



System Librarian's Guide – Search

Version 22

CONFIDENTIAL INFORMATION

The information herein is the property of Ex Libris Ltd. or its affiliates and any misuse or abuse will result in economic loss. DO NOT COPY UNLESS YOU HAVE BEEN GIVEN SPECIFIC WRITTEN AUTHORIZATION FROM EX LIBRIS LTD.

This document is provided for limited and restricted purposes in accordance with a binding contract with Ex Libris Ltd. or an affiliate. The information herein includes trade secrets and is confidential.

DISCLAIMER

The information in this document will be subject to periodic change and updating. Please confirm that you have the most current documentation. There are no warranties of any kind, express or implied, provided in this documentation, other than those expressly agreed upon in the applicable Ex Libris contract. This information is provided AS IS. Unless otherwise agreed, Ex Libris shall not be liable for any damages for use of this document, including, without limitation, consequential, punitive, indirect or direct damages.

Any references in this document to third-party material (including third-party Web sites) are provided for convenience only and do not in any manner serve as an endorsement of that third-party material or those Web sites. The third-party materials are not part of the materials for this Ex Libris product and Ex Libris has no liability for such materials.

TRADEMARKS

"Ex Libris," the Ex Libris bridge, Primo, Aleph, Alephino, Voyager, SFX, MetaLib, Verde, DigiTool, Preservation, URM, Voyager, ENCompass, Endeavor eZConnect, WebVoyage, Citation Server, LinkFinder and LinkFinder Plus, and other marks are trademarks or registered trademarks of Ex Libris Ltd. or its affiliates.

The absence of a name or logo in this list does not constitute a waiver of any and all intellectual property rights that Ex Libris Ltd. or its affiliates have established in any of its products, features, or service names or logos.

Trademarks of various third-party products, which may include the following, are referenced in this documentation. Ex Libris does not claim any rights in these trademarks. Use of these marks does not imply endorsement by Ex Libris of these third-party products, or endorsement by these third parties of Ex Libris products.

Oracle is a registered trademark of Oracle Corporation.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Ltd.

Microsoft, the Microsoft logo, MS, MS-DOS, Microsoft PowerPoint, Visual Basic, Visual C++, Win32,

Microsoft Windows, the Windows logo, Microsoft Notepad, Microsoft Windows Explorer, Microsoft Internet Explorer, and Windows NT are registered trademarks and ActiveX is a trademark of the Microsoft Corporation in the United States and/or other countries.

Unicode and the Unicode logo are registered trademarks of Unicode, Inc.

Google is a registered trademark of Google, Inc.

Copyright Ex Libris Limited, 2014. All rights reserved.

Document released: December 31, 2014

Web address: <http://www.exlibrisgroup.com>

Table of Contents

1	SYSTEM SETTINGS	4
1.1	Sort Limit and Sort Order of Records.....	4
1.2	Tree View Display.....	5
1.3	Search Limits.....	6
1.4	Limit and Format for Separate Display of Linked Records.....	6
2	LOGICAL BASES	7
3	FIND QUERIES	8
3.1	Boolean Operators Setup.....	9
4	HOLDINGS INFORMATION	10
4.1	Holdings Note for Serials.....	10
5	DISPLAYING AND INDEXING COPY INFORMATION	10
6	COLUMN HEADINGS	11
6.1	pc_tab_short.lng - Brief List.....	11
6.2	pc_tab_col.lng.....	12
6.3	tab_col.dat.....	13
6.4	Multi-field Search.....	14
7	MENUS	14
7.1	Index Menus.....	14
7.2	Formats Menu.....	15
7.3	Sort Order Menu.....	16
8	LOCATE FUNCTION	16
9	DEFAULT TEXT FOR LINKS	17
10	SEARCH FUNCTION TABLES	18

1 System Settings

You define Search function system settings in the `pc_server_defaults` table, located in the `alephe` directory. This table includes Search function defaults.

The elements listed below are defined in the default parameter section of the `pc_server_defaults` table.

1.1 Sort Limit and Sort Order of Records

The following lines in `pc_server_defaults` refer to Search function sort settings:

```
setenv pc_sort_field      "01  D02  A"
setenv pc_sort_field_aut  "03  A01  D"
setenv pc_sort_field_sub  "01  D02  A"
setenv pc_sort_field_lcc  "DISPLAY"
```

The maximum number of records that can be sorted is 1000.

