

# 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 of the UNNS Admissibility Framework (UNNS-ADM). UNNS-ADM-A provides implementation requirements and support tools necessary to realize UNNS-ADM and UNNS-ADM-B in practice. It does not define admissibility criteria, evaluate structures, or perform analytical operations. Its role is strictly infrastructural: to enforce, record, and preserve admissibility decisions without introducing new semantics or interpretive power.

## 1 Scope and Role

UNNS-ADM-A defines the minimal technical capabilities required to support:

- creation of UNNS-ADM-B registry entries,
- enforcement of immutability and versioning rules,
- verification of admissibility documentation completeness,
- traceable linkage between registry entries and downstream analyses.

UNNS-ADM-A is subordinate to UNNS-ADM (theory) and implements the normative specification defined in UNNS-ADM-B.

## 2 Normative Position

The admissibility stack is ordered as follows:

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

UNNS-ADM-A may not introduce criteria, heuristics, thresholds, or decisions not explicitly defined in UNNS-ADM or UNNS-ADM-B.

## 3 Non-Goals

UNNS-ADM-A explicitly does not:

- determine admissibility or marginality,

- execute recursive generators,
- apply UNNS operators or chambers,
- rank, score, or prioritize candidates,
- infer or suggest interpretations.

Any such functionality constitutes a violation of scope.

## 4 Core Functional Requirements

### 4.1 Registry Infrastructure

UNNS-ADM-A shall provide a registry backend capable of:

- storing immutable registry entries,
- assigning globally unique persistent identifiers,
- recording timestamps and version metadata,
- linking superseding and superseded entries,
- preserving full entry history.

### 4.2 Schema Enforcement

UNNS-ADM-A shall enforce schema completeness as defined in UNNS-ADM-B. Entries missing required fields shall be rejected at submission time without evaluating admissibility content.

### 4.3 Immutability Enforcement

Once recorded, a registry entry must not be modified or deleted. Corrections or updates shall be realized exclusively by:

new entry  $\rightarrow$  supersedes prior entry.

### 4.4 Version Control

UNNS-ADM-A shall support:

- registry version identifiers (UNNS-ADM-B.vX.Y),
- backward-compatible and incompatible schema evolution,
- coexistence of entries created under different registry versions.

## 5 Admissibility Documentation Support

UNNS-ADM-A shall support structured documentation for:

- discovery protocols (A6 compliance),
- admissibility justifications (A1–A7),
- verifier identification and verification methods,
- marginality annotations and sensitivity analyses.

UNNS-ADM-A must not evaluate the correctness of justifications, only their presence and structural completeness.

## 6 Enforcement Interfaces

### 6.1 Instrumental Reference Interface

UNNS-ADM-A shall provide a reference mechanism by which instrumental analyses can:

- cite a registry entry ID,
- verify admissibility status,
- verify temporal precedence of registry entry creation.

UNNS-ADM-A may reject references to inadmissible or non-existent entries.

### 6.2 Compliance Validation

UNNS-ADM-A shall expose a compliance validation function that checks whether a given analysis output satisfies the enforcement protocol defined in UNNS-ADM-B, without inspecting analytical content.

## 7 Audit and Traceability

UNNS-ADM-A shall provide audit facilities enabling:

- retrieval of full registry history,
- inspection of supersession chains,
- verification of timestamps and version metadata,
- detection of post hoc modifications.

Audit functionality must be read-only.

## 8 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 access.

Authorization mechanisms must be auditable.

## 9 Archival Guarantees

UNNS-ADM-A shall support:

- persistent identifier resolution,
- redundant storage across independent locations,
- integrity verification (checksums),
- format migration for long-term preservation.

## 10 Failure Semantics

Technical failures of UNNS-ADM-A (e.g. storage errors, submission rejection) must be clearly distinguished from admissibility failure. No technical failure may be recorded as an admissibility verdict.

## 11 Non-Guarantees

UNNS-ADM-A does not guarantee:

- correctness of admissibility judgments,
- scientific validity of registered structures,
- success of downstream analyses.

## 12 Conclusion

UNNS-ADM-A provides the minimal technical infrastructure necessary to realize the UNNS Admissibility Framework without introducing interpretive authority. By strictly separating enforcement from evaluation, it preserves the methodological guarantees established by UNNS-ADM and UNNS-ADM-B.