Audit Trail Process for the Agnosia Knowledge Graph

Technical Specification (v0.1)

Objective. Build a knowledge graph that produces decision-grade claims with a scalable audit trail by (1) using free/low-cost secondary sources to discover and index evidence, (2) clustering supporting evidence into *independent primary clusters*, (3) promoting claims through evidence tiers, and (4) continuously calibrating confidence using human review.

Design promise. Users generally do not need to read primary sources. They need structural credibility: independent evidence clusters, scope, recency, contradiction awareness, and a path to deeper verification when required.

1. Ontology framing

Every claim is categorized under an upper-ontology output type. This influences claim templates, evidence expectations, and what “auditability” means in the UI.

Upper ontology types

Entities: stable things (objects, categories, physical or institutional entities).

Relations: formal relationships (equivalence, implication, causality, dependency, correlation).

Events/Processes: time-indexed change (historical events, procedures, policy shifts, economic dynamics).

Domain map

Key	Label	Under upper ontology
mathematics	Mathematics & Computational Sciences	relations
natural_sciences	Natural Sciences	entities
engineering	Engineering & Applied Sciences	relations
social_sciences	Social Sciences & Human Behavior	entities
health_medicine	Health & Medicine	entities
philosophy_religion	Philosophy, Religion & Ethics	relations
history	History & Cultural Studies	events_processes
business_economics	Business, Economics & Law	events_processes
languages_literature	Languages & Literature	events_processes
arts	Arts, Music & Performance	events_processes
vocational	Applied & Vocational Skills	events_processes
interdisciplinary	Interdisciplinary & Emerging Fields	events_processes

2. What “audit trail” means

An audit trail is the information required to reproduce and defend a claim’s support status. The system is designed so that most users can trust a claim without reading primaries, while auditors and high-stakes users can validate the evidence standard and process.

Audit trail must provide: (a) trust calibration, (b) reproducibility, and (c) accountability.

3. Claim tiers and guarantees

Every claim is assigned a tier based on evidence quality, independence, scope completeness, recency, and contradiction handling.

Provisional: derived from secondary convergence and metadata-level evidence mapping; confidence is capped; best for exploration.

Supported: meets independence and scope requirements using primary identifiers and credible evidence types.

Audited: meets strict evidence floors and has a reproducible verification record (human or controlled verifier); required for high-stakes claim classes.

4. Core data model (conceptual)

Claim: claim_id, domain_key, upper_ontology_type, claim_type, claim_text, scope, confidence_tier, p_error, contradiction_status, last_verified_at, evidence_summary.

Source: source_id, source_type, identifiers (DOI/PMID/arXiv/NCT/CIK+Accession/etc.), license_flags, issuer, timestamp/version.

Evidence edge: source → (supports|contradicts|qualifies|defines) → claim, plus extraction_method and scope_alignment_score.

Independence cluster: grouping of sources with shared upstream origin (trial, dataset series, statute version, filing accession, etc.).

5. Methodology: from secondary sources to independent primary clusters

Secondary sources are used to discover candidate claims and enumerate primary references with stable identifiers. Independence clustering prevents “echo chamber confidence” by collapsing derivative references into upstream-origin groups.

Workflow

• Collect candidate references from secondary sources.

• Canonicalize primary identifiers (DOI/PMID/arXiv/NCT/etc.).

• Expand a citation-graph neighborhood (references, co-citations) where available.

• Cluster primaries by upstream origin (shared trial/dataset/statute version/filing accession; plus citation overlap signals).

• Search for high-quality counterevidence and attach contradiction edges.

• Compute effective evidence strength and assign tier.

6. Scoring and equations (adaptive via human review)

We model confidence via estimated probability of error. Parameters are learned per (domain, claim_type) and updated as humans find errors.

6.1 Probability-of-error model

```
P(error | c) = σ( β0 − βP·EP(c) − βS·ES(c) + βK·K(c) + βT·T(c) + βA·A(c) )
```

σ(x)=1/(1+e^(−x)); EP and ES are effective primary/secondary evidence; K is contradiction; T is time-sensitivity; A is scope mismatch.

Confidence is Conf(c) = 1 − P(error | c).

6.2 Effective evidence (independence-weighted)

```
EP(c) = Σ_i ( w_i^P · q_i^P · r_i^P · m_i^P ) over primary clusters

ES(c) = Σ_j ( w_j^S · q_j^S · r_j^S · m_j^S ) over secondary clusters
```

w = independence weight; q = quality; r = recency; m = scope match.