The default sort order for the results of a **Search** query are defined by the parameter below (the staff user can change the sort order of a set when the set is displayed in Brief Table format):

```
setenv pc_sort_field      "01---D02---A"
```

01 and 02 are codes taken from column 1 of the `tab_sort` table. In this case, 01 refers to the Year field and 02 refers to the Author field. To sort the records by a different field, make sure the field is defined in `tab_sort` (see the Tables Section at the end of this chapter for an explanation of `tab_sort`). Here is an example from the table:

```
!1 2    3    4    5    6    7  8    9    10 11 12    13 14 15    16 17 18
19
!!-!!-!!!!-!!!!-!!-!!!!-!!!!-!!-!!!!-!!!!-!!-!!!!-!!!!-!!-!!!!-
!!!!-!!-!!!
01 95 008          260## c
08 04

02 01 1#### a
00 00
```

The letter "D" for the Year field stands for "descending" order, which means that the latest dates will be listed first (for example, 1999 will be listed before 1998). The letter "A" stands for "ascending" order and means that the order of the Authors will be from A to Z.

The default sort order for the results of a **Browse** query are defined by the parameters below (the staff user cannot change this sort order):

```
setenv pc_sort_field_aut  "03---A01---D"
setenv pc_sort_field_sub  "01---D02---A"
```

The "aut" and "sub" in the above parameters refer to the Browse Indexes as defined in tab00.lng. In this case, "aut" refers to Author and "sub" refers to Subject. You can define other parameters, but you must use the three-letter code of an index that is defined in tab00 (see the Tables Section at the end of this chapter for an explanation of tab_00.lng).

The character strings

```
02  A03  A
```

are defined in the same way as in the Search query.

If you want to display lists of records sorted by specific heading codes in the Show node, ensure that the following line appears in `pc_server_defaults`:

```
setenv pc_sort_field_<heading code> "DISPLAY"
```

This line can be repeated for different codes.

For example, the line `setenv pc_sort_field_lcc "DISPLAY"` appears in `pc_server_defaults`. A user browses by LC classification number and gets a list sorted by ascending LC classification numbers. When he moves selected records to the Show node, the sorting stays the same (that is, a sort by ascending LC classification numbers).

1.2 Tree View Display

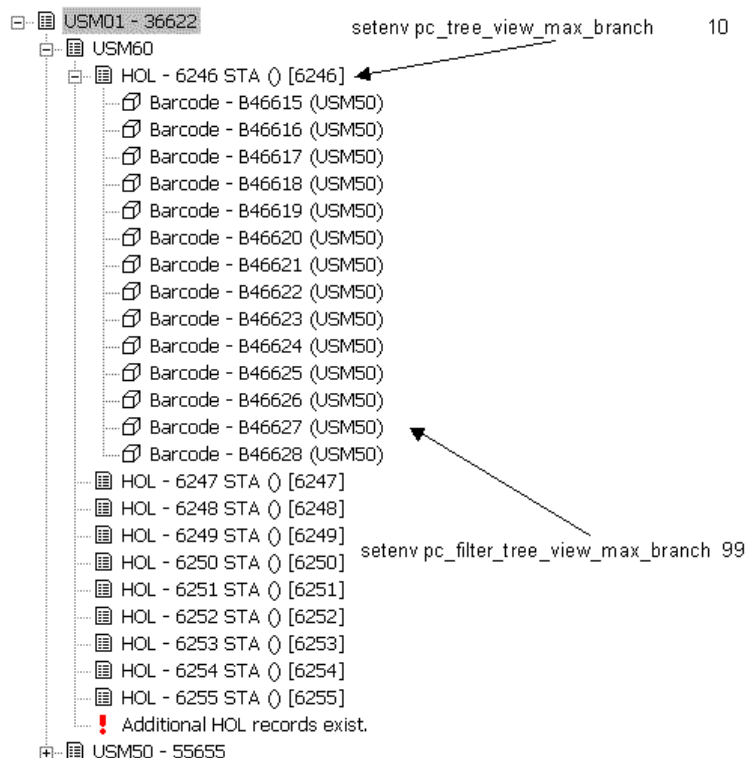
The following line in the table refers to Search function and cataloging module Tree View settings:

```
setenv pc_tree_view_max_branch 99
```

`pc_tree_view_max_branch` defines the limit after which the display of the navigation tree nodes in the Cataloging module and in the Overview tree of the Search tab is truncated with an appropriate message. This parameter limits either the number of ADM or HOL links of the given BIB record. If it is set to 5, only 5 ADM records that are linked to this BIB are displayed. The following parameter:

```
setenv pc_filter_tree_view_max_branch 99
```

defines the limit after which the display of the children of the navigation tree nodes in the Overview tree are truncated with an appropriate message. If both parameters exist, you will be able to see more information. In both cases the trees are limited to 800 lines.



