A Tale of Two APIs

Annamarie Klose Hrubes
Digital Initiatives Librarian
William Paterson University

What are APIs?

In computer programming, an **application programming interface (API)** is a set of subroutine definitions, protocols, and tools for building application software. In general terms, it is a set of clearly defined methods of communication between various software components.

Web APIs are the defined interfaces through which interactions happen between an enterprise and applications that use its assets. An API approach is an architectural approach that revolves around providing programmable interfaces to a set of services to different applications serving different types of consumers. When used in the context of web development, an API is typically defined as a set of Hypertext Transfer Protocol (HTTP) request messages, along with a definition of the structure of response messages, which is usually in an Extensible Markup Language (XML) or JavaScript Object Notation (JSON) format.

- Application programming interface,
https://en.wikipedia.org/wiki/Application_programming_interface

How can you use APIs?

In the library world, vendors have created APIs that allow us to get better access to their data. As a librarian, you can retrieve a large set of records (GET command) or add/edit a large set of records (POST command). We'll focus on the GET command.

Some data download scenarios with the GET command:

- MARCXML or Dublin Core records from World Cat
- Records on a specific subject in Digital Public Library of America (DPLA)
- Statistics related to journals that your university's faculty have published in*

More examples:
https://www.programmableweb.com/news/49-library-apis-librarything-worldcat-search-and-bookshare/2012/11/06
https://libraries.mit.edu/scholarly/publishing/apis-for-scholarly-resources/
http://guides.lib.berkeley.edu/information-studies/apis

Downsides

APIs allow us to retrieve, manipulate, and provide data faster and better but there are some pitfalls to be aware of.

1. Technical errors - Poor design of API or underlying information resource. Computers are only as smart as the people who program them.
2. Data integrity issues - The data contained in the information resource is flawed or incorrect.

The moral of this story is to sample API data and examine it before engaging in large-scale projects with it.

Be aware that APIs beat manual labor but your downloaded data sets might require data clean-up efforts.

Use sampling to get to know your API

When you data gives you questions… Examine the results.

Verify unusual results, e.g.

- Does *Journal of Celtic Linguistics* really have a CiteScore of 0.00? Yes
- Does *Pediatric Health* really have no CiteScore? Yes, coverage discontinued in Scopus.
- Why is an article indexed in Scopus but I'm not getting data for its journal? Well, sometimes this happens.
- Is there something wrong with NULL results?

Patterns will emerge that will help you understand the data you receive in the future.

However, you might also find errors...

API reliability as illustrated by 2 examples

For my project, I used two APIs to acquire bibliographic information and statistics related to journals that WPU faculty have published in. I used this data for a database the university uses for annual reporting of faculty achievements.

