ENA Portal API v2

Programmatic access to data held within ENA

EMBL-EBI
May 2023

Introduction to ENA Portal API

The main function of the ENA Portal API is to provide access for users to search against all available data in ENA. In this API we introduce the term "data portal". This is a way to allow a search to be performed against different sets of data, which can include a mixture of public and pre-publication (private) data. This concept of data portal, as well as the data portals currently available, is described in more detail in the following chapter "Performing a search".

As the API provides access to pre-publication data, user authentication is required when performing searches for such data. Any Webin (submission) or DCC (data hub) account is supported. If you wish to search against only public data, you can search without passing credentials. By default, any username provided that **does not** begin with "Webin-" or "dcc_" is assumed to be an anonymous access.

The portal API is available from https://www.ebi.ac.uk/ena/portal/api. If you use this URL within a web browser, you will see some documentation regarding the different functions available, as well as forms for the different endpoints which allow you to send requests. However, beyond providing ease of use for testing, we don't recommend you use the API via a web browser as there are other tools available more appropriate for that purpose (such as Advanced Search within the ENA Browser).

Access to the portal API will likely be through either inclusion within scripts or using a tool such as ***wget*** and ***curl***. When using any of these options, it is important to remember that when performing authenticated searches, a request header is needed for supplying the username and password. For anonymous searches on public data, username and password are not required.

For example, to download a search result using *curl* (anonymously):

```
curl -X GET --header 'Accept: application/json'
'https://www.ebi.ac.uk/ena/portal/api/search?<search_definition>'
```

The same search with authentication:

```
curl -X GET --header 'Accept: application/json' -u username:password
'https://www.ebi.ac.uk/ena/portal/api/search?<search_parameters>'
```

There are three important points to note about the above command:

1. The search URL is bound by single quotes (or double quotes if you're using curl in a Windows command line). Due to the search URL consisting of several parameters, joined by the "&" character and often containing double quotes around text field

values, this is required for the entire URL to be read by *curl*. This is also true when using *wget*.

2. HTTPS is used in the URL instead of HTTP.
3. The above command will write the data to the stream (it will be printed to your screen and not saved). To save into a named output file, you should use the "-o" option in *curl* (or the "-O" option in *wget*). For example:

```
curl -o output.txt -X GET --header 'Accept: application/json' -u
username:password
'https://www.ebi.ac.uk/ena/portal/api/search?<search_parameters>'
```

Endpoints

While there is one main function of the API, several additional endpoints are available to support searches. These are listed in the table below.

ENA Portal API endpoints

Endpoint	Purpose	Authentication Header required
doc	Download the latest documentation for the API.	No
search	Perform a search against a single data group (result) of the data available in ENA. This is optionally against all public data in ENA or against a subset of data (which can be both public and private) defined by a "data portal". Both GET and POST are supported.	Yes for a data portal specific search. No to search across all public data.
results	Fetch a list of all results available to search against. The list will be different based on the data portal identified in the request.	No
searchFields	Fetch a list of all searchable fields for a given result.	No
returnFields	Fetch a list of all returnable fields for a given result.	No
controlledVocab	Fetch a list of all available controlled vocabulary values for a given field.	No
accessionTypes	Fetch a list of accession types that can be used to tweak a search query by including or excluding specific accessions	No
count	Provides a count of rows matching the search parameters in a given search. Both GET and POST are supported	Yes for a data portal specific search. No to search across all public data.
filereport	Provides a report of data file information, including download URLs, byte sizes, MD5 checksums	Yes for a data portal specific search. No to search across all public data.
filereportcount	Provides only a count of the matching records available for a file report	Yes for a data portal specific search. No to search across all public data.

Endpoint	Purpose	Authentication Header required
links/sample	Fetch a list of all accessions of a result type linked with a given sample	No. Only for public data.
links/study	Fetch a list of all accessions of a result type linked with a given study	No. Only for public data.
links/taxon	Fetch a list of all accessions of a result type linked with a given taxon	No. Only for public data.

The doc endpoint will redirect you to the latest version of this API documentation:

https://www.ebi.ac.uk/ena/portal/api/v2.0/doc

All other endpoints are described in the following chapters.

Performing a search

A search is performed via the `/search` endpoint. Several parameters are available, but only one is required: result. All available parameters are listed in the table below.