In the example above, only 10 holdings can be displayed. A message, *Additional HOL records exist*, is displayed to let the user know that there are more holdings records available for this bibliographic record. Up to 99 items linked to each Holding record can be displayed.

1.3 Search Limits

There are no limits in creating a set in the Search Function. There are, however, limits for displaying and sorting a set. This means that:

- There is no maximum number of documents permitted in a set created after a Search query.
- There is no maximum number of words that are "collected" when truncation is used (when, for example, *find a?* retrieves all words beginning with *a*).
- There is no maximum number of hits that can be saved on the server.
- The number of hits from a set that display and are sorted can be set in the `set_result_set_limit` environment variable in `pc_server_defaults`. It can be between 1000-20000. If this variable does not exist, the default is 1000.

1.4 Limit and Format for Separate Display of Linked Records

A BIB record can have a number of bibliographic links to other BIB records. The Full view in the lower pane in the Show node can display a single line, pointing to a set of records, instead of a number of linked lines. The set is shown in the Brief list table.

Column 16 of `edit_doc_999.lng` defines after how many link lines the single line, pointing to a set, is displayed. If the column is empty, the default is 99 links.

In Column 15, you can determine a Brief list format for every bibliographic link. If a format is not defined in the table, the program displays the default format type.

Example:

```
!          1          1          1
!1   2   3         4         5 6 7         8         9 0 11  2  13  4 15
16
!!-!!!!!!-!-!!!!!!-!-!-!!!!!!-!-!-!!!!!!-!-!-!!!!!!-!-!-!!!!!!-!-!-!-
!!
## UP          D LUplink          Y Q      E          00
02

## PAR          D LParall          Y Q      E          00
02

## DN          D LDown          Y Q      E          00
02
```

2 Logical Bases

You can group bibliographic records into logical bases that the user can search and browse through. Examples of logical bases are records for periodicals or audio-visual materials, or materials about History, or France. When the user performs a Browse or Find Query, the system automatically excludes items that are not included in the logical base.

A given bibliographic record may be included in any number of bases but a base can be defined within one library only. A base cannot be defined to include more than one library.

Your ALEPH system must have word indexes built in order for the base option to work. You can update the word indexes using the Update Word Index (manage-01) service from the Services menu in the Cataloging module.

In order to define a logical base, edit `tab_base.lng`. Following is a sample from the table:

```
!          1          2          3          4          5          6          7  8
9
!!!!!!!!!!!!!!!!!!!!!!!!-!-!!!!!!-!-!-!!!!!!-!-!-!!!!!!-!-!-!!!!!!-!-!-!!!!!!-
!-!!!!!!>
MONOGRAPHS          Monographs (USM01)  USM01          USM01 Y
wfm=BK
USM01          ExLibris University          USM01 N
LINC          Lincoln Library          USM01          USM01 Y
wsl=ulinc
EDUC          * Education Library          USM01          USM01 Y
wsl=ueduc
USM01_PAC          USM01 PAC          USM01          USM01 N
wrd=alldoc
uments not wst=(suppressed or deleted)
HISTORY          * History          USM01          USM01 Y
wti=history
```

Column 1 is the code identifying the logical base. When displaying screens in the OPAC, the system looks for files that use this code as an extension to the file name (for example, find-a-3-USM01). If this extension is not found, it uses one of the extensions defined in columns 4,5, or 6 of this table instead.

Column 2 Limits access to signed-in users by means of an asterisk (*). This is relevant for Web OPAC only.

Column 3 is the base name that is displayed to the user.

Column 4 lists the alternative extensions to the filename. If there are no OPAC files with the extension defined in column 1, the system uses the files which have an extension defined in columns 4, 5, or 6.

Columns 4-6 are also used for referring to data format tables (`edit_doc`, `edit_doc_999.lng`, `edit_doc_999_<aut_library>.lng`). These tables can be duplicated, with a base extension (for example, `edit_doc_999.eng.mono`).

Column 7 is the code of the physical library in which the logical base is defined.

Column 8 Column 8 defines whether or not Z0102 is used to create a logical base.

Column 9 is the Find command that the system uses to create a logical base of records. Once this base exists, all Scan and Find queries that the OPAC user performs are carried out on the records in the base (and all other records are excluded). In the above example, the HISTORY base includes all records that include the word "history".

To set the bases that are available in (the drop-down list for the Bases field in the Find and Browse nodes), follow these steps:

1. Open the ALEPHADM client.
2. From the Configuration menu, click **Libraries Manager**. The Libraries Manager window is displayed.
3. Open the fourth tab, Bases for Search, and add or delete lines. The values in Base for Search and Library Code must match values in the system's `tab_base.lng` table.
4. Click **OK** to save the changes. The client's `alephcom\tab\searbase` file is updated.

