System Librarian's Guide – Web OPAC

Version 23

ExLibris The bridge to knowledge

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, 2018. All rights reserved.

Document released: April 2018

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

Table of Contents

1 INTRODUCTION 8

2 SETTING UP THE SYSTEM 8

2.1 aleph_start Parameters 21

3 PATRON REGISTRATION AND PROFILES 22

3.1 Patron Registration 22

3.2 Patron Profiles 24

3.3 Interface Language 25

3.4 Brief View of Records 26

3.5 Full View of Records 27

3.6 Additional Optional Fields 27

4 HTML FILES AND THE INTERFACE 27

4.1 ALEPH and CSS 28

4.2 Include Files 29

4.3 File Extensions 29

4.4 Menu Bar 30

4.5 Encryption of patron connection information using SSL 30

4.6 Setting up the Browse and Search Screens 31

4.7 Result Screens 36
4.7.1 Brief View (Table Format) 37
4.7.2 Other Brief Format Options 41

4.8 Full View 42

4.9 Holdings Screens 48

4.10 User Screens 49

4.11 Submitting a Photocopy Form 50
4.11.1 General 50
4.11.2 Photocopy Submission for Article Document 50

4.12 ILL Screens 51

4.13 Setting up the Special Request Function....53
4.14 Support For Right-To-Left Text Display In All Formats....56
4.15 Adding an SFX Link Button....57
4.16 Setting Up the Refine Window....60
4.17 Photocopy Information....61

5 SORT 61

6 FILTERS 62

6.1 Setting Up Bases and Filters....65

7 ERROR MESSAGES 69

8 DEFAULT TEXT FOR LINKS 70

9 RANK 71

10 ADDING A NEW INDEX 71

11 SETTING UP THE SDI FUNCTION 74

12 LOGICAL BASES 75

12.1 Logical Base Counter....76
12.2 Denied Libraries....78

13 PATRON ADM LIBRARY (SINGLE/MULTIPLE) 79

14 SYNONYMS DICTIONARY FOR WORD SEARCHING (OPTIONAL) 79

15 SETTING UP SAVE/MAIL ENCODING OPTIONS 80

16 SAVE/MAIL ON MULTI-APACHE TOPOLOGY 81

17 LOCAL OWN CHECK PROCEDURE 82

18 ACQ REQUEST FOR LIBRARY WITH MORE THAN ONE BIB LIBRARY 83

19 APPLYING ADM FILTER IN OPAC-ACQ REQUESTS 84

19.1 Activating the ADM Library Filter 84

19.2 ADM Library Filter Routine 85

20 LOG FILE EXPANSION 85

21 SETTING UP BIB REQUEST POLICY 86

21.1 Group Definition and Availability 86

21.2 Pickup Location List 86

21.3 Item Level Request 87

22 SETTING UP MY E-SHELF 87

22.1 My e-Shelf and Course Reading Addresses 88

22.2 Setting the Maximum Number of Records in a Folder 89

22.3 Setting Up the Note Column in the Records Which Were Not Requested Table 90

22.4 Setting Up the Submitted a Hold Request Column in My e-Shelf 90

23 SEARCH KEYWORD IN HEADINGS 91

23.1 Keyword Search in Heading List Functionality Via Web OPAC-Browse 91

23.2 Search Keyword in Heading via Web OPAC-Search/Browse 92

23.3 Search Keyword in Heading via GUI-Search-Browse 93

23.4 Search Rules and Logical Operators for Keyword Search in Headings 93

23.4.1 Truncation 94

23.4.2 Logical (Boolean) Operators 94

23.4.3 Parentheses 94

23.4.4 Phrase Searches 95

23.4.5 Using the "-" Symbol 95

23.5 Setting Up Keyword Search in Heading List Functionality 95

24 WEB OPAC AND OPENSEARCH™ COMPATIBILITY 98

25 WEB OPAC SECURITY 99

25.1 XSS Validation 99

25.2 CSRF Valuation 99
25.3 Session Fixation 100
25.4 Apache Security Vulnerability Patch 101

26 WEB OPAC ACCESSIBILITY 101

26.1 Adding Descriptions to Images 101
26.2 Defining Titles with the Heading Attribute 102
26.3 Bulleted and Numbered Lists 102
26.4 Indicating Table Headers 103
26.5 Assigning Labels to Input Form Elements 104
26.5.1 Text Fields 104
26.5.2 Text Area Fields 105
26.5.3 Checkbox Fields 105
26.5.4 Radio Button Fields 106
26.5.5 Drop-Down List Fields 106
26.6 Color Indications 108
26.7 Additional Changes for Aleph Web Files 108
26.8 Skipping Repetitive Links 110
26.9 Exceptions 111

