# SADAR Governance and Conformance

## SADAR Governance and Conformance

Governance and Conformance

SADAR Documentation — Governance

*Draft — May 2026*

## Purpose

This document is the orientation page for the SADAR governance and conformance model: the roles that steward the standard, the separation between governance authority and operational participation, the three-step ladder of **conformance — certification — authorization** that produces SADAR's institutional-trust path, and the architectural patterns that recur at every layer of the SADAR ecosystem.

The normative basis lives across two main sources. The OpenSemantics.org Charter is the canonical source for the steward's authority, the SADAR Certification Program, working group structure, and contribution process. The SADAR Conformance Specification is the canonical source for the criteria implementations are evaluated against. This page treats the governance and conformance model as a coherent topic across both — the institutional substrate that everything else in SADAR depends on.

This document is descriptive in nature, intended to capture the governance architecture for inclusion in subsequent normative specification revisions. A normative requirements summary appears at the end. Open items at the end of the document enumerate concerns that require pinning down in the formal specification.

Audience: enterprise architects, security architects, federation operators, and standards practitioners evaluating how SADAR's governance produces the trust properties the protocol layer depends on. Implementers reasoning about what conformance means and what it takes to participate in public federation as a provider.

## What the Governance Model Covers

SADAR's governance model addresses three concerns. First, who owns the specification and how it evolves. Second, how implementations demonstrate that they meet the specification, and what role institutional attestation of that demonstration plays. Third, how participation in public federation is gated through authorization, and how authorization composes with the cryptographic protocol layer to produce the two-path trust resolution that runs through SADAR.

These concerns are organizational rather than protocol-level. SADAR's protocol mechanisms — manifest signing, bilateral matching, replication with home-registry attribution, the SADAR Context Token — establish *authenticity* through cryptography. The governance model establishes *trustworthiness* through institutional accountability. Both are required: cryptographic verification without institutional trust gives provenance but no judgment; institutional trust without cryptographic verification gives trust assertions that cannot be validated.

This page covers the governance architecture at orientation depth. The detailed certification procedures, working group operations, and contribution process live in the OpenSemantics.org Charter. The detailed conformance criteria live in the SADAR Conformance Specification. Cross-references appear throughout.

## The Three-Role Governance Model

SADAR's governance model defines three distinct roles. Each has its own scope of authority and its own relationship to the operational layer. The roles are organizationally independent within a public federation: a single organization MUST NOT simultaneously occupy a governance role and an operational role for the same federation.

### Steward and Authorizing Body

The Steward is the organization that owns the SADAR specification — OpenSemantics.org — and is also the Authorizing Body. As Authorizing Body, OpenSemantics.org issues certifications and authorizes participation in public federation. The Authorizing Body MAY designate independent assessors to evaluate implementations against the conformance specification, but the Authorizing Body retains the certification and authorization authority itself.

Combining the Steward and Authorizing Body roles in a single organization is acceptable provided the structural safeguards required for governance independence are maintained. The Steward MUST NOT operate a Registry-of-Registries (RoR) or any registry that participates in public federations it authorizes; financial and operational independence between the governance function and the operational layer MUST be preserved.

Steward / Authorizing Body responsibilities include:

- Maintaining the authoritative specification documents and managing their evolution under the Community Specification License 1.0.
- Defining what conformance means and which features are mandatory or optional in the SADAR Conformance Specification.
- Issuing certifications to implementations evaluated as conformant, directly or through designated assessors.

- Authorizing certified implementations to participate in public federation as providers by listing them in the canonical Directory of Authorized Registries.
- Investigating disputes about specification interpretation or about conduct under the federation regime, and deauthorizing or revoking certification of parties whose conduct warrants it.
- Stewarding the bootstrap algorithm set, the mandatory feature boundary, and other conformance-relevant constants as technologies evolve.

## Operational Entities

Operational entities are the organizations that run components of the SADAR ecosystem. They participate within the governance regime, not as part of it.

| Role | What it does |
|---|---|
| Registry operator | Operates a SADAR registry that publishes and serves manifests for agents, tools, business functions, business processes, and other registered entries. |
| Registry-of-Registries operator | Operates a registry that holds registry descriptors only, used to resolve cross-registry discovery. The Directory of Authorized Registries is the canonical RoR for the SADAR public federation, operated by OpenSemantics.org. |
| Publisher | Authors and signs manifests for entries the publisher owns: agents, tools, business functions, business processes, and registry descriptors for registries the publisher operates. |
| Consumer | Operates agents that consume content through SADAR registries and participate in agentic interactions governed by the protocol primitives. |

The independence between operational and governance roles is structural: an organization operating a registry cannot simultaneously be the body that decides whether their registry is conformant or authorized.