In the Browse node, only the headings that are relevant for the logical base are displayed. An exception to this is when the code of the search database is the same as a library code (for example, base code "USM01"). In this case, the Browse list will include all headings, both linked and unlinked. In order to filter the Browse list to include only linked headings, use a base code that is different than the library code (for example, USMXX).

3 Find Queries

To define the types of Find Queries that are available to the user and the order in which they are displayed, go to the ALEPHCOM/TAB directory on the client and

open the ALEPHCOM.INI file. Go to the section headed [SearchFind]. Following is a sample of the relevant line:

```
FindTypes=S,C,F
```

In this example, all types of Find Queries are available, Advanced Search, Multi-base CCL Search and Multi Field Search. The corresponding tabs are displayed in this order (S,C,F), from left-to-right. To change the order in which the tabs are displayed, change the order of the letters above. To make an option unavailable to the user, remove its letter from the list above. Note that the letters are written with commas between them, but without any spaces.

Another line in [SearchFind] is:

```
SelTab=C
```

The content of this line is filled automatically. It defines the last Find query that was used the last time that the GUI client was closed.

In the Search function, the Advanced, Multi-base CCL and Multi-field queries allow the user to select the Words Adjacent option. This instructs the user that the words in the search string must be in the same field (word index) and appear next to each other in the record.

You can determine whether or not the "Words Adjacent" option is selected by default when the user activates the Find Query screen. To do this, open the ALEPHCOM.INI file on the client and go to the section headed [SearchFind]. Following is a sample of the relevant line:

```
FindAdjacency=Y
```

If you want the "Words Adjacent" option to be selected by default, type **Y** to the right of the equal (=) sign. If you do not want the "Words Adjacent" box to be checked by default, type **N**.

Note that this also controls the default setting of the "Words adjacent" option that appears on the right-hand side of the Refine screen.

The AutoShow parameter in the same section sets the threshold of the number of documents that are displayed in Brief or Full format when FIND is used. If the set contains the same number of records, or less than the number set in AutoShow, the records are automatically displayed in full format in the Show node after a FIND query is run. Otherwise, the results appear in a set in the List of Sets in the Find node.

3.1 Boolean Operators Setup

You can set up different Boolean operators according to different interface languages in the `./alephe/tab/tab_ccl_boolean_operators` table. This allows you to use different operators, based on language, other than the standard AND, OR and NOT. For example, you can add the Danish word *Eller* which corresponds to *or*.

!1	2	3
!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!>		
###		OR
### ~		NOT
### +		AND
### &		AND
DAN IKKE		NOT
DAN OG		AND
DAN ELLER		OR

Column 1 contains a language code, column 2 contains a string and column 3 contains a CCL operator.

4 Holdings Information

When you click the *Items* button on the Full record, the List of Items is displayed with holdings information such as Description, Item Status, Sublibrary, Collection and Call Number. In addition, if the item is currently on loan, the due date is shown.

4.1 Holdings Note for Serials

The USMARC 866 holdings note field which appears in the holdings record may be displayed in the Full view of the Bibliographic record in the Search function. To set this up, follow these steps:

1. To ensure that these records will be merged (for display purposes only), edit the `tab_expand` table. Make sure that the following line appears in the table:

```
GUI-DOC-D expand_doc_bib_hol_USM
```
2. After editing the `tab_expand` table, to ensure that the 866 field is included in the Full view of the BIB record, edit the `edit_doc_999.lng` table.

5 Displaying and Indexing Copy Information

Information about copies can be displayed together with the linked bibliographic record in special ALEPH fields called Z30-1, Z30-2, LOC, PSTS and PST (depending on the program used to compile the information). These fields do not actually exist in the bibliographic record; rather, the system compiles the information from the subscription, holdings and item records and displays it in the bibliographic record as if they were regular fields. Therefore, the Z30-1, Z30-2, LOC, PSTS and PST fields are referred to as "virtual fields."

These virtual fields may also be indexed so that the user may search them using the Find or Browse options in the same manner that he can search subjects, titles, and authors.

There are four different programs for expanding copy information into the bibliographic record. Each program does this in a particular way and the library should decide which program best suits its needs. Following are the existing programs:

- expand_doc_bib_z30
- expand_doc_bib_psts
- expand_doc_bib_loc_usm
- expand_doc_bib_loc_n (together with expand_doc_sort_x)

Refer to Indexing - Chapter 9 Expand Routines, Tables and Indexing Expanded Fields (Expand Record section) for a complete explanation of these programs and how to set up the relevant table (tab_expand).

