AOS Specification Quality Standard

Document: specification-quality-standard · Source revision or record: AOS-QS v0.1, 18 September 2026

AOS Specifications v1.0 · Edition dated 3 November 2026

Original architecture and specifications: Mark Sweeny, Governance Intelligence. © 2026 Mark Sweeny.

AOS-authored material is licensed under Creative Commons Attribution 4.0 International (CC BY 4.0).
https://creativecommons.org/licenses/by/4.0/

Preserve third-party credits. Referenced external works remain subject to their own terms.

Canonical collection: https://www.governanceintelligence.co.uk/aos

Publication copy with editorial notices, public references and disclosure edits where applicable.

Table row 1
Document control: Identifier
Value: AOS-QS

Table row 2
Document control: Version
Value: 0.1

Table row 3
Document control: Date
Value: 18 September 2026

Table row 4
Document control: Status
Value: First draft for project review; not an approved organisational governance instrument

Table row 5
Document control: Project owner
Value: Mark Sweeny

Table row 6
Document control: Prepared by
Value: Codex, at the project owner's request

Table row 7
Document control: Independent reviewer
Value: Not yet assigned

Table row 8
Document control: Approval record
Value: No approval recorded

Table row 9
Document control: Applies to
Value: Specifications and capability annexes identified in SPECIFICATION_MAP.md

Table row 10
Document control: Change history

Value: 0.1 - initial issue, including template and review checklist

1. Purpose and status

This standard defines how AOS specifications are to be written, organised, reviewed and controlled so that an AI coding agent can implement an authorised scope without inventing material architectural or governance decisions. It supports PROJECT_OBJECTIVE.md and the component structure in SPECIFICATION_MAP.md.

It makes two assessments distinct:

- Specification quality: whether requirements are identifiable, understandable, traceable, consistent, controlled and verifiable.
- Design adequacy: whether those requirements are the right ones for the organisation, technically feasible, faithful to AOS principles and supported by the necessary human decisions.

A specification can pass the first assessment and fail the second. Neither assessment, alone, proves that implemented software works or that an operating institution governs well.

The numbered QS rules below are the proposed mandatory rules of this project standard once adopted. Examples and explanatory paragraphs are informative unless incorporated into a numbered rule. This first draft does not retrospectively declare existing documents conformant, approve proposed system capabilities, or authorise deployment.

2. External reference basis

The external references inform the approach; the QS rules and gates are project-specific provisions, not reproduced standard clauses.

Table row 1
Reference: ISO/IEC/IEEE 29148:2018 - Requirements engineering (https://www.iso.org/obp/ui?_escaped_fragment_=iso:std:iso-iec-ieee:29148:ed-2:v1:en)
Relevance and limit: Addresses requirements processes, information items, their contents and format guidance. It provides the principal reference direction for requirements documentation.

Table row 2
Reference: ISO/IEC/IEEE 42010:2022 - Architecture description (https://www.iso.org/standard/74393.html)
Relevance and limit: Addresses the structure and expression of architecture descriptions, separately from the architectures described. It does not prescribe a document medium. It informs the shared architecture and relationships between component specifications.

Table row 3
Reference: INCOSE - Guide to Writing Requirements (https://www.incose.org/resources-publications/)
Relevance and limit: Practical guidance on clear, verifiable requirements, including writing patterns and quality checks. It informs the writing discipline.

Reference pages checked on 18 September 2026. This version uses public descriptions and available guidance; it is not a clause-by-clause conformity assessment against the full standards. No ISO, IEEE or INCOSE certification or endorsement is claimed. Any later external-conformity claim requires the applicable full texts, a defined assessment scope, a mapping to their requirements and evidence of the assessment.

Markdown is the proposed authoritative authoring format. Readable Word/PDF exports may be supplied when needed. Neither polished presentation nor a particular file type establishes specification quality.