Parameters for the search endpoint

Parameter	Data type	Description
result	string	The result type (data set) to search against
query	string	A set of search conditions joined by logical operators (AND, OR, NOT) and bound by double quotes. If none are supplied, the full result set will be returned.
field	string	Specific to the /count endpoint. If provided, an aggregation is run on the field. It is only supported for a set of fields.
fields	string	A list of fields (comma separated) to be returned in the result. If none are supplied, the accession and description/title of the main result object will be returned. To get ALL available fields for the result, you can use the convenience value of “fields=all”.
dataPortal	string	The data portal ID.
limit	integer	The maximum number of records to retrieve. The default value is 100,000. If the full result set is to be fetched, the limit should be set to 0.
dccDataOnly	boolean	Whether to limit the search to only records that are in the DCC datahubs that the requesting user has access to. Options are true and false.
includeMetagenomes	boolean	Whether to include public metagenome data in the search. By default, this is false, so these are not included. Note that any metagenome data associated with a DCC hub will always be included in a search against that DCC.
format	string	What format the results should be returned as: TSV or JSON. By default, a TSV report is provided.
download	boolean	Whether to download the result as a file, rather than read it from the stream. By default, this is false. Options are true/false.
excludeAccessionType	string	The accession type to exclude accessions
excludeAccessions	string	A list of accessions that you would like to be excluded from the results of your query
includeAccessionType	string	The accession type to include accessions

Parameter	Data type	Description
includeAccessions	string	A list of accessions that you would like to be included with the results of your query
searchCurations	boolean	Whether to search in third party curations submitted to the ELIXIR Data Clearinghouse. By default this is false.

Data portals

A "data portal" defines the set of data that is available for searching. There are currently four data portals available: *ena*, *pathogen*, *faang*, and *metagenome*. If no data portal is defined in the request, the ENA portal will be used by default.

- *ENA portal*

The ENA portal covers all public data held within ENA.

- *Pathogen portal*

As there is not yet a public repository defining all pathogenic taxa and strains, the pathogen portal contains all data from the Bacteria, Virus, Kinetoplastida and Amoebozoa lineages from the NCBI taxonomic tree. The pathogen portal can be accessed anonymously, in which case only public data can be searched. Alternatively, a user can authenticate their search using either a Webin account or a DCC (data hub) account. When using a Webin account, all data hubs that the user is registered with will be included in the search.

- *Faang portal*

Public data associated with the FAANG (Functional Annotation of Animal Genomes) project.

- *Metagenome Portal*

Metagenomic data including raw data originating from environmental samples and assemblies of classes 'primary metagenome', 'binned metagenome' and 'Metagenome-Assembled Genome'. The metagenome portal includes access to pre-publication data for specific users.

Note:

For authentication, usernames that begin with "dcc_" or "Webin-" are accepted. Any other values are ignored and the search is treated as an anonymous search.

Results

A result is a set of data that can be searched against and returned. These are largely based on the different data types available within ENA, however there are also some sub-divisions created by the data release/distribution practice.

Results may be added at any time to a data portal and the documentation may not yet be updated to those available. You can find the latest list of results available for a data portal using the `/result` endpoint. This requires just one parameter: dataPortal.

e.g.

https://www.ebi.ac.uk/ena/portal/api/results?dataPortal=ena

https://www.ebi.ac.uk/ena/portal/api/results?dataPortal=pathogen

Results available for the ena data portal

Result	ENA
read_run	Y
read_experiment	Y
read_study	Y
analysis	Y
analysis_study	Y
sample	Y
study	Y
assembly	Y
sequence	Y
wgs_set	Y
tsa_set	Y
tls_set	Y
coding	Y
noncoding	Y
taxon	Y

Sample result

Samples are of particular importance within the read and analysis domains, where all of the metadata describing the experiment and analysis are only available within the sample record. Assembled and annotated sequences (and their feature level products, coding and noncoding) traditionally hold sample annotation within their source feature, however samples are also gaining a more important role within these data types. This is due to the wider scope of information sample records can hold, as well as standardisation of fields (and formats) for communities based on sample checklists. Assemblies are one area that are gaining larger links to sample records.

Study result

While the *read_study* and *analysis_study* results cover all studies that are referenced by read and analysis data, the *study* result covers all available studies, including those that

are linked to genome assemblies and other assembled and annotated sequences. The other main difference is that for the study result only study-specific fields are available to be searched and returned while the other two also have data-specific (read/analysis) fields.

Read results

There are three different results that can be searched within the read data based on the three data types: runs (*read_run* result), experiments (*read_experiment* result) and studies (*read_study* result). These data types are all linked and therefore the same fields are available for searching against for all three results. However the *read_experiment* and *read_study* results are subsets of the *read_run* result and therefore only a subset of the read_run fields are returnable for each.

Analysis results

There are two different results that can be searched within the analysis data based on the two data types: analyses (*analysis* result) and studies (*analysis_study* result). As for the read results, these have the same fields available for searching, but the *analysis_study* result is a subset of the *analysis* result.

Assembly result

While all other data types have only a single public version, there can be multiple public assembly versions. The *assembly* result returns information on the latest assembly version. This is something to bear in mind when searching against the assembly name as the record returned may not be the version that matches the name.

