

The AUGMANITAI Concept Registry: A Public, Versioned, Author-Attributed Concept Registry - Reference Implementation of the Universal Concept Layer (v0.1, September 2026)

Andreas Ehstand

Independent Researcher

ORCID: 0009-0006-3773-7796

License: CC BY 4.0

Prepared with AI assistance; content reviewed and approved by the author.

Research working paper. No legal, medical, financial or other advice.

Part I — English

1. Problem

Agent ecosystems are increasingly standardizing transport and discovery. They are becoming better at moving messages, exposing capabilities, locating resources and connecting tools across technical boundaries. These mechanisms answer operational questions: where a resource is, how it can be reached and which request retrieves it. This is necessary infrastructure, but it does not by itself establish what the terms carried through that infrastructure mean.

The meaning layer remains comparatively unstructured. There is no public, versioned and organization-spanning infrastructure that consistently records what a term means, how its meaning changes and who coined or formulated it. Definitions may exist in prose, metadata or local documentation, yet their identity, provenance and revision history are rarely handled as one retrievable object. A term can therefore circulate widely while the conditions needed to interpret it remain scattered.

The consequences are meaning drift, audit gaps and disputes about origin. Meaning drift occurs when identical labels acquire incompatible interpretations without a visible version boundary. Audit gaps arise when a person or agent can retrieve a term but cannot reconstruct the definition, attribution or evidence used at the time of a decision. Disputes about origin

become harder to assess when authorship, first formulation and evidence are detached from the definition. These are governance problems at the level of language, not merely failures of transport.

2. Design Principles

The AUGMANITAI Concept Registry treats terms as versioned building blocks. A concept record is not only a label; it is a compact unit containing identity, bilingual wording, definition, version, status, authorship, dates, evidence anchors, retrieval addresses and license information. This makes the term addressable as an object whose current state can be inspected by a person or a machine.

Semantic versioning describes changes in meaning. A version has the form `MAJOR.MINOR.PATCH`. `MAJOR` marks a break in meaning: users must not assume that the revised concept is interchangeable with the preceding major version. `MINOR` marks a refinement that makes the meaning more precise without replacing its core. `PATCH` marks a wording change that preserves the intended meaning. The rule turns revision into explicit information instead of leaving readers to infer whether a changed sentence also changed the concept.

Stable retrieval addresses support ordinary reads. The registry uses web addresses that can be retrieved with `GET`. The registry index, schema and individual concept records are separately addressable. A stable slug keeps retrieval independent from presentation labels in either language. This is intentionally a small interface: a consumer can fetch the index, follow a concept URL and validate the returned record without requiring a special browsing session.

Author coupling is mandatory. The `coined_by` field binds attribution to the concept record rather than leaving it to surrounding prose. In `v0.1`, authorship is therefore part of the minimum record, alongside the date of first formulation. This coupling does not ask a registry consumer to settle every possible historical claim automatically; it ensures that the claim being made is explicit, transportable and open to inspection with its supporting anchors.

Every concept carries evidence anchors. The `anchors` field connects a concept to dated evidence references. Each anchor contains a type, a reference and a date, so provenance is represented as structured data rather than an untraceable note. The field is required by the `v0.1` schema. The schema does not yet specify `minItems`, however, and thus does not technically reject an empty anchor array. The design principle is one or more anchors per concept; closing the validation gap belongs in the conformance work toward `v1.0`.

Human and machine views express the same content. The human registry page explains the model and lists the six concepts, while the index and concept files expose structured JSON. A `human_page` and a `canonical_url` connect records to readable context. The two representations need not have identical syntax, but their identities, versions, definitions and attribution should remain aligned. A discrepancy between the readable and structured forms is consequently a synchronization defect, not an alternative interpretation.

3. The Data Model

The v0.1 schema is a closed object: `additionalProperties` is `false`, and all listed record fields are required. This makes unexpected fields detectable and gives producers and consumers a shared minimum shape.