3. Document control and structure

QS-01 - Controlled identity. Every specification shall state its stable identifier, title, version, date, status, scope, owner, author, reviewer, approval record and applicable AOS-QS version. An unassigned role or absent approval shall be stated explicitly. Such absence is permissible during drafting but not where required by a readiness gate.

QS-02 - Common structure. Every specification shall use the template in Appendix A or provide a section-to-template mapping. A section may be marked not applicable only with a reason and a recorded reviewer determination. Related details may be held in controlled annexes rather than duplicated.

QS-03 - Stable identity and history. Requirements, interfaces, decisions and verification cases shall have stable identifiers that are not reused for different meanings. Changes shall preserve the prior approved baseline and record the author, reason and affected identifiers. Editorial changes shall be distinguished from changes to behaviour, authority, data use or acceptance conditions.

Recommended identifier patterns are AOS-11-REQ-001, AOS-02-IF-001, AOS-DEC-001 and AOS-11-VER-001. The prefixes identify ownership, not deployment units. The D01-D13 questions in the specification map remain source references until individual decision records are created.

QS-04 - Explicit normative status. Specifications shall distinguish requirements from rationale, illustrative examples, assumptions, proposals and unresolved decisions. “Shall” denotes a mandatory obligation; “shall not” denotes a prohibition; “may” denotes an explicitly optional behaviour. “Should” is reserved for non-binding guidance and shall not carry a build-critical obligation. Draft requirement wording does not confer approval merely by using “shall”.

4. Requirement writing and records

QS-05 - One obligation. Each requirement shall identify the responsible system component or human role and express one independently verifiable obligation. Independently changeable or testable obligations shall have separate identifiers. A condition and its required response may form one requirement.

QS-06 - Precise meaning. Requirements shall define or reference the conditions, observable result, timing and boundaries necessary to interpret them consistently. Terms, units, time origins, ranges and error states shall be explicit where relevant. Words such as “promptly”, “appropriate”, “secure”, “robust” and “sufficient” shall not substitute for defined criteria.

QS-07 - Appropriate level. Each requirement shall be classified as a stakeholder need, system requirement, component requirement, interface requirement or design constraint. Component requirements shall link to their parent requirement or an explicit derivation decision. A mandated technology or implementation choice shall have a recorded justification; otherwise the specification shall describe required behaviour and leave implementation discretion visible.

QS-08 - Complete requirement record. Each requirement shall have the following attributes, either locally or through an unambiguous link to a shared register:

Table row 1
Attribute: Identity and statement
Required information: Stable ID and normative text.

Table row 2
Attribute: Level and category
Required information: For example, component/security, interface/data or system/governance.

Table row 3
Attribute: Source and derivation
Required information: Source ID, version and precise locator; parent requirement or decision where derived.

Table row 4
Attribute: Rationale
Required information: Why the obligation is necessary and what failure it addresses.

Table row 5
Attribute: Accountability
Required information: Requirement owner and authority that approves it. These may differ.

Table row 6
Attribute: Status and release scope
Required information: Proposed, reviewed, approved, superseded or rejected; relevant release or configuration.

Table row 7
Attribute: Dependencies
Required information: Referenced interfaces, parameters, assumptions and blocking decisions. Explicit “none” is acceptable.

Table row 8
Attribute: Verification
Required information: Method, linked verification case and objective acceptance condition.

Table row 9
Attribute: Change record
Required information: Origin and subsequent material changes.

QS-09 - No hidden defaults. A parameter, threshold or permission not yet decided shall be marked unresolved, with an owner and the affected requirements identified. Illustrative values shall be labelled as examples and excluded from build instructions. Configurable parameters shall specify their permitted range or validation rule, authorisation, versioning and behaviour when absent, expired or invalid.

QS-10 - Source precision. Requirements shall distinguish directly adopted source obligations, interpretations and new engineering decisions. Source references shall include a heading, principle text locator, section or other durable location; a filename alone is insufficient for an individual requirement. Legal and supplier-dependent requirements shall record jurisdiction or product scope, verification date and a review trigger where applicable. Conflicting sources shall have an explicit resolution record before dependent requirements are approved.