Note that for the indexing and display of copy information, it is necessary to set up all the relevant tables:

- For indexing purposes: tab00.lng, tab01.lng, tab11_acc, tab11_ind and tab11_word.
- For display purposes: edit_field.lng, edit_doc_999.lng, and so on.

6 Column Headings

You may define the column headings and column widths for tables of information. To do so, you need to edit one of three files, depending on the type of screen.

6.1 pc_tab_short.lng - Brief List

The columns for the Brief List are defined by editing the pc_tab_short.lng file. Following is an extract from the file:

```

!1      2              3  4      5      6      7      8      9  10
11 12  13
14
!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!-!!!!!!-!!!!!!-!!!!!!-!!!!!!-!!!!!!-!!!!-!!!-
!!!-!!!-!!!-
!!!
00 Doc no.          1 SUM                                010 01
C08
00 Call No         2 050##  LOC##                          015 01
C01
00 Author          3 100##a 700##a                          025 02
C02
00 Title           4 245##  240##                          030 03
C03
00 Year            5 008      260##c                          010 03
C03 008 004
00 $1              6 BASE1                                005 01
C01
00 $2              7 BASE2                                005 01
C01

```

Column 1 is the display format. The standard is 00.

Column 2 is the text of the column heading that is displayed to the user.

Column 3 is the column number (from left to right).

Columns 4-8 are for tag + indicator + subfield. You may enter up to five field tags. If the system does not find the first field in the record, it will look for the alternatives

you define here. The # character may be used as a placeholder for indicators in positions 4 and 5. For example, entering 245## designates 2451, 2452 and 24514. BASE1/BASE2 are system codes that can be used to display whether the record is present in a particular logical base. The logical bases for each column are configured in \$alephe_tab/tab_base_count, col. 4 and 5.

Column 9 is the percentage of the page width that you want the column to take up. The number must be 3 digits. For numbers less than 100, enter leading zeros, for example, 025. You may remove a column heading from the table by entering 000.

Column 10 is the font as defined in the FONT.INI file on the PC. The relevant lines in font.ini start with ListBox. If the line has ListBox## in the first column of font.ini it will relate to all font numbers. That means they will all be displayed, some according to the line starting with ListBox##.

Column 11 is the color as defined in the ALEPHCOM.INI file on the PC. The relevant section is [TextListBox]. Please note that after effecting changes you have to restart the server to see your new color setup.

Column 12 is the starting character, that is, the position number of the first character in the field that should be displayed. In the example above, the first character of the Year field that will be displayed is the character in position number 8. The number you enter must be 3 digits. For numbers less than 100, enter leading zeros, for example, 008. If you want the starting character to be the character in position number one, leave this column blank.

Column 13 is the total number of characters of the field that should be displayed. In the example above, four characters of the Year field will be displayed, those in positions 8, 9, 10, and 11. The number you enter must be 3 digits. For numbers less than 100, enter leading zeros, for example, 004. If you want the entire contents of the field to be displayed, leave this column blank.

Column 14 contains a language code. It will act as a filter on the field. It uses a subfield 9 that contains a language code as a filter.

6.2 pc_tab_col.lng

pc_tab_col.lng defines the columns of information that are displayed in list windows in the GUI clients.

In order to define column headings, edit the bibliographic library table

pc_tab_col.lng.

For more information about pc_tab_col.lng, see the discussion of GUI lists and column headings in the ALEPH User Guide - General chapter - Using and Customizing Your GUI.

The following is a list of the Search Function windows which use pc_tab_col.lng for formatting data, and their identifiers (Column 1 in pc_tab_col.lng).

Identifier	Search Function Windows
CORRECT_HEADING_1	Correct Heading
CORRECT_HEADING_2	Correct Heading

PC_COM_DOC	Navigation Tree
PC_SEAR_ADM_LIBRARY	Global Items List of Record <record #>
PC_SEAR_FULL_000	MARC Tags (tab01)
PC_SEAR_FULL_99	Full+Link (edit_doc_999)
PC_SEAR_FULL_XXX	Catalog Card (edit_doc)
PC_SEAR_HEADINGS	Brief Records list
PC_SEAR_ITEM_LIST	Item list
PC_SEAR_MARC_OPAC	Document code text
PC_SEAR_REFERENCE	Reference List
PC_SEAR_SCAN	Browse List of <ACC name>
PC_SEAR_SCAN_I	Browse List of <index name>
PC_SEAR_SCAN_W	Browse List of <Word index>
PC_SEAR_SCAN_Z39	Browse List of <Z39 index>

Note: the definition for the List of Sets and Cross Set windows are in the pc client in:
ALEPHCOM\tab\<language>\tab_col.dat.

