaining_uncertainty,  
  requested_authorization:ActionOrGrantProposal|null,  
  deadline:Timestamp|null, deadline_policy_ref, while_waiting,  
  refresh_conditions:List<TypedPredicate>, status
```

A null recommendation requires a nonempty recommendation_reason; no preference or proposal is fabricated. Deadline policy implements §7.3, including overdue status, retained obligation [SR4], and deduplicated escalation. Response types distinguish preference, investigation request, rejection, and scoped authorization. An acknowledgment alone is not a grant.

C14 Foundational and audit objects

```
DECLARATION = declaration_id, authenticated_issuer, goal, scope,  
  constraints, success_criteria, existing_grants, effective_period  
USER_DECISION = decision_id, briefcase_ref, selected_option_or_instruction,  
  authenticated_issuer, issued_at, evidence_version, decision_type  
GRANT = grant_id, authenticated_issuer, allowed_operations, targets,  
  parameter_bounds, resource_bounds, expiry, delegability, revocation  
COMMIT_RECEIPT = receipt_id, delta_id, proposal_digest, admission_path,  
  adjudication_refs:List<ID>, before_state, after_state,  
  committing_authority, committed_at, invalidated_dependencies,  
  urgent_details:UrgentDetails|null  
UrgentDetails = issuer_auth_ref, restrictive_authority_ref, reason,  
  evidence_refs, affected_scope, reconciliation_obligation_id,  
  reconciliation_due_at, independent_owner, escalation_policy_ref  
RECONCILIATION_RECEIPT = receipt_id, urgent_commit_ref, assessment_ref,  
  disposition, justification, evidence_refs, issued_at, authority_ref
```

For ordinary commits, adjudication_refs is nonempty and urgent_details null. For urgent restrictions, the list is explicitly empty when no prior endorsement exists; urgent_details is mandatory and non-null. The mandatory reconciliation receipt later references the original urgent receipt without altering it. Overdue review follows §7.2; restoration needs a new authorized commit. Every accepted state transition has an immediate receipt.

A confession [SR20] records affected claim/action/receipt, omitted check or deviation, supporting evidence, unknowns, cost, consequence, status, and reconsideration condition. A disclosure incident records released item/recipient references, containment and notification attempts, dependent artifacts, acknowledgments, unknown copies, and accountable incident owner. These auxiliary records are authenticated and disclosure-governed; neither independently establishes factual truth.

C15 Trace contracts

Ordinary:

event → matched obligation → work → action proposal → pre-action adjudication
→ dispatch → readback → post-outcome adjudication → immutable delta
→ semantic-commit endorsement → commit receipt

Urgent restriction:

authenticated restrictive event → immutable urgent delta + atomic commit receipt
→ affected dispatch refused → mandatory reconciliation receipt by deadline
→ ordinary endorsed correction/confirmation as required

User judgment:

unresolved assessment → Bench → explicit user decision → scoped grant
→ new action assessment → dispatch and readback

The action proposal exists before its pre-action assessment. Ordinary closure occurs only at a supported commit. No receipt erases dissent, unknown effects, or a missed deadline.

References

The v3 source checks dated 2026-09-13 are retained. Versioned documentation describes its cited release; preprints are not validated empirical results. Original actor and truth-maintenance linked PDFs were not fully retrievable during that review, so related descriptions remain limited to foundational concepts. New BDI foundation references supplement, rather than replace, Jason's implementation documentation.

- **R1.** Hewitt, C., Bishop, P., and Steiger, R. (1973). *A Universal Modular ACTOR Formalism for Artificial Intelligence*. IJCAI. [Paper](#).
- **R2.** LangChain. *LangGraph persistence*. [Documentation](#).
- **R3.** CrewAI. *Flows*, version 1.15.21. [Documentation](#).
- **R4.** W3C (2013). *PROV-DM: The PROV Data Model*. [Recommendation](#).
- **R5.** Jason project. *suspend internal action*. [API](#).
- **R6.** de Kleer, J. (1986). *An Assumption-Based TMS*. Artificial Intelligence. [Paper](#).
- **R7.** NIST. *Reference monitor*. [Glossary](#).
- **R8.** Aureli, F., et al. (2008). *Fission-Fusion Dynamics: New Research Frameworks*. Current Anthropology. [Paper](#).
- **R9.** Haesevoets, R., Weyns, D., Holvoet, T., and Joosen, W. (2009). *A Formal Model for Self-Adaptive and Self-Healing Organizations*. SEAMS, 116–125. [Paper](#).