Example of insufficient wording: “The system shall escalate serious welfare issues quickly.”

Improved drafting pattern: “When an event meets trigger WEL-01 in approved policy version [identifier], the routing service shall deliver it to that policy's designated review queue within [approved interval] of event receipt.”

The second sentence is still a draft until its referenced trigger, policy and interval are resolved. Separate requirements would address delivery evidence, acknowledgment, unavailable recipients, deadline breach and closure. This example supplies no operational welfare threshold.

5. Architecture, data and interfaces

QS-11 - Architectural context. Each specification shall identify its purpose, stakeholders, concerns, responsibilities, exclusions, dependencies and trust/authority boundaries. The shared architecture shall maintain consistent component, information-flow, execution and deployment views, or justify why a view is unnecessary. Diagrams shall identify their scope and use defined terms; they shall not silently introduce requirements absent from the controlled text or contracts.

QS-12 - Interface completeness. Each interface shall identify producer, consumer, owner, purpose, contract version, inputs, outputs, permissions, preconditions and failure responses. Contracts shall define relevant schemas, units, identifiers, valid/invalid values, timing, acknowledgment, duplicate handling, ordering, retries and revocation behaviour. Each item shall be specified or

explicitly judged not applicable. Both sides shall reference the same authoritative contract.

QS-13 - Data lifecycle. Specifications shall define or reference collection purpose, provenance, permitted access and inference, consent or other authorised basis, retention, correction, deletion and audit treatment for their data. They shall distinguish observations, inferences, proposals, approvals and outcomes. Requirements for persistent evidence shall be reconciled with data-minimisation and deletion requirements rather than allowing one to override the other by implication.

QS-14 - Behaviour and failure coverage. Workflows shall describe normal operation, exceptions, invalid inputs, dependency failure, interrupted execution, safe fallback and recovery. Stateful processes shall identify states, permitted transitions and the conditions preventing transition. Consequential handoffs shall state when responsibility transfers, what proves receipt, and what happens if receipt or action does not occur.

QS-15 - Measurable operating qualities. Performance, availability, recovery, capacity, accessibility and security requirements shall identify relevant operating conditions and an assessment method. Claims such as “real time”, “always available” or “impossible to bypass” shall not stand without defined limits, assumptions and evidence criteria. A quality may be assessed by structured inspection or analysis where a numerical measure would be inappropriate.

6. AOS-specific specification obligations

These rules concern what must be made explicit and reviewed. The substantive authority comes from the adopted AOS principles and the organisation's approved instruments.

QS-16 - Human and agent authority. Every consequential workflow shall identify what the system can execute, what it must not execute, what requires human judgement, and the authority governing each transition. Human review shall have a named accountable role, required evidence, time conditions and an outcome record. A checkbox or signature shall not be described as proof that substantive judgement occurred.

QS-17 - Constitutional fidelity. Each component shall carry a documented applicability assessment against Foundational Principles v4, AOS Reasoning Principles v6 and the Governance Intelligence Tests, using their controlled source versions. This assessment shall address the 1+5 distinction, permanent human Mode 2, boardroom exclusion, purpose limits, protected escalation, uncertainty and feedback where applicable. Exceptions or conflicts require explicit resolution by the appropriate authority; the specification author cannot silently waive them.

QS-18 - Probabilistic behaviour. Where AI inference affects an output or action, the specification shall distinguish it from deterministic rules and enforcement. It shall define the intended task, allowed outputs, prohibited behaviour, confidence meaning, evaluation conditions, failure response and model-change review. Quality claims shall identify evaluation data, metrics or human-review rubric, acceptance thresholds and relevant limitations. A model's self-reported confidence shall not be treated as validated calibration without evidence.