6.3 tab_col.dat

In order to define column headings for the List of Sets and Cross Set screens, go to the client's ALEPHCOM\tab\ENG directory (for English headings; other headings are located in the appropriate ALEPHCOM\tab\<language>directory.) Open the tab_col.dat file. Following is a sample of the relevant lines in that file:

```

1           2 3           4 5 6 7 8 9 10
SEAR_CROSS_SET  L Find Request  01 060 01 C04  Find request
SEAR_CROSS_SET  L Base Name      02 020 01 C02  Base name
SEAR_CROSS_SET  L No. Records    03 020 01 C03  No. records
SEAR_RESULT_LIST L Request        01 050 01 C01  Request
SEAR_RESULT_LIST L Database       02 030 02 C02  Database
SEAR_RESULT_LIST L No. Records    03 020 03 C03  No. records

```

Column 1 is the unique code by which the system identifies this set of column headings. The code SEAR_CROSS_SET refers to Cross Set; SEAR_RESULT_LIST refers to the List of Sets, both from the Find node.

The definitions of the other columns are the same as pc_tab_col.lng. See the discussion of GUI lists and column headings in the ALEPH User Guide - General chapter - Using and Customizing Your GUI.

6.4 Multi-field Search

To define the list of fields for a Find Multi-Field query, edit the `pc_tab_sear.lng` file. There is a limit of 8 FF lines per base. Following is a sample from that file:

```
!           2           3           4           5
!!--!!!!!!-!-!!!!!!-!!!!!!-!!!!!!
FF  USM01           L System number           SYS
FF  USM01           L Titles                   WTI
FF  USM01           L Exact titles                 TIT
FF  USM01           L Authors                     WAU
FF  USM01           L Uniform Titles              WUT
FF  USM01           L Barcode                     BAR
```

Key to the `pc_tab_sear.lng` Table:

- **Column 1 - Code**
This is the unique code by which the system identifies the set of menu choices. To define Fixed fields to search in, enter FF in this column.
- **Column 2 - Code**
Enter here the database associated with the menu option you are defining. You must use a database code that was assigned in `tab_base`.
- **Column 3 - ALPHA**
ALPHA code. Must always be L.
- **Column 4**
This is the text of the field that is displayed to the user.
- **Column 5**
This is the index code for the search query taken from `tab00.lng`.

7 Menus

You can determine the choices that are available in the following menus of the Search function:

- List of indexes for the Find Query, Browse Query and Direct Query.
- List of formats for displaying and printing bibliographic records.
- List of sort order options.

7.1 Index Menus

To define the list of indexes for Find and Browse queries, edit the `pc_tab_sear.lng` file. Following is an extract from that file:

```
!1           2           3           4           5
!!--!!!!!!-!-!!!!!!-!!!!!!-!!!!!!
SC  USM01           L Titles                   TIT
SC  USM01           L Related Titles          TRT
SC  USM01           L Authors                 AUT
```

SC	USM01	L Series	SRS
SC	USM01	L General Subjects	SUB
FI	USM01	L Anywhere in the record	WRD
FI	USM01	L System number	SYS
FI	USM01	L Titles	WTI
FF	USM01	L System number	SYS
FF	USM01	L Titles	WTI
FF	USM01	L Exact titles	TIT
FF	USM01	L Authors	WAU
FF	USM01	L Uniform Titles	WUT
FF	USM01	L Barcode	BAR

Column 1 is the unique code by which the system identifies the set of menu choices. For the Find Query menu, enter the code FI; for the Browse Query menu, enter SC; for the fixed field (Multi-Field) Query menu, enter FF;

Column 2 is the database code. Enter here the database associated with the menu option you are defining. You must use a database code that was assigned in `tab_base.lng`.

Column 3 is the code for the character set of the menu choice. The standard is L, which stands for Latin.

Column 4 is the text of the menu choice that is displayed to the user. Enter here the name of a word index that was defined in `tab00.lng`.

Column 5 is the code for the above index as it was defined in `tab00.lng`, column 2.

7.2 Formats Menu

To define the list of formats for displaying or printing bibliographic records, edit the `pc tab_sear.lng` file. Following is an extract from that file:

```

!1          2          3          4          5
!!--!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!
FO  USM01          L Full+Link          099
FO  USM01          L MARC Tags          001
FO  USM01          L MARC Public View   002
FO  USM01          L Catalog Card      037
FO  USM01          L Citation          040
PF  USM01          L Name tags         002
PF  USM01          L MARC tags         001
PF  USM01          L MARC Public View   099
PF  USM01          L Catalog Card      037
PF  USM01          L Citation          040

```

Column 1 is the unique code by which the system identifies the set of menu choices. To define **display** formats in the Show node, enter FO in this column. For **print** formats, enter PF. For the **record** formats available when clicking the **Print/Send** button, enter PF.

