Aurora Workflow Orchestration – Method Specification v1.2.1

Shawn C. Wright, Aurora Research Initiative / Waveframe Labs Division

October 2025

Aurora Workflow Orchestration (AWO)

Method Specification — v1.2.1 (Scaffold)

Maintainer: Waveframe Labs

Preface

This document defines the **normative specification** for Aurora Workflow Orchestration (AWO).
It replaces descriptive or philosophical language with enforceable procedural logic.
All future automation layers (e.g., CRI-CORE) must validate conformance against these requirements.

Interpretation of Compliance Language
- **MUST** — absolute requirement for AWO-compliant repositories.
- **SHOULD** — strong recommendation; deviations must be justified in documentation.
- **MAY** — optional behavior permitted for flexibility.

---

1. Introduction

1.1 Purpose

Aurora Workflow Orchestration (AWO) establishes a formal, falsifiable framework for conducting reproducible AI-assisted research.
It defines the structural and procedural rules by which reasoning processes—whether human, synthetic, or hybrid—are documented, attested, and version-controlled.

This specification is **methodological**, not philosophical.
It governs the organization, validation, and archival of reasoning artifacts so that every claim produced under AWO can be independently verified.

1.2 Scope

This document applies to all research workflows that:

- Integrate AI or automated reasoning systems as active participants in the research process.
- Produce verifiable artifacts such as manifests, runs, and audit logs.
- Intend for those artifacts to be **reproducible, falsifiable, and citable**.

It defines the **minimum structural and procedural requirements** for an AWO-compliant repository, including file hierarchy, provenance recording, versioning, and attestation rules.

AWO does **not** specify runtime behavior or enforcement mechanisms.
Those are defined in successor frameworks such as **CRI-CORE**, which must implement this specification as their normative foundation.

1.3 Objectives

The objectives of the AWO standard are to:

1. Encode the **scientific method** as a verifiable workflow rather than a descriptive ideal.
2. Replace subjective credibility with **objective auditability**.
3. Ensure that every reasoning artifact—data, model, or decision—can be traced to its origin.
4. Provide a foundation for automated reproducibility enforcement systems.
5. Support both manual and fully automated orchestration without altering compliance semantics.

1.4 Relationship to Other Documents

- The **AWO Whitepaper** provides conceptual background and philosophical rationale.

- The **AWO Adoption Guide** describes practical implementation and onboarding.
- This **Method Specification** defines the normative requirements that all AWO artifacts must satisfy.

Where discrepancies occur, **this specification takes precedence**.

1.5 Normative References

- **AWO Whitepaper v1.1** (Waveframe Labs)
- **Aurora Workflow Orchestration Adoption Guide v1.2.1**
- **Architecture Decision Records (ADR-0001 – ADR-0017)** — authoritative design decisions underlying AWO's structural, governance, and lifecycle model.
- **CRI-CORE Design Notes** (draft, forthcoming)
- **ISO/IEC Directives Part 2** — interpretation of compliance terms ("shall," "should," "may")

1.6 Status of This Version

Version 1.2.1 represents the **finalized methodological form** of AWO under Waveframe Labs governance.
Future revisions may clarify or extend definitions for CRI-CORE compatibility but will not alter the normative logic without an explicit version increment.

---

2. Definitions

This section defines the key entities and concepts used throughout the Aurora Workflow Orchestration (AWO) standard.
All terms are **normative** unless otherwise specified.

Wherever applicable, definitions align with terminology used in the AWO Whitepaper and will later be cross-referenced to CRI-CORE schema identifiers.

2.1 Core Entities

Run
A discrete, traceable research execution instance.
Each Run represents a bounded reasoning process that produces one or more verifiable artifacts and is identified by a unique timestamp or run ID.
All Runs must be immutable once attested.

Provenance
The complete, chronological lineage of data, logic, parameters, and decisions leading to a result.
Provenance includes all intermediate steps, transformations, and validations necessary to reproduce a Run.
In CRI-CORE, this concept maps to the `provenance-ledger` schema.

Artifact
Any persistent output generated within an AWO process.
Artifacts include reports, manifests, ADRs, checksums, datasets, logs, or schema validation results.
Artifacts must be versioned, hashable, and linkable to a Run.

Attestation
A confirmation—human, automated, or hybrid—that artifacts produced during a Run are complete, correct, and verified against defined falsifiability criteria.
Attestations are recorded in `approval.json` and form the evidentiary basis for repository integrity.

ADR (Architecture Decision Record)
A structured document that records a significant reasoning or design choice, the context in which it was made, and its consequences.
ADRs form the canonical log of epistemic evolution and are numbered sequentially (`ADR-0001` to `ADR-NNNN`).
Each Run must reference at least one ADR.

Manifest (Falsifiability Manifest)
A declaration of the hypothesis, predicted outcomes, and explicit disproof conditions for a Run.
The Manifest defines what constitutes falsification before execution.
It serves as the precondition for attestation and must be stored under `/docs/`.

2.2 Secondary Concepts

Repository
The complete version-controlled environment in which all AWO artifacts are stored.
Every AWO-compliant Repository must maintain a standard directory structure defined in Section 4.