27 LINK TO GOOGLE BOOKS SERVICE 111

28 FORGOT PASSWORD 112

28.1 Reset Password by Answering a Predefined Question 112
28.2 Reset Password by Getting an Email 113

29 WEB OPAC-RELATED TABLES 115

30 CANCELLING HOLD REQUESTS 123

31 STAFF REQUEST AND REQUESTER NAME FIELD 124

31.1 Requester Name Field on Request Creation Forms 124
31.2 Requester Name Information in My Library Card 126
31.3 Sort Based on Requester Name Information 127

32 SERVER SIDE SCRIPTS IN THE ALEPH OPAC 127

32.1 Script Input 128

32.2 Script Output 128

33 LIST OF WEB OPAC FILES 128

33.1 List of Web OPAC Files 131

1 Introduction

You can run the Web OPAC using the default settings in your ALEPH 500 installation. However, most libraries choose to customize the Web OPAC in line with their policies and needs.

As Systems Librarian, you can modify HTML screens, display formats, messages, access points and command codes. A working knowledge of HTML is necessary in order to change your Web OPAC's screen settings. No additional programming skills are required. Any changes you make are updated in the system immediately.

2 Setting Up the System

You can customize the display of the OPAC interface by modifying the parameters listed in the table below. These parameters are defined in the default parameter section of the `www_server.conf` table in the `alephe` directory. Edit the `www_server.conf` table. You can disable a setting by placing a # at the start of the relevant line.

The following table lists the default parameters together with a brief explanation of their associated functionality.

Default Parameter	**Set this parameter to ..**
`allow_create_new_profile`	Allow the creation of new personal profiles
`create_statistics`	Enable statistical analysis of Web server use
`direction_support`	Set up right-to-left text display
`due_date_format`	Set up the due date format of loans
func_in_log	Set up the name of the function (program) to the ending of each transaction in the Web server log file.
`hol_filter_op`	
`hol_sort_op`	Determine the filter and sort options of the Holding records list
`ill_total_limit_period`	Define period limits for ILL requests
`Keep_z39_gate`	Prevent the automatic start of the z39_gate
`Mark_match_words_prefix`	
`Mark_match_words_suffix`	Format highlighted search terms

Default Parameter	Set this parameter to ..
patron_notice_cash	Check the patron's cash transaction after successfully logging into the Web OPAC and display the 'Pending Cash Transaction' notice if unpaid cash transactions are detected.
patron_notice_delinq	Check the patron's delinquencies after successfully logging into the Web OPAC and display the 'Delinquency Notice' if delinquencies are detected.
patron_notice_overdue	Check the patron's loaned items after successfully logging into the Web OPAC and display the 'Overdue Items' notice if overdue items are detected.
patron_expiry_alert	Check if any of the patron's local accounts (z305) expire in N days. The number of days is defined in patron_expiry_alert variable (Possible values: numbers between 1- 999). If one (or more) of the accounts expire during this period, a message is displayed in the patron notice page, alerting the patron about the coming expiration and the 'Library Card' includes a link for extending the user's account.
server_default_charset	Set up a default character set for non-UTF-8-enabled Web browser
set_default_z110_create_order	Control the order in which a result set is populated
set_prox_limit	Limit the number of documents that are checked for proximity
set_result_set_limit	This parameter defines the find command limit
set_word_limit	Define the maximum number of words retrieved when truncation is used
sfx_base_url	Define the IP of your SFX server
sfx_fetch_address	Set up your SFX server to fetch additional data from an ALEPH server
sfx_issn_routine	The format from which the ISSN component of the SFX link will be constructed
Show_copyrights	Control display of copyright information

Default Parameter	Set this parameter to ..
www_a_s_time_out	Define the length of user session before timeout
www_bor_registration_ends	Define the valid period of the patron's address
www_browse_previous	Set up the Browse list display
www_con_lng	Set the default language of the OPAC interface
www_course_email_address	Set up the destination e-mail address for course reading material
www_dn_link_display	Set up down link display
www_dn_view_display	Set this parameter to View down links display
www_f_sfx	
www_f_sfx_dim	Set up display of SFX button
www_feedback_email_address	Define the e-mail address to which the feedback will be sent from the Web OPAC
www_item_schedule	Determine the number of days items can be booked in advance
www_lib_info_link	Enable user to see more information about a branch library
www_myeshelf_limit	Set the maximum number of records allowed in a My e-Shelf folder (including BASKET)
www_login	Set the screen to appear after log-in
www_no_items_display	Determine the number of items displayed per page
www_parallel_search_base	Define the databases searched when Multi-Base search is selected in the Web OPAC
www_short_max_no_ext_links	Limit the number of electronic locations in the Brief display
www_short_max_no_locations	Determine the maximum number of location links in the Brief list
www_sort_field	Define the default sort order for Browse results
www_sort_field_aut	

