

# ScienceBase Catalog v2 Search for Developers

Note: Try this PYSB Jupyter [notebook](#) first, where you can try live searches.

## Basics

### Folder/community Search

### Extent/Spatial Query Search

### Allowed Search Filters

#### Lucene Query

Everything in Sciencebase Catalog is an item.

If you know the item id

<https://www.sciencebase.gov/catalog/item/4fb51859e4b04cb937751c8c>

Search for Items containing the word "trout" using the keyword q search.

<https://www.sciencebase.gov/catalog/items?q=trout>

Find all the descendants of an item using the ancestors parameter.

<https://www.sciencebase.gov/catalog/items?q=&format=json&filter=ancestors=5952995fe4b062508e3c770d>

## Search for Items in Folder/Community

Search folder id (itemID) from URL:

<https://www.sciencebase.gov/catalog/item/4fb51859e4b04cb937751c8c>

Search for all items in folder:

<https://www.sciencebase.gov/catalog/items?q=&items?q=&filter=parentId=4fb51859e4b04cb937751c8c>

Search for all items in multiple folders:

<https://www.sciencebase.gov/catalog/items?q=&items?q=&filter=parentId=4fb51859e4b04cb937751c8c&filter=parentId=4f4e4771e4b07f02db47e1e4&conjunction=parentId=OR>

## Search with Filters:

### Filters

Perform a filter search by adding the parameter *filter=*.

*Should* instead of *filter* performs an "OR" search

### browseType

Find items by ScienceBase browse type.

filter:browseType=ArcGIS Service Definition

### browseCategory

Find items by ScienceBase browse category.

'filter': 'browseCategory=Data

### party

```
{'filter': 'party=(name:FT COLLINS SCI CTR)', 'fields': 'contacts'}
```

### partyWithName

```
{'filter': 'partyWithName=17368_FT COLLINS SCI CTR')}
```

### itemIdentifier

```
{"type": <type>, "scheme": <scheme>, "key": <key>}
```

```
filter=itemIdentifier={"type": "DOI", "scheme": "https://www.sciencebase.gov/vocab/category/item/identifier"}
```

### facets

```
{'filter': 'facets.facetName=Shapefile'}
```

### dateRange

```
format is dateRange={"choice": ?, "start": ?, "end": ?, "dateType": ?}dateRange={"dateType": "End", "choice": "year"}
```

```
filter=dateRange={"start": "2014-03-04", "end": "2020-03-04", "dateType": "End Date"}
```

```
# Find all items created in the past month
```

```
response = sb.find_items({'filter': 'dateRange={"dateType": "creation", "choice": "month"}'})  
print("Found %s items" % response['total'])
```

### tags

```
Simple tag name search filter=tags=water
```

```
Format is tags={"name": name, "type": type, "scheme": scheme}tags=water
```

```
filter=tags={"name": "water"}
```

### Filter Conjunction

```
Find items tagged with either "water" or "birds"
```

```
'filter': ['tags=water', 'tags=birds'], 'conjunction': 'tags=OR'})
```

```
# Can also accomplish the same thing using 'should' instead of 'filter'
```

```
response = sb.find_items(  
'should': ['tags=water', 'tags=birds']  
)
```

### Multiple Filters Example:

```
https://sciencebase.gov/catalog/items?filter=tags={"type": "Data Type", "scheme": "NGGDPP"}&q=&max=20&filter0=facets.facetName=NGGDPPItem&filter1=browseCategory=Physical Item
```

### Allowed Filters:

```
"ancestors":
"ancestorsExcludingLinks":
"tags":
"tagsIq":
"parentIid":
"parentIidExcludingLinks":
"parentIidIncludingLinks":
"linkParentIid":
"ids":
"itemIdentifier":
"itemIdentifierKeys":
"itemIdentifierTypes":
"itemIdentifierSchemes":
"files.name":
"files.originalMetadata":
"facets.files.originalMetadata":
"webLinks":
"party":
"contactNameLike":
"projectStatus":
"dateRange":
"systemType"
"spatialQuery":

"spatialQueryOnExtents":

"spatialQueryOnXFields":

"extentQuery":
```

## Parameters:

lq

q

format (default/blank is html)

e.g. format=json

includeFeatures

GET /catalog/items?includeFeatures=1&q=&format=json&fields=&max=10&loadInstances=true&filter=parentIid=4f4e476ae4b07f02db47e13b HTTP/1.1"  
200 10334

```
if (params.includeFeatures) {
  query.should(allQuery)
} else {
  query.must(allQuery)
}
```

onlyWritable

writable

(both are permissions-based limitations)

getList

?

oldItemType

oldBasicItemType

folderSearchId

folderId

parentId

```

searchExtent
_featured
enableHighlighting
max
offset
sort
order

facets
facetDetail
facetSort == "alphabetical"
facetTagTypes
facetTagSchemes