- `id` is the non-empty stable identifier of the concept. In the reference record it equals the slug, but the schema does not require those strings to be identical.
- `slug` is the address-friendly name. Its pattern permits lower-case letters, digits and single hyphen-separated segments.
- `term_de` and `term_en` hold the non-empty German and English labels. Keeping both in one record prevents language variants from becoming disconnected entries.
- `version` is the semantic version string. The schema enforces three non-negative integer components and documents the meaning of MAJOR, MINOR and PATCH.
- `status` records the concept's declared lifecycle state as a non-empty string. The schema leaves the status vocabulary open; the example uses `stable`.
- `coined_by` records the attributed author as a non-empty string. It is the mandatory author coupling in the record.
- `first_formulated` gives the earliest claimed formulation date at the precision supported by the source: year, year-month or full date.
- `definition_de` and `definition_en` contain the non-empty definitions in German and English. In the reference implementation these are concise, one-sentence formulations.
- `anchors` is an array of evidence objects. Each object is closed and requires non-empty `type`, `ref` and `date` fields. The date is intentionally a string, allowing the source's original date expression to be retained.
- `canonical_url` identifies the concept's canonical public context, while `human_page` points to a human-readable page. Both must have URI form.
- `license` states the reuse terms for the record's definition and metadata.
- `updated` records the record-level update date and must use a full calendar-date format.

The following figure reproduces the supplied reference record for *Gedankenvererbungslehre / Thought Inheritance*. It shows the complete v0.1 object, including multiple evidence types and date precisions.

```
{
  "id": "gedankenvererbungslehre",
  "slug": "gedankenvererbungslehre",
  "term_de": "Gedankenvererbungslehre",
  "term_en": "Thought Inheritance",
  "version": "1.0.0",
  "status": "stable",
  "coined_by": "Andreas Ehstand",
  "first_formulated": "2026-02",
  "definition_de": "Rahmenwerk für die bewusste Dokumentation und Übertragung der Denk-  
Struktur eines Menschen – Wissen, Entscheidungs-Muster, gedankliche Rahmen –  
über die eigene Lebenszeit hinaus.",
```

```

"definition_en": "A framework for the conscious documentation and transfer of an
    individual's cognitive structure – knowledge, decision heuristics, intellectual
    frameworks – beyond their lifetime.",
"anchors": [
  {
    "type": "archive",
    "ref": "https://doi.org/10.5281/zenodo.22211952",
    "date": "21.02.2026"
  },
  {
    "type": "internet-archive",
    "ref": "Schnappschüsse des Internet-Archivs (neutraler Dritter), samt Prüfsummen",
    "date": "20. Februar 2026"
  },
  {
    "type": "internet-archive",
    "ref": "Schnappschüsse des Internet-Archivs (neutraler Dritter), samt Prüfsummen",
    "date": "27. Februar 2026"
  },
  {
    "type": "bitcoin-blockchain",
    "ref": "OpenTimestamps",
    "date": "20.-22. Februar 2026"
  }
],
"canonical_url": "https://augmanitai.com/prioritaet",
"human_page": "https://doi.org/10.5281/zenodo.22211952",
"license": "CC BY 4.0 - one-sentence definitions and metadata, consistent with the
    published Priority and Attribution Register (Ehstand, 2026)",
"updated": "2026-09-02"
}

```

The example also clarifies the distinction between several dates. `first_formulated` records the claimed origin at the precision available from the source. Anchor dates record the dates associated with particular evidence items and may preserve localized formatting or a range. `updated` records when the JSON concept record itself was updated. These fields should not be collapsed because they answer different audit questions.

4. The Reference Implementation

The reference implementation has been live at <https://augmanitai.com/registry/> since 2 September 2026 and publishes six concept records, all listed as version 1.0.0 in the supplied index:

- *Gedankenvererbungslehre / Thought Inheritance*:
`https://augmanitai.com/registry/gedankenvererbungslehre.json`
- *The Fifth Stage / Architectural Cognitive Externalization*:
`https://augmanitai.com/registry/fifth-stage.json`
- *Cognitive Legacy Documentation*: `https://augmanitai.com/registry/cognitive-legacy-documentation.json`

- *Machine Coaching*: <https://augmanitai.com/registry/machine-coaching.json>
- *Universal Concept Layer*: <https://augmanitai.com/registry/universal-concept-layer.json>
- *Ehstand-Test*: <https://augmanitai.com/registry/ehstand-test.json>

Two collection-level addresses complete the simple retrieval surface:

<https://augmanitai.com/registry/index.json> is the machine-readable directory, and <https://augmanitai.com/registry/schema.json> is the validation schema. The directory names the registry, the Universal Concept Layer v0.1 specification, the maintainer and ORCID, the generation date, license, schema address, human page and the six concept summaries. A consumer can therefore begin with the collection and proceed to an individual record through the published URL.