## Independence as an Architectural Property

The separation of governance and operational roles is not a procedural convenience; it is an architectural property of SADAR's institutional-trust path. The trust properties that allow heterogeneous, independently-operated registries to federate depend on the credibility of the certification and authorization regime. If the regime is operated by a party with operational interests in the outcome, the regime cannot credibly attest to conformance — the attestation becomes a statement of competitive interest rather than a statement of conformance.

This is the same principle that underlies the separation of standards bodies, certification authorities, and operators in established industries. In SADAR's context, independence is

normative: a single organization MUST NOT combine governance and operational roles for the same federation.

## The Recursive Architectural Pattern

SADAR's architecture exhibits a recursive pattern at every layer. The same identity, manifest, lifecycle, authentication, and federation primitives apply to publisher-to-agent registration, to registry-to-RoR registration, and (operationally) to RoR catalog inclusion of authorized registries. The differences between layers are in the role each entry plays, not in the mechanism by which entries are identified, signed, verified, and lifecycle-managed.

Every entry within the SADAR ecosystem shares the following common properties:

- It has an `owning_entity` — the organization accountable for the entry, attributed within the SADAR entity hierarchy.
- It has a durable signing keypair held by its owning entity.
- It has a publisher-signed manifest published through a registry.
- Its verification key is anchored to a trust anchor declared in its manifest.
- It is subject to the universal lifecycle: TTL, deprecation history, renewal, revocation.
- It authenticates and is authenticated using the SADAR authentication primitives.

| Layer | Role | Anchored at |
|---|---|---|
| 0 — Governance | Steward / Authorizing Body. Owns the specification, certifies conformance, authorizes participation in public federation. | OpenSemantics.org. Independent of operational layers. |
| 1 — RoR | Registry-of-Registries. Holds registry descriptors only; resolves cross-registry discovery. | Itself a registered entry with a manifest, owning entity, and verification key. Replicates with peer RoRs under the standard registry federation contract. |
| 2 — Registry | Holds publisher-signed manifests for agents, tools, business functions, business processes, and other entries. | Itself a registered entry, listed in the Directory when authorized for public federation as a provider. |
| 3 — Entries | Agents, tools, business functions, business processes — the operational artifacts of capability and process. | Published through a home registry; replicated to peer registries through bilateral federation under the SADAR replication contract. |

At each layer, the entries above are the publishers and consumers of the layer below. The same protocol patterns apply at each layer; the role of each entry differs, but the mechanism is

identical. This produces a uniform mental model for operators, a uniform implementation surface for software, and a uniform set of properties for security analysis.

## Conformance, Certification, and Authorization

The institutional-trust path in SADAR is built on three distinct concepts that together gate participation in public federation. The three are sequential: each presupposes the one before it. Treating them as a single concept produces architectural ambiguity; treating them as three concepts produces a clean three-step ladder with well-defined transitions.

### Conformance

Conformance is functional compliance with the SADAR specification. Every registry — public, private, federated, internal-only — MUST implement the protocol correctly to be a SADAR registry at all. Conformance is a functional property of the implementation; without it, the registry is not SADAR.

The SADAR Conformance Specification distinguishes between mandatory features (which MUST be implemented for conformance) and optional features (which are declared in the implementation's manifest and bilaterally matched with counterparties at runtime). An implementation that omits a mandatory feature is non-conformant regardless of what optional features it implements.

This binary distinction is itself a security property. Optionalism at the conformance level fragments the ecosystem: every consumer would have to handle every combination of declared and undeclared support, and the federation would degrade into a set of bilateral matrices rather than a coherent network. Mandatory features at the conformance level are the discipline that makes broad federation tractable.

### Certification

Certification is the Authorizing Body's formal attestation that a specific implementation has been evaluated and found to meet the conformance specification. It is the institutional record that converts demonstrated functional compliance into a documented, verifiable attestation that the protocol rules are being followed.

