

Brief Search

Brief Search

General Purpose

The search/brief request performs a search request.

Interface Type

X-Service

Table of Contents

- [Brief Search](#)
 - [General Purpose](#)
 - [Interface Type](#)
 - [Table of Contents](#)
 - [Product Version Compatibility](#)
 - [Input Parameters Description](#)
 - [Input Parameters](#)
 - [Notes](#)
 - [Output Parameters](#)
 - [Input Example](#)
 - [Output Example](#)

Product Version Compatibility

Version 2.0 and later

Input Parameters Description

The first four parameters in the list below are only relevant for restricted scopes. A restricted scope is authorized according to the following parameters: **institution**, **group**, and **onCampus**. Because the institution parameter defines which PC key is sent to Primo Central, it is most important when searching in Primo Central.

Input Parameters

Parameter	Description
institution	The institution code. Relevant for restricted scopes or for when searching against Primo Central.

ip	The client IP. Will help to identify the institution in cases the institution was not identified (according to the IP range associated with institution) Not relevant for Deep Links
group	Name of the user group. Relevant for restricted scopes.
onCampus	On/off campus. Relevant for restricted scopes. possible value: false or true Not relevant for Deep Links

query

The query parameter is a combination of three parameters that are separated by a comma:

&query=<field>,<precision>,<value>

- 1) field - The field to search against
- 2) precision - A precision operator (i.e exact/contains/begins_with)
- 3) value - Any value

Examples:

&query=facet_topic,exact,Nonfiction

&query=any,contains,queens+of+the+stone+age (the value contains more than one term in it)

Note: The query parameter can be used more than once in the URL thus causing for an **AND** correlation between the

For example:

&query=creator,contains,comm&query=facet_creator,exact,United+States

1) <field> possible values:

The field names to be used are field names used by the search engine. As such their names are a bit different than what you would expect. The below will list the relevant field names:

a) any -

Will search against most of the fields at once.

Example: **&query=any,contains,queens**

b) regular fields -

(taken from the code table: "[Basic Index Fields](#)"):

desc, title, creator, sub (subject), rtype (resource type), isbn, issn, rectype (record type), dlink (down link), ftext (full text), general, toc, fmt, lang, cdate (creation date), sid (source ID), rid (record ID), addsrcrid (additional source ID), addtitle, pnstype, alttitle, abstract, and fiction

Example: **&query=rid,exact,PRIMO1234**

b) facet fields -

(taken from the code table: "[Facets Code Fields](#)"):

facet_creator, facet_lang, facet_rtype, facet_pfilter, facet_topic, facet_creationdate, facet_dcc, facet_lcc, facet_rvk, facet_tlevel, facet_domain, facet_fsize, facet_fmt, facet_frbrgroupid, facet_frbrtype, facet_local1 - facet_local50

Example: **&query=facet_creator,william+s+burroughs**

2) <precision> possible values:

NOTE: for facets, the only possible precision value is exact (i.e. **&query=facet_topic,exact,Nonfiction**)

- a) exact** - When searching for a phrase (more than one term, i.e "queens of the stone age") and we want an exact match for this phrase.

Note: When querying for one term there is no difference between exact and contains

Example: **&query=title,exact,queens+of+the+stone+age**

- b) contains** - When searching for one term, will return results for any field containing this term. When searching for a phrase, will return results for any field containing at least one of the given terms. For example:

&query=title,contains,queens+of+the+stone+age

will return titles like:

"the **age** of life", "kings and **queens**" or
"age and health of **queens**"

- c) begins_with** - Works only against the sort title field (swstitle)!

The swstitle was not mentioned above. The swstitle is the title under the sort section in the PNX.

Note: When using this precision, there is no meaning to the field you are using, as it will always search against the swstitle field!

3) <value> possible values:

Any value is allowed.