The human index carries structured `DefinedTermSet` markup. Its `hasDefinedTerm` collection contains the same six entries as `DefinedTerm` objects, with names, alternate names, descriptions and JSON URLs. This makes the page simultaneously a readable overview and a structured declaration that the records belong to one term set. It also creates a concrete consistency requirement: the table, the structured markup and `index.json` should describe the same registry membership.

The site's `llms.txt` supplies the wider machine-facing context. It describes AUGMANITAI as an independent research practice, identifies the public lexicon and its machine-readable data, links the priority and canonical-terms page, and points to the Universal Concept Layer paper and its plain-text versions. Since 2 September 2026 it also lists the registry directly: the human registry page, the machine-readable index and the record schema each appear as dedicated addresses, so a machine reader can reach the registry from the site's central machine-facing entry point.

5. What the Registry Is Not

- It is not a data catalogue: it registers meaning-bearing concepts and provenance, not collections of datasets or their storage locations.
- It is not a metrics semantic layer: it versions definitions of concepts, not business measures, calculation rules or reporting dimensions.
- It is not a company ontology: it does not model the entities, processes and internal relationships of a particular organization.

6. Open Questions Toward v1.0

Third-party namespaces. A public, organization-spanning layer needs a way for third parties to publish identifiers without collisions or implied ownership by the reference maintainer. Namespace syntax, delegation, transfer and persistence rules need to be defined while keeping concept URLs readable.

Resolution interface. GET retrieval works for known addresses, but a broader layer needs a defined way to resolve an identifier, language label or version request to the relevant record. The interface must distinguish an exact version from the latest compatible version and must define responses for unknown, moved or superseded concepts.

Conformance tests. The JSON schema establishes a structural baseline, but conformance also includes cross-file conditions: a listed concept should resolve, its identity and version should match the directory, human and machine definitions should remain aligned, and the evidence-anchor principle should be enforced. Test fixtures can make these expectations reproducible.

Connection to agent toolchains. Agent workflows need a predictable point at which concept resolution occurs: when reading an instruction, selecting a tool, recording a decision or producing an audit trail. Integration should preserve the exact concept identifier and version that informed an action, rather than copying only a label whose meaning may later change.

7. References

1. Andreas Ehstand, *The Universal Concept Layer: Language as Versionable Code — A Meaning Layer between Humans, Language Models, Agents, and Robots*: <https://doi.org/10.5281/zenodo.22172128>
 2. AUGMANITAI Concept Registry — *Universal Concept Layer v0.1 Reference Implementation*: <https://augmanitai.com/registry/>
 3. *Priority & Canonical Terms*: <https://augmanitai.com/prioritaet>
-

Teil II — Deutsch

1. Problem

Agenten-Ökosysteme standardisieren zunehmend Transport und Auffinden. Sie werden besser darin, Nachrichten zu übertragen, Fähigkeiten sichtbar zu machen, Ressourcen zu lokalisieren und Werkzeuge über technische Grenzen hinweg zu verbinden. Diese Mechanismen beantworten operative Fragen: Wo befindet sich eine Ressource, wie lässt sie sich erreichen und mit welcher Anfrage wird sie abgerufen? Das ist notwendige Infrastruktur, legt aber noch nicht fest, was die darin transportierten Begriffe bedeuten.