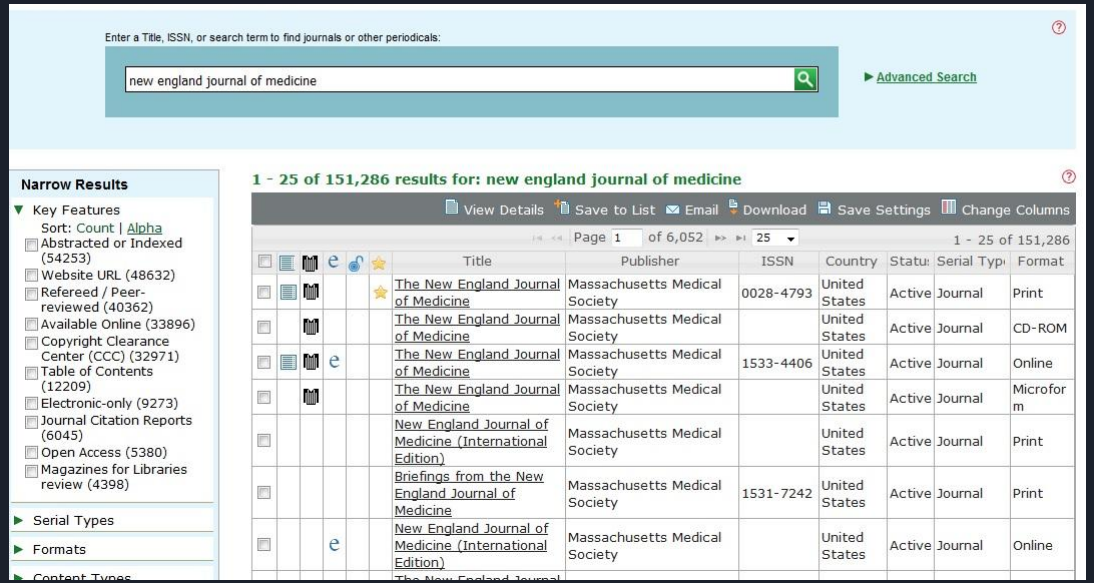
Structure of Resource

Even when APIs work as advertised, you need to be aware of the underlying structure of the information resource.

That means understanding how the resource works. APIs have rigid requirements. They either do what you want or don’t. Depending on your needs, you may need to use multiple APIs at once!

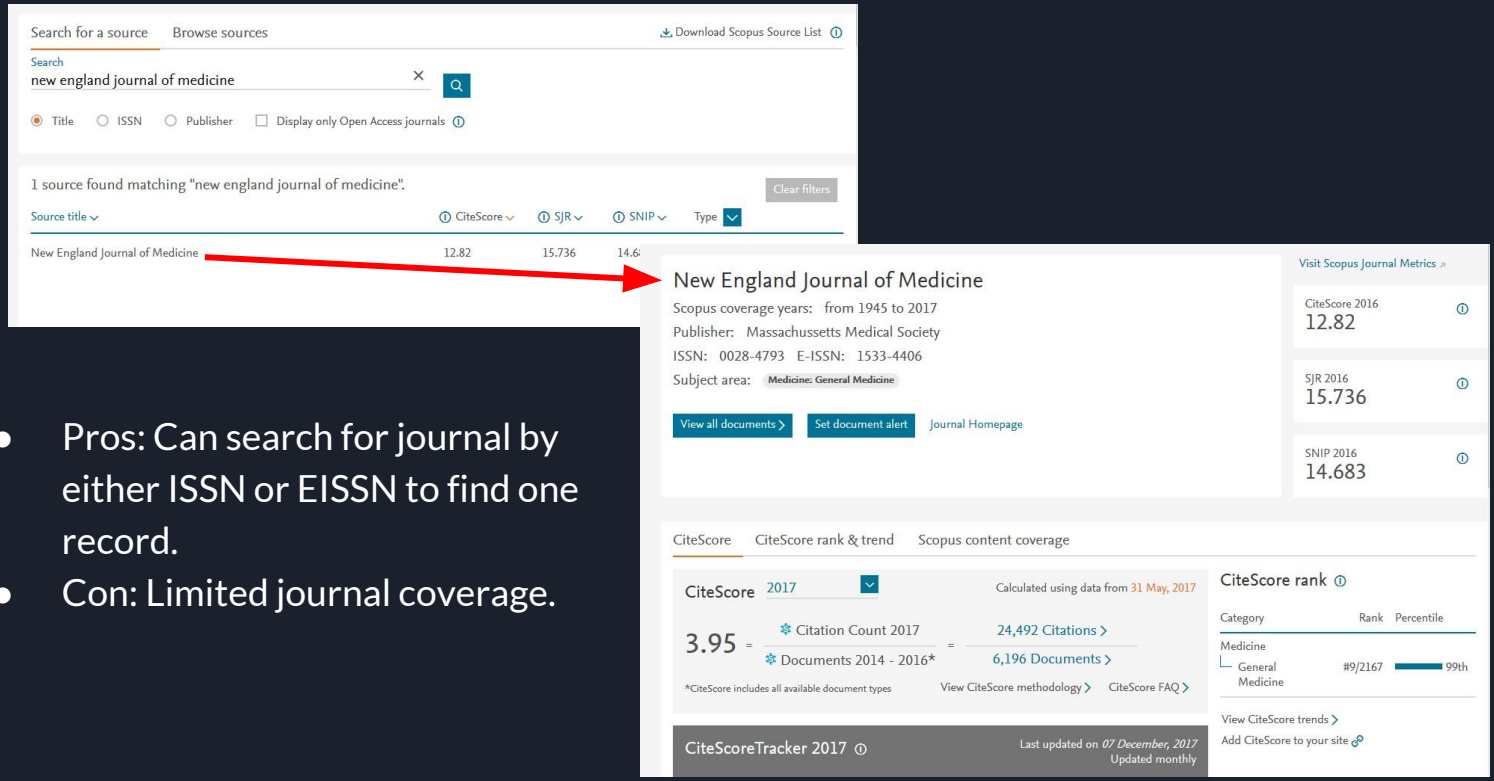
APIs can only be reliable if you understand the structure that you are working with. Reliability is a function of how predictable they are.