Conformance and certification are distinct: a registry can be functionally conformant without being certified — implementations exist before they are evaluated, and operators of internal-only registries may choose to remain conformant without seeking the institutional attestation. Certification is the bridge from functional compliance to institutional trust.

Certification is implementation-specific (a software product, a deployed service, a hosted offering) rather than vendor-specific; a single vendor may have multiple certified implementations and may have implementations that are not certified. The certification process is governed by the OpenSemantics.org Charter at the operational level. The Authorizing Body MAY designate independent assessors to perform evaluations; final certification authority remains with the Authorizing Body itself. At the architectural level, SADAR requires that:

- Certification evaluates the implementation against a specific published version of the Conformance Specification.
- Certification produces a verifiable record discoverable through the Authorizing Body's certification records.
- Certification expires on a schedule defined in the Charter, requiring periodic re-certification.
- Certification records are publicly discoverable, allowing any party to verify the current certification status of an implementation.

## Authorization

Authorization is the right to participate in public federation as a provider. It is a separate Authorizing Body act that requires certification as a precondition: an uncertified registry cannot be authorized, and a certified registry MAY be authorized but is not automatically. Authorization is the gating decision for the public-federation provider role.

An authorized registry is listed in the Directory of Authorized Registries. Listing in the Directory is the operational signal of authorization, and federation candidates discover one another through that listing. A registry not listed in the Directory is not authorized for public federation, regardless of any other claims it may make.

Authorization gates a specific role: serving as a home registry for content that flows into public federation. A non-authorized registry MAY still consume content from authorized registries through bilateral federation (subject to the authorized registry's federation policy), but MUST NOT serve as the home registry for content that will be discoverable to public-federation peers. This directional asymmetry is the structural enforcement of the institutional-trust path: home-registry attribution flows only from registries that the Authorizing Body has institutionally vouched for.

## The Three-Step Ladder Applied to Operational Modes

The three concepts compose into a clean classification of operational modes. The combination of conformance, certification, and authorization status determines what role a registry can play.

| Operational mode | Conformant | Certified | Authorized | Role permitted |
|---|---|---|---|---|
| Internal-only (private) | Required | Optional (operator's choice) | Not eligible | Consumer only. May consume from authorized registries through bilateral federation. MUST NOT be a home registry for public-federation content. |
| Authorized non-federated | Required | Required | Not granted (operator's choice or Authorizing Body's) | Consumer with verifiable conformance. Same role limits as Internal-only; certification provides third-party-validated conformance claim. |
| Authorized federated | Required | Required | Granted; listed in Directory | Full provider in public federation. May serve as home registry for content discoverable to public-federation peers. |

All three modes are SADAR registries; all three implement the same protocol; all three meet the same conformance baseline. They differ in the institutional treatment of that conformance and in the role each is permitted to play in the public federation. The protocol mechanics enforce the role limits structurally: a consumer following the verification chain at a peer registry confirms institutional trust against the Directory, and content from a non-authorized home registry fails that confirmation. The architectural enforcement is automatic; no separate “is this registry allowed to be a home” check is required.

## Deauthorization and Certification Revocation

Authorization MAY be revoked (deauthorization) by the Authorizing Body. Certification MAY also be revoked, separately. The two acts are decoupled: the Authorizing Body MAY do either, both, or neither in response to investigated conduct, depending on what the conduct reflects on.

| Action | When applied | Effect |
|---|---|---|
| Deauthorization without certification revocation | Conduct that violates federation norms but does not reflect on conformance — contractual disputes, federation-policy violations, conduct issues unrelated to protocol implementation. | Registry transitions from Authorized federated to Authorized non-federated. Still certified; removed from the Directory; no longer a public-federation provider. |
| Certification revocation with deauthorization | Conformance failures detected after certification, or conformance violations significant enough to | Registry transitions to Internal-only or to a non-conformant state if the operator does not remediate. Removed from the Directory and from the certification records. |

| Action | When applied | Effect |
|---|---|---|
| | invalidate the institutional attestation. | |
| Certification revocation without deauthorization | Not a typical case; included for completeness. Would arise only if certification expires while authorization has not yet been formally revoked. | Registry's certification record is no longer current, which makes its authorization no longer valid since authorization requires current certification. Effectively equivalent to deauthorization plus certification revocation, with the Authorizing Body potentially handling them in either order. |

The decoupling matches real-world dispute scenarios. An Authorized Registry whose conduct produces a federation dispute might still be functionally conformant; deauthorization addresses the conduct without invalidating the technical attestation. A registry whose conformance has been found wanting needs both: deauthorization (exit the public federation) and certification revocation (the technical attestation no longer holds).