```

### Lucene Query

lucene query on the Item JSON model using lq parameter. Requires the q parameter to be specified as well, use " if you don't want results restricted.

```

#_exists_ lucene query
items?scheme=BASIS&q=&lq=_exists_:facets.annualGuidance

```

```

# Find items that are tagged with "birds" AND "water, OR are tagged with "WY"
response = sb.find_items({
'q': "",
'lq': '(tags.name(+birds) AND tags.name(+water)) OR tags.name(+WY)'
})

```

```

# Find items with "USGS Data Release Products" in the title, and "This" in the body
response = sb.find_items({
'q': "",
'lq': '(title:"USGS Data Release Products" AND body:"data products")'
})

```

```

# Find items that have an extent field. This will find footprinted items.
response = sb.find_items({
'q': "",
'lq': '_exists_:extents'
})

```

```

# Find items containing files flagged as original metadata
response = sb.find_items({
'q': "",
'lq': 'files.originalMetadata:true'
})

```

### Search Result Control

```

sort -- sort by title, dateCreated, lastUpdated, or firstContact
order -- asc desc
max -- max number of items to return per request. Note, ScienceBase enforces an upper limit of 1000.
offset -- paging parameter. Return items beginning at offset.
fields -- specifies which fields in the sbJSON to return, e.g. title,summary,distributionLinks,webLinks,previewImage
fieldset --
loadInstances -- loads the result items from mongoDB vs. just responding with the elasticsearch response. Some fields, such as distributionLinks, are only available from mongoDB, thus the loadInstances parameter would be required to fetch those.

```

```
# Return three items from the third page of search results for a "water" keyword search, and only return the
# 'title' field
response = sb.find_items({
  'q': 'water',
  'offset': 3,
  'max': 3,
  'fields': 'title'
})
```

### Search Facets

These parameters control the searchFacets section of the results JSON, allowing you to retrieve elasticsearch facet information about your query results. These are not to be confused with facets in the Item sbJSON.

enableFacets -- enable the return of the searchFacets section of the results JSON.

facets -- specifies which search facets to return

facetSize -- specifies max the number of entries returned for each search facet

facetTagTypes -- return search facets for the given tag types, e.g. facetTagTypes=Order,Family,Taxon

facetTagSchemes -- return search facets for the given tag schemes

```
# Search by the keyword "water." Return the searchFacets section for browseCategory.
```

```
response = sb.find_items({
  'q': 'water',
  'enableFacets': True,
  'facets': 'browseCategory'
})
```

```
# Find SDC-tagged items
```

```
items?filter=tags={"type":"Harvest Set","name":"usgs science data catalog (sdc)"}&q=&lq=files.originalMetadata:true
items?should=ancestorsExcludingLinks=<itemid>&should=ids=<itemid>
```

## Spatial Query

This query type is a query filter in the following form:

```
https://www.sciencebase.gov/catalog/items/?q=&filter=spatialQuery=<query>
```

Spatial search supports only the following types of shapes in any format:

- point and multipoint
- linestring
- polygon and multipolygon
- envelope

The query supports either WKT or GeoJSON input spatial data in one of four different forms:

- Raw WKT

```
POINT(-105 43)
```

- JSON with WKT and relation

```
{
  wkt: "POINT(-105 43)",
  relation: "disjoint"
}
```

- Raw GeoJSON

```
{
  type: "point",
  coordinates: [-105.078056, 40.559167]
}
```

- JSON with GeoJSON and relation

```
{
  shape: {
    type: "point",
    coordinates: [-105.078056, 40.559167]
  },
  relation: "disjoint"
}
```

The relation field must be one of:

- intersects
- disjoint
- within

# Extent polygon search

```
items?q=&filter=spatialQuery={"wkt":"POLYGON%20((-102 37,-102 41,-109 41,-109 37,-102 37),(-104 38,-106 38,-105 39,-104 38))","relation":"intersects","fields":"Extents"}
```

# [Maps.usgs.gov/search](https://maps.usgs.gov/search) spatialQuery

```
items?q=water&browseCategory=Map&max=20&format=json&fields=webLinks,title,summary
```

## Search via geoJSON

```
'filter': 'spatialQuery={type:"envelope",coordinates:[[-104.7756918907963,42.49482654800248],
[-99.85381689079756,40.11845545879961]]}'
})
```

## extentQuery

Find items whose footprint intersects Colorado (extent 36) with a buffer of .125 degrees.

```
{'filter': 'extentQuery={"extent":36,"relation":"intersects","buffer":".125"}}
```

## Extent Search

This query type is a query filter in the following form:

```
https://www.sciencebase.gov/catalog/items/?q=&filter=extentQuery=<query>
```

There are two forms the contents of the query can take:

- 
- Extent ID

```
36
```

- JSON with extent ID and relation type

```
{
  extent: 36,
  relation: "disjoint"
}
```

The `relation` field must be one of:

- `intersects`
- `disjoint`
- `within`

## URL for Extents

```
https://www.sciencebase.gov/catalog/extent/2004981
Accept: application/vnd.geo+json for geojson
```