Ulrich's Web - Evaluating the resource structure



- Pro: Expansive list of journals
- Neutral: Journals have different records depending on their format. Multiple records for one journal.
- Con: Ulrich's uses the same field for ISSN and EISSN numbers.

Scopus - Evaluating the resource structure



- Pros: Can search for journal by either ISSN or EISSN to find one record.
- Con: Limited journal coverage.

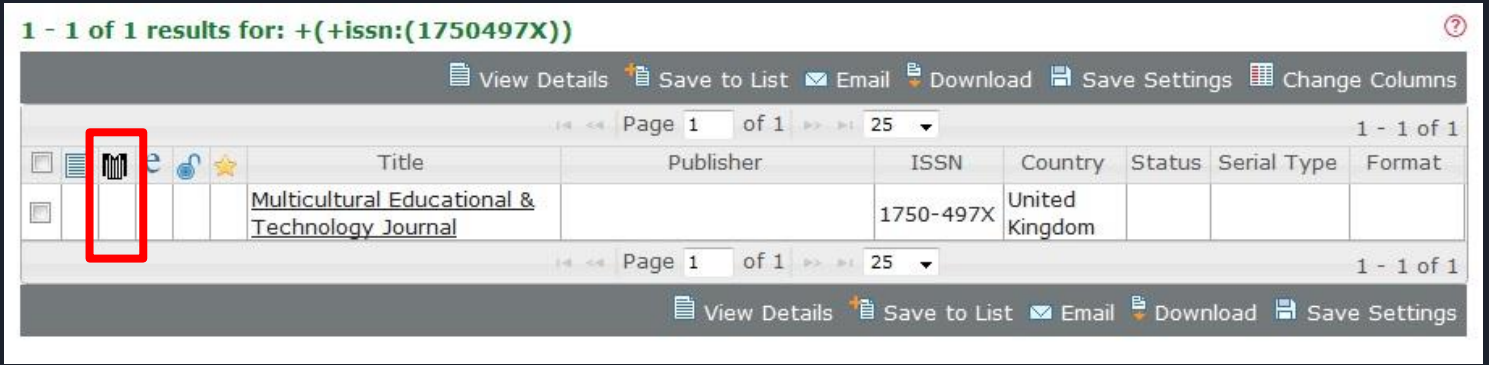
Ulrich's Web - API Design Issues

GET http://ulrichsweb.serialssolutions.com/api/[API Key]/search?query=issn:1750497X

{"requestStart":null,"statusMessage":null,"results":[{"keyFeatures":[],"rss":[],"tocImg":null,"languagesCSV":null,"contentTypesCSV":null,"formatsCSV":null,"availableOnlineImg":null,"openAccessImg":null,"openAccessUrl":null,"serialTypes":[],"titleId":"625520","title":"Multicultural Educational & Technology Journal","toc":false,"historicTitle":true,"refereed":false,"availableOnline":false,"openAccess":false,"reviewed":false,"price":null,"contentTypes":[],"frequency":null,"languages":[],"issn":"1750-497X","publisher":null,"startYear":null,"coden":null,"otherFeatures":[],"deweyNumbers":[],"lcnumber":null,"historicTitleImg":null,"serialTypesCSV":null,"refereedImg":null,"reviewedImg":null,"id":"550876","country":"United Kingdom","description":null,"subject":[],"formats":[],"status":null}],"numberOfRecords":1,"requestSortOrder":null,"requestSortIndex":null,"requestRows":null,"totalRecords":1,"requestQuery":"issn:1750497X","status":"Success"}

