Simplified Query Syntax

Open Data Portal API

Contents

Open Data Portal Patent API Syntax

Reserved Words

API Defined Keywords

Below are the list of reserve key words either used as query parameter in GET search request or as an parameter in POST json request.

q	parameter to search a value across multiple field or opensearch simple query string query
filters	parameter for filter search result
rangeFilters	parameter to filter data based on range of values
sort	sort parameter in search request
fields	limit the response fields to be returned in response
limit	number of results to be returned. Default size is 25
offset	starting position of the records
facets	Parameter that supply list of fields to be faceted (aggregated) upon
:	Works as separator for range filter values in GET search request. Refer to

Opensearch Defined Operators/Keywords

Below is the list of reserve key words and operators defined in opensearch. These keywords are mainly applied in q parameter which accepts query in **domain-specific language** (DSL) simple query string query

Operator/Keyword	Explanation
AND	Query string of 'Patented **AND** Design'
OR	Query string of 'Patented OR Design' means search for word Patented OR Design

NOT	Defines NOT operator
*	Prefix query or wildcard search
?	Wild card search single character
"	Wraps several terms into a phrase (e.g. "Patented Design").
:	Works as separator between fields name and it's value for simple query
>, <, >=, <=	Less than, Greater than, Greater than and Less than

Patent File Wrapper Data Search Guide

POST search request

POST /api/v1/patent/applications/search

Below is the sample POST search request and explanation of different parameters in request body.

```
{
  "q": "applicationMetaData.applicationTypeLabelName:Utility",
  "filters": [
    {
      "name": "applicationMetaData.applicationStatusDescriptionText",
      "value": [
        "Patented Case"
      ]
    }
  ],
  "rangeFilters": [
    {
      "field": "applicationMetaData.grantDate",
      "valueFrom": "2010-08-04",
      "valueTo": "2022-08-04"
```

```
      }
    ],
    "sort": [
      {
        "field": "applicationMetaData.filingDate",
        "order": "desc"
      }
    ],
    "fields": [
      "applicationNumberText",
      "correspondenceAddressBag",
      "applicationMetaData.filingDate"
    ],
    "pagination": {
      "offset": 0,
      "limit": 25
    },
    "facets": [
      "applicationMetaData.applicationTypeLabelName",
      "applicationMetaData.applicationStatusCode"
    ]
  }
```

Note:

All parameters (q, filters, rangeFilters, fields, facets, pagination) are optional. Even a blank POST request ({}) is also a valid POST search request.

A blank POST request is equivalent to below request.

POST {}	{

<table>
<tr><td></td><td>

```
"q": null,

"filters": null,

"rangeFilters": null,

"sort": [

  {

    "field": "applicationMetaData.filingDate",

    "order": "desc"

  }

],

"fields": null,

"pagination": {

  "offset": 0,

  "limit": 25

 }

}
```

</td></tr>
</table>

q parameter

q parameter of POST request supports two forms of search.

Free Form Search – Search a value against all searchable fields in data set. In another way free form search is more relevant when you know what to search but not where (which data field) to search.

Example:

<table>
<tr><td>GET</td><td>https://api.uspto.gov/api/v1/patent/applications/search?q=Design</td></tr>
<tr><td>POST</td><td>

```
{

   "q": "Design",
```

</td></tr>
</table>

```
       "pagination": {

         "offset": "0",

         "limit": "25"

        }

       }
```

Simple Query String Query - q parameter accepts search query in *domain-specific language (DSL)* simple query string query.

GET	https://api.uspto.gov/api/v1/patent/applications/search?q= applicationMetaData.applicationTypeLabelName:Design
POST	```{

    "q": "applicationMetaData.applicationTypeLabelName:Design ",

    "pagination": {

      "offset": "0",

      "limit": "25"

     }

    }``` |

q parameter query examples

Free Form Search

In free form search, search phrase is searched across all searchable fields.

