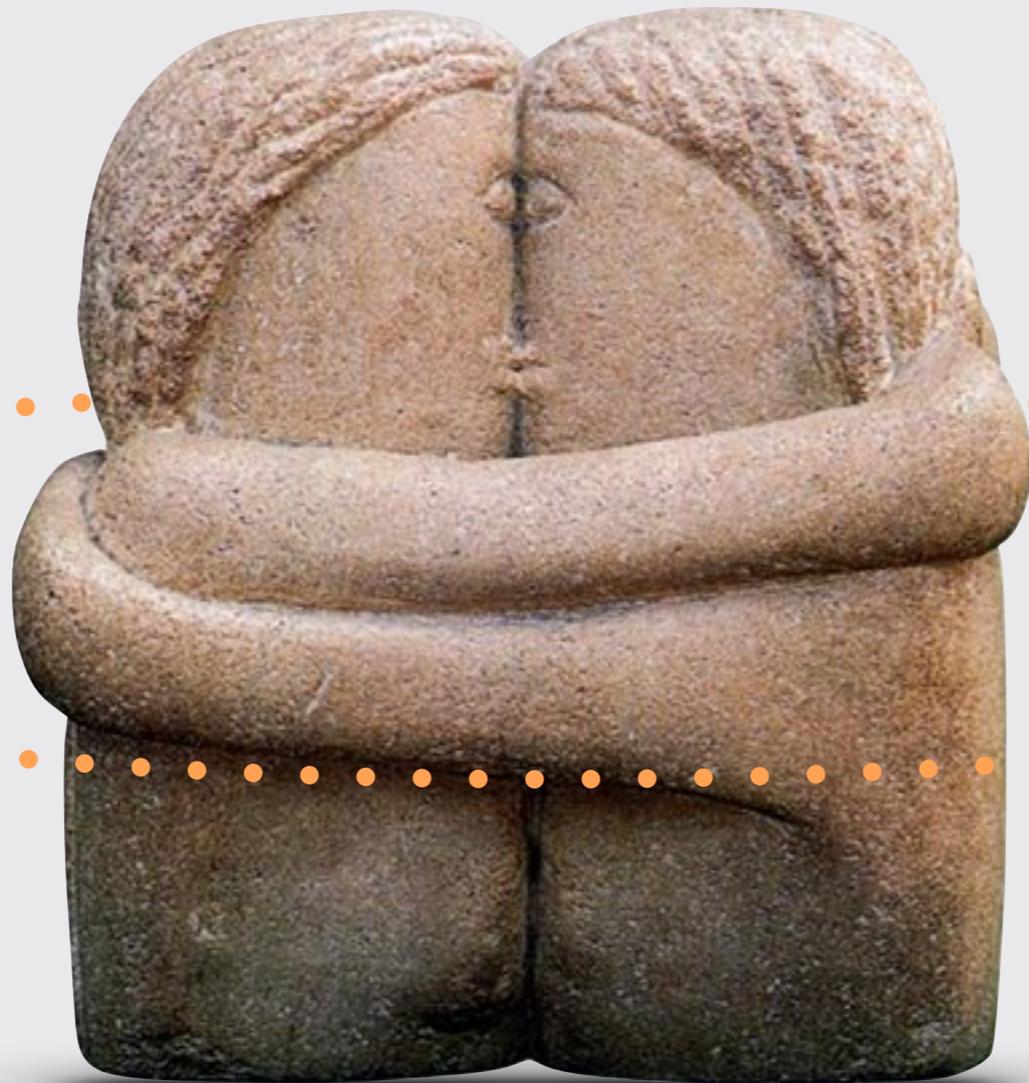**ZENON**

Zenon

Dédalo

author:
Wigg-Wolf

ZENON

{"resultCount": 34,
    "records":[{
        "authors":{"primary":{"**Wigg-Wolf, David**.":[]}},
        "formats":["Book"],
        "id":"001034714",
        "title":"Coins and ritual in late Iron Age and early
        Roman sanctuaries in the territory of the Treveri.",
}],"status":"OK"}

OPENAPI
INITIATIVE

swagger

https://zenon.dainst.org/api?swagger

Explore

Zenon

Record
Show/Hide | List Operations | Expand Operations

GET /record   Fetch records

Search
Show/Hide | List Operations | Expand Operations

GET /search   Search the index

Implementation Notes

Search the index with given terms and filters. POST method may also be used if sending a long request.