Default Parameter	Set this parameter to ..
www_sort_field_sub	Define the default sort order for Browse results
www_sort_limit	Limits the sorting of result sets
www_update_session_id	Define the default OPAC user profile
www_user_z37_sort_routine	
www_user_z37_sort_order	
www_user_z38_sort_routine	
www_user_z38_sort_order	Configure the hold/photo/loan sort order
www_z36h_max_record_limit	Limit the number of loans displayed in the Loan History list
www_z37h_max_record_limit	Limit the number of records displayed in the Hold History list
www_z37_course_request_days	Define the number of days before a course that a hold request becomes active
www_z37_date_to	Define the end of the time slot when a hold request is created
www_z37_recall_type	Set up the recall status for patron hold requests
ichain_mode	Use the ichain option

```
setenv www_no_items_display          100
```
This parameter determines the number of items displayed in one page of the Holdings Window.

If no value is defined for this parameter, the default value is 40 items displayed per page.

```
setenv www_server_timeout            300
```
This parameter defines the maximum number of seconds for the www_server timeout. It must be in the range of 5-300. Recommended values are in the range of 5-120.

```
setenv www_myeshelf_limit            100
```
This parameter is used to determine the maximum number of records that can be stored in each folder of the My e-Shelf function.

If this parameter is not defined, the default is 500 records. If the defined number is higher than 500 (for example, 1000), the system will consider it to be the default value of 500, which is the maximum that can be defined in this parameter.

```
setenv www_sort_field          "01---D02---A"
```

This setting defines the default sort order for the results of a Search query (the OPAC user can change the sort order of a set when the set is displayed in Brief Table Format). The numbers 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.

The letter "D" after 01 in 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.

You can, of course, set the order for any field to be ascending or descending.

```
setenv www_sort_field_aut      "03---A01---D"
setenv www_sort_field_sub      "01---D02---A"
```

These settings define the default sort order for the results of a Browse query (the OPAC user cannot change this sort order).

The "aut" and "sub" in the above parameters refer to the Browse Indexes as defined in tab00.lng, column 2. In this case, "aut" refers to Author and "sub" refers to Subject. You can define other parameters, but you must use the code of an index that is defined in tab00.lng.

The character strings are defined in the same way as in the Search query (setenv www_sort_field).

```
setenv www_browse_previous  Y
```

You can set the Browse list of the headings index to display in two ways:

- The entry that matches the user's query heads the list.
- The entry immediately preceding the user's query heads the list.

The starting point of the list is determined by the above setting, whereby Y (default) displays the list headed by the preceding entry, and N displays the list headed by the entry.

We recommend that the Browse list be set to display the immediately preceding entry, as it helps the user understand where he is if the entry that he requested does not appear in the list.

```
setenv www_item_schedule        14
```

This setting determines the number of days of the Reserve Item Schedule that are accessible in the OPAC for advance booking. In the above example, a user is able to book an item up to 14 days in advance.

```
setenv www_dn_link_display          01
```

Down links in a record can be displayed in one of two ways:

- Separate Entry For Each Down Link
- Referral To Links Window

The value of the above setting determines which of the two options is used. This value is defined by the system librarian. If the number of records found is less than, or equal to, the value defined, there will be a separate entry for each down link in the Full View of the Bibliographic Record. Otherwise, the down links are displayed as a referral to other windows that contain the down links. These down links are often used with analytic records.

The text that displays when a record has more than the number of links defined in www_dn_link_display is set in the first line in `tab05.lng`, for example:

```
NONE       Click to view related record
```

```
setenv www_f_sfx          "&#60;img src=&icon_path/f-sfx.gif border=0
alt='Use SFX services'&#62;"
setenv www_f_sfx_dim      "&#60;img src=&icon_path/f-sfx-dim.gif
border=0 alt='No SFX services are available'&#62;"
```

The *www_f_sfx* and *www_f_sfx_dim* variables define whether or not an SFX button will be displayed in the Web OPAC and if it will be dimmed or not, depending on the availability of SFX services.

In the Full display (direct-head, full-set-head), the SFX button can be made sensitive to different SFX services. The placeholder for the SFX icon is $1100. The services that are checked are defined in tab_sfx. For example, if you want the SFX button to be available only if full text is available, the tab_sfx setup is:

```
USM01                www_f_sfx_ft         12 N getFullTxt
```

Note that although the SFX button can be displayed on the list of items (placeholder $1500 on item-global-body) and on the Brief list (www_tab_short), this SFX button cannot be made sensitive.

```
setenv www_dn_view_display Y
```

This parameter, if set to Y, allows "View down links" to display in the full view of a record in Web OPAC. This creates a list based on the z103-lkr-text-n sorted by z103-sort. It also displays a link “Create set of down-linked records". This link creates a new set in the results list containing the linked records.

