

For Developers: API documentation (v1.0, 14 September 2018)

This API allows anyone to download the XML and PDF of specific articles as well as to download links to the XML of the entire corpus of articles. The content is indexed using [Apache Solr](#), which is open-source software that allows simple queries to be built in order to search and retrieve the content required.

Each article is assigned a unique DOI that can be resolved at [doi.org](#). An example of a DOI for a Gates Open Research article is <https://doi.org/10.12688/gatesopenres.12838.2>. Solr queries will return the DOIs of articles that match the search. See below for details of how to build a Solr query, and the parameters and terms that you can use to refine your search.

Once you have the DOI of the articles you are interested in, the XML or PDF can be retrieved thus:

Request XML for a specific DOI:

```
https://gatesopenresearch.org/extapi/article/xml?doi={doi}
```

For example:

```
https://gatesopenresearch.org/extapi/article/xml?doi=10.12688/gatesopenres.12838.2
```

Request PDF for a specific DOI:

```
https://gatesopenresearch.org/extapi/article/pdf?doi={doi}
```

For example:

```
https://gatesopenresearch.org/extapi/article/pdf?doi=10.12688/gatesopenres.12838.2
```

To download links to the XML of the entire corpus, use: <https://gatesopenresearch.org/published-xml-urls>

Building a Solr query

You do not need to register on the site to get an API key, you can just build the Solr query as described and run it.

The base HTTP GET request is:

```
https://gatesopenresearch.org/extapi/search?q={query}
```

Each query will need the parameter 'q', followed by the query parameters and/or terms required.

Multiple fields can be searched using the Boolean operators AND, OR and ANDNOT.

Tables of valid query parameters and terms at the end of this page.

The response to the GET request can be delivered as an XML or a JSON file. By default, you will receive an XML file. To get a JSON file, use `wt=json`:

```
https://gatesopenresearch.org/extapi/search?q=R_TI:"immunization"&wt=json
```

Some example queries

To get all articles with “immunization” in the abstract:

```
https://gatesopenresearch.org/extapi/search?q=R_ABS:"immunization"
```

To get all articles with “immunization” or “vaccine” in the abstract:

```
https://gatesopenresearch.org/extapi/search?q=R_ABS:"immunization" OR R_ABS:"vaccine"
```

To add in another variable, for example, you only want articles with “immunization” in the abstract and that have an author affiliated with a certain institution:

```
https://gatesopenresearch.org/extapi/search?q=R_ABS:"immunization" AND R_INS:"World Health Organization"
```

And here, we want articles with “vaccine” in the title that were published after a specific date:

```
https://gatesopenresearch.org/extapi/search?q=R_TI:"vaccine" AND R_PUD:[1528270617465 TO *]
```

If you want to get results containing more than one study type or more than one article type, you must use the operator OR not AND, for example:

```
https://gatesopenresearch.org/extapi/search?q=R_TY:"CASE_REPORT" OR R_TY:"RESEARCH_ARTICLE"
```

For a tutorial on the syntax of building regular Solr queries, go to [SolrTutorial.com](https://solr.apache.org/tutorial.html).

Limitations on requests

Frequency of requests

Please note, user can only make 100 requests per 60 seconds; if the requests exceed this, an unauthorized status 401 will be returned.

Number of results per request

There is a limit of 100 results per request.

The file will include the total number of pages that the Solr query would return if there was no limit set so that you can then run further requests to get all the results.

For example, if a request returns >100 results (there is no need to put a page number to get the first page of results as that is the default of the parameter – see the Parameters and terms table below), then the file will have the following information at the top:

```
"totalNumberOfPages" : 2,  
"numberOfResultsInPage" : 100
```

This indicates that you only need to run one more request with the page number filter set to 2 to get the rest of the results (append &page=2 to the query).

Parameters and terms

Query parameters

Name	Type	Mandatory	Information
q	String	Y	Query terms (see the table below)
page	Integer	N	1,2,3 Retrieve a specific page from the results (e.g. return page 3); default is page 1
rows	Integer	N	Number of results to include per page; default/maximum is 100
start	Integer	N	Start point in page; default is 0
wt	String	N	Values: 'xml', 'json' Type of file retrieved; default is 'xml'

Query terms

The filters correspond to the filtering and advanced search options found on Gates Open Research's [browse pages](#).

Filter	What the filter will retrieve
R_TI	Title
R_AU	Author
R_INS	Institution
R_AF	Author affiliation
R_CI	Competing interests
R_ABS	Abstract text
R_FT	Full text
R_TE	All (can be any option in this table)
R_RE	Reviewer name
R_RA	Reviewer affiliation
R_RP	Reviewer report

R_GIT	Grant funder
R_CM	Commenter name
R_CO	Comment text
R_CA	Commenter affiliation
R_CL	Study type – <i>see list of options below*</i>
R_TY	Article type – <i>see list of options below[#]</i>
R_SUB	Subject
R_PUD	Publication date of article (version 1) (in milliseconds) – <i>see how to convert to time in ms below^{\$}</i>
R_LU	Publication date of most recent version (in milliseconds) – <i>see how to convert to time in ms below^{\$}</i>

***Article types**

CASE_REPORT	RESEARCH_ARTICLE
CLINICAL_PRACTICE_ARTICLE	RESEARCH_NOTE
CORRESPONDENCE	SOFTWARE_TOOLS
DATA_NOTE	STUDY_PROTOCOL
METHOD_ARTICLE	SYSTEMATIC_REVIEW
OPEN_LETTER	

#Study types

NEGATIVE	CLINICAL_TRIAL
REANALYSIS	REGISTERED_REPORT

^{\$}Dates

In order to convert a date to a time value in milliseconds (ms), follow the instructions given here:

<http://www.ruddwire.com/handy-code/date-to-millisecond-calculators/#.Wx0yA0gvw4k>