

UNNS-ADM-A: Technical Realization of the UNNS Admissibility Framework

UNNS Research Collective

31 December 2025

Abstract

This document specifies UNNS-ADM-A, the technical realization layer supporting the UNNS Admissibility Framework (UNNS-ADM) and the Empirical Candidate Registry (UNNS-ADM-B). UNNS-ADM-A defines the minimal technical requirements, interfaces, and enforcement mechanisms necessary to implement admissibility recording and auditability in practice. It introduces no new admissibility criteria and performs no analytical evaluation. Its role is strictly infrastructural and enforcement-oriented.

1 Purpose and Scope

UNNS-ADM-A exists to realize, in technical form, the normative requirements defined by UNNS-ADM and UNNS-ADM-B. It provides infrastructure and support tools that ensure:

- admissibility records are complete, immutable, and auditable,
- admissibility documentation precedes instrumental analysis,
- registry semantics are enforced mechanically rather than by convention,
- historical integrity of admissibility decisions is preserved.

UNNS-ADM-A is not an analytical system and does not participate in scientific inference.

2 Normative Position

The admissibility stack is strictly ordered:

UNNS-ADM (theory) \rightarrow UNNS-ADM-B (registry specification) \rightarrow UNNS-ADM-A (technical realization) \rightarrow in

UNNS-ADM-A is subordinate to UNNS-ADM and UNNS-ADM-B and may not reinterpret, extend, or weaken their requirements.

3 Non-Goals

UNNS-ADM-A explicitly does not:

- determine or revise admissibility criteria,
- judge correctness of admissibility justifications,
- execute recursive generators or simulations,
- apply UNNS operators or chambers,
- rank, score, filter, or optimize candidate structures,
- infer significance, relevance, or observability.

Any system behavior beyond these boundaries constitutes scope violation.

4 Core System Components

UNNS-ADM-A shall minimally comprise the following logical components:

- **Registry Core:** immutable storage of UNNS-ADM-B entries,
- **Schema Validator:** enforcement of entry completeness and structure,
- **Identifier Authority:** generation and persistence of unique entry IDs,
- **Version Manager:** handling of registry and schema versions,
- **Reference Interface:** validation of admissibility references by downstream systems,
- **Audit Interface:** read-only inspection of registry history.

These components may be implemented jointly or separately, but their functional roles must remain distinct.

5 Registry Core Requirements

The registry core shall support:

- immutable storage of entries once accepted,
- permanent retention of inadmissible and superseded entries,
- explicit supersession links between entries,
- timestamping at entry creation time,
- association of each entry with a registry version identifier.

Deletion or modification of existing entries is prohibited.

6 Schema Enforcement

UNNS-ADM-A shall enforce the schema defined in UNNS-ADM-B at submission time.

- Entries missing required fields shall be rejected.
- Structural validation must occur prior to registry acceptance.
- No semantic evaluation of admissibility claims is permitted.

Schema enforcement is binary: pass or reject.

7 Identifier and Versioning System

7.1 Identifier Requirements

Each registry entry shall be assigned a globally unique, persistent identifier. Identifiers must:

- be stable across system migrations,
- never be reused,
- resolve to exactly one registry entry.

7.2 Registry Versioning

UNNS-ADM-A shall support registry versions of the form:

UNNS-ADM-B.vX.Y

where:

- X denotes backward-incompatible changes,
- Y denotes backward-compatible extensions or clarifications.

Entries created under older versions must remain interpretable.

8 Admissibility Documentation Handling

UNNS-ADM-A shall support structured storage of:

- discovery protocols and A6 compliance documentation,
- axiom-by-axiom admissibility justifications,
- verifier identification and verification methods,
- marginality annotations and sensitivity analyses.

UNNS-ADM-A shall verify presence and structure only, not correctness.

9 Reference and Enforcement Interfaces

9.1 Admissibility Reference Validation

UNNS-ADM-A shall expose an interface enabling downstream systems to:

- verify existence of a registry entry ID,
- verify admissibility status,
- verify that the registry entry predates the analysis referencing it.

References to inadmissible or non-existent entries must be rejected.

9.2 Compliance Validation

UNNS-ADM-A shall provide a compliance check that determines whether a given analysis artifact satisfies UNNS-ADM-B enforcement requirements, without inspecting analytical content.

10 Audit and Traceability

UNNS-ADM-A shall provide read-only audit capabilities, including:

- retrieval of any registry entry by ID,
- traversal of supersession chains,
- inspection of timestamps and version metadata,
- verification of immutability guarantees.

Audit access must not permit modification.

11 Access Control

UNNS-ADM-A shall implement access control consistent with UNNS-ADM-B:

- public read access,
- restricted write access for authorized submitters,
- no modification privileges.

Authorization events must be logged.

12 Archival and Integrity Guarantees

UNNS-ADM-A shall support:

- persistent identifier resolution (e.g. DOI or equivalent),
- redundant storage across independent locations,
- integrity verification using cryptographic checksums,
- controlled format migration for long-term preservation.

13 Failure Semantics

Technical failures (e.g. submission rejection, storage error, network failure) must be explicitly distinguished from admissibility failure.

No technical failure may be recorded or interpreted as an admissibility verdict.

14 Non-Guarantees

UNNS-ADM-A does not guarantee:

- correctness of admissibility assessments,
- scientific validity of registered structures,
- success or relevance of downstream analyses.

15 Conclusion

UNNS-ADM-A provides the minimal and sufficient technical infrastructure required to realize the UNNS Admissibility Framework in practice. By enforcing immutability, traceability, and temporal precedence without introducing interpretive authority, it preserves the methodological guarantees established by UNNS-ADM and UNNS-ADM-B.