Sequence result

Contains assembled and annotated sequences. This data set does not include whole genome shotgun (WGS) or transcriptome analysis (TSA) or Targeted Locus Study (TLS) sequences. These are instead separated into the contig set results described below.

Note that due to this segmentation of data, a sequence that has been suppressed since the last release can still be returned within the *sequence_release* result.

Contig set results

Whole genome shotgun (WGS) and transcriptome analysis (TSA) and Targeted Locus Study (TLS) sequences are represented as sets grouped by a common set prefix, with a set of annotations that describes all sequences in the set. These sets can contain a few to millions of sequences and are usually required as a whole as opposed to as individual sequences. For this reason, these sequences are treated differently from all other assembled and annotated sequences. There are three results covering this data: *wgs_set* and *tsa_set and tls_set*. When performing a search against one of these results, the search is being made against the master record, holding all common information on the set. As such, the accession returned in the resultant report is the master accession. However the files that are linked within these results contain the full set (with the exception of the master_file field), allowing users to obtain all sequences of interest in one go.

Coding result

Coding records are generated from the CDS features within assembled and annotated sequences, and also within contig set sequences.

Noncoding result

Noncoding records are generated from several features within assembled and annotated sequences and contig set sequences; mostly the different RNA features.

Specifying the required output

Choosing which fields to return

By default, the output from a search will be tab separated format (TSV) report consisting of two columns, the accession and title/description. If wishing to retrieve a different set of columns in the report, the *fields* parameter should be used. A comma-separated list of returnable fields for the result should be supplied. The order of the columns will match the order of the request. For example, to drop the description field and fetch the scientific name, strain and collection date for a sample along with the accession, one would use:

```
fields=sample_accession,scientific_name,strain,collection_date
```

If the accession column for the result is not listed, it will be prepended to the list of requested fields. For example, the following list of fields will produce exactly the same report as the example given above:

```
fields=scientific_name,strain,collection_date
```

To get ALL available fields for the result, you can use the convenience value of "fields=all".

Sort order of the result

In most cases, the search results should be ordered based on the accession and this is therefore the default behaviour. But this is not guaranteed due to backend search optimisations, and users should NOT depend on the API returning fully ordered results.

Setting the format of the output

The output from the search is presented as a tab separated report with the first row providing the field IDs as column headers. This is the most useful format for human users (as opposed to fully programmatic/scripted use) as it can be imported into spreadsheet programs like Excel. Many script writers may also find it a preferred option due to simple splitting of the columns based on a tab. However, some programmatic users might prefer JSON formatted output. This can be requested using the *format* parameter:

```
format=json
```

Size of results

A search could return a few or millions, or even billions of results. The API does not support pagination of results, and will always return all results from the start to end, unless a "limit" parameter is passed. We strongly recommend that you pass a small limit

value when testing the API or experimenting building up your search queries. e.g. limit=1000

Pathogen portal specific search options

By default a search against the pathogen portal will look at all public data plus any pre-publication data that is shared within any of the data hubs (DCC accounts) that are authorised for the user account. In some cases, only the data within the data hub(s) should be searched. In this case, the *dccDataOnly* parameter should be used and set to true. In the case of a user accessing the API from a DCC account, `dccDataOnly=true` will result in the search being performed against all public and pre-publication data associated with that DCC data hub. If a Webin account was being used which was associated with two DCC accounts, the search would cover all data associated with both of the accounts.

We may add an option at a later date to refine searches to a single DCC data hub (or subset) to which a Webin account is registered.

Building a query

When no query is defined, all records from the selected result will be displayed. In most cases, however, a subset of those records are required. To define this subset, there are a number of filter fields available.

The query can be built from any number of fields, with logical operators and parentheses used to order the execution of each. Any text or controlled vocabulary values used within the query must be bound by double quotes. For example:

```
query=booleanField=true AND (stringField="value" OR cvField="CV1")
```

Standard filter types

The majority of searchable fields use a standard data type. The table below lists all available operators for each of these filter types.

Filter type	Operators
boolean	=
controlled vocabulary	=, !=
date	=, !=, <, <=, >, >=
number	=, !=, <, <=, >, >=
text	=, !=

Text searches are case insensitive and a wildcard character (*) can be used at the beginning or end of a string value for partial matching.

To fetch a list of all available values for controlled vocabulary fields, the following endpoint is available: `/controlledVocab?field=<fieldname>`

Function filter types

In addition to the standard filter types listed above, there are two query filters that are based on functions: geospatial and taxonomic.

Geospatial

All geospatial coordinates are represented in decimal degrees.