Die Bedeutungs-Ebene bleibt demgegenüber wenig strukturiert. Es gibt keine öffentliche, versionierte und organisationsübergreifende Infrastruktur, die durchgängig festhält, was ein Begriff meint, wie sich seine Bedeutung verändert und wer ihn geprägt oder formuliert hat. Definitionen können in Fließtext, Metadaten oder lokaler Dokumentation vorhanden sein; ihre

Identität, Herkunft und Änderungsgeschichte werden jedoch selten als ein gemeinsam abrufbares Objekt behandelt. So kann ein Begriff weite Verbreitung finden, während die Voraussetzungen für seine Auslegung verstreut bleiben.

Die Folgen sind Bedeutungs-Drift, Audit-Lücken und Streit um Herkunft. Bedeutungs-Drift entsteht, wenn gleiche Bezeichnungen unvereinbare Auslegungen annehmen, ohne dass eine sichtbare Versionsgrenze besteht. Audit-Lücken entstehen, wenn ein Mensch oder Agent zwar einen Begriff abrufen kann, aber die bei einer Entscheidung verwendete Definition, Zuschreibung oder Beweislage nicht mehr rekonstruieren kann. Streit um Herkunft lässt sich schwerer prüfen, wenn Urheberschaft, Erst-Formulierung und Belege von der Definition getrennt sind. Es handelt sich um Governance-Probleme auf der Ebene der Sprache, nicht nur um Fehler beim Transport.

2. Design-Prinzipien

Das AUGMANITAI Concept Registry behandelt Begriffe als versionierte Bausteine. Ein Begriffs-Datensatz ist nicht nur eine Bezeichnung, sondern eine kompakte Einheit aus Identität, zweisprachiger Benennung, Definition, Version, Status, Urheberschaft, Datumsangaben, Beweis-Ankern, Abruf-Adressen und Lizenzinformation. Dadurch wird der Begriff zu einem adressierbaren Objekt, dessen aktueller Stand von Menschen und Maschinen geprüft werden kann.

Semantische Versionierung beschreibt Änderungen der Bedeutung. Eine Version folgt dem Muster `MAJOR.MINOR.PATCH`. `MAJOR` bezeichnet einen Bedeutungsbruch: Nutzer dürfen die überarbeitete Fassung nicht ohne Weiteres als austauschbar mit der vorherigen Hauptversion behandeln. `MINOR` bezeichnet eine Präzisierung, die die Bedeutung genauer fasst, ohne ihren Kern zu ersetzen. `PATCH` bezeichnet eine Änderung der Formulierung bei gleichbleibender beabsichtigter Bedeutung. Die Regel macht Revision zur ausdrücklichen Information, statt Lesern die Vermutung zu überlassen, ob ein geänderter Satz auch den Begriff verändert hat.

Stabile Abruf-Adressen unterstützen gewöhnliche Lesezugriffe. Das Register nutzt Web-Adressen, die per `GET` abrufbar sind. Register-Index, Schema und einzelne Begriffs-Datensätze sind getrennt adressierbar. Ein stabiler Slug entkoppelt den Abruf von der Darstellungs-Bezeichnung in beiden Sprachen. Die Schnittstelle bleibt bewusst klein: Ein Verbraucher kann den Index laden, einer Begriffs-Adresse folgen und den zurückgegebenen Datensatz prüfen, ohne eine besondere Browser-Sitzung zu benötigen.

Die Autor-Kopplung ist verpflichtend. Das Feld `coined_by` bindet die Zuschreibung an den Begriffs-Datensatz, statt sie einem Begleittext zu überlassen. In `v0.1` gehört die Urheberschaft damit zusammen mit dem Datum der Erst-Formulierung zum Mindestbestand jedes Datensatzes. Diese Kopplung verlangt von einem Register-Nutzer nicht, jede denkbare historische Behauptung automatisch zu entscheiden; sie sorgt dafür, dass die aufgestellte Behauptung ausdrücklich, transportierbar und zusammen mit ihren Beweis-Ankern prüfbar ist.

Jeder Begriff trägt Beweis-Anker. Das Feld `anchors` verbindet einen Begriff mit datierten Beleg-Referenzen. Jeder Anker enthält einen Typ, eine Referenz und ein Datum, sodass Herkunft als strukturierte Angabe und nicht als unauffindbare Notiz erscheint. Das Feld ist nach dem v0.1-Schema verpflichtend. Das Schema enthält allerdings noch kein `minItems` und weist deshalb ein leeres Anker-Array technisch nicht zurück. Das Design-Prinzip lautet ein oder mehrere Anker je Begriff; die Validierungs-Lücke zu schließen, gehört zur Konformitätsarbeit in Richtung v1.0.

