Current Approaches to Providing Search APIs

Authorities Summary

Quick summary of primary information about each authority.

Authority	Results Format	Extent of Data Returned	Example Search API URL	Documentation	Comments
AGROVOC	JSONLD - skos ontology	URI, skos:prefLabel	http://agrovoc.uniroma2.it/agrovoc/rest/v1/search/?query=*milk*&lang=en&maxhits=10	SKOSMOS REST APIs	Additional Information
Getty	JSON, XML				
ISNI	RDF/XML, JSON-LD	URI, schema, dcterms, madsrdf, umbel (plus some supporting ontologies)	http://isni-acc.oclc.org/sru/DB=1.2/?query=pica.na+%3D+%22Piketty%2C+Thomas%22&operation=searchRetrieve&recordSchema=isni-rdf https://isni-url-acc.oclc.org/isni/0000000120247615.jsonld	ISNI SRU Search API Guidelines and Examples and ISNI Enquiry Response	See ISNI below
Library of Congress - Search API	HTML, ATOM/XML, ATOM/JSON	URI (with some formats), preferred label	http://id.loc.gov/search/?q=camus+peste&format=json http://id.loc.gov/search/?q=camus+peste&format=atom	See "Searching" and "Search Limits..."	See LC Search API below
Library of Congress - Suggest API	JSON	Preferred label, URI	http://id.loc.gov/authorities/names/suggest/?q=camus		See LC Suggest API below
Library of Congress - Label Search	HTTP Headers	Preferred label, URI	http://id.loc.gov/authorities/names/label/Camus, Albert, 1913-1960	See "Known-label search".	See LC Label Search below
MeSH	JSON	Valid subject headings in a variety of granularity; Heading context; Entry terms; URIs	https://id.nlm.nih.gov/mesh/lookup/descriptor?label=diabetes&match=contains&limit=10	MeSH RDF API https://id.nlm.nih.gov/mesh/swagger/ui#/	The API builder includes matching parameters and maximum results returned See MeSH below about SPARQL queries.
OCLC - FAST	RDF/XML	SKOS, Schema.org, Dublin Core	http://experimental.worldcat.org/fast/search?query=cql.any+all+%22dogs%22&maximumRecords=10&startRecord=1&sortKey=usage&recordSchema=info:srw/schema/1/rdf-v2.0&stylesheet=/fast/xsl/dummyResponse.xsl	https://platform.worldcat.org/api-explorer/apis/fastapi/LinkedData/SRUSearch	
OCLC - VIAF	RDF/XML, JSON	SKOS	http://www.viaf.org/viaf/search?query=cql.any+=+%22Jane%20Austen%22&maximumRecords=5&httpAccept=application/json	https://platform.worldcat.org/api-explorer/apis/VIAF	
Share-VDE	JSON (for the future Share-VDE APIs); XML (for Stardog APIs)	xml/rdf	http://stardog.share-vde.org/Share-VDE_prod/query?query=select*where{graph?g{?s?p?o}}=&limit=10	See: 1. Share-VDE Stardog databases 2. Share-VDE CKB query examples 3. Stardog APIs	See Share-VDE API below. If you need to query the system for testing, please, ask me for user/pw information.
Wikidata					
add additional authorities					

Additional Information

Details that are helpful in understanding and working with authorities.

Agrovoc

AGROVOC is a controlled vocabulary covering all areas of interest of the Food and Agriculture Organization (FAO) of the United Nations, including food, nutrition, agriculture, fisheries, forestry, environment etc. It is published by FAO and edited by a community of experts.

Stores data in Fuseki triple store and provides SKOSMOS rest API for searching and browse. Includes endpoints...

- SKOSMOS REST APIs - search API (in table above),
- **ONLINE** : SKOSMOS : Search & browse
- **File for DOWNLOAD**. Two distributions are provided: (a) CORE , which only includes AGROVOC concepts (RDF and NT format), and (b) Linked Open Data (LOD) which includes AGROVOC concepts + links to outside resources (NT and NQ format).
- **Public SPARQL endpoint**, served by Fuseki version 3.6.0

Information about these can be found at How to Access AGROVOC (Technical Documentation for all Endpoints).

Getty

Linked open data: SKOS ontology

All data should be available.

AAT - Cut down amount of data to a more useful profile. Re-express information in CIDOC CRM? rather than SKOS. Museums use CRM. Want to have CRM and simplified SKOS.