Geospatial functions

Function	Description	Parameters	Example
geo_box1	All locations within a box defined by the lower left (SW) and upper right (NE) points	SW latitude, SW longitude, NE latitude, NE longitude	geo_box1(-20, 10, 20, 50)

Function	Description	Parameters	Example
geo_box2	All locations within a box defined a centre point and a radius in km	latitude, longitude, radius (km)	geo_box2(35, 100, 300)
geo_circ	All locations within a circle defined by a centre point and a radius in km	latitude, longitude, radius (km)	geo_circ(35, 100, 300)
geo_lat	All locations within a latitude range given by a latitude and a radius in km	latitude, radius (km)	geo_lat(0, 100)
geo_north	All locations north of a given latitude (inclusive)	latitude	geo_north(80)
geo_south	All locations south of a given latitude (inclusive)	latitude	geo_south(-80)
geo_point	An exact latitude/longitude position	latitude, longitude	geo_point(9.12, -79.7)

Taxonomy

Three functions are available for performing taxonomic searches. These make it possible to filter on a single taxon (via NCBI taxon ID or scientific name) or a branch of the NCBI taxonomic tree.

Taxonomic functions

Function	Description	Parameters	Example
tax_eq	All records that match the given NCBI taxonomy identifier	NCBI taxon ID	tax_eq(9606)
tax_tree	All records that match the given NCBI taxonomy identifier or are descendants of it	NCBI taxon ID	tax_tree(2759)
tax_name	All records that match the given NCBI scientific name	NCBI scientific name	tax_name(“Homo%20sapiens”)

Searchable fields

The set of fields available for searching is dependent on the data portal and the result type. These fields are listed here. Note that fields may be added to the API faster than they are added to this documentation.

A full list of the latest result fields can be fetched from the following endpoint:

```
/searchFields?result=<resultId>
```

By default the searchable fields for that result for the ENA data portal will be returned. To request the fields for a specific portal, the *dataPortal* parameter needs to be used. For example to fetch the *sample* fields for the *pathogen* portal:

```
/searchFields?result=sample&dataPortal=pathogen
```

Sample fields

The set of fields searchable against samples is specific to a data portal. This is due to the different priorities of information to be searched, and therefore indexed, for each use case. Sample fields can also be used for refining data within the read and analysis results.

ENA sample fields

https://www.ebi.ac.uk/ena/portal/api/searchFields?result=sample

Pathogen sample fields

https://www.ebi.ac.uk/ena/portal/api/searchFields?result=sample&dataPortal=pathogen

Read fields

The following fields can be used to search against any of the read results: *read_run*, *read_experiment* and *read_study*. In addition to the fields in the table below, any *sample* fields can also be used in the search query.

Read search fields

https://www.ebi.ac.uk/ena/portal/api/searchFields?result=read_run

Analysis fields

The following fields can be used to search against any of the analysis results: *analysis* and *analysis_study*. In addition to the fields in the table below, any sample fields can also be used in the search query.

Analysis search fields

https://www.ebi.ac.uk/ena/portal/api/searchFields?result=analysis

Assembly fields

As the *assembly* result represents the latest public version of the assembly, the majority of fields that can be searched are specific to the latest version. The *assembly_name* field however contains all assembly version names, therefore a search for an assembly name may not return the version that matches the name.

Assembly fields

https://www.ebi.ac.uk/ena/portal/api/searchFields?result=assembly

Sequence fields

The fields in the table below are searchable for both the *sequence_release* and *sequence_update* results. Most of the source feature qualifiers are available to search against.

Sequence search fields

https://www.ebi.ac.uk/ena/portal/api/searchFields?result=sequence

Contig set fields

The fields in the table below are searchable for both the *wgs_set* and *tsa_set* results. The information common to the whole WGS set (therefore contained within the set's master record) is available for each record. Any individual sequence-level source feature information that differs from the master record cannot be searched for.

Contig set search fields

https://www.ebi.ac.uk/ena/portal/api/searchFields?result=wgs_set

https://www.ebi.ac.uk/ena/portal/api/searchFields?result=tsa_set

https://www.ebi.ac.uk/ena/portal/api/searchFields?result=tls_set

Coding fields

The fields in the table below are searchable for the *coding* result. As for searches against sequences, most of the source feature qualifiers are available to search against. Selected CDS-feature specific information has also been included.

Coding search fields

https://www.ebi.ac.uk/ena/portal/api/searchFields?result=coding

Noncoding fields

The fields in the table below are searchable for the *noncoding* result. As for searches against sequences, most of the source feature qualifiers are available to search against. Selected RNA-feature specific information has also been included.

Noncoding search fields

https://www.ebi.ac.uk/ena/portal/api/searchFields?result=noncoding

Returnable fields

While there is a large overlap between the searchable and returnable fields for each result, there are some differences. The returnable fields for each result are listed here. Note that fields may be added to the API faster than they are added to this documentation.