Values including commas in them should **replace the comma with a space**.

query_inc (Multiple facets - Query include)	<p>Supported from version V4.0 only. Query a facet including multiple values:</p> <p>&query_inc=<field>,<precision>,<value1, value2,...></p> <p>Example: &query_inc=facet_topic,exact,women,science</p> <p>field - any facet fields - (taken from the code table: "Facets Code Fields"): facet_creator, facet_lang, facet_rtype, facet_pfilter, facet_topic, facet_creationdate, facet_dcc, facet_lcc, facet_rvk, facet_tlevel, facet_domain, facet_fsize, facet_fmt, facet_frbrgroupid, facet_frbrtype, facet_local1 - facet_local50</p> <p>precision - The precision operator MUST only hold the "exact" value (other values are not supported).</p> <p>value1, value2.. - any value or multiple values separated by a comma. (Values including commas in them should repeat comma with a space)</p> <p>The query created will generate an OR query between the values: i.e. (facet_topic=value1 OR facet_topic=value2 OR ...)</p> <p>You can use the query_inc multiple times in the query: i.e. &query_inc=facet_topic,exact,women,science&query_inc=facet_lang,exact,eng,heb</p>
query_exc (Multiple facets - Query exclude)	<p>Supported from version V4.0 only. Query a facet excluding multiple values:</p> <p>&query_exc=<field>,<precision>,<value1, value2,...></p> <p>Example: &query_exc=facet_topic,exact,women,science</p> <p>field - any facet fields - (taken from the code table: "Facets Code Fields"): facet_creator, facet_lang, facet_rtype, facet_pfilter, facet_topic, facet_creationdate, facet_dcc, facet_lcc, facet_rvk, facet_tlevel, facet_domain, facet_fsize, facet_fmt, facet_frbrgroupid, facet_frbrtype, facet_local1 - facet_local50</p> <p>precision - The precision operator MUST only hold the "exact" value (other values are not supported).</p> <p>value1, value2.. - any value or multiple values separated by a comma. (Values including commas in them should repeat comma with a space)</p> <p>The query created will generate an AND query between the values: i.e. (facet_topic=value1 AND facet_topic=value2 AND ...)</p> <p>You can use the query_exc multiple times in the query: i.e. &query_exc=facet_topic,exact,women,science&query_exc=facet_lang,exact,eng,heb</p>
indx	The position number of the record to begin from when retrieving the results (For example, in order to display page #2, we want to set indx=11, meaning start retrieving records from position 11 in set)
bulkSize	The amount of records to retrieve from the position set for indx (In order to display page #2 when we display 10 records in a page: indx=11 bulkSize=10)
dym	<p>Whether to include in the response a "Did you mean" suggestion. The response will include (if at all) only 1 suggestion!</p> <p>Possible values: true or false Default: false</p>

highlight +
displayField

You can define to "highlight" the search term you have requested in the retrieved results.
(Below, highlighting the title when searching for the term: "book")



Article

☆ **BOOK**
Patricelli Eddy
Islands Magazine, April-May, 2012, Vol.32(3), p.21(1)
Full text available

[View Online](#) [Details](#) [Reviews & Tags](#) [More](#)

highlight

Indication whether highlights by server is required.

Possible values: true or false

Default: false

If highlight=true, must supply a **displayField** (one or more) to indicate what fields to highlight.

displayField

defines in which field from the Display section of the PNx to highlight, if it contains the search term.

Possible values: PNx field names from the Display section.

You can define more than 1 displayField:

&highlight=true&displayField=title&displayField=creator

Note: the highlighting is done by wrapping the relevant section in the PNx with a tag,
i.e. BOOK

more

Applicable only for remote search. It simulates the "Get More" functionality.

Possible values: true or false

Default: false

lang

Hints input languages to search engine for language recognition.

"eng" or "fre" or "ger" (Code for the representation of name of language conform to ISO-639)

sortField

Specifies the type of sort to perform. If this parameter is not specified or left empty, the system uses the relevance so than the relevance sort, Primo supports the following types of sorts:

- 1) **stitle** - performs Title sort.
- 2) **sdate** - performs Date sort.
- 3) **screator** - performs Author sort.
- 4) **popularity** - Performs Popularity sort.