Operator	Query	Explanation
Single Phrase Search	"q": "Patented"	Search 'Patented' across all searchable fields
OR (Space)	"q": "Patented Design"	Search 'Patented' OR 'Design' across all searchable fields

OR	"q": "Patented OR Design"	Search 'Patented' OR 'Design' across all searchable fields
Search Phrases (")	"q": "\"Patented Case\""	Search 'Patented Case' across all searchable fields
AND	"q": "Patented AND Design"	Search 'Patented' AND 'Design' across all searchable field
NOT	"q": "\"Patented Case\" NOT Design"	Search 'Patented Case' across all searchable fields but exclude records where 'Design' is found
Wildcard * (Zero or More)	"q": "Technolo*"	Search across all fields for value that contains 'Technolo'
Wildcard ? (Any Single Character)	"q": "Nordi?k "	Search across all fields for value 'Nordi?k' where ? could any one single character

Query String Query Examples

Operator	Query	Explanation
	"q": "applicationMetaData.entityStatusData.businessEntityStatusCategory:Small"	Find data where entity status category is small
OR Single field multi-value	"q": "applicationMetaData.applicationTypeLabelName:(Design OR Plant)"	Search data where application type label is either 'Design' or 'Plant'
OR multi-field	applicationMetaData.applicationTypeLabelName:Design OR applicationMetaData.entityStatusData.:Micro	Search data where applicationTypeLabelName is Design OR businessEntityStatusCategory is Micro
AND multi-field	"q": "applicationMetaData.applicationStatusDescriptionText \"Patented Case\" AND applicationMetaData.entityStatusData.businessEntityStatusCategory:Micro"	Search data where applicationStatusDescriptionText is 'Patented Case' and businessEntityStatusCategory is Micro

Wildcard *	"q": "applicationMetaData.firstApplicantName:Technolog*"	Search data where firstApplicantName contains 'Technolog'
Wildcard ?	"q": "applicationMetaData.examinerNameText:ANDERS?N"	Search data where examinerNameText matches 'ANDERS?N' where ? can be replace with any single chanreacter
Ranges >=	"q": "applicationMetaData.applicationStatusDate:>=2024-02-20"	Search data where applicationStatusDate is greater than or equal to 2024-02-20
Ranges number[min TO max]	"q": "applicationMetaData.applicationConfirmationNumber:[2700 TO 2710]"	Search data where applicationConfirmationNumber is from 2700 to 2710
Ranges dates[min TO max]	"q": "applicationMetaData.filingDate:[2024-01-01 TO 2024-08-30]"	Search data where filingDate is from 2024-01-01 to 2024-08-30

filters parameter

Filter request parameter is used for narrowing down search result.

Rules

1- Filters parameter is optional.
2- Filter name which is an actual dataset field and field value(s) must be provided.
3- Multiple filters can be added, and each filter is equivalent to AND condition.
4- Multiple values can be added for a single field.
5- A qualified field name must be added

Example

Search dataset where applicationStatus is 'Patented Case'	**"filters": [** { "name": "applicationMetaData.applicationStatusDescriptionText",

	"value": ["Patented Case"] }]

filters parameter query examples

Single Filter	{
	
	 "filters": [
	
	  {
	
	    "name": "applicationMetaData.applicationTypeLabelName",
	
	    "value": ["Utility"]
	
	  }
	
	]
	
	}

Multiple Filter	{
	
	 "filters": [
	
	  {
	
	    "name": "applicationMetaData.entityStatusData.businessEntityStatusCategory",
	
	    "value": ["Small"]
	
	  },
	
	  {
	
	    "name": "applicationMetaData.applicationStatusCode",
	
	    "value": ["150","164"]

	``
	  }
	
	  ]
	
	}
	``

Single Filter Multi-Value	```
	{
	
	  "q": "",
	
	  "filters": [
	
	    {
	
	      "name": "applicationMetaData.applicationTypeLabelName",
	
	      "value": ["Design","Re-Issue"]
	
	    }
	
	  ]
	
	}
	``

rangeFilters

Range filters are used to narrow down search result by supplying range value

Rules –

1- Range filters is optional.
2- Field which is an actual dataset field and field values must be provided.
3- Multiple range filters can be added, and each range filter is equivalent to AND condition.
4- rangeFilters must include '**valueFrom**' and '**valueTo**'.
5- Search result is **inclusive** of **valueFrom** and **valueTo**.
6- Range filters is valid for date fields and number fields only.
7- Combination of date field and number field range filter request can be added.