A full list of the latest result fields can be fetched from the following endpoint:

`/returnFields?result=<resultId>`

By default the returnable fields for that result for the ENA data portal will be retrieved. To request the fields for a specific portal, the *dataPortal* parameter needs to be used. For example to fetch the *sample* fields for the *pathogen* portal:

`/returnFields?result=sample&dataPortal=pathogen`

Sample fields

The *sample* result is unique amongst all results in that it is custom built for each data portal. Each data portal has different priorities with respect to the plethora of sample attributes, giving rise to a different, targeted, set of fields for each. If a field is not available for the data portal you are using, please contact our helpdesk using https://www.ebi.ac.uk/ena/browser/support to request its addition. If there is sufficient use of and/or demand for the field, we will add it.

These fields listed for each data portal below are available when searching against the *sample* result, but also when searching against the read and analysis results.

ENA *sample* result fields

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=sample

Pathogen *sample* result fields

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=sample&dataPortal=pathogen

Read fields

In addition to the fields listed below for *read_run* searches, any *sample* field can also be returned. Unlike for samples, the available columns for each of these results are consistent across all data portals.

read_run fields for all data portals

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=read_run

read_experiment fields for all data portals

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=read_experiment

read_study fields for all data portals

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=read_study

Analysis fields

In addition to the fields listed below for *analysis* searches, any *sample* field can also be returned. As with read results, the available columns for each of these results are consistent across all data portals.

analysis fields for all data portals

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=analysis

analysis_study fields for all data portals

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=analysis_study

Assembly fields

The following fields can be returned for the *assembly* result, based on the latest public version of the assembly.

assembly fields

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=assembly

Sequence fields

The following fields are available to fetch for the *sequence* result.

sequence fields

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=sequence

Contig set fields

The fields available to fetch for the *wgs_set* and *tsa_set and tls_set* results differ slightly.

wgs_set fields

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=wgs_set

tsa_set fields

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=tsa_set

tls_set fields

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=tls_set

Coding fields

The following fields are available to fetch for the *coding* result.

coding fields

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=coding

Noncoding fields

The following fields are available to fetch for the *noncoding* result.

noncoding fields

https://www.ebi.ac.uk/ena/portal/api/returnFields?result=noncoding

Field definitions

Attribute names can vary greatly, especially those used to describe samples. We map all similar fields into a single representation, and normalise the format of the data wherever possible. To assist in determining which fields are of interest in building search queries and result reports, a full listing of all fields and their definitions are detailed below.

Field	Description
accession	Accession number
allele	Name of the allele for the given gene
altitude	Altitude (in metres)
analysis_accession	Analysis accession number
analysis_alias	Submitter's name for the analysis
analysis_title	Brief analysis description
analysis_type	Type of sequence analysis
anticodon	Location of the anticodon of tRNA and the amino acid for which it codes
artificial_location	Indicates location is modified to adjust for the presence of a frameshift or internal stop codon
assembly_level	Assembly level (contig, scaffold, chromosome, complete genome)
assembly_name	Genome assembly name. When searching, considers all all live versions of the assembly. When returning, the name of the latest version.
assembly_title	Brief assembly description
base_count	Number of base pairs
bio_material	Identifier for biological material including institute and collection code
breed	Breed
broad_scale_environmental_context	Report the major environmental system the sample or specimen came from. The system(s) identified should have a coarse spatial grain, to provide the general environmental context of where the sampling was done (e.g. in the desert or a rainforest). We recommend using subclasses of EnvO's biome class: http://purl.obolibrary.org/obo/ENVO_00000428. EnvO documentation about how to use the field: https://github.com/EnvironmentOntology/envo/wiki/Using-ENVO-with-MIxS.
broker_name	Name of broker for the submission
cell_line	Cell line from which the sample was obtained
cell_type	Cell type from which the sample was obtained

Field	Description
center_name	Submitting centre
checklist	ENA metadata reporting standard used to register the biosample (Checklist used)
codon_start	Indicates the offset of the first complete codon relative to the first base of the coding feature
collected_by	Name of the person who collected the specimen
collecting_institute	Name of the institution to which the person collecting the specimen belongs. Format: Institute Name, Institute Address
collection_date	Date that the specimen was collected
country	Locality of sample isolation: country names, oceans or seas, followed by regions and localities
cram_index_aspera	Aspera links for CRAM index files
cram_index_ftp	FTP links for CRAM index files
cram_index_galaxy	Galaxy links for CRAM index files
cultivar	Cultivar (cultivated variety) of plant from which sample was obtained
culture_collection	Identifier for the sample culture including institute and collection code
dataclass	Sequence data class
depth	The distance below the surface of the water at which a measurement was made or a sample was collected (in metres)
description	Brief sequence description
dev_stage	Sample obtained from an organism in a specific developmental stage
ec_number	Enzyme commission number for enzyme product of sequence
ecotype	A population within a given species displaying traits that reflect adaptation to a local habitat
elevation	The elevation of the sampling site as measured by the vertical distance from mean sea level (in metres)
embl_file	Flat file for the set
environment_biome	Environment (biome). Biomes are defined based on factors such as plant structures, leaf types, plant spacing, and other factors like climate. Examples include: desert, taiga, deciduous woodland, or coral reef
environment_feature	Environmental feature level includes geographic environmental features. Examples include: harbor, cliff, or lake
environment_material	The environmental material level refers to the material that was displaced by the sample, or material in which a sample was embedded, prior to the sampling event. Examples include: air, soil, or water
environmental_medium	Report the environmental material(s) immediately surrounding the sample or specimen at the time of sampling. We recommend using subclasses of 'environmental material'