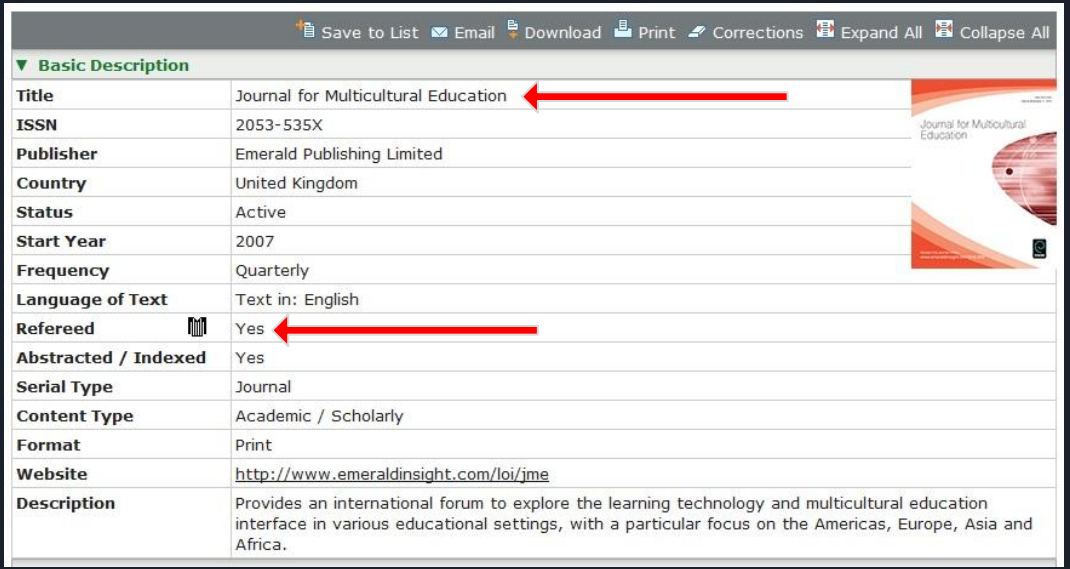
Ulrich's Web - API Design Issues

When you look up this ISSN manually in Ulrich's Web, you find in the search results that this journal is not peer-reviewed, i.e. no refereed shirt is indicated.



Ulrich's Web - API Design Issues

However, clicking on the record in the summary list leads to the current version of the journal (different title and ISSN) that *is* peer-reviewed.



Save to List | Email | Download | Print | Corrections | Expand All | Collapse All

▼ Basic Description

Title	Journal for Multicultural Education
ISSN	2053-535X
Publisher	Emerald Publishing Limited
Country	United Kingdom
Status	Active
Start Year	2007
Frequency	Quarterly
Language of Text	Text in: English
Refereed	Yes
Abstracted / Indexed	Yes
Serial Type	Journal
Content Type	Academic / Scholarly
Format	Print
Website	http://www.emeraldinsight.com/loi/jme
Description	Provides an international forum to explore the learning technology and multicultural education interface in various educational settings, with a particular focus on the Americas, Europe, Asia and Africa.

Scopus’ Serial Title - API Design Issues

The same ISSN lead to different results. The Scopus’ Serial Title API recognized the older ISSN (for *Multicultural Education & Technology Journal*) but gave me scores for the newer journal title (*Journal for Multicultural Education*). That was perfect for my purpose. It may not be for others.

Data Integrity Issues

Errors

- Some peer-reviewed journals were listed as FALSE for refereed. Some of those have since been corrected by Ulrich's Web.
- One ISSN was recorded incorrectly in a journal article indexed by Scopus. While NULL results sometimes are because of no data, this NULL was an error that required manual correction to retrieve data.
- Without adequate information, it's difficult to tell if prior versions of journals were not peer-reviewed while new ones are. FALSE values may be, well, FALSE.
- Of hundreds of records, some had to be corrected or entered manually.

Recommendations

- Understand the underlying resource structure
- Study the API documentation carefully to construct queries
- Conduct sampling to review structure and data integrity.
- Compare data provided from different APIs to decide which to use for what. For instance, the publisher listed was different between Ulrich's Web and Scopus. The latter favored the larger, parent company of publishers.
- If necessary, clean up data after download. Excel, Google Refine, or coding languages (e.g. Python) might be utilized for bulk corrections. Some corrections might require manual work.

Thank you!

Annamarie Klose Hrubes

klosehrubesa@wpunj.edu