Role
A functional agent—human or synthetic—responsible for a specific epistemic operation within the reasoning lifecycle (see Section 3).
Roles are procedural, not hierarchical.

Conformance
The degree to which an AWO repository satisfies all mandatory requirements defined in this specification.
Conformance is binary (pass/fail) for each clause but may include graded compliance levels ("Minimum," "Standard," "Full") as defined in the Adoption Guide.

Attestation Record
The recorded output of a completed review or validation step, stored as a structured file (`approval.json`) under the corresponding Run directory.
It includes participant identity, timestamp, and signature or digital hash.

2.3 Future Schema Alignment

All defined entities in this section will be mapped to corresponding CRI-CORE schema classes in later versions.
Cross-references will be introduced once the enforcement layer is finalized.

TODO: Refine definitions list and cross-link to CRI-CORE schema references after CRI draft publication.

---

3. Roles and Responsibilities

AWO distinguishes between procedural roles to ensure accountability and non-circular validation.

Primary Roles: - **Researcher:** Executes runs and maintains artifacts.
- **Maintainer:** Oversees repository integrity and version control.
- **Reviewer:** Performs verification and attestation of completed runs.

TODO: Add explicit permissions/responsibilities (who can sign approvals, tag releases, modify manifests).

---

4. Repository Requirements

Every AWO project MUST follow a consistent repository layout to ensure verifiability and interoperability.

Required Directories:

```
/docs/        → manifests, specs, reports
/decisions/   → ADRs (0001-NNNN)
/logs/        → timestamped workflow notes
/runs/        → attested run artifacts
/figures/     → diagrams, lifecycle visuals
```

TODO: Add detailed artifact rules and cross-link schema expectations.

---

5. Lifecycle and Run Phases

Each research cycle proceeds through four canonical phases:

1. **Fan-out (Planning)** — Define hypotheses, manifests, ADRs.
2. **Consensus (Execution)** — Perform runs and collect data.
3. **Attestation (Verification)** — Approve or reject based on falsifiability criteria.
4. **Archival (Publication)** — Freeze results, compute checksums, tag releases.

TODO: Create table describing inputs/outputs for each phase.

---

6. Artifacts and File Rules

Every run MUST produce a verifiable set of artifacts:

File	Description	Required
`workflow_frozen.json`	Captures executed parameters and inputs.	Yes
`report.md`	Describes outcomes, metrics, and observations.	Yes
`approval.json`	Signed validation record by human reviewer.	Yes
`SHA256SUMS.txt`	Hash registry for all outputs.	Yes
`manifest.json` or `manifest.md`	Defines falsifiability boundaries.	Yes

TODO: Add versioning, format validation (JSON schema references), and CRI-CORE integration hooks.

---

7. Compliance Language

This section defines the mandatory, recommended, and optional behaviors for implementers.

Level	Definition	Enforcement
MUST	Required for compliance.	Hard validation
SHOULD	Recommended unless documented exception.	Warning
MAY	Optional feature.	No enforcement

TODO: Map existing AWO clauses to each compliance level.

---

8. Governance and Attestation

Each run requires human or automated attestation of validity and completeness.

Core Requirements: - Runs MUST include `approval.json` with reviewer signature and timestamp.
- Attestation MAY include checksum verification and peer confirmation.
- Failed attestations MUST be logged under `/logs/attestation_failures/`.

TODO: Specify acceptable digital signature methods and verification workflows.

---

9. Release and Versioning

AWO-compliant repositories MUST version all outputs and preserve immutability.

Release Requirements: - Each release corresponds to a reproducible state of the repository.
- Tags MUST follow semantic versioning (e.g., `v1.2.1`).
- Releases MUST attach PDF artifacts, SHA256SUMS, and ADR references.
- Released runs MUST NOT be altered post-publication.

TODO: Add instructions for checksum regeneration and Zenodo linkage.

---

10. Licensing and Attribution

AWO uses dual licensing to separate executable and textual components.

- **Code:** Licensed under Apache 2.0.

- **Documentation:** Licensed under CC BY 4.0.
- Attribution MUST include author, ORCID, and concept DOI in derivative works.

TODO: Add structured attribution metadata schema reference.

---

11. Falsifiability Manifests

Each experiment MUST include a falsifiability manifest before execution.

Manifest Contents: - Hypothesis statement
- Predicted outcomes
- Disproof criteria
- Experimental plan
- Acceptance thresholds
- Known risks

TODO: Formalize manifest schema for CRI-CORE parsing.

---

12. Conformance Checklist

Each repository MUST pass the following before claiming AWO compliance:

☐ Standard directory structure present.

☐ At least one signed run in `/runs/`.

☐ ADRs and falsifiability manifests linked.

☐ SHA256SUMS.txt present at root.

☐ PDF artifacts built successfully.

☐ CHANGELOG includes version reference.

☐ README links to Whitepaper, Method Spec, Adoption Guide.

TODO: Add automated compliance script references (future CRI module).

---

13. Appendix C — Rationale Summary (Reserved)

TODO: When the Method Spec text is finalized, reintroduce Appendix C summarizing why each rule exists in concise bullet form.

(Placeholder retained for structural continuity.)

End of Specification — Aurora Workflow Orchestration (AWO) v1.2.1 Scaffold