# Database Design V1

**1. researchers​**

* researcher\_id (PK, VARCHAR)
* canonical\_name (VARCHAR) – Standardized full name.
* latest\_affiliation (VARCHAR) – Most recent institution

**2.papers​**

**Available:**

* paper\_id (PK, VARCHAR) – Unique paper full ID
* title (TEXT) – Paper title
* abstract (TEXT) – Abstract text
* year (INT) – Publication year.
* doi (VARCHAR) – Digital Object Identifier.
* venue\_id (VARCHAR, FK to venues.venue\_id) – Conference/journal ID
* code\_repo (VARCHAR) – research reproducibility, url from PapersWithCode

**To be calculated:**

* paper\_type(VARCHAR) – inferred from full id

**3. researcher\_paper​ (Many-to-Many Join)**

**Available:**

* researcher\_id (PK, FK to researchers.researcher\_id).
* paper\_id (PK, FK to papers.paper\_id).
* affiliation (TEXT) – Institution at time of publication.

**To be calculated:**

* author\_order (INT) – Position in author list (1 = first author). -> needs better design for co-authorship （x）

**potential:**

* paper – track (official data or semantic analysis)
* paper – award (multiple nominees, multiple awards)
* paper – keyword
* researcher – affiliation history (找同事，
* venue\_id
* paper\_id – cited by, year

**derivable metrics:**

* total publications
* earliest publication
* most recent publication
* **citation count -> google scholar, semantic scholar…**

# Database Design V2

**1. researchers​**

* researcher\_id (PK, VARCHAR)
* canonical\_name (VARCHAR) – Standardized full name.
* affiliation (TEXT) – Most recent institution

**2.papers​**

* doi (**PK,** VARCHAR) – Digital Object Identifier.
* paper\_id (VARCHAR) – Unique paper full ID
* title (TEXT) – Paper title
* abstract (TEXT) – Abstract text
* year (INT) – Publication year.
* venue\_id (VARCHAR) – Conference/journal ID
* code\_repo (VARCHAR) – research reproducibility, url from PapersWithCode
* paper\_type(VARCHAR) – inferred from full id

**3. researcher\_paper​ (Many-to-Many Join)**

* researcher\_id (PK, FK to researchers.researcher\_id).
* paper\_doi (PK, FK to papers.doi).

**potential:**

* researcher – affiliation history (找同事，
* paper\_id – cited by, year
* **venue\_id – research field**

**Needs LLM or undisclosed data:**

* paper – track (official data or semantic analysis)
* paper – award (multiple nominees, multiple awards)
* paper – keyword

**derivable metrics:**

* total publications
* earliest publication
* most recent publication
* **citation count （scholarly, semantic schlar）\*not suitable for database**

# Useful Links

* <https://serpapi.com/> (pull cites from google scholar)

keyf0587c06e34e0315632fa6cc74eeda3805183300174eec55bfc41521289318d9

* <https://github.com/scholarly-python-package/scholarly>
* database: https://railway.com/

# Useful Metadata from ACL Anthology

## Paper Object

**abstract: MarkupText**

address: str

attachments: list

**authors: list[NameSpecification]**

**awards: list[str]**

**bibkey: str (unique for each paper)**

bibtype: str

citeproc\_dict: dict

collection\_id: str

csltype: str

deletion: PaperDeletionNotice

**doi: str**

editors: list[NameSpecification]

errata: list

**full\_id: str -> infer paper type, collection, volume, etc**

full\_id\_tuple: tuple

**id: str**

ingest\_date: None type

is\_deleted: bool

is\_frontmatter: bool

issue: str

journal: str

language: str -> ISO 639-2 code or IETF

language\_name: None type

month: str -> from parent volume

note: None type

**pages: str -> infer paper length if not classified**

paperswithcode: PapersWithCodeReference -> repeatability?

parent: Volume

pdf: PDFReference

publisher: str

revisions: list

root: Anthology

thumbnail: PDFThumbnailReference

**title: MarkupText**

venue\_ids: list with 1 item(s)

videos: list

volume\_id: str

web\_url: str

**year: str -> from parent volume**

## Person Object

**canonical\_name: Name**

comment: str

**id: str (unique for each author)**

is\_explicit: bool

**item\_ids: list[tuple (collection\_id -> str, volume\_id -> str, paper\_id -> str)]**

**names: list[Name]**

parent: Anthology

## Name Object

**first: str**

**last: str**

script: str (non-latin name variants)

\*Use **.as\_full()** to display in appropriate format

## NameSpecification Object

**name: Name**

**id: str**

**affiliation: str -> default None**

**variants: list[Name]**