Field	Description
	(http://purl.obolibrary.org/obo/ENVO_00010483). EnvO documentation about how to use the field: https://github.com/EnvironmentOntology/envo/wiki/Using-ENVO-with-MIxS . Terms from other OBO ontologies are permissible as long as they reference mass/volume nouns (e.g. air, water, blood) and not discrete, countable entities (e.g. a tree, a leaf, a table top).
environmental_package	MIGS/MIMS/MIMARKS extension for reporting of measurements and observations obtained from one or more of the environments where the sample was obtained
environmental_sample	Identifies sequences derived by direct molecular isolation from an environmental DNA sample
event_label	Label given to sampling event
exception	Indicates that the coding region cannot be translated using standard biological rules
experiment	A brief description of the nature of the experimental evidence
experiment_accession	Experiment accession number
experiment_alias	Submitter's name for the experiment
experiment_title	Brief experiment title
experimental_factor	Experimental factors are essentially the variable aspects of an experiment design which can be used to describe an experiment, or set of experiments, in an increasingly detailed manner
fasta_file	FASTA file for the set
fastq_aspera	Aspera links for FASTQ files
fastq_bytes	Size (in bytes) of FASTQ files
fastq_ftp	FTP links for FASTQ files
fastq_galaxy	Galaxy links for FASTQ files
fastq_md5	MD5 checksum of FASTQ files
first_public	Date when made public
function	Function attributed to a sequence
gene	Symbol of the gene corresponding to a sequence region
gene_synonym	Synonymous, replaced, obsolete or former gene symbol
genome_representation	Whether the genome assembly is a full or partial genome
geo_accession	GEO accession
germline	Indicates whether the sample is an unrearranged molecule that was inherited from the parental germline
haplotype	Combination of alleles that are linked together on the same physical chromosome
host	Natural (as opposed to laboratory) host to the organism from which

Field	Description
	sample was obtained
host_body_site	Name of body site where the sample was obtained from, such as a specific organ or tissue
host_common_name	Common name of the natural host organism from which the sample was obtained
host_genotype	Genotype of the host
host_gravidity	Whether or not subject is gravid, including date due or date post-conception where applicable
host_growth_conditions	Literature reference giving growth conditions of the host
host_phenotype	Phenotype of the host
host_scientific_name	Scientific name of the natural (as opposed to laboratory) host to the organism from which sample was obtained
host_sex	Physical sex of the host
host_status	Condition of host (eg. diseased or healthy)
host_subject_id	A unique identifier by which each subject can be referred to, de-identified
host_tax_id	NCBI taxon ID of the host
identified_by	Name of the taxonomist who identified the specimen
inference	A structured description of non-experimental evidence
influenza_test_method	Method by which the current assessment of a sample as flu positive/negative is made
influenza_test_result	Classification of a sample as flu positive or negative based on the test performed and reported
instrument_model	Instrument model used in sequencing experiment
instrument_platform	Instrument platform used in sequencing experiment
investigation_type	The study type targeted by the sequencing
isolate	Individual isolate from which sample was obtained
isolation_source	Describes the physical, environmental and/or local geographical source of the sample
keywords	Keywords associated with sequence
lab_host	Scientific name of the laboratory host used to propagate the source organism for the sample
last_updated	Date when record was last updated
library_layout	Sequencing library layout
library_name	Sequencing library name
library_selection	Method used to select or enrich the material being sequenced
library_source	Source material being sequenced