The display works in accordance with the `www_dn_link_display` variable.

```
setenv set_word_limit           1000
```

set_word_limit defines the maximum number of words that are retrieved when truncation is used (for example, find a? runs a FIND query on all words beginning with *a*). If a find limit exists, this is the number of distinct words included in a given search. The maximum value allowed is 9999.

```
setenv mark_match_words_prefix      "<span class=text2 id=normalb>
setenv mark_match_words_suffix      "</span>
```

In the Full View of record display (controlled by edit_doc_999.lng), the system highlights the search terms that have been used to find the record. This includes highlighting words from a FIND query and the entry from a browse list. The format of the highlighting is defined using HTML prefix and suffix coding. The above example uses the exlibris.css cascading style sheet to set bold highlighting.

The system default is:
```
mark_match_words_prefix "<b>"
mark_match_words_suffix "<b>"
```
The highlighting can be turned off by defining a non-HTML prefix and suffix. For example:
```
mark_match_words_prefix "<x>"
mark_match_words_suffix "<x>"
```
The highlighting of particular words can be turned off by listing them in tab06.

```
setenv www_user_z37_sort_routine      00
setenv www_user_z37_sort_order        A
setenv www_user_z38_sort_routine      00
setenv www_user_z38_sort_order        A
```

The above display parameters allow you to configure the hold/photo/loan sort order in the Web OPAC:

The routines available are:

z37 (Holds):
00 = z37-priority z37-request-date z37-open-date z37-open-hour

01 = z30-sublibrary z30-item-status z30-collection z37-open-date z37-open-hour

02 = z30-sublibrary z30-item-status z30-collection z37-status z37-open-date z37-open-hour

04 = sort using group ID and sequence, priority, request interest date and request date

05= sort using open date and requester name

z38 (Photocopy):

00 = z38-open-date
08 =z38-open-date and z38-requester-name

The sort orders available are:
A = Ascending Order
D = Descending Order

```
setenv www_short_max_no_ext_links "02"
```
This parameter limits the number of electronic locations that can appear in the Brief Display format in the Electronic Location column.

```
setenv www_short_max_no_locations "03"
```
This parameter determines the maximum number of location links in the Web Short Display (Brief List).

If this limit is exceeded, a message displays. The message is set in lines 9553 and 9564 of `/alephe/error.lng/www_f_heading`

The location information is limited to 5000 characters.

```
setenv sfx_base_url    http://demo.exlibrisgroup.com:9003/demo
```

Defines the IP address of the SFX server.

The sfx-base-url variable is ADM library-sensitive. This means that different URLs can be defined for different ADM libraries, in the following manner:

```
setenv sfx_base_url       http://demo.exlibrisgroup.com:9003/demo
setenv sfx_base_url_usm50 http://demo.exlibrisgroup.com:9004/demo
setenv sfx_base_url_usm51 http://ram47:8888/exlibris
```

When a patron logs into the Web OPAC, his session identifiers include which base URL will be used for the SFX link. The base URL will be set in the following manner

- If there is a Web cookie named "BASE-URL", the value of that cookie should contain the SFX host name and instance, in the format *http://host-name:port/instance*. In that case, the value of the cookie will set the base URL, and the `www_server.conf` variable is irrelevant.

- If there is no Web cookie, the ADM-specific `www_server.conf` variable will be used, according to the patron's profile ADM library. For example, if the patron has the ADM library USM50 set as the ADM library in his profile (or his group profile or ALEPH profile), the `www_server.conf` variable *sfx_base_url_usm50* will be used to set the patron's base URL.

- If there is no matching ADM-specific `www_server.conf` variable, the *sfx_base_url* variable will set the base URL.

`setenv sfx_fetch_address ,Ip=whitman.exlibris.com,Port=9909`

The Ex Libris SFX server can retrieve additional data from an ALEPH server. For this purpose, the OpenURL contains a link to the full document record in the ALEPH database, in the "pid" parameter. The setup for the pid is defined in the display parameter above. You must define ",Ip=" followed by your ALEPH server address and your Z39.50 port.

`sfx_issn_routine`

This variable defines the field from which the ISSN component of the SFX link will be constructed. This parameter has the following configuration options:

00 – Extract the last 022 $$a

01 – Extract the last 022 $$y if the default routine does not find any ISSN.