Example: Date Range

Search dataset where grantDate is from **2010-08-04** to **2022-08-04**	```
	"rangeFilters": [
	
	  {
	
	    "field": "applicationMetaData.grantDate",
	
	    "valueFrom": "2010-08-04",
	``

	"valueTo": "2022-08-04"
	}
	]

Example: Number Range

Filter dataset where applicationStatusCode is between **140** and **150**	**"rangeFilters": [**
	{
	"field": "applicationMetaData.applicationStatusCode",
	"valueFrom": "140",
	"valueTo": "150"
	}
	]

rangerFilters query examples

Single Range Filter	```{ ```
	``` "rangeFilters": [```
	``` { ```
	``` "field": "applicationMetaData.filingDate", ```
	``` "valueFrom": "2019-08-04", ```
	``` "valueTo": "2019-08-05" ```
	``` } ```
	```] ```
	```} ```
Multiple Range Filters	```{ ```
	``` "rangeFilters": [```

```
    {
      "field": "applicationMetaData.filingDate",
      "valueFrom": "2019-08-04",
      "valueTo": "2019-08-05"
    },
    {
      "field": "applicationMetaData.applicationStatusCode",
      "valueFrom": "150",
      "valueTo": "200"
    }
  ]
}
```

sort parameter

Sort data set based on specified fields.

Rules –

1- Sort parameter is optional.
2- When no sort order provided default sort option on field filingDate in desc order is applied.
3- Field to be sorted on and sort order must be provided.
4- Text fields are not valid fields for sorting.
5- Acceptable value for order are desc, asc, DESC and ASC.

Sample Request:

Sort data set by filingDate in descending order	**"sort": [**
	
	  {
	
	    "field": "applicationMetaData.filingDate",
	
	    "order": "desc"
	
	  }
	
	]

Sort parameter query examples

| Sort On Single Field | ```
{
  "sort": [
    {
      "field": "applicationMetaData.filingDate",
      "order": "desc"
    }
  ]
}
``` |
|---|---|
| Sort On Multiple Fields | ```
"sort": [
    {
      "field": "applicationMetaData.filingDate",
      "order": "desc"
    },
    {
      "field": "applicationMetaData.applicationStatusCode",
      "order": "asc"
    }
]
``` |

fields parameter

fields parameter is used to limit the list of data fields to be included in search response.

Rules

1- Fields parameter is optional.
2- When no fields parameter provided all available data fields are returned.
3- Include the list of fields to be included in api response.
4- List of fields must be a valid field in dataset.
5- Invalid field name will not result in error rather it will be ignored, and only valid fields will be included in response.

Example

Filter dataset where applicationTypeLabelName is 'Re-Issue' but only **include fields** in response 'applicationMetaData.filingDate', 'applicationMetaData.inventorBag', 'applicationNumberText'	```{ "filters": [{ "name": "applicationMetaData.applicationTypeLabelName", "value": ["Re-Issue"] }], "fields": ["applicationMetaData.filingDate", "applicationMetaData.inventorBag", "applicationNumberText"] }```

Pagination (offset & limit) parameter

Limit the search result by providing pagination details

Rules

1- Pagination section of post request is optional.
2- When no pagination details added in request default pagination values are in effect.
3- Default offset value is 0 and default limit value is 25.
4- When any of the pagination parameter (offset or limit) is missing default value is applied.

Example

Limit the search result to 25 records starting from 0	**"pagination":** {
	"offset": 0,
	"limit": 25
	}

facets parameter

facets parameter is added in search request is added to apply faceting (aggregation) on supplied list of fields. Search result includes the field name and count of different type of values. Below is the sample of faceting result in search result response.