Menschen- und Maschinen-Fassung drücken denselben Inhalt aus. Die menschlich lesbare Register-Seite erklärt das Modell und führt die sechs Begriffe auf; Index und Begriffs-Dateien stellen strukturierte JSON-Daten bereit. `human_page` und `canonical_url` verbinden Datensätze mit lesbarem Kontext. Beide Fassungen müssen nicht dieselbe Syntax haben, doch Identitäten, Versionen, Definitionen und Zuschreibungen sollen übereinstimmen. Eine Abweichung zwischen lesbarer und strukturierter Fassung ist folglich ein Synchronisationsfehler, keine alternative Auslegung.

3. Das Datenmodell

Das v0.1-Schema beschreibt ein geschlossenes Objekt: `additionalProperties` ist `false`, und alle aufgeführten Datensatz-Felder sind verpflichtend. Damit werden unerwartete Felder erkennbar, und Erzeuger wie Verbraucher erhalten eine gemeinsame Mindestform.

- `id` ist die nichtleere stabile Kennung des Begriffs. Im Referenz-Datensatz entspricht sie dem Slug; das Schema verlangt diese Gleichheit jedoch nicht.
- `slug` ist der adressfreundliche Name. Sein Muster erlaubt Kleinbuchstaben, Ziffern und durch einzelne Bindestriche getrennte Segmente.
- `term_de` und `term_en` enthalten die nichtleeren deutschen und englischen Bezeichnungen. Beide im selben Datensatz zu führen verhindert, dass Sprachvarianten zu voneinander getrennten Einträgen werden.
- `version` ist die semantische Versions-Zeichenfolge. Das Schema erzwingt drei nichtnegative ganzzahlige Bestandteile und dokumentiert die Bedeutung von MAJOR, MINOR und PATCH.
- `status` hält den erklärten Lebenszyklus-Stand als nichtleere Zeichenfolge fest. Das Schema lässt das Status-Vokabular offen; das Beispiel verwendet `stable`.
- `coined_by` nennt den zugeschriebenen Autor als nichtleere Zeichenfolge. Das Feld bildet die verpflichtende Autor-Kopplung im Datensatz.
- `first_formulated` gibt das beanspruchte Datum der frühesten Formulierung in der von der Quelle gestützten Genauigkeit an: Jahr, Jahr-Monat oder vollständiges Datum.
- `definition_de` und `definition_en` enthalten die nichtleeren Definitionen auf Deutsch und Englisch. In der Referenz-Umsetzung sind dies knappe Ein-Satz-Formulierungen.
- `anchors` ist eine Liste von Beweis-Objekten. Jedes Objekt ist geschlossen und verlangt die nichtleeren Felder `type`, `ref` und `date`. Das Datum bleibt bewusst eine Zeichenfolge, damit die ursprüngliche Datumsangabe der Quelle erhalten werden kann.
- `canonical_url` bezeichnet den kanonischen öffentlichen Kontext des Begriffs; `human_page` verweist auf eine menschlich lesbare Seite. Beide Angaben müssen die Form einer URI

haben.

- `license` nennt die Bedingungen für die Weiterverwendung von Definition und Metadaten.
- `updated` hält das Aktualisierungsdatum des Datensatzes fest und muss das Format eines vollständigen Kalenderdatums verwenden.