The valid values for this parameter are defined with the **Index Field** column in the **Sort Fields Config** mapping table, below. The direction of the sort is defined with the **Direction** column, which means that you can sort only in one direc

> Mapping Tables

Update for Owner:

Installation

Sub System :

Front End

Table Name :

Sort Fields Config

Mapping Table Rows

Enabled	Direction	Code*	Index Field	Description
<input type="checkbox"/>				
<input checked="" type="checkbox"/>	DESC	popularity	popularity	Search results sort by popularity
<input checked="" type="checkbox"/>	ASC	title	stitle	Search results sort by title
<input checked="" type="checkbox"/>	DESC	date	sdate	Search results sort by date
<input checked="" type="checkbox"/>	ASC	author	screator	Search results sort by author

loc

NOTE: when not configured, defaults to: **local,scope:(<institution code>)**
where <institution code> is the value entered for the parameter, institution

Supports 3 different search targets:

- 1) Local
- 2) Deep Search
- 3) Remote (Metalib) - If the results from a remote site are not in the MetaLib data format, Primo displays a link to the Additional Results section, which appears first in the results.

In order to add a target to search on, add the following template to the URL:

loc=<target>,<location reference>

<target> possible values:

- 1) **local** - For a local search
- 2) **adaptor** - For a deep search
- 3) **remote** - For a remote search against Metalib

<location reference> possible values:

- 1) for local - define the scope to search on
Example: **loc=local,scope:(VOLCANO)**
- 2) for adaptor - need to define the name of the requested adaptor
Example: **loc=adaptor,primo_central_multiple_fe**
- 3) for remote - need to define a quick set name
Example: **loc=remote,Business and Sciences**

A single request will let you define as many search targets as you want, meaning, all 3 can be done in a single request

Example for single search target:

http
://il-primo03:1702/PrimoWebServices/xservice/search/brief?institution=PRIMO&query=any,contains,book&indx=20&b
loc=local,scope:(VOLCANO)

Example for multi search (two) targets:

http
://il-primo03:1702/PrimoWebServices/xservice/search/brief?institution=PRIMO&query=any,contains,book&indx=20&b
loc=local,scope:(VOLCANO)&loc=adaptor,primo_central_multiple_fe

Example for multi search (three) targets:

http
://il-primo03:1702/PrimoWebServices/xservice/search/brief?institution=PRIMO&query=any,contains,book&indx=20&b
loc=local,scope:(VOLCANO)&loc=adaptor,primo_central_multiple_fe&loc=remote,Business and Sciences

Multiple scopes

You can search multiple scopes in 2 ways:

- 1) **loc=local,scope:(VOLCANO,NORTH)**
- 2) Use multiple loc parameters:
&loc=local,scope:(VOLCANO)&loc=local,scope:(NORTH)

There is a limitation to the order and which scopes can be used in such queries.

The limitation is according to the "Search Scopes" you have defined for each view.

To be sure what your "Search scopes" are, you can view the following file on your FE machine:

\$primo_dev/ng/jaguar/home/system/conf/**search_schema.xml**

under the <filters> tag you will find all your configured search scopes:

```
<filters>
  <filter>facet_frbtype:(7 OR 6)</filter>
  <filter>NOT scope:("PRIMO")</filter>
  <filter>scope:(NORTH) OR scope:(demo_digitool) OR scope:("CENTRAL")</filter>
  <filter>scope:(BBR)</filter>
</filters>
```

You are allowed to use only those scopes listed in the file and only use the order as in the file.

Mixing the order will not work, so **&loc=local,scope:(NORTH, demo_digitool, "CENTRAL")** will work but **&loc=local,scope:(demo_digitool, NORTH, "CENTRAL")** will not!

Searching against Primo Central

Searching against Primo Central is done via an adaptor.

Therefore as mentioned above, you will need to add the following:

loc=adaptor,primo_central_multiple_fe

pyrCategories	<p>Allows you to personalize your results by disciplines.</p> <p>Possible values: agriculture_forestry, anthropology, arts_humanities, biology, business, chemistry, computer_science, earth_sciences, economics, education, engineering, geography, languages_literature, law, library_information_science, mathematics, medicine, nursing, pharmacy_therapeutics_pharmacology, philosophy_religion, physics, political_sciences, psychology, public_health, sciences, social_sciences, sociology, veterinary_medicine</p> <p>Example: &pyrCategories=chemistry;business</p>
pyrDegree	<p>Allows you to personalize your results by degree.</p> <p>Possible values: graduate, librariystaff, researcher, undergraduate</p> <p>Example: &pyrDegree=undergraduate</p>

Notes

Retrieving only full text results:

query=facet_tlevel,exact,online_resources_PC_TN.

