

# Federated Search API Documentation

December 4, 2018

Tamara Marnell, ILS & Discovery Services Librarian, Central Oregon Community College  
[tmarnell@cocc.edu](mailto:tmarnell@cocc.edu)

## Vendor APIs

### Credo Reference

**Documentation:** <https://developer.credoreference.com/>

**Type:** REST API

**Returns:** JSON

**Requires:** Access token provided by the vendor

### EBSCOhost

**Documentation:** <http://support.ebsco.com/eit/ws.php>

**Type:** REST API

**Returns:** XML

**Requires:** EIT profile set up by the vendor

**Notes:**

- For results to match the results seen in the database, the EIT profile must have the same settings as the profile used for database access.

### Ex Libris

**Documentation:** <https://developers.exlibrisgroup.com/>

**Type:** REST API

**Returns:** JSON

**Requires:** API key set up in the Developers Network

**Notes:**

- The “Did You Mean” suggestion is available in the [XService API Brief Search](#), which requires IP address registration on the vendor side and returns XML.

### Gale

**Documentation:**

<https://drive.google.com/open?id=1PKPQaCUBs8EuOkNiprGp81bnKWssNQVE>

**Type:** SRU

**Returns:** XML

**Notes:**

- Results cannot be returned in the same order they appear within a database.

- “The Basic Search formula varies slightly for each product and consists of multiple criteria for running the search. This is why you are seeing different results between the Basic Search in product and the basic SRU call...our current SRU server also has its own default sort order that cannot be modified at this time.” - Cengage support response to case 01905277, August 15, 2018

## Infobase

**Documentation:**

<http://support.infobaselearning.com/index.php?/videofod/Knowledgebase/Article/View/2093/633/streaming-video-search-api-integration-guide>

**Type:** REST API

**Returns:** XML

**Notes:**

- Currently there is no way to limit the number of results returned.

## ProQuest

**Documentation:** [https://drive.google.com/open?id=1I5vOFxkMqeUnxLW2Qcsk\\_LXab70vY\\_9C](https://drive.google.com/open?id=1I5vOFxkMqeUnxLW2Qcsk_LXab70vY_9C)

**Type:** SRU

**Returns:** XML

**Requires:** IP authentication set up by the vendor

**Notes:**

- Currently it is not possible to link to a keyword search, only to a database where users must replicate the search themselves.

## Statista

**Documentation:** <https://www.statista.com/download/Statista-API.pdf>

**Type:** REST API

**Returns:** XML

**Requires:** API key provided by the vendor

## Classification/Subject APIs

### OCLC

#### Classify API

The Classify API returns FAST classifications given a keyword.

**Documentation:** [http://classify.oclc.org/classify2/api\\_docs/index.html](http://classify.oclc.org/classify2/api_docs/index.html)

**Type:** REST API

**Returns:** XML

**Example:** <http://classify.oclc.org/classify2/Classify?heading=bananas&summary=true>

## FAST API

The FAST API returns a full FAST heading given an identifier, which includes Library of Congress subject heading IDs.

**Documentation:**

<https://www.oclc.org/developer/develop/web-services/fast-api/linked-data.en.html>

**Type:** REST API

**Returns:** XML

**Example:** <https://experimental.worldcat.org/fast/826418/rdf.xml>

**Notes:**

- There are three domains to choose from that all point to the same service: id.worldcat.org, fast.oclc.org, and experimental.worldcat.org. On March 29, 2019 id.worldcat.org began sporadically returning 404 errors for every query. Switching to fast.oclc.org resolved the issue at COCC.

## Library of Congress

The Linked Data Service provides access to ontologies, controlled vocabularies, and other lists for bibliographic description. It can be used to get LC classifications using a subject heading ID. Then a script can consult a local database of recommended resources by classification (e.g., for the subject heading Bananas with classification QK495.M78, the database can map QK to botany resources to recommend).

**Documentation:** <https://id.loc.gov/techcenter/>

**Returns:** XML (when requested in the header of the request, i.e., "Accept: application/rdf+xml")

**Notes:**

- The request must include a User Agent in the header, even an empty or nonsense one, or the request will be denied (e.g., "User-Agent: 0")

**Example:** <http://id.loc.gov/authorities/subjects/sh85011450>

## Autocomplete

The Wikipedia API can be used to get the titles of Wikipedia articles that match keywords. When coupled with [jQuery UI Autocomplete](#), this API can be used to offer the titles as autocomplete suggestions to users as they type.

**Documentation:** [https://www.mediawiki.org/wiki/API:Main\\_page](https://www.mediawiki.org/wiki/API:Main_page)

**Type:** REST API

**Returns:** JSON

**Example:** <https://en.wikipedia.org/w/api.php?action=opensearch&search=jane+aust>