QS-19 - Evidence without invented reasoning. Audit requirements shall specify evidence actually captured: inputs, source/version references, applied rules, actions, outputs, uncertainty and approved human rationale. They shall distinguish contemporaneous records from later explanations. Requirements shall not depend on recovering an AI's private internal reasoning or treat a generated retrospective explanation as proof of the actual decision path.

QS-20 - Controlled learning and interpretation. Specifications shall distinguish learning from outcomes, changing an analytical model, changing an operational parameter and amending constitutional authority. Each shall have its own permissions and review path. Revealed preferences and precedent shall not silently acquire authority to amend founding purpose. Where common agent calibration is adopted, its agent definition and GI oversight specification shall cross-reference each other.

7. Traceability and verification

QS-21 - Two-way traceability. The suite shall support navigation from source need or authorised decision to requirement, responsible component, interface and verification case, and back again. Shared rules shall have one authoritative definition.

Unallocated requirements, requirements without a source/derivation, incompatible duplicate definitions and verification cases without linked requirements shall be identified during review.

QS-22 - Verification definition. Every approved requirement shall have at least one defined verification method and an acceptance condition. Each verification case shall identify:

- Case ID and requirement IDs.
- Method: test, inspection, analysis or demonstration; combinations are permitted.
- Preconditions, configuration, representative inputs and dependencies.
- Procedure or assessment steps.
- Expected results and objective pass/fail criteria, or a defined judgement rubric and qualified reviewer where appropriate.
- Evidence to retain and the role responsible for evaluating it.

At specification review, these are planned verification cases, not claims of successful execution. After implementation, the result, tested versions, date, evidence and assessor shall be added to a separate execution record.

QS-23 - System-level coverage. The suite shall specify end-to-end cases for interactions that component tests cannot establish, including conflicting signals, uncertain evidence, withdrawn authority/consent, mandatory escalation, partial failure and recovery. Safety, privacy and authority controls shall include negative and attempted-bypass cases. Longitudinal human governance assessments shall remain distinct from software pass/fail results.

8. Review, readiness and change

QS-24 - Separate recorded assessments. Every review shall record specification quality and design adequacy separately, with scope, exact document versions, reviewer identity, method, evidence and findings. Automated checks may assist either review but shall not be described as independent human assurance. An author self-review shall be labelled as such; it does not count as the separate reviewer check at Gate 3.

Review gates

Table row 1
Gate: G1 - Draft organised
Minimum evidence: Identifier, owner or explicit vacancy, scope, source baseline, template coverage and visible unknowns.
Meaning: Suitable for collaborative drafting.

Table row 2
Gate: G2 - Reviewable
Minimum evidence: Proposed requirements have complete records or linked blocking issues; interfaces, verification cases and source conflicts are visible; author checklist completed.
Meaning: Sufficiently explicit for substantive review.

Table row 3
Gate: G3 - Ready for authorised build
Minimum evidence: Separate quality and design assessments completed by identified people with the relevant competence/authority; no unresolved blocking findings in build scope; dependencies and contract versions fixed; approval and bounded build scope recorded.
Meaning: A coding agent may implement the specified scope under the project's authorisation. Not operational deployment approval.

Table row 4
Gate: G4 - Implementation verified
Minimum evidence: Execution evidence against the approved baseline, resolved failures, integration results and recorded residual limitations.
Meaning: Evidence that the implementation meets its specified requirements. Not proof of organisational effectiveness or permission to go live.

QS-25 - Readiness determination. A document shall not claim a gate it has not evidenced. Gate 3 shall require every applicable QS rule to pass, subject only to recorded permitted tailoring under QS-27, and a separate positive design-adequacy determination for the build scope. Unresolved decisions may remain outside that scope only where the excluded behaviour and dependency effects are explicitly bounded and reviewed. Missing operational deployment approvals need not prevent a properly bounded build; missing decisions that determine what must be built do prevent readiness for that scope.

Finding classifications

Table row 1
Class: Critical
Interpretation: A defect could conceal or permit a material breach of authority, safety, privacy or the permanent human decision boundary.
Effect: Blocks Gate 3 for affected scope and dependencies.

