

See WorldCat API examples in action:

www.oclc.org/applicationgallery/

Additional OCLC Web Services to consider:

- WorldCat Search API
- WorldCat Registry APIs
- OpenURL Gateway (part of WorldCat Registry)
- xID Services (xISBN, xISSN, xOCLCnum)
- Terminology Services

Join the OCLC Developer Network

The **OCLC Developer Network** seeks to create a space where developers and librarians can connect. It is designed as a collaborative, two-way communication group where members directly influence what OCLC Web Services are created and enhanced.

- Join the Developer Network by joining the **WC-DEVNET-L listserv** at <https://www3.oclc.org/app/listserv/>
- Read the Developer Network blog at <http://www.worldcat.org/devnet/blog/>
- Follow our tweet stream at <http://twitter.com/oclcdevnet>

WorldCat Basic API

What it is: Developer-level access to WorldCat—for limited data.

What it does: Search WorldCat and receive results for items in libraries. (WorldCat Basic API is a scaled-down version of the WorldCat Search API.)

What you get:

- Information about books, video, music and more in WorldCat
- Information about authors, titles, ISBNs and OCLC numbers
- Records in standard bibliographic citation formats (APA, Chicago, Harvard, MLA, and Turabian)
- A link back to WorldCat.org for geographically-sorted library information

Who can use it: Anyone and everyone for noncommercial use. The WorldCat Basic API requires a unique Key—your own personal “wskey”

Usage limits: 1,000 queries/day

Query Protocols: OpenSearch

Record Formats: Atom and RSS

Where to access: <http://www.worldcat.org/wcpa/content/affiliate/>

Documentation: <http://www.oclc.org/developer/services/WCBasic>

Why you love it: it’s open to all. Anyone can build an app or mash-up that encourages library discovery and use.

How to use the WorldCat Basic API



Access

WorldCat Basic API requests require an access key, provided by OCLC automatically through the Service Configuration module. Include your assigned “wskey” parameter in each request to the API, as shown in the example links.

Getting A Search Result

The form of an OpenSearch request is likely to be something like:

```
http://www.worldcat.org/webservices/catalog/search/openserach?q=[search terms]&wskey=[your key]
```

The complete pattern for an OpenSearch request would be:

```
http://worldcat.org/webservices/catalog/search/openserach?q=[query]&format=[atom|rss]&start=[start position]&count=[maximum number of records to return]&cformat=[citation format]&wskey=[your key]
```

Only the query value “q” would be required. Queries are sent as strings of keywords. All other values are optional. The default values expected are: format=atom, start=1, count=10. The search result presented will be limited to the first 100 records of that result.

Some examples

- A search for civil war, returning a result with the default Atom format, starting position, and count: [http://www.worldcat.org/webservices/catalog/search/openserach?q=civil%20war&wskey=\[key\]](http://www.worldcat.org/webservices/catalog/search/openserach?q=civil%20war&wskey=[key])
- A search for civil war, returning a result in the RSS format, starting at position 6, with a count of 5 records: [http://www.worldcat.org/webservices/catalog/search/openserach?q=civil%20war&format=rss&start=6&count=5&wskey=\[key\]](http://www.worldcat.org/webservices/catalog/search/openserach?q=civil%20war&format=rss&start=6&count=5&wskey=[key])
- A search for civil war, returning a result in the Atom format, including an MLA-formatted citation for each record: [http://www.worldcat.org/webservices/catalog/search/openserach?q=civil%20war&format=atom&cformat=mla&wskey=\[key\]](http://www.worldcat.org/webservices/catalog/search/openserach?q=civil%20war&format=atom&cformat=mla&wskey=[key])

Sample responses

See a Sample RSS response at <http://www.oclc.org/developer/documentation/worldcat-basic-api/rss-xml-sample>

See a Sample Atom response at <http://www.oclc.org/developer/documentation/worldcat-basic-api/atom-sample>

Requesting Formatted Citations

The API provides a way to obtain bibliographic citations, formatted in HTML for display in a Web browser. The supported bibliographic citation formats are APA, Chicago, Harvard, MLA, and Turabian.

The formatted citation result will be returned as a string of plain text, however it includes HTML formatting, so could be inserted directly into an HTML application. For example:

```
<p class="citation_style_TURABIAN">McPherson, James M. <i>Battle Cry of Freedom: The Civil War Era</i>. New York: Oxford University Press, 1988. </p>
```

results in a formatted citation such as:

```
McPherson, James M. Battle Cry of Freedom: The Civil War Era. New York: Oxford University Press, 1988.
```

If ‘all’ is specified as the cformat, all available citation formats will be returned in a single string.

Interpreting OpenSearch Responses

OpenSearch responses include title, the first author, a link, ISBN, and the OCLC number, whether the responses are in RSS or Atom format. For example, an Atom-formatted response:

```
<title>OCLC Worldcat Search: civil war</title>
<link href="http://worldcat.org/webservices/catalog/search/worldcat/openserach?q=civil+war&start=1&count=5&format=atom"/>
<subtitle>Search results for civil war at http://worldcat.org/webservices/catalog/</subtitle>
```

The title offers a human-readable string that could be used to present a label for the search result, the link field contains a URL that represents the current search in the web service, and the subtitle gives a brief annotation for the search.

In addition, responses include some OpenSearch response elements that are used to extend the RSS and Atom syndication formats. The additional metadata can be helpful for result set context and navigation, including the result size, starting position, number of items, and the search terms. For example:

```
<opensearch:totalResults>322066</opensearch:totalResults>
<opensearch:startIndex>1</opensearch:startIndex>
<opensearch:itemsPerPage>5</opensearch:itemsPerPage>
<opensearch:Query role="request" searchTerms="civil war" startPage="1"/>
```

Other response elements differ, depending on the requested format.

For Atom responses, these elements are especially useful:

The link elements with rel attributes of alternate, self, first, next and last include pre-built URLs for navigation through the search result.