`setenv show_copyrights Y`

This parameter determines whether or not copyright information is displayed when a patron uses an external link via the Web OPAC. If it is set to Y, copyright information is shown to the patron if the 856## subfield $$4 is not cataloged.

If `show_copyrights` is set to N the system does not show copyright information when a patron uses an external link via the Web OPAC.

`setenv www_con_lng ENG`

This sets the default language of the Web OPAC interface.

`setenv www_course_email_address`

This determines the e-mail address to which course reading material will be sent (from the Web OPAC) for adding to the Course Reading module. If the library decides to use an alternative mail, it must be defined in `/alephe/www_f_eng/myshelf-course`. In this case, the system ignores the address in `www_server.conf`.

If you have a multi-ADM environment, it might be useful to define an alternative e-mail address of a Course Reading administrator.

`setenv create_statistics Y`

This setting defines whether or not records are to be written in the Z34 (IP Statistics) table for statistical analysis of Web server use. If it is set to Y, the "last-z34-sequence" counter must be defined in the Z52 table (UTIL G/2) of the OPAC library.

`setenv server_default_charset iso-8859-1`

If the Web server detects that the Web OPAC browser is not UTF-8 enabled (that is, the browser is not Netscape 6+ or Microsoft Internet Explorer 5+), the system overrides the UTF-8 parameter in the HTML files, and the data is sent back from the server in a standard character set. The character set to be used is defined in this line.

`setenv www_z37_recall_type 03`

A Web OPAC user can request an item by placing a hold request. The user's request automatically receives the status set in the above parameter. The patron cannot change this status.

These are the available statuses:

01 = Regular
02 = Rush
03 = No Recall

If you choose 02 (Rush), all hold requests made in the Web OPAC automatically receive the highest level priority of "00", the lowest being "99".

`setenv www_z37_course_request_days 20`

This parameter defines the number of days prior to the beginning of a course that a hold request, on an item that needs to be placed in the Course Reserves module, becomes active.

`setenv www_z36h_max_record_limit 30`

From the User Information screen in the Web OPAC, users can view the history of their loans sorted by due dates in descending order, that is, a due date of 15th July, 2004 precedes a due date of 20th June, 2004. This line allows you to set a limit to the number of previous loans that displays in this list. If the line does not appear in the table or if it is commented out, the default number of loans is 80.

```
setenv www_z37h_max_record_limit          30
```

From the User Information screen in the Web OPAC, users can view the history of their hold request sorted by History Date in descending order, so that a request that moved to history on August 30, 2011 precedes a request that moved to history on August 20, 2011. This line allows you to set a limit to the number of hold requests that are displayed in this list. If this line does not appear in the table or if it is commented out, the default number of holds is 80.

```
setenv www_update_session_id          "UPDATE_ALEPH"
```

This parameter defines the default user profile for OPAC users.

```
setenv set_default_z110_create_order          "D"
```

The display of a result set is limited to a maximum of 1000 records. When the result list is over the maximum number, the *set_default_z110_create_order* environment variable comes into play. If it is set to *"D"*, the displayed set includes the 1000 highest record numbers, and if it is set to *"A"*, the displayed set includes the lowest 1000 record numbers.

```
setenv hol_filter_op            BASE
setenv hol_sort_op              BASE
```

These parameters determine the filtering and sorting options of the Holding records list on the Holding Information display. HOL records are filtered according to the search base and the parameters given in `tab_base.lng.conf`. The rest of the HOL records are sorted again, according to the list in `tab_base.lng.conf`.

If the variables are not present in www_server.conf or if the value of hol_filter_op and hol_sort_op in tab_base.lng.conf are "NO-FILTER" and "NO-SORT" respectively, the HOL records are not filtered or sorted.

```
setenv allow_create_new_profile             Y
```

If set to Y, ALEPH creates a personal profile for the user (if a personal profile does not exist) when the user requests that his preferences are saved.

```
setenv set_prox_limit      100
```

This parameter places a limit on the number of documents that will be checked for proximity. The default is 500.

If the value of the set_prox_limit parameter is less then the number of results retrieved before the proximity filter is selected, then none of the results are checked for proximity. For example, the set_prox_limit value is 100, and a user searches for *art !4 history*. If he retrieves more than 100 records, none of the results will be checked for proximity.

If the ADJACENCY_TYPE parameter in aleph_start is set to 0, (that is, word pairs are not built), when a Search query includes Adjacency, the query is parsed as %0, and the set_prox_limit is active.