```
........
"facets": {
    "applicationMetaData.entityStatusData.businessEntityStatusCategory": [
      {
          "value": "Regular Undiscounted",
          "count": 4828697
      },
      {
          "value": "Small",
          "count": 1813470
      },
      {
          "value": "Micro",
          "count": 167301
      }
    ]
```

```
}
.........
```

Rules

1- Facets parameter is optional
2- Faceting (aggregation) on text field is not supported.
3- List of fields of fields should be separated by comma (**,**)

Example

Aggregate the result based on applicationTypeLabelName and applicationStatusCode	**"facets": [**
	
	    "applicationMetaData.applicationTypeLabelName",
	
	    "applicationMetaData.applicationStatusCode"
	
	]

GET search request

GET /api/v1/patent/applications/search

Note – Some character e.g. Space ( ) , Quotation (") cannot be part of URL and must be UTF-8 character encoded. Most of the REST api test tools (e.g. Thunder Client) encodes these characters before submitting request to server. Please refer this to get some more information about URL encoding.

Below are the list of some common characters and it's encoded values.

Character	Encoded Value
Space	%20
"	%22

#	%23
%	%25
<	%3C
>	%3E
|	%7C

q query parameter

q query parameter is used to specify multiple arguments delineated by regular expressions directly in the query string.

Free Form Search – Used this form when you know what to search but don't know where to search

Simple Query String Query Search – Use this form when you know what to search and where to search.

Sample Query

Free Form Search	https://api.uspto.gov/api/v1/patent/applications/search?q=Utility	Search phrase Utility across all searchable fields.
Simple Query String Search	https://api.uspto.gov/api/v1/patent/applications/search?q= applicationMetaData.applicationTypeLabelName:Utility	Search Phrase Utility against field applicationTypeLabelName

q query parameter examples

AND	https:// api.uspto.gov/api/v1/patent/applications/search?q=Utility%20AND%20Design

AND	https://api.uspto.gov/api/v1/patent/applications/search?q=applicationMetaData.applicationTypeLabelName:Utility%20AND%20Design
OR	Search Small OR Micro https://api.uspto.gov/api/v1/patent/applications/search?q=Small%20OR%20Micro
OR	https://api.uspto.gov /api/v1/patent/applications/search?q= q=applicationMetaData.entityStatusData.businessEntityStatusCategory:%20Small%20OR%20Micro
Multiple tokens as single phrase (")	Search phrase "Patented Case" Free Form Search https://api.uspto.gov/api/v1/patent/applications/search?q=%22Patented%20Case%22
	Search phrase "Patented Case" Query String Query Search https://api.uspto.gov/api/v1/patent/applications/search?q=applicationMetaData.applicationStatusDescriptionText:%22Patented%20Case%22
Wildcard (*) Zero or More	https://api.uspto.gov/api/v1/patent/applications/search?q=Googl*
	https://api.uspto.gov/api/v1/patent/applications/search?q=applicationMetaData.inventorBag.inventorNameText:Googl*
Wildcard (?) Single	https://api.uspto.gov/api/v1/patent/applications/search?q=Nordi?k

Chara cter	https://api.uspto.gov/api/v1/patent/applications/search?q=applicationMetaData.firstApp licantName:Nordi?k

Filters query parameter

Sample Query

filter=applicationMetaData.firstInventorToFileIndicator Y https://api.uspto.gov/api/v1/patent/applications/search? filters=applicationMetaData.firstInventorToFileIndicator%20Y	Filter dataset where firstInventorToFileIndicator value is Y

filterQuery parameter examples

https://api.uspto.gov/api/v1/patent/applications/search? filters=applicationMetaData.entityStatusData.businessEntityStatusCategory%20Micro	Single Filter Single Value
Filter data where applicationTypeLabelName is Re-Issue and businessEntityStatusCategory is Small https://api.uspto.gov/api/v1/patent/applications/search? filters=applicationMetaData.applicationTypeLabelName%20Re-Issue&filters=applicationMetaData.entityStatusData.businessEntityStatusCategory%20Small	Multiple Filter And Each Filter has Single Value To Search
applicationMetaData.applicationTypeCode UTL,DES	Single Filter

https://api.uspto.gov/api/v1/patent/applications/search? filters=applicationMetaData.applicationTypeCode%20UTL%2CDES	Multivalu e

rangeFilters query parameter

Rules

1- Add rangeFilter by adding field name followed by range from and to value
2- From and to value must be separated by : (Colon)
3- Search results are inclusive of from and to value

Sample Query

Example	rangeFilters=applicationMetaData.grantDate 2010-01-01:2011-01-01 https://api.uspto.gov/api/v1/patent/applications/search? rangeFilters=applicationMetaData.grantDate%202010-01-01%3A2011-01-01

rangeFilters query examples

Filter data where grant date is between 2010-01-01 and 2011-01-01 https://api.uspto.gov/api/v1/patent/applications/search?rangeFilters= applicationMetaData.grantDate%202010-01-01%3A2011-01-01	Date Rang e Filter
Filter data where applicationStatusCode is between 150 and 153 https://api.uspto.gov/api/v1/patent/applications/search?rangeFilters=applicationMetaDat a.applicationStatusCode%20150%3A153	Num ber rang e filter

sort query parameter

Rules

1- Sort query parameter is optional.
2- Use sort query parameter followed by field name and sort order
3- Field name and sort order must be separated by space
4- To add multiple sort option, add repeated sort query parameter

applicationMetaData.applicationStatusDate asc https://api.uspto.gov/api/v1/patent/applications/search?sort=applicationMetaData.applicationStatusDate%20asc	Sort on single field
https://api.uspto.gov/api/v1/patent/applications/search?sort=applicationMetaData.applicationStatusDate%20asc&sort=applicationMetaData.grantDate%20desc	Sort on multiple field

fields query parameter

Rules

1- Use query parameter fields followed by comma (,) separated list of fields.
2- Qualified field name must be used. e.g. to include patentNumber use applicationMetaData.patentNumber
3- If include all fields of a parent field, adding only parent field name will include all available field will be included. e.g. Adding correspondenceAddressBag in field list will include all child fields as well.
4- Wildcard can also be added to include all matching fields. e.g. to include all fields that has Date, *Date* can be added in field list.
5- Field names are case sensitive

Sample Query