- **R10.** Model Context Protocol. *Architecture*, 2025-06-18. [Specification](#).
- **R11.** A2A project. *Agent2Agent Protocol Specification*. [Specification](#).
- **R12.** Hewitt, C. (2015 version). *Actor Model of Computation: Scalable Robust Information Systems*. arXiv:1008.1459v38. [Review](#).
- **R13.** Li, X., et al. (2026). *MemTX: Transactional Belief Commit for Stateful Agent Memory*. arXiv:2607.23929v2; abstract-level comparison. [Preprint](#).
- **R14.** He, J., and Yu, D. (2026). *Beyond Memory: A Transactional Continuity Kernel for Long-Lived AI Agents*. arXiv:2608.11632v1; abstract-level comparison. [Preprint](#).

- **R15.** Rundgren, A., Jordan, B., and Erdtman, S. (2020). *JSON Canonicalization Scheme*. RFC 8785. [RFC](#).
- **R16.** Microsoft. *Event Sourcing pattern*. [Documentation](#).
- **R17.** Prakken, H. (2009). *Models of Persuasion Dialogue*. [Chapter](#).
- **R18.** Khodyakov, D., Grant, S., Kroger, J., and Bauman, M. (2023). *RAND Methodological Guidance for Conducting and Critically Appraising Delphi Panels*. [Guidance](#).
- **R19.** Erman, L., Hayes-Roth, F., Lesser, V., and Reddy, D. (1980). *The Hearsay-II Speech-Understanding System*. [Paper](#).
- **R20.** Ongaro, D., and Ousterhout, J. (2014). *In Search of an Understandable Consensus Algorithm*. [Paper](#).
- **R21.** McBurney, P., and Parsons, S. *Retraction and Revocation in Agent Deliberation Dialogs*. ULCS-07-016; record/abstract inspected. [Institutional record](#).
- **R22.** Bratman, M. E. (1987). *Intention, Plans, and Practical Reason*. [Publisher reissue record](#).
- **R23.** Rao, A. S., and Georgeff, M. P. (1995). *BDI Agents: From Theory to Practice*. ICMAS, 312–319. [Original paper](#).

Appendix D Human judgment and AI execution

Review status: v4 wording and engineering choices await human review. The user reviewed v3 and requested a shorter, more visual revision, supplying a two-seat AI corrections list. This is evidence of revision requests, not approval of the rewritten artifact or independent proof of its claims. AUTHORSHIP-EVIDENCE-v4.json binds this revision's evidence while preserving the earlier sidecars.

Contribution	Evidence and limit
Human conceptual judgments	Actor foundation; three assessment roles; shared/specialized skills; governing semantics; semantic closure; continued reentry; briefcase; slower distillation; payload routes; useful discovery beyond revenue; generic audience; retained obligations. Detailed excerpts remain in AUTHORSHIP-EVIDENCE-v2.json and AUTHORSHIP-EVIDENCE-v3.json.
AI execution and proposals	Manuscript prose, organization, diagrams, payload details, engineering rules, and evaluation design were generated or revised by AI. Applying a human concept does not establish human approval of its wording or implementation.
Literature	Established concepts belong to their cited authors. Citation verifies attribution, not this integrated protocol.
Latest review	User requested concision and broader visuals, and relayed subsection corrections attributed to Kimi/GLM review. These are review inputs, not a claim that both seats verified v4.

The inherited, non-exhaustive 16-decision register records **12/16 (75%) with explicit human conceptual evidence** and **4/16 (25%) as detailed AI proposals without an evidenced human ruling**. This is a selected, unweighted denominator—not authorship, ownership, approval, or confidence. The latest feedback does not retroactively convert those ratios into artifact approval. No percentage of factual correctness or prose authorship is calculated.

Prior positive feedback remains scoped to its version and subject: v2 section 6 and the contribution account received positive comments; other specified sections required revision. Current human review is needed particularly for the emergency authority/abuse policy (§7.2), disclosure remediation (§4.3), exact contracts (Appendix C), and evaluation tradeoffs (§10). No v4 wording approval is evidenced.

A future review receipt should name artifact/section digest, precise subject, decision type (concept endorsement, wording approval, revision, rejection, or effect authorization), reviewer identity/assurance, timestamp, and supporting message. Silence or approval of a neighboring concept is insufficient. Material revision leaves prior approval attached to the historical artifact. Signatures authenticate statements, not truth; authenticated messages can evidence review when their assurance is stated. Detailed history remains in sidecars instead of repeating throughout the paper.