If the ADJACENCY_TYPE parameter in aleph_start is set to 2, then set_prox_limit is not relevant for a Search query including Adjacency.

```
setenv direction_support          Y
```

Supports right-to-left, right-aligned and right-oriented (for Hebrew and Arabic) text display in all formats in Web OPAC files.

```
setenv www_login                    "find-b"
```

Sets the default page which appears after a user has signed in. Valid values for this variable are find-a, find-b, find-c, find-m and find-d.

```
setenv www_z37_date_to              M003
```

This variable defines the end of the time slot (Dates of Interest) when a hold request is created. The definition must include one character that describes the time period (D-day, M-month, Y-year), followed by a three-digit value for the number of time units.

For example, M003 creates a hold request that starts from the current date and ends in another three months.

If the variable is not defined properly, the default time slot is three months from the current date.

```
setenv www_lib_info_link            Y
```

If set to Y, a user can click on an "About library" hypertext link in order to see more information about a branch library.

Note: this setting does not influence ITM4 (links to remote library items, with no circulation information, set in the edit_doc_999.lng table of the library's tab directory).

```
setenv www_parallel_search_base USM01-UNI01
```

This parameter defines the databases searched by a parallel find command (find-m-acc). The find command is activated from the ALEPH services menu "Find M documents in the database". In this example, the parallel find command will search USM01 and UNI01.

```
setenv due_date_format              2
```

The due date format of loans which appear in the Holdings window are displayed according to the following formats:

- 1 - Display a single effective due date, based on the RECALL-METHOD.
- 2 - Display the recall-due-date, if there is one (even if it is later); otherwise, display the regular due date.
- 3 - Always display the z36_due_date; (for recall methods 2 and 3 this means that the recall-due-date is never displayed).
- 4 - Display both: the regular due date and (if there is one) the recall-due-date.

`setenv www_bor_registration_ends 20050101`

Defines the default date that is entered in the *Address valid to* field in the New Borrower Registration form if no date has otherwise been entered. This date represents the end of the borrower's address validation.

`setenv ill_total_limit_period 1001`

This variable defines the date from which the counting of an ILL request is started. The ill_total_limit_period format is *mmdd* (in this example, 1001 is the first of October). All GUI and Web OPAC ILL requests created between these dates are counted and compared to the patron's ILL total limit. The patron is blocked if he exceeds its limit within this period. To override this calculation, either comment out this variable or enter 0000.

`setenv www_a_s_time_out 100`

Defines the time period (in seconds) that elapses after usage is discontinued before the user is disconnected from the network.

The above definition means that a user can stay logged in for 100 seconds.

The predefined default setting in an ALEPH installation is 600 (secs.).

`www_feedback_email_address`

This parameter defines to which e-mail address the feedback will be sent from the Web OPAC. If this parameter is not defined in `www_server.conf`, define it on the feedback page (`../alephe/www_f_eng/feedback`) in the Web OPAC as follows:

```
</div>
<form name=form1
      method=post
      action="&server_f"
      onSubmit="my_replace(document.form1.text)";>
<input type=hidden name=func value=feedback>
<input type=hidden name=file_name value=find-b>
<input type="hidden" size=50 name="EMAIL" value="Please fill here the
library email address">

<div class=text3>
    Your constructive feedback will be greatly appreciated.
</div>
```

Note that defining this line does not cause it to display on the Feedback page.

`setenv set_result_set_limit 2000`

This parameter defines the find command limit. The number of hits for a find command has the default of 1000. It is now possible to change this limit to a number in the range of 1000-20000 by setting the `set_result_set_limit` variable.

To facilitate this, you can change the following line in `short-head-2` from:

```
Records $0100 -$0200 of $0300 (maximum display and sort is 1000
records)
```

To:

```
Records $0100 -$0200 of $0300 (maximum display and sort is
$4500 records)
```

```
setenv keep_z39_gate                Y
```

This parameter prevents the automatic start of `z39_gate`.

For every incoming request, the Web server checks if `z39_gate` is up and running. If the gate is down, the gate server process will be started automatically. To prevent the gate server process starting automatically, set `keep_z39_gate` to "N".

The default is "Y".

```
setenv www_sort_limit             800
```

This parameter limits the sorting of result sets in the Web OPAC.

It contains a number of up to six digits; its default value is 999999 (that is, no sort limit). If the number of hits returned by the search query exceeds the sort limit, the result set will not be sorted.

Note

The maximum number of documents a result set can contain is controlled by the environment variable "set_result_set_limit", set either in www_server.conf or in pc_server_defaults. Make sure that the sort limit is not greater than "set_result_set_limit".