Field	Description
library_strategy	Sequencing technique intended for the library
local_environmental_context	Report the entity or entities which are in the sample or specimen's local vicinity and which you believe have significant causal influences on your sample or specimen. We recommend using EnvO terms which are of smaller spatial grain than your entry for "broad-scale environmental context". Terms, such as anatomical sites, from other OBO Library ontologies which interoperate with EnvO (e.g. UBERON) are accepted in this field. EnvO documentation about how to use the field: https://github.com/EnvironmentOntology/envo/wiki/Using-ENVO-with-MIxS.
location	Geographic location of isolation of the sample. Latitude and longitude are given in decimal degrees. When using latitude and longitude in the geospatial search functions, positive and negative values should be given to represent direction. When returned in the results report, N/S and E/W are displayed with latitude and longitude.
locus_tag	A submitter-supplied, systematic, stable identifier for a gene and its associated features
map	Map position of feature
marine_region	Geographical origin of the sample as defined by the marine region
marker	Marker classification
master_file	Flat file for the set master
mating_type	Mating type of the organism from which the sequence was obtained
mol_type	in vivo molecule type of the sequence
nominal_length	Average fragmentation size of paired reads
ncbi_reporting_standard	NCBI metadata reporting standard used to register the biosample (Package used)
old_locus_tag	Deprecated submitter-supplied, systematic, stable identifier for a gene and its associated features
operon	Name of the group of contiguous genes transcribed into a single transcript
organelle	Membrane-bound intracellular structure from which the sequence was obtained
other_pathogens_result	Classification of a sample as positive or negative based on the test performed and reported
other_pathogens_tested	Classification of pathogenic organisms other than influenza virus tested in the current assessment of a sample
parent_accession	Parent sequence accession number
parent_study	Parent study accession number
ph	pH measurement

Field	Description
plasmid	Name of naturally occurring plasmid from which the sequence was obtained
product	Name of the product associated with the feature
project_name	Name of the project within which the sequencing was organised
protein_id	A stable protein identifier issued by INSDC
protocol_label	The protocol used to produce the sample
pseudo	Indicates whether the feature is non-functional
pseudo_gene	Indicates that this feature is a pseudogene
read_count	Number of reads
receipt_date	Date on which the sample was received
region	Geographical origin of the sample as defined by the specific region name followed by the locality name
related_sample_accession	Reference to sample(s) that the sample is derived from (derived_from), are equivalent to (same_as), to host sample from symbiont (symbiont_of), included in a group sample (composed_of). The referenced sample(s) should be registered in INSDC. E.g. related_sample_accession="SAMEA111458031:derived_from" to bring the given sample with the given relation; related_sample_accession="SAMEA111458031:*" to bring all the relations of the given sample; related_sample_accession="*:derived_from" to bring all samples which have derived_from relation; related_sample_accession=":" to bring all the samples with all the relations.
ribosomal_slippage	Indicates ribosomal slippage (change to an alternative reading frame) during protein translation
rna_class	Classification of RNA
run_accession	Run accession number
run_alias	Submitter’s name for the run
salinity	Salinity of water at the time of taking the sample
sample_accession	Sample accession number
sample_alias	Submitter’s name for the sample
sample_collection	The method or device employed for collecting the sample
sample_title	Brief sample title
sampling_campaign	The activity within which this sample was collected
sampling_platform	The large infrastructure from which this sample was collected
sampling_site	The site/station where this sample was collection
scientific_name	Scientific name of the organism from which the sample was derived
secondary_sample_accession	Secondary sample accession number

Field	Description
secondary_study_accession	Secondary study accession number
sequence_md5	MD5 checksum of the sequence
sequencing_method	Sequencing method used
serovar	Serological variety of a species characterized by its antigenic properties
set_files	Flat file for the set
sewage_type	Type of sewage based on origin
sex	Sex of the organism from which the sample was obtained
specimen_voucher	Identifier for the sample culture including institute and collection code
sra_aspera	Aspera links for NCBI SRA format files
sra_bytes	Size (in bytes) of NCBI SRA format files
sra_ftp	FTP links for NCBI SRA format files
sra_galaxy	Galaxy links for NCBI SRA format files
sra_md5	MD5 checksum of NCBI SRA format files
standard_name	Accepted standard name for a feature
strain	Strain from which sample was obtained
study_accession	Study accession name
study_alias	Submitter's name for the study
study_description	Detailed sequencing study description
study_title	Brief sequencing study description
sub_species	Name of sub-species of organism from which sample was obtained
sub_strain	Name or identifier of a genetically or otherwise modified strain from which sample was obtained
submission_accession	Submission accession number
submitted_aspera	Aspera links for submitted files
submitted_bytes	Size (in bytes) of submitted files
submitted_format	Format of submitted reads
submitted_ftp	FTP links for submitted files
submitted_galaxy	Galaxy links for submitted files
submitted_host_sex	Physical sex of the host as provided by the submitter. This is in contrast to the "host_sex" field which contains standardised values.
submitted_md5	MD5 checksum of submitted files
submitted_sex	Sex of the organism from which the sample was obtained. This is in contrast to the "sex" field which contains standardised values.
target_gene	Targeted gene or locus name for marker gene studies
tax_division	Taxonomic division of the organism from which the sample was

Field	Description
	obtained
tax_id	NCBI taxon ID of the organism from which the sample was obtained
taxonomy	A virtual field representing NCBI taxonomic classification. Searchable using one of the taxonomy functions
temperature	Temperature of the sample at time of sampling
tissue_lib	Tissue library from which sample was obtained
tissue_type	Tissue type from which the sample was obtained
topology	Sequence topology: circular or linear
trans_splicing	Indicates exons from two RNA molecules are ligated in intermolecular reaction to form mature RNA
transl_except	A single codon translation that does not conform to genetic code
transl_table	Indicates the genetic code table used if other than universal genetic code table
variety	Variety (varietas, a formal Linnaean rank) of organism from which sample was derived

RESTful API Rate Limit

In order to ensure a smooth and fair user experience, we have implemented rate limits on our REST services.