Federation relationships the deauthorized registry had with peers continue under their current TTLs but are not renewed. The institutional-trust path that supported those relationships has been withdrawn; new relationships are not established; existing relationships expire on their own schedule. Deauthorization propagates through the standard SADAR lifecycle and push mechanism.

# Two-Path Trust Resolution

Trust in SADAR is resolved through two independent paths that compose to give a complete trust assessment. Both paths are available; consumers and admins use both, and the integrity of the federation depends on both being functional.

## Cryptographic Provenance

The first path is cryptographic verification of provenance. Every artifact in SADAR carries signatures verifiable independently against verification keys anchored in trust anchors that the verifying party recognizes. The verification chain typically extends from the artifact through the publisher's signature, to the publisher's manifest, to the publisher's verification key (typically a JWKS endpoint), to the trust anchor declared in the publisher's manifest.

This path establishes *authenticity* — is this artifact really what it claims to be, signed by the party it claims to be signed by, with no tampering since signing? Cryptographic provenance answers this independently of any institutional knowledge of the parties involved.

### Institutional Trust

The second path is institutional trust through authorization and federation membership. A consumer trusts a registry implicitly because (a) the consumer's home-registry admin configured federation with that registry from candidates listed in the Directory, exercising organizational judgment, and (b) the Authorizing Body has authorized the registry, attesting that it is certified and meets the requirements for public-federation participation. Both are institutional acts that establish trust at a layer above cryptography.

This path establishes *trustworthiness* — should this party be trusted in the first place, even if their artifacts are cryptographically authentic? Institutional trust answers this by reference to the authorization regime and the federation establishment decisions of admins. The institutional-trust check is verified specifically against current authorization (Directory listing), not against certification alone — a certified-but-not-authorized registry does not pass the check for public-federation content.

### Why Both Paths Are Required

Cryptographic verification without institutional trust gives provenance but no judgment. A consumer can confirm an artifact was signed by a specific entity with cryptographic certainty, but has no basis for deciding whether that entity should be trusted. Institutional trust supplies that judgment.

Institutional trust without cryptographic verification gives trust assertions that cannot be validated. A claim that a registry is authorized is meaningless unless the consumer can verify that the artifacts they receive actually originated from that registry. Cryptographic verification supplies the validation.

Both paths are necessary, neither is sufficient, and they fail independently. Deauthorization affects the institutional path while leaving the cryptographic record of past interactions intact. Trust-anchor compromise affects the cryptographic path while leaving the institutional standing of the publisher unchanged. The federation's resilience depends on these failure modes being independently detectable and addressable.

## Universal Lifecycle

Every discoverable artifact in SADAR is subject to the same lifecycle. The mechanism applies uniformly to manifests, registry descriptors, federation entries, certification records, authorization status, home-registry bindings, and any other artifact consumers retrieve, cache, or rely upon.

- **TTL.** Every discoverable artifact carries a TTL declared by its publisher. Risk-appropriate TTLs are an operator decision: high-risk relationships use shorter TTLs to drive more

frequent re-authorization. TTL expiry is a re-authorization checkpoint, not a security failure.

- **Deprecation.** Deprecation is the publisher's signal that an artifact is approaching end-of-life. Deprecation history is durable: a deprecated version remains discoverable through history queries even after it is no longer current, because cryptographic artifacts (such as bindings) may still reference the deprecated version by hash and may need to be reasoned about for forensic purposes. New relationships, however, MUST NOT be established against deprecated versions.
- **Revocation.** Revocation is the publisher's (or, where applicable, the Authorizing Body's) immediate withdrawal of an artifact's validity. A revoked artifact is no longer authoritative, even if not yet expired by TTL.
- **Renewal as re-authorization.** Renewal in SADAR is substantive re-authorization, not automatic re-issuance. When a TTL expires and a consumer re-resolves an artifact, the resolution flow re-evaluates the underlying authorization conditions: the publisher's signing key is still valid, the publisher is still authorized in their federation, the entity has not been deprecated or revoked, and so on.