`ichain_mode`

Setting this variable to Y will run the www server in ichain-mode. When in this mode, the web server session is synchronized with the HTTP "Authorization" header. In other words, if there is no "Authorization" entry in the HTTP header, or if the header contains invalid login information, the ALEPH session will be that of a guest. If the "Authorization" entry contains valid login information, then that information will be automatically used to log in the patron.

```
setenv patron_notice_cash           Y
```

When this parameter is set to Y, the system checks the patron's cash transaction after the patron logs into WWW OPAC and displays the “Pending Cash Transaction” notice if unpaid cash transactions are detected. (HTML notice page: ./alephe/www_f_eng/patron-notice).

```
setenv patron_notice_delinq         Y
```

When this parameter is set to Y, the system checks the patron's delinquencies after the patron successfully logs into WWW OPAC and displays the “Delinquency” notice if delinquencies are detected. (HTML notice page: ./alephe/www_f_eng/patron-notice).

```
setenv patron_notice_overdue        Y
```
When this parameter is set to Y, the system checks patrons loaned items and displays the “Overdue Items” notice if the patron currently is holding overdue items. (HTML notice page: ./alephe/www_f_eng/patron-notice).

```
setenv patron_expiry_alert   030
```
When a patron logs on to OPAC, the system checks if any of the patron’s local accounts (z305) expire in N days. The number of days is defined in the `patron_expiry_alert` variable (Possible values: numbers between 1- 999).
If one (or more) of the accounts expire during this period, a message is displayed in the patron notice screen, alerting the patron about the coming expiration and the Library Card displays a link for extending the user's account. The extension period is defined in tab31 (current date plus number of days defined in tab31 columns 12-14). Depending on tab18.<lng>, the user is charged for the amount specified for cash transaction 029.

Accounts that are already expired are not extended.

```
setenv header_no_server_info        Y
```
When this parameter is set to Y, the www server does not display information (“ALEPH/5.20”) in the HTTP headers.

2.1 aleph_start Parameters

To ensure the display of authority records with multiple XYZ subfields use the following variable:

```
setenv include_all_aut Y
```

If the parameter is set to "Y", all entries which are connected to authority records will be displayed in the browse list of the Web OPAC, even if there are no records connected to this access. This applies only to codes using Z01 and not to those using Z0102. Z0102 continues to use the XYZ mechanism.
If the parameter is set to "N", the entries will not display. For example, the following will not be displayed:

```
450 0 L $$aFolk$$xCollectors and collecting$$xFolk and stories
```

This variable is optional and its default is N.

The following variable effects how ALEPH sends bibliographic information to the SFX server when using the SFX button:

```
sfx_first_isbn
```

If the BIB record has more than a single ISBN, the variable can be used to set which ISBN of the BIB record will be sent to SFX. Possible values are:

Y – Only the first ISBN of the BIB record will be sent to the SFX server.

N – Only the last ISBN of the BIB record will be sent to the SFX server. This is the default value if nothing is specified.

This environment variable is relevant for USMARC only.

3 Patron Registration and Profiles

3.1 Patron Registration

The patron registration screen is determined by the bor-new HTML file. The patron has to fill in the mandatory fields on this screen in order to complete his registration. If certain non-mandatory fields are not filled, their values are taken from the following places:

- The Patron ID field value is taken from the USR00 library's last-bor-id-1 counter.
- The Patron Verification field value is taken from the USR00 library's last-bor-verify-2 counter.
- The Address Valid From field's value contains the current date
- The Address Valid To field's value is a default value set in the program. If at least one of the "Address Type", "Address valid from" and "Address valid to" fields is empty, then the system assigns default values in the following way.

  If the ADM library to which the patron's registered home library belongs has a non-empty tab_bor_address, the values are taken from it. Otherwise, they are taken from the tab100 variable DEFAULT-ADDRESS.
- The system may be set up to allow the registering patron to register also an additional ID. This can be done by adding the following to the HTML page bor-new:

```
<tr>
 <td class=td2 width=20% nowrap>Additional ID:</td>
 <td class=td2> </td>
 <td class=td2>
     <input name=N_ADDITIONAL_ID_DATA
      size=30 maxlength=20 value="$6800">
 </td>
</tr>
<tr>
 <td class=td2 width=20% nowrap>Additional ID Ver:</td>
 <td class=td2> </td>
 <td class=td2>
     <input type=password name=N_ADDITIONAL_ID_VER
```

```
      size=30 maxlength=20>
  </td>
 </tr>
 <tr>
  <td class=td2 width=20% nowrap>Add.ID Ver.Repeat:</td>
  <td class=td2> </td>
  <td class=td2>
      <input type=password name=N_ADDITIONAL_ID_VER_CHECK
       size=30 maxlength=20>
  </td>
 </tr>
```

The Additional ID field is optional, and the verification is required only if the ID has been filled in.

The type of ID that will be added is set in the usr_library's tab100 variable ADDITIONAL-ID-TYPE.