It helps us in maintaining optimal performance and preventing overload on our servers. By regulating the number of requests from individual users, we can ensure that everyone gets a consistent and responsive experience. It also acts as a protective measure against malicious activities such as DDoS attacks and brute-force attempts.

At present we have set the upper limit at 50 requests per second which we think should be sufficient for most use-cases. If the number of requests breaches this limit then the subsequent requests may be rejected with the error "Too Many Requests" (HTTP status code 429).

Examples

All examples given below show the ENA Portal API URL only. Examples of using these URLs with *curl* are given in the Introduction section.

Fetch available results

Fetch the list of results that can be searched against in the pathogen data portal.

```
https://www.ebi.ac.uk/ena/portal/api/results?dataPortal=pathogen
```

Fetch searchable fields

Fetch the list of fields that can be searched for the assembly result.

```
https://www.ebi.ac.uk/ena/portal/api/searchFields?result=assembly
```

Fetch returnable fields

Fetch the list of fields that can be returned in the report for the analysis result

```
https://www.ebi.ac.uk/ena/portal/api/returnFields?result=analysis
```

Fetch controlled vocabulary

Fetch the list of controlled vocabulary for the checklist field. Note that the value and label contain different content. The value column is what should be used in the search query but the label column gives the title of each checklist.

```
https://www.ebi.ac.uk/ena/portal/api/controlledVocab?field=checklist
```

Fetch the list of controlled vocabulary for the instrument_model field. Note that as the value column contains understandable values, the label column holds the same information.

```
https://www.ebi.ac.uk/ena/portal/api/controlledVocab?field=instrument_model
```

Search against public samples

- Find all public samples in the ENA data portal. Return a list of accession and title, ordered by accession.

```
https://www.ebi.ac.uk/ena/portal/api/search?result=sample&limit=0
```

- Find all public samples in the pathogen data portal. Return a list of accession, tax ID, scientific name, collection date, country and location.

```
https://www.ebi.ac.uk/ena/portal/api/search?result=sample&dataPortal=pathogen&fields=sample_accession,tax_id,scientific_name,collection_date,country,location
```

Authenticated search for a dataPortal and results download

- Using CURL, submit a search with the username and password.

```
curl -X GET --header 'Accept: text/plain' -u 'username:password'
'https://www.ebi.ac.uk/ena/portal/api/search?result=read_run&limit=0&d
ataPortal=pathogen'
```

Search against data hub (DCC) data only

- Find all read data that is registered with data hub dcc_chopin. Return a list of the run accessions with FTP FASTQ file links. Include the MD5 checksums for the files. Note that this example URL doesn't include the data hub account information, this must be included in the request header. Note also that the URL has been changed to HTTPS so that the account information is secure during transit.

https://www.ebi.ac.uk/ena/portal/api/search?result=read_run&fields=fastq_ftp,fastq_md5&dccDataOnly&limit=0

To build a query for data hub(DCC) data only, one can also use the Advanced Search GUI.

Open the ena browser and navigate to Search -> Advanced Search. In the first page, choose the appropriate data type and click Next.

This will take you to the Query Builder page. There you can search for your fields in the "Type to filter query params" box and build a query.

If you go further next you will go to the Fields page, where you can select the fields you would want to filter for your search results.

Once you have provided your options, you can copy the request using Copy Curl Request Option and add authorization details to it and retrieve your results.

Search for read data using sample fields

- Find all public Salmonella read data collected in 2016. Return a list of the run accessions with FTP FASTQ file links. Include the MD5 checksums for the files.

```
https://www.ebi.ac.uk/ena/portal/api/search?result=read_run&query=coll
ection_date>=2016-01-01%20AND%20collection_date<=2016-12-31%20AND%20ta
x_tree(590)&fields=fastq_ftp,fastq_md5&limit=0
```

- Perform the same search, but return the sample accession and collection date in addition to the read file data. Note that while the above search will return one row for each run accession, the following search will return one row for each run-sample combination.

```
https://www.ebi.ac.uk/ena/portal/api/search?result=read_run&query=coll
ection_date>=2016-01-01%20AND%20collection_date<=2016-12-31%20AND%20ta
x_tree(590)&fields=fastq_ftp,fastq_md5,sample_accession,collection_dat
e&limit=0
```