## Authentication Uniformity

All authentication in SADAR uses the same primitives, regardless of the layer at which authentication occurs. The same protocol an agent uses to authenticate to another agent is the protocol a registry uses to authenticate to a peer registry, an RoR uses to authenticate to a peer RoR, and any party uses to authenticate to a push endpoint.

This uniformity produces a uniform implementation surface (one authentication library applied to every channel), a uniform security analysis (one set of cryptographic properties evaluated once), and a uniform operator experience (one set of authentication concerns configured once per entity). The protocol primitives are mTLS at the transport layer, JWS-signed messages with publisher-verifiable signatures, the SADAR Context Token for cross-trust-boundary authorization context, and time-of-use credentials issued at the relationship layer. Detailed specification of these primitives lives in the relevant companion documents (SCT Operations, Trust Models, searchAndInvoke Telemetry and Authentication).

## Boundaries — What's Not Here

This page covers governance and conformance at orientation depth. Several adjacent topics live in dedicated documents:

| Topic | Where it lives |
|---|---|
| Specification ownership and contribution process | How OpenSemantics.org stewards the SADAR specification, the working group structure, the contribution process, and the operational procedures for the SADAR Certification Program. See OpenSemantics.org Charter. |
| Conformance criteria in detail | The specific mandatory and optional features, the bootstrap algorithm set, the test suites, and the validation rules an implementation is evaluated against. See SADAR Conformance Specification. |
| Trademark policy | Use of the SADAR mark, certification mark authorization, and enforcement. See OpenSemantics.org Trademark Policy. |
| Registry architecture | The registry itself — what it does, what it doesn't do, the six registry types, the three foundational principles. See Registry Overview. |
| RoR architecture | How a Registry-of-Registries operates, the Directory of Authorized Registries, registry descriptors, and the operational shape of an RoR. See Registry of Registries. |
| Federation establishment and policy | How registries discover federation candidates, evaluate compatibility, establish bilateral agreements, and administer per-registry policy. See Federation Establishment and Policy. |
| Replication mechanics | How manifests retain verifiable provenance when replicated across federated registries, including the home-registry binding. See Replication and Manifest Provenance. |

# Normative Requirements Summary

The following requirements use the terminology of **RFC 2119** / RFC 8174 (“MUST,” “SHOULD,” “MAY”, etc.) as it will appear in the normative specification revision.

## Governance Independence

1. **R-GOV-1.** The Steward / Authorizing Body MUST maintain organizational and financial independence from operational entities (registries, RoRs) within the public federations it authorizes.
2. **R-GOV-2.** The Authorizing Body MUST NOT operate any registry or RoR within the public federations it authorizes.
3. **R-GOV-3.** The Authorizing Body MAY designate independent assessors to evaluate implementations against the Conformance Specification. Final certification and authorization authority remains with the Authorizing Body itself.

## Conformance

4. **R-CONF-1.** A registry MUST be functionally conformant to the SADAR specification to be considered a SADAR registry, regardless of operational mode.
5. **R-CONF-2.** Conformance requires implementing every feature designated as mandatory in the SADAR Conformance Specification.
6. **R-CONF-3.** Optional features SHALL be declared in the implementation's manifest and bilaterally matched with counterparties at runtime.
7. **R-CONF-4.** Conformance is required of all SADAR registries; certification and authorization are NOT required for a registry to operate as a SADAR registry.

## Certification

8. **R-CERT-1.** A registry seeking certification MUST be evaluated by the Authorizing Body (or its designated assessors) against a specific published version of the SADAR Conformance Specification.
9. **R-CERT-2.** Certification MUST produce a verifiable record discoverable through the Authorizing Body's certification records.
10. **R-CERT-3.** Certification MUST be time-bounded and require periodic re-certification per the schedule defined in the OpenSemantics.org Charter.
11. **R-CERT-4.** Certification MAY be revoked by the Authorizing Body if conformance failures are detected. Certification revocation is independent of authorization status: the Authorizing Body MAY revoke certification with or without simultaneous deauthorization.