Column 2 is the database code. Enter here the database associated with the menu option you are defining. You must use a database code that was assigned in `tab_base.lng`. **Column 3** is the code for the character set of the menu choice. The standard is L which stands for Latin.

Column 4 is the text of the menu choice that is displayed to the user. Available options include: Full+Link, Catalog Card, MARC Tags, and Name Tags.

Column 5 is the code for the above display format. Enter 099 for Full+Link, 037 for Catalog Card, 001 for MARC Tags and 002 for Name Tags. Codes are defined in `edit_doc.lng`.

7.3 Sort Order Menu

The user may choose to sort bibliographic records by specific fields. To define the list of fields that the user may choose from, edit the `pc_tab_sear.lng` file. Following is a sample from that file:

```

!1          2          3          4          5

!!--!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!
SO  USM01          L Year          01
SO  USM01          L Call number  04
SO  USM01          L Author       02
SO  USM01          L Title        03
SO  USM01          L Type         06

```

Column 1 is the unique code by which the system identifies the set of menu choices. For the Sort Order menu, always enter SO.

Column 2 is the database code. Enter here the database associated with the menu option you are defining. You must use a database code that was assigned in `tab_base.lng`.

Column 3 is the code for the character set of the menu choice. The standard is L which stands for Latin.

Column 4 is the text of the menu choice that is displayed to the user. Enter here the name of a field that was defined in `tab_sort`.

Column 5 is the code for the above field as it was defined in column 1 of the `tab_sort` table.

For example, the following line appears in `tab_sort`:

```

!1 2 3 4 5 6 7 8 9 10 1 12 13 4 15
16 7 18 9
!!-!!-!!!!!!-!!!!!!-!-!!!!!!-!!!!!!-!-!!!!!!-!!!!!!-!-!!!!!!-!!!!!!-!-!!!!!!-
!!!!!!-!-!!!!!!-
01 95 008          260## c
08 04

```

Column 4 of `pc_tab_sear.lng` will be Year and column 5 will be 01.

8 Locate Function

The Locate function of the Search function enables you to find records in other databases that are similar to the record you have selected. As System Librarian, you are responsible for setting up the criteria that the system uses in order to determine which records are similar. (For example, your library can decide that if the records have the same words in the title and author fields, then the records are "similar.")

You can define the criteria by editing the `tab_locate` table. For more information see the Locate function in the Cataloging guide.

For example, the following lines determine the fields that will be checked for similarity when searching the Library of Congress:


```

!           1           2           3           4
5
                6
!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
LOC           245## a           wti=
locate_str_1
LOC           100## a           wau=
locate_str_0

```

Column 1 contains the database code as it appears in the first column of tab_base.lng. Columns 2 and 3 contain the fields and subfields that will be checked for similarity. The Find command resides in Column 4. Column 5 contains the Extract function that defines how the contents of the field are handled.

To determine the database that will be available for the Locate function, edit the ALEPHCOM\TAB\LOCATE.dat file. The databases on the above list are marked automatically by the system according to the user's last choice. The parameter that defines the marked databases can be found in the ALEPHCOM\TAB\ALEPHCOM.ini file:

```
[SearchLocate]
SelectedBase=LOC
```

This means that the Library of Congress database will be marked when the Locate window is opened.

9 Default Text for Links

The cataloger uses the LKR field to create links from the bibliographic record he is currently cataloging to another record. The following table shows several examples of links that can be created by the cataloger and appended to a record in the LKR field:

Subfield	Description	
a	Value	Use
	UP	"Up" link to another BIB type record. A record may have only one link of this type.
	DN	"Down" link to another BIB type record. Multiple links of this type are possible.
	PAR	Parallel link from BIB record to BIB record
	ANA	Analytic link between BIB records of different levels
	ITM	Link from BIB record to ADM record (used for items bound together)

When no other text is available (the Z103-LKR-TEXT-N field is empty), the default text that appears next to the LKR (link) field in the Full View of the bibliographic record is defined through the library's tab/tab05.lng table as follows:

```
! 1           2
```

```
!!!!!!-----!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
NONE          "text"
```

For example:

```
! 1                      2
!!!!!!-----!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
NONE          Click to view related record
```

NONE is reserved for the Search function both from GUIs and from the Web OPAC.

The LKR fields can be displayed according to various sorts. For more information see `tab_fix_z103`.