Die folgende Abbildung gibt den bereitgestellten Referenz-Datensatz für *Gedankenvererbungslehre* / *Thought Inheritance* wieder. Sie zeigt das vollständige vo.1-Objekt einschließlich mehrerer Belegtypen und Datumsgenauigkeiten.

```
{
  "id": "gedankenvererbungslehre",
  "slug": "gedankenvererbungslehre",
  "term_de": "Gedankenvererbungslehre",
  "term_en": "Thought Inheritance",
  "version": "1.0.0",
  "status": "stable",
  "coined_by": "Andreas Ehstand",
  "first_formulated": "2026-02",
  "definition_de": "Rahmenwerk für die bewusste Dokumentation und Übertragung der Denk-  
Struktur eines Menschen – Wissen, Entscheidungs-Muster, gedankliche Rahmen –  
über die eigene Lebenszeit hinaus.",
  "definition_en": "A framework for the conscious documentation and transfer of an  
individual's cognitive structure – knowledge, decision heuristics, intellectual  
frameworks – beyond their lifetime.",
  "anchors": [
    {
      "type": "archive",
      "ref": "https://doi.org/10.5281/zenodo.22211952",
      "date": "21.02.2026"
    },
    {
      "type": "internet-archive",
      "ref": "Schnappschüsse des Internet-Archivs (neutraler Dritter), samt Prüfsummen",
      "date": "20. Februar 2026"
    },
    {
      "type": "internet-archive",
      "ref": "Schnappschüsse des Internet-Archivs (neutraler Dritter), samt Prüfsummen",
      "date": "27. Februar 2026"
    },
    {
      "type": "bitcoin-blockchain",
      "ref": "OpenTimestamps",
      "date": "20.-22. Februar 2026"
    }
  ],
  "canonical_url": "https://augmanitai.com/prioritaet",
  "human_page": "https://doi.org/10.5281/zenodo.22211952",
  "license": "CC BY 4.0 - one-sentence definitions and metadata, consistent with the  
published Priority and Attribution Register (Ehstand, 2026)",
  "updated": "2026-09-02"
}
```

Das Beispiel verdeutlicht außerdem den Unterschied zwischen mehreren Datumsangaben. `first_formulated` hält den beanspruchten Ursprung in der aus der Quelle verfügbaren Genauigkeit fest. Anker-Daten bezeichnen die zu einzelnen Belegen gehörenden Zeitangaben und können eine lokalisierte Schreibweise oder einen Zeitraum bewahren. `updated` nennt das Datum, an dem der JSON-Begriffs-Datensatz selbst aktualisiert wurde. Diese Felder sollten nicht zusammengeführt werden, weil sie unterschiedliche Audit-Fragen beantworten.

4. Die Referenz-Umsetzung

Die Referenz-Umsetzung ist seit dem 2. September 2026 unter <https://augmanitai.com/registry/> live und veröffentlicht sechs Begriffs-Datensätze, die im bereitgestellten Index jeweils mit Version 1.0.0 geführt werden:

- *Gedankenvererbungslehre / Thought Inheritance*:
<https://augmanitai.com/registry/gedankenvererbungslehre.json>
- *The Fifth Stage / Architectural Cognitive Externalization*:
<https://augmanitai.com/registry/fifth-stage.json>
- *Cognitive Legacy Documentation*: <https://augmanitai.com/registry/cognitive-legacy-documentation.json>
- *Machine Coaching*: <https://augmanitai.com/registry/machine-coaching.json>
- *Universal Concept Layer*: <https://augmanitai.com/registry/universal-concept-layer.json>
- *Ehstand-Test*: <https://augmanitai.com/registry/ehstand-test.json>

Zwei Adressen auf Sammlungs-Ebene vervollständigen die einfache Abruf-Oberfläche: <https://augmanitai.com/registry/index.json> ist das maschinenlesbare Verzeichnis, <https://augmanitai.com/registry/schema.json> das Validierungs-Schema. Das Verzeichnis nennt Register, Spezifikation Universal Concept Layer v0.1, Verantwortlichen und ORCID, Erzeugungsdatum, Lizenz, Schema-Adresse, menschlich lesbare Seite und die sechs Begriffs-Kurzangaben. Ein Verbraucher kann somit bei der Sammlung beginnen und über die veröffentlichte URL zu einem einzelnen Datensatz gelangen.

Der menschlich lesbare Index enthält eine strukturierte `DefinedTermSet`-Auszeichnung. Deren Sammlung `hasDefinedTerm` führt dieselben sechs Einträge als `DefinedTerm`-Objekte mit Namen, Alternativnamen, Beschreibungen und JSON-Adressen. Damit ist die Seite zugleich eine lesbare Übersicht und eine strukturierte Erklärung, dass die Datensätze zu einem Begriffs-Set gehören. Daraus entsteht auch eine konkrete Konsistenz-Anforderung: Tabelle, strukturierte Auszeichnung und `index.json` sollen dieselbe Register-Zugehörigkeit beschreiben.