ULAN, TGN - Move to same ontology.

Person is a person instead of a representation of a person. This avoids confusion and mistaken use of a concept of a person as a person.

Suggests sticking to a concept of a person as opposed to the real person.

Data across ontologies becomes more difficult. Validations become a challenge. Request data from a specific ontology and only get the data in the ontology as opposed to mixing them.

ISNI

ISNI is the ISO certified global standard number for identifying the millions of contributors to creative works and those active in their distribution. The mission of the ISNI International Agency (ISNI-IA) is to assign to the public name(s) a persistent unique identifying number in order to resolve the problem of name ambiguity in search and discovery; and diffuse each assigned ISNI across all repertoires in the global supply chain so that every published work can be unambiguously attributed to its creator wherever that work is described. By achieving these goals, the ISNI will act as a bridge identifier across multiple domains and become a critical component in Linked Data and Semantic Web applications.

ISNI as Linked Data is still in test phase, but a first version is expected to be released, soon.

- The on-line search facility is based on SRU (See above) and allows multiple search keys as well as multiple result formats (xml, rdf/xml, jsonld)
- Linked Data serializations can be retrieved using content negotiation on the ISNI canonical form (isni.org/isni/{ISNI})
- A data dump for testing purposes will soon be available.

LC Search API

LC's HTML results and ATOM results (either as XML or JSON) are actually serializations of the output from LC's internal search API. There is no daylight, therefore, between the types of searches that can be performed by a human using the HTML UI or by a machine making requests of the service. LC Search supports generic keyword searching, with no limits of any kind, but only labels and a few other key data points are actually searched. Notes, for example, are not searchable. It supports limiting searching to specific datasets and date ranges (undocumented, which is an oversight). It supports targeted searches for specific labels, tokens (such as LCCNs), or codes. It supports wildcards, boolean keywords, and negation. Results are sorted by relevancy and, depending on the type of search, results may be weighted to preference "higher value" resources, such as Names, Subjects, or BF Resources, versus resources that play a supporting role in library data, such as regional encodings or broadcast standards.   With the use of key search limits /filters and wildcarding, it is theoretically possible to make the generic search service closely mimic the LC Suggest API, at least in terms of results (not sure if the sort order can be manipulated easily).   The Search API's output is intentionally generic.

LC Suggest API

LC's Suggest API has received minor enhancements over the years, but is largely the same as implemented back in 2009-2010, when the service was still hosted at lcsh.info. Its output format is identical. It is formally undocumented (and how it has managed this long without being formally documented is a bit of a mystery) but LC readily and willingly shares information about not only its existence but how it works. LC's Suggest API is a left-anchored, wild-carded search of preferred labels, codes (for example a search for "deu" will return "German" from the ISO-6392 dataset), or tokens (such as "sh12345," which is an LCCN). Results are ordered alphabetically (based on codepoint). There is some support for limiting based on MADS/RDF type (PersonalName, Topic, Geographic, etc.). Notably, it does not search variant/alternate labels. Under the hood, the code is 100% distinct from the code that runs the Search API, mostly so it can better focus on its purpose, which is to return typeahead suggestions as quickly as possible. The service's output is outdated, a form of JSON that is difficult to work with given today's typeaheads. It is also limited to only the URI and label of the hit.

LC Label API

LC's Label API attempts to match a string representing nominally the authorized label and alternatively a variant label to a resource. It returns the URI. LC's Label API has received minor, undocumented enhancements since it was released in 2009-2010. It is sensitive to spaces and punctuation and only returns a result of the result is strong, though it is not without error. The response is pure HTTP, with the Location header being that of a matched entity. A 404 is returned if no match was found.

MeSH (Medical Subject Headings)

MeSH also provides a SPARQL endpoint http://id.nlm.nih.gov/mesh/sparql, a SPARQL endpoint option on the API page https://id.nlm.nih.gov/mesh/swagger/ui#/, and a user friendly SPARQL endpoint editor https://id.nlm.nih.gov/mesh/query. SPARQL query results can be returned in a variety of formats including RDF.

Share-VDE API

1. Description of Share-VDE databases available on Stardog triplestore.

2. Instructions for Share-VDE libraries on how to access databases on Stardog triplestore, with some query example.

3. Stardog indications on HTTP APIs.

We are currently working on SVDE backend infrastructure that will support an ad hoc set of APIs specific to SVDE.