10 Search Function Tables

GENERAL

1. Table Extensions

You can use different tables for the Search function in GUIs and in the Web OPAC by using the file extensions: `[filename].PC` and `[filename].WWW`

Example:

```
tab_sub_library.lng.PC OR tab_sub_library.lng.WWW
```

LIBRARY TABLES

1. `edit_doc.lng`

`edit_doc.lng` is used to define the display of document information. It defines a concatenation of a number of paragraphs. It is used in conjunction with `edit_paragraph.lng`.

2. `edit_doc_999.lng`

`edit_doc_999.lng` defines the fields to be included in a display of a bibliographic record. It defines the full format for printing and saving in GUIs and in the Web OPAC.

This display includes hypertext linking.

3. `edit_doc_999_aut_xxx1n.lng`

`edit_doc_999_aut_xxx1n.lng` defines the display of AUT records for cross-reference display. The display of the record is defined in the `edit_doc_999_aut_xxx1n.lng` table where "xxx1n" is the code of the AUT library (for example, USM10).

4. `edit_field.lng`

`edit_field.lng` defines the display of a tag. It is used in conjunction with `edit_paragraph.lng` and `edit_doc.lng`. If a tag is not defined in `edit_field.lng`, it will be displayed as is. Accordingly, tags entered in the database without punctuation need to be defined here. You can define up to 10 filters for each subfield per tag name.

5. `edit_paragraph.lng`

`edit_paragraph.lng` defines the display of a paragraph of data from the bibliographic record. The `edit_paragraph` table defines a concatenation of a number of fields. It is used in conjunction with `edit_field.lng` and `edit_doc.lng`.

Note that formats 001-099 are used by the BIB_FORMAT program for direct display of specific BIB information (for example, on Patron List of Loans, Search function of Items, and so on.)

6. `pc_tab_sear.lng`

This table is used to define options for the Search function search as follows:

- **SC** - link to the ACCess code as defined in `tab00` in the `tab` directory of the library.
- **FI** - link to the WORD files as defined in `tab00` in the `tab` directory of the library.
- **FO** - link to `edit_doc` format (must be expressed in three digits).
- **SO** - link to field tag codes as defined in `tab_sort` table in the `tab` directory of the library.
- **PF** - link to `edit_doc` format for print/save formats.
- **FF** - fields for Find - Multi-field Search.

7. `pc_tab_short.lng`

This table defines the document fields that are included in the Brief display in the upper pane of the Show node.

8. `tab00.lng`

`tab00.lng` defines the system index files. There should be one such table for each language defined.

9. `tab01.lng`

`tab01.lng` defines per language the type and name of the library and tag definitions for the bibliographic records.

10. `tab05.lng`

This table defines the caption to be displayed in the Web OPAC before `$$n` and `$$m`.

11. `tab100`

`tab100` is the central configuration table for system-level, server-level and library-level variables.

Example:

```
AUTO-TRUNCATE-Z01-FIND
```

This variable controls the automatic truncation option in Browse.

```
CORRECT-HEADING-ORDER
```

This variable defines the order of subfields when adding or updating a specific subfield of a heading via GUI-Search-Browse - Correct heading.

Possible values: 1 or 0.

1 (default) – The correct heading of an existing subfield remains in the original order. The new subfield is added at the end of the field.

0 – The new or modified subfield is added at the beginning of the field.

12. `tab11_acc`

The `tab11_acc` table is used to assign fields to headings indexes.

13. `tab11_ind`

The `tab11_ind` table is used to assign fields to direct indexes.

14. `tab11_word`

The `tab11_word` table is used to assign fields to word indexes.

15. `tab20`

This table defines the headings (including see references) that are built in the ACC table of the bibliographic base, based on the fields of the authority record that "matches" the BIB ACC Headings.

16. `tab22`

This table defines the fields that are included in the "short-doc" (z13) - and also defines the fields in the Basket Brief format.

17. tab_aut

This table defines which heading files are linked to an authority database and which authority database is searched.

18. tab_filing

The `tab_filing` table defines which filing procedures are used when building the filing key for Heading (Z01), Index (Z11) entries and Sort keys (Z101). The filing procedures identifier of these is set in col. 5 of `tab00<lng >`.

19. tab_sort

`tab_sort` defines fields for sorting. This table is used in conjunction with:

- o `pc_tab_sear.lng` (option SO)
- o option-sort HTML screen (for Web OPAC)
- o `www_server.conf`
- o `pc_server_defaults`
- o `tab01.lng` - document record fields (for correct filing of z101 sort keys).

20. tab_word_breaking

This table defines routines for filing of headings - word breaking procedures.

21. tab_z30_sort

`tab_z30_sort` defines items (z30 sort types) in various ALEPH functions. It is located in the library's `data_tab` directory. The instance which affects the Search function is SEARCH.