6.3 Primary-to-secondary exchange rate

```
1 unit of primary evidence ≈ (βP / βS) units of secondary evidence (within a specific domain and claim type).
```

6.4 Confidence caps and promotion thresholds

```
Conf(c) ≤ C_sec(domain, claim_type) unless EP(c) ≥ τP(domain, claim_type).
```

7. Human review loop

Human review is the control system that makes confidence meaningful and corrects failure modes.

7.1 Sampling strategy (active learning)

```
Priority(c) = Impact(c) × P(error | c) × Uncertainty(c)
```

7.2 Review outputs

Reviewers label claims (correct/incorrect/ambiguous) and tag failure modes (scope error, staleness, non-independence, extraction error, missed contradiction, etc.).

7.3 What gets updated

Weights (β), source quality priors (q), independence heuristics (w), staleness decay (r), scope matching (m), and policy caps/floors (C_sec, τP).

8. Initial secondary sources and access guide

These sources bootstrap the pipeline and enable independence clustering across domains without proprietary licensing.

Source	What we use it for	How to attain access to start pipeline	Auth/key	Cost
OpenAlex	Scholarly works metadata + citations backbone	Use public API or download snapshot	No	Free
OpenCitations (COCI)	Open DOI-to-DOI citation edges	Use public REST API	No	Free
Crossref	DOI resolution + metadata normalization	Use Crossref REST API	No	Free
Wikipedia dumps	Topic routing + citation harvesting	Download Wikimedia dumps	No	Free
Europe PMC	Biomed metadata + links	Use Europe PMC REST API	No	Free
PubMed (E-utilities)	Biomed indexing (PMIDs/abstracts)	Use NCBI Entrez E-utilities	No	Free
ClinicalTrials.gov	Trial registry IDs for clustering	Use CTG v2 API	No	Free
PMC OA subset / datasets	OA full text where licenses permit	Use PMC OA dataset access (FTP/cloud)	No	Free
arXiv	Preprint metadata (math/CS/physics)	Use arXiv API and/or OAI-PMH	No	Free
NASA ADS	Astro/physics index + citations	Create account and generate token	Yes	Free
IETF / RFC Editor	Standards + dependency chains	Pull rfc-index.xml and RFC texts	No	Free
W3C specs (GitHub)	Web standards metadata + specs	Use W3C GitHub org repos/lists	No	Free
SEC EDGAR	Filings metadata + XBRL endpoints	Use SEC EDGAR APIs	No	Free
FRED	Macro data series + provenance	Request FRED API key	Yes	Free
Congress.gov	Legislative data	Request Congress.gov key via api.data.gov	Yes	Free
FederalRegister.gov	Regulatory notices/rules metadata	Use Federal Register REST API	No	Free
CourtListener	Opinions/dockets metadata	Use CourtListener API/bulk data	Varies	Free/low-cost
Open Library	Book/work/edition metadata	Use Open Library APIs or dumps	No	Free

9. Locating domain-specific secondary sources (ongoing process)

We continuously add secondary sources per domain. The selection criterion is: citation hygiene + stable identifiers + structured access.

9.1 Selection checklist

• Explicit primary references (not just narrative).

• Stable identifiers (DOI/PMID/arXiv/NCT/standard IDs/filing IDs).

• Structured access (API, data dump, OAI-PMH) or consistent machine-readable pages.

• Licensing clarity (what can be stored vs link-only).

• Authority signals (curation, editorial controls, institutional backing).

• Update cadence appropriate for the domain.

9.2 Discovery playbook

• Identify the domain's indexing bodies (registries, bibliographic databases, standards bodies, statistical agencies).

• Search for "API", "data dump", "OAI-PMH", "open citations", and "metadata" endpoints.

• Prefer sources that emit primary identifiers over those that only provide prose references.

• Define for each source: emitted IDs, what counts as a primary, and independence clustering rules.

10. Pipeline summary

• Ingest secondary sources (metadata, references, identifiers).

• Extract candidate claims using domain-aware templates.

• Attach evidence edges (supports/contradicts/qualifies/defines).

• Canonicalize and deduplicate sources via stable identifiers.

• Form independence clusters and compute effective evidence.

• Score P(error) and assign tier with caps/floors.

• Publish claim cards (scope, tier, cluster summary, recency, contradictions).

• Run the human review loop and recompute.