Die `llms.txt` der Website liefert den weiteren maschinenorientierten Kontext. Sie beschreibt AUGMANITAI als unabhängige Forschungspraxis, nennt das öffentliche Lexikon und seine maschinenlesbaren Daten, verlinkt die Prioritäts- und Begriffs-Seite und verweist auf das Universal-Concept-Layer-Papier sowie dessen Klartext-Fassungen. Seit dem 2. September 2026 führt sie zusätzlich das Register selbst auf: Die menschlich lesbare Register-Seite, das

maschinenlesbare Verzeichnis und das Datensatz-Schema erscheinen jeweils als eigene Adressen, sodass eine Maschine das Register direkt vom zentralen maschinenorientierten Einstiegspunkt der Website aus erreichen kann.

5. Was das Register NICHT ist

- Es ist kein Daten-Katalog: Es registriert bedeutungstragende Begriffe und ihre Herkunft, nicht Sammlungen von Datensätzen oder deren Speicherorte.
- Es ist kein Kennzahlen-Semantic-Layer: Es versioniert Begriffs-Definitionen, nicht geschäftliche Messgrößen, Berechnungsregeln oder Berichts-Dimensionen.
- Es ist keine Firmen-Ontologie: Es modelliert nicht die Entitäten, Prozesse und internen Beziehungen einer bestimmten Organisation.

6. Offene Punkte Richtung v1.0

Namensräume für Dritte. Eine öffentliche, organisationsübergreifende Schicht benötigt ein Verfahren, mit dem Dritte Kennungen ohne Kollisionen oder eine unterstellte Eigentümerschaft des Referenz-Verantwortlichen veröffentlichen können. Syntax, Delegation, Übertragung und Beständigkeit von Namensräumen müssen bestimmt werden, während Begriffs-Adressen lesbar bleiben.

Auflösungs-Schnittstelle. Der Abruf per GET funktioniert für bekannte Adressen; eine breitere Schicht benötigt zusätzlich ein festgelegtes Verfahren, um eine Kennung, Sprachbezeichnung oder Versionsanfrage zum passenden Datensatz aufzulösen. Die Schnittstelle muss eine exakte Version von der jüngsten kompatiblen Version unterscheiden und Antworten für unbekannte, verschobene oder abgelöste Begriffe definieren.

Konformitäts-Tests. Das JSON-Schema schafft eine strukturelle Grundlage, doch Konformität umfasst auch Bedingungen zwischen Dateien: Ein aufgeführter Begriff soll erreichbar sein, Identität und Version sollen mit dem Verzeichnis übereinstimmen, Menschen- und Maschinen-Definition sollen synchron bleiben, und das Beweis-Anker-Prinzip soll technisch erzwungen werden. Test-Datensätze können diese Erwartungen reproduzierbar machen.

Anbindung an Agenten-Werkzeugketten. Agenten-Abläufe benötigen einen vorhersehbaren Punkt für die Begriffs-Auflösung: beim Lesen einer Anweisung, bei der Auswahl eines Werkzeugs, beim Aufzeichnen einer Entscheidung oder beim Erstellen eines Audit-Pfads. Die Integration sollte die genaue Begriffs-Kennung und Version bewahren, die eine Handlung beeinflusst hat, statt nur eine Bezeichnung zu kopieren, deren Bedeutung sich später verändern kann.

7. Referenzen

1. Andreas Ehstand, *The Universal Concept Layer: Language as Versionable Code — A Meaning Layer between Humans, Language Models, Agents, and Robots*:
<https://doi.org/10.5281/zenodo.22172128>
2. AUGMANITAI Concept Registry — Universal Concept Layer v0.1 Referenz-Umsetzung:
<https://augmanitai.com/registry/>
3. Prioritäts- und Begriffs-Kanon: <https://augmanitai.com/prioritaet>