Table row 2
Class: Major
Interpretation: Missing or inconsistent information requires the builder to invent material behaviour, or prevents verification.
Effect: Blocks Gate 3 for affected scope and dependencies.

Table row 3
Class: Minor
Interpretation: A defect that does not change required behaviour or prevent reliable implementation/verification.
Effect: Must be corrected, or receive a documented exception with owner and disposition before Gate 3.

Table row 4
Class: Observation
Interpretation: Improvement suggestion without a current rule failure.
Effect: Does not block; retain for consideration.

An average score shall not override a blocking finding. Findings shall identify the QS rule or substantive source concerned, evidence, affected requirements, severity rationale, owner, resolution and closure verification. The reviewer shall distinguish format defects from content concerns; both may block readiness.

QS-26 - Change impact. Material changes after approval shall identify affected requirements, interfaces, source mappings, tests and downstream documents, and reopen the relevant review and approval. A changed specification shall not inherit its predecessor's readiness claim automatically. A release manifest shall identify the compatible document, contract and parameter versions supplied to the builder.

QS-27 - Proportionate tailoring. A combined document, external controlled register or reasoned non-applicability decision may replace template duplication if the required information remains accessible. Tailoring shall identify the rule, reason, evidence, affected scope, reviewer and approver. It shall not waive necessary authority, conceal uncertainty, remove traceability or permit an unverifiable mandatory requirement. Critical or major defects shall not be relabelled as tailoring to achieve readiness.

QS-28 - Builder handover check. Before Gate 3, a reviewer other than the specification author shall walk through a normal case, a relevant exception and an authority-boundary case using only the controlled handover package. The reviewer shall record any point requiring an unstated material decision. The package shall state what may be built, what is excluded, which implementation choices remain discretionary and when the builder must stop and report a gap. An AI walkthrough may supplement but not replace this reviewer check.

9. Audit method and evidence pack

A reviewer applying this standard first agrees the scope: a template, one component, a release subset or the whole suite. Passing a template review does not establish that populated specifications meet the same standard.

The review pack contains the baseline manifest, source register, specifications and annexes, requirement/traceability registers, interface catalogue, open decisions, planned verification cases, review findings and approval records. Executed verification evidence is added when assessing Gate 4.

For Gate 3, identifier integrity, required fields, links and unresolved-status checks cover all in-scope requirements. Human review covers all consequential safety, privacy, authority and constitutional requirements and all shared interfaces. Any sampling of remaining requirements must identify the population, selection, exclusions and limitations; the resulting conclusion must not imply exhaustive human review. This is the project's proposed review policy, not an ISO-mandated sampling rule.

An audit report should give separate conclusions for document control, requirement expression, traceability, interface consistency and verification readiness, followed by the separate design-adequacy conclusion. Use pass, fail, not assessed, or not applicable with rationale. Include counts and specific findings rather than a single quality percentage.

No assessment has yet been conducted against this first version. Initial adoption requires project review of the rules, review roles and gates; it does not require pausing ordinary drafting.

Appendix A - Reusable specification template

Copy this structure for each mapped specification. Bracketed entries are placeholders, never default decisions.

Table row 1
Control field: ID / title / version / date
Entry: [Specify]

Table row 2
Control field: Status / gate achieved
Entry: [Draft; none until evidenced]

Table row 3
Control field: Owner / author / reviewer
Entry: [Name or explicit vacancy]

Table row 4
Control field: Approver and approval record
Entry: [Reference or not approved]

Table row 5
Control field: AOS-QS version / source baseline
Entry: [Specify]

Table row 6
Control field: Release scope / dependency manifest
Entry: [Specify]

Table row 7
Control field: Change history
Entry: [Version, change, reason, affected IDs]

1. Purpose and scope: intended outcome, stakeholders, responsibilities, exclusions and deployment assumptions.
2. Authority and sources: precise references, source applicability, interpretations, decisions and unresolved conflicts.
3. Context and architecture: component relationships, relevant views, trust boundaries and human responsibilities.