The URL syntax here is the as the one used in VuFind user interface. It is possible to make a search in VuFind and copy the query parameters here to make the same search via the API.

Response Class (Status 200)

Response containing result count, records and/or facets.

Model | Example Value

SearchResponse {
    resultCount (integer): Number of results,
    records (Array[Record], optional): Records,
    facets (Array[Facet], optional): Facets,
    status (string): Status code = ['OK']
}

Record {
    accessRestrictions (Array[string], optional): Access restriction notes,
    authors (Array[Authors], optional): Deduplicated author information including main, corporate and secondary authors,
    awards (Array[string], optional): Award notes,
    bibliographicLevel (string, optional): Bibliographic level = ['Monograph', 'Serial', 'MonographPart', 'SerialPart', 'Collection', 'CollectionPart', 'Unknown'],
    bibliographyNotes (Array[string], optional): Bibliography notes,
    callNumbers (Array[string], optional): Call numbers,
    childRecordCount (integer, optional): Number of child records,
    cleanDoi (string, optional): First valid DOI,
    cleanIsbn (string, optional): First valid ISBN favoring ISBN-10 over ISBN-13 when possible,
    cleanIssn (string, optional): Base portion of the first listed ISSN,
    cleanOclcNumber (string, optional): First OCLC number,
    containerEndPage (string, optional): End page in the containing item,
    containerIssue (string, optional): Issue number of the containing item,
    containerReference (string, optional): Reference to the containing item,
    containerStartPage (string, optional): Start page in the containing item,
    containerTitle (string, optional): Title of the containing item,
    containerVolume (string, optional): Volume of the containing item,
    corporateAuthors (Array[string], optional): Main corporate authors,
    dedupIds (Array[string], optional): IDs of all records deduplicated with the current record,
    edition (string, optional): Edition,
    findingAids (Array[string], optional): Finding aids,
    formats (Array[string], optional): Formats,
    fullRecord (Array[string], optional): Full metadata record (typically XML),
    generalNotes (Array[string], optional): General notes,
    geoLocations (Array[string], optional): Geographic locations (e.g. points, bounding boxes),
    hierarchicalPlaceNames (Array[string], optional): Hierarchical place names concatenated for display,
    hierarchyParentId (Array[string], optional): Parent record IDs for hierarchical records,
    hierarchyParentTitle (Array[string], optional): Parent record titles for hierarchical records,
    hierarchyTopId (Array[string], optional): Hierarchy top record IDs for hierarchical records,
    hierarchyTopTitle (Array[string], optional): Hierarchy top record titles for hierarchical records,
    humanReadablePublicationDates (Array[string], optional): Publication dates in human-readable format,
    id (string, optional): Record unique ID (can be used in the record endpoint),
    institutions (Array[string], optional): Institutions the record belongs to,

---

Dédalo Web REST API 1.0.8 [1.0.8]

[Base url: valencia.numisdata.org/dedalo/lib/dedalo/publication/server_api/v1/json]

https://valencia.numisdata.org/dedalo/lib/dedalo/publication/server_api/v1/docu/ui/json/json.php?code=456asd4asdsakljasduas753asdhj874asdAqheFv7wXM&host=valencia.numisdata.org

Builds Public Dédalo data for publication.
Use this online tool to test your data and remember always write your *Auth* code when executes a try call.
This API is focused to Oral History archives and is for general pourposes.
If you need special features or custom results, you are wellcome to join API develop team contacting with the source developers in webmaster@render.es

Aditional Dédalo publication info - Website

Schemes

[HTTPS ⌄]

Info Informative calls   ⌄

GET /tables_info Info about tables schema

Records Generic data get functions   ⌄

POST /records Get records from DDBB

Thesaurus Specific hierarchy and thesaurus functions   ⌄

POST /reel_terms Get reel indexed terms

POST /fragment_from_index_locator Build fragment of text and video from a index locator

POST /thesaurus_root_list Get first level of terms from thesaurus

POST /thesaurus_random_term A random thesaurus term name

POST /thesaurus_search Search into thesaurus term

POST /thesaurus_autocomplete Predictive autocomplete search

POST /thesaurus_term Get thesaurus term objects (one or various)

POST /thesaurus_indexation_node Indexation object with all required info

POST /thesaurus_video_view_data Video player data

POST /thesaurus_childrens Search all childrens of current term

POST /thesaurus_parents Search all parents of current term

Free Search functions in transcription text   ⌄

POST /free_search Find word/s in DDBB

Demo