```
https://api.uspto.gov/api/v1/patent/applications/search?
fields=applicationNumberText%2CapplicationMetaData.patentNumber%2CapplicationMetaData.
applicationTypeCode
```

fields query examples

https://api.uspto.gov/api/v1/patent/applications/search?fields=eventDataBag	Include all fields defined under eventDataBag
Continuity https://api.uspto.gov/api/v1/patent/applications/search?fields=%2AContinuity%2A	Include fields that has term Continuity in field name

pagination (offset & limit) query parameter

Rules

1- If no pagination (offset & limit) value provide default value 0 for offset and 25 for limit is applied.
2- If any of the value from offset and limit is missing default value is applied for that field.
3- Only non-negative integer values are valid for offset and pagination.

offset=10&limit=50 https://api.uspto.gov/api/v1/patent/applications/search?offset=10&limit=50	Offset - Position in the dataset of a particular record Limit – Total number of records to be returned

Field Dictionary

To view list of data fields and dataset json structure please refer to schema document here.

FAQs

1- Is it possible to search multiple values for a field data?
   Yes. To search multiple values for a field add list of values as an array to filter parameter as shown below.

```
"filters": [
   {
      "name": "applicationMetaData.applicationTypeLabelName",

      "value": [ "Design","Re-Issue" ]

   }
```

2- Are all parameters in POST /search request are mandatory?
   No. All parameters are optional. When no parameters are provided default values are set to some of the parameters. Even an empty json request payload is a valid requet.

   An empty ({}) POST request is equivalent to below to as below.

Blank Request	Equivalent To

```
{}                          {
                              "q": null,

                              "filters": null,

                              "rangeFilters": null,

                              "sort": [

                                {

                                  "field": "applicationMetaData.filingDate",

                                  "order": "desc"

                                }

                              ],

                              "fields": null,

                              "pagination": {

                                "offset": 0,

                                "limit": 25

                              }

                            }
```