From Primo V4.4 this also works in a blended search (up to then it worked only against a Primo Central search).

Retrieving only peer reviewed results from Primo Central:

query=facet_tlevel,exact,peer_reviewed

Working with facet_creationdate

Example: query=facet_creationdate,exact,2012-YEAR

The value should be built using the following structure: <number>-<unit>

The unit can be:

- 1) YEAR
- 2) MONTH
- 3) WEEK

Limitation on number of results

When querying Primo Central via API, you are limited to requesting records within the first 2,000 results. That is, if you are retrieving records in order to display 10 records per page, you can only display the first 200 pages.

Any request that you send to retrieve records beyond the first 2,000 records will result in an empty result set.

Record IDs retrieved from Primo Central

Record IDs returning from Primo Central will hold a leading "TN_" in front of them.

For example: TN_jstor1234.

In order to query against such record IDs, you must remove the leading TN_, so in our case: **rid,exact,jstor1234**

Output Parameters

Element	Description	Value
segments	Root element of the XML tree.	no value
jagroot	<p>Element to contain one or several sub-elements.</p> <p>Structure: Sub-element of segments.</p>	no value

result	Element to contain one or several sub-elements. Structure: Sub-element of jagroot.	no value
querytransformations	Element to contain one or several sub-elements. Structure: Sub-element of result.	no value
querytransformation	Element to contain one or several sub-elements. Structure: Sub-element of querytransformations.	no value
action	did_u_mean. Structure: Sub-element of querytransformation.	any value
query	Alternative spelling of the search term. Structure: Sub-element of querytransformation.	any value
facetlist	Element to contain one or several sub-elements. Structure: Sub-element of result.	no value
accurate_counters	Defines if the facets counters are accurate or approximate. Structure: Sub-element of facetlist.	"true" or "false" Default: true
facet	Element to contain one or several sub-elements. Structure: Sub-element of facetlist.	no value
name	Name of the facet. Structure: Sub-element of facet.	any value
count	Number of facet values within one facet. Structure: Sub-element of facet.	any number
facet_values	Element to contain one or several sub-elements. Structure: Sub-element of facet.	no value
key	Name of the facet value. Structure: Sub-element of facet_values.	any value
value	Number of records within one facet value. Structure: Sub-element of facet_values.	any value
docset	Element to contain one or several sub-elements. Structure: Sub-element of result.	no value
firsthit	Index of the first record to be returned by the search. Structure: Sub-element of docset.	any value
lasthit	Index of the last record to be returned by the search. Structure: Sub-element of docset.	any value
hit_time	Time used by the search engine to find results. Structure: Sub-element of docset.	any number
total_time	Time used by the search engine to retrieve results. Structure: Sub-element of docset.	any number
totalhits	Number of results found by search. Structure: Sub-element of docset.	any number
doc	Element to contain one or several sub-elements. Structure: Sub-element of docset.	no value
no	Serial; number of the record in the bulk. Structure: Sub-element of doc.	any value
rank	The score of the query for the document. Structure: Sub-element of doc.	any decimal number
id	Index ID of the record in the search engine. Structure: Sub-element of doc.	any number
local	Indicates whether search was done locally or remotely. Structure: Sub-element of doc.	"true" or "false" Default: true
GETIT	Element to contain one or several sub-elements. Structure: Sub-element of doc.	no value

GetIt1	Primary GetIt link. Structure: Sub-element of GETIT.	any value
GetIt2	Secondary GetIt link. Structure: Sub-element of GETIT.	any value
pnx:PrimoNMBib	Element to contain one or several sub-elements. Structure: Sub-element of doc.	no value
record	Details about how the internal structure of a PNX can be found in the <i>Primo Technical Guide</i> .	

Input Example

<div>http://primo.com:170X/PrimoWebServices/xservice/search/brief?institution=WESTERN&onCampus=false&que</div>
--

Output Example

Brief Search Example Output