4. Requirements: stable requirement records using the QS-08 attributes; distinguish mandatory behaviour from supporting explanation.
5. Data and interfaces: controlled contract references, data lifecycle and field/schema definitions.
6. Behaviour and authority: workflows, states, permissions, prohibitions, human decisions, escalation and feedback.
7. Operating qualities: uncertainty, security, privacy, accessibility, capacity, performance and observability.
8. Failure and recovery: abnormal cases, fallback, partial execution, interruption and restoration.
9. Verification and acceptance: case records under QS-22; component and integration coverage; human assessment boundaries.
10. Dependencies and open decisions: owner, affected IDs, resolution gate/date, consequences and excluded scope.
11. Build and handover: approved scope, build dependencies, bounded discretion, stop conditions, configuration and migration requirements.
12. Review and control: checklist, findings, tailoring, separate quality/design conclusions, approvals and change impact.

Reusable decision record

ID | question | alternatives | affected requirement/interface IDs | decision owner | required authority | evidence/source | due gate/date | status | resolution and rationale | approval/date | superseded decision

Reusable review finding

ID | baseline | quality/content classification | rule/source | evidence and location | affected IDs | severity and rationale | owner | required correction | resolution evidence | closure reviewer/date

Appendix B - Review checklist

For each row record pass/fail/not assessed/not applicable, evidence location, reviewer and finding IDs. A tick without evidence is insufficient.

Table row 1
Check: Identity, ownership, status, versions and approval records are explicit.
QS rules: 01, 03

Table row 2
Check: Template coverage and any tailoring are visible and justified.
QS rules: 02, 27

Table row 3
Check: Requirements, rationale, examples and proposals are distinguishable.
QS rules: 04

Table row 4
Check: Requirements are singular, precise and expressed at the correct level.
QS rules: 05-07

Table row 5
Check: Each requirement has all mandatory attributes.
QS rules: 08

Table row 6
Check: No illustrative or unresolved value masquerades as an approved default.
QS rules: 09

Table row 7

Check: Sources, derivations and conflicts are precisely recorded.
QS rules: 10

Table row 8
Check: Architecture views and component responsibilities agree.
QS rules: 11

Table row 9
Check: Producer and consumer use complete, compatible interface contracts.
QS rules: 12

Table row 10
Check: Data purposes, provenance, access and lifecycle are explicit.
QS rules: 13

Table row 11
Check: Exceptions, handoffs, failure and recovery are specified.
QS rules: 14

Table row 12
Check: Operating qualities have assessable criteria and stated assumptions.
QS rules: 15

Table row 13
Check: Human/agent authority and AOS principle applicability are explicit.
QS rules: 16-17

Table row 14
Check: Probabilistic behaviour has an evaluation and failure specification.
QS rules: 18

Table row 15
Check: Audit evidence is distinguished from generated explanations.
QS rules: 19

Table row 16
Check: Learning, interpretation and amendment have distinct controls.
QS rules: 20

Table row 17
Check: Source-to-requirement-to-verification links work in both directions.
QS rules: 21

Table row 18
Check: Verification plans specify pass conditions; execution is not falsely claimed.
QS rules: 22

Table row 19
Check: Cross-domain, negative and attempted-bypass cases are covered.
QS rules: 23

Table row 20
Check: Quality and design reviews are separately evidenced; independence is accurately described.
QS rules: 24

Table row 21
Check: Gate claims match evidence, findings and authorised scope.

QS rules: 25

Table row 22
Check: Changes and dependent baselines are controlled.
QS rules: 26

Table row 23
Check: A separate reviewer completed the builder handover walkthrough.
QS rules: 28

Review conclusion fields: baseline and scope; method and coverage; quality conclusion; design-adequacy conclusion; outstanding findings; permitted tailoring; excluded capabilities; gate achieved; reviewer; approver; date.