## Authorization

12. **R-AUTH-1.** Authorization to participate in public federation as a provider MUST be granted by the Authorizing Body and signaled by listing in the Directory of Authorized Registries.
13. **R-AUTH-2.** Authorization is contingent on current certification. A certified registry MAY be authorized; an uncertified registry MUST NOT be authorized.
14. **R-AUTH-3.** Only authorized registries MAY serve as home registries for content flowing into public federation. A non-authorized registry MAY consume content from authorized registries through bilateral federation, subject to the authorized registry's federation policy, but MUST NOT serve as a home registry for content discoverable to public-federation peers.
15. **R-AUTH-4.** Authorization MAY be revoked (deauthorization) by the Authorizing Body. Deauthorization is independent of certification revocation: the Authorizing Body MAY do either, both, or neither in response to investigated conduct.

16. **R-AUTH-5.** Deauthorization SHALL be propagated through the standard lifecycle mechanism (deprecation, push notification, Directory update, re-resolution at consulting parties).

### Two-Path Trust Resolution

17. **R-TRUST-1.** Consumers and admins MUST verify both the cryptographic provenance and the institutional trust paths when establishing new relationships.
18. **R-TRUST-2.** Verification of cryptographic provenance MUST extend through the artifact's signature chain to a recognized trust anchor declared in the relevant manifest.
19. **R-TRUST-3.** Verification of institutional trust MUST confirm that the home registry of public-federation content is currently authorized (i.e., listed in the Directory of Authorized Registries). Certification alone is not sufficient; current authorization is required.

### Universal Lifecycle

20. **R-LC-1.** Every discoverable artifact MUST carry a TTL declared by its publisher.
21. **R-LC-2.** Consumers MUST re-resolve artifacts upon TTL expiry; cached use beyond TTL is non-conformant.
22. **R-LC-3.** Renewal at TTL expiry MUST be substantive re-authorization, not automatic re-issuance.
23. **R-LC-4.** Deprecation history MUST be durable; deprecated versions remain discoverable for forensic and verification purposes.
24. **R-LC-5.** Revocation SHALL be propagated through the push channel for operational responsiveness, in addition to being discoverable through TTL-driven re-resolution.

### Authentication Uniformity

25. **R-AUTHN-1.** All channels carrying SADAR protocol traffic MUST use mutual TLS.
26. **R-AUTHN-2.** The SADAR authentication primitives (mTLS, JWS-signed messages, time-of-use credentials, SCT) MUST be applied uniformly across all SADAR channels regardless of the layer at which the channel operates.
27. **R-AUTHN-3.** Implementations MUST NOT introduce special-purpose authentication mechanisms for federation, replication, or governance interactions.

## Open Items

The following items require resolution in subsequent specification or operational documentation work.

### Mandatory vs. Optional Feature Boundaries

The Conformance Specification defines which features are mandatory and which are optional. The boundary between these categories is a meaningful design decision that affects ecosystem coherence; subsequent specification work will revisit this boundary as the specification evolves.

### Bootstrap Algorithm Set

The cryptographic algorithms that all conformant implementations MUST support — the bootstrap set — are specified in the Conformance Specification and managed by the Authorizing Body as technologies evolve. Subsequent specification work will define the migration mechanism for evolving the set, the deprecation cadence for legacy algorithms, and the post-quantum migration path.

## Where to Learn More

OpenSemantics.org Charter — the canonical source for the steward's authority, the SADAR Certification Program, working group structure, and contribution process.

SADAR Conformance Specification — the canonical source for mandatory and optional features, the bootstrap algorithm set, test suites, and validation rules.

OpenSemantics.org Trademark Policy — use of the SADAR mark, certification mark authorization, and enforcement.

Registry Overview — the architectural framing for registries, the six registry types, and the three operational modes.

Registry of Registries — the federation-layer component that resolves cross-registry discovery, including the canonical Directory of Authorized Registries.

Federation Establishment and Policy — how registries discover federation candidates, evaluate compatibility, and administer per-registry policy.

Replication and Manifest Provenance — how manifests retain verifiable provenance when replicated across federated registries.

2. Scope §5.1.6 — Registry Protocol: registration, validation, immutability, error responses.

2. Scope §5.1.11 — Registry Topology, Federation, and Directory.

2. Scope §5.1.12 — Registry Isolation: the normative constraints establishing the registry as a discovery-time component only.