- You can set up the registration page (bor-new) to require the registering patron to fill in the home library. To do this, add the following to the bor-new page:

```
<select name=M_Z303_HOME_LIB>
   <option value=" "></option>
   <option value=LIB1 $$2400-S"Library 1"> Library 1</option>
   <option value=LIB2 $$2400-S"Library 2"> Library 2</option>
</select>
```

If you do not want the home library to be a mandatory field, either use N_Z303_HOME_LIB instead of M_Z303_HOME_LIB, or replace:

```
<option value=" "></option>
```

With:

```
<option value=EMPTY $$2400-S"EMPTY"></option>
```

In the event that the home library is not filled in, the library that is registered with patron privileges (‘Library Branch’) will be used as the patron’s home library.

If the above section is not added to the bor-new page, the patron will not be asked to fill in a home library. The library that is registered with patron privileges (‘Library Branch’) will be used as the patron’s home library.

- A local patron record (Z305) is created for the sublibrary chosen by the user. The settings for the patron's Local Privileges and expiry date are defined in the bor-new HTML file. An ALEPH local patron is automatically created, with privilege and date values taken from `tab31` in the ADM library. For more information on Patron Registration, see the Patrons chapter in the Circulation guide.

3.2 Patron Profiles

Patron profiles are set in the Circulation GUI, either by using the Patrons / Patron Profiles menu commands, or by using the Patron bar. You must define a default user profile (ALEPH) for Web OPAC users. You can also define individual or group user profiles for library patrons. An individual profile has the same ID as the user, and does not have to be entered in the profile field of the user record. A group profile must be entered in the profile field of the user record. These profiles set preferences for display, language, database and home library, as well as permissions for specific utilities and libraries. A comprehensive explanation of the default user profile setup is contained in the Patron Profiles section of the Circulation - Patrons chapter.

You can also allow the automatic setup of new profiles (see explanation of setenv allow_create_new_profile.

There are two types of Web OPAC users: identified and unidentified.

A user who has not identified himself (by signing in) is assigned a default profile. If the system librarian has set up profile(s) that match the IP address of the user workstation, the profile that most closely matches the IP address is used. If there is no matching IP address profile, the default profile is ALEPH.

When a user identifies himself, if the user has a matching personal profile (that is, the profile ID is the same as the user ID), this profile is used. If there is no matching profile, the system uses the group profile registered in the user's profile field. If the profile field is blank, the IP or ALEPH profile remains in effect.

Both unidentified and identified users can override the default user profile when they access the Web OPAC by setting their own preferences. Identified and unidentified users differ as follows:

- Identified users who have a personal profile - A user who has signed in can enter his own preferences and save them, using the Display Formats window accessed via Preferences on the menu bar. When an identified user clicks End Session on the menu bar, the display format available for the next user reverts to the default user profile.

- Identified users who have a group profile or Unidentified users - Any user can enter his own preferences, using the Display Formats window accessed via Preferences link on the menu bar. If the user has been assigned a group profile, or if the user has not signed in, the changes in preferences cannot be saved.

- Identified users who do not have a profile listed in their user record - When the user enters his preferences, if the profile field of the user's record is blank, a profile record is created (in which his preferences are saved and permissions are defaulted to the permissions of the default profile), and the profile field is updated.

User profiles are contained in the Z61 table. Apart from setting user preferences for the display and retrieval of information in the Web OPAC, the Z61 record is also used to define the list of logical bases to which the user is denied access. This table also determines authorizations such as:

- Whether or not the user has access to electronic locations (mainly field 856) from the Web OPAC.
- Permission to save sets on the server.
- Permission to create and send lists for Course Reading.
- SDI permission.
- Special Request permission.
- Address update permission.

The Z61 record also includes:

- Denied bases.
- The display format of the records.
- The number of records displayed per page and the order in which records are sorted for display.
- The default sublibrary for holdings display in OPAC.

To ensure that the default sublibrary for holdings is displayed, check that the following line in `edit_doc_999.lng` is present:

```
## ITM5                 D LHoldings (5)        Y       E
```

The User Profile can be created through the Profile option in the Circulation module. The only fields that can be changed by the user through the Web OPAC are those related to the display format. Other fields, such as the denied records (base), can be changed and updated only through the Circulation module.

Usually, the Z61 table is not converted and it is built in the system online.

3.3 Interface Language

You can set up the Web OPAC interface in numerous languages. Each language interface has its own set of HTML files (in the `www_f_lng` directory), where *lng* is replaced by the three-letter code of the language. Tables which contain formats for data display (located in the library's tab directory) must have the proper language extension as well (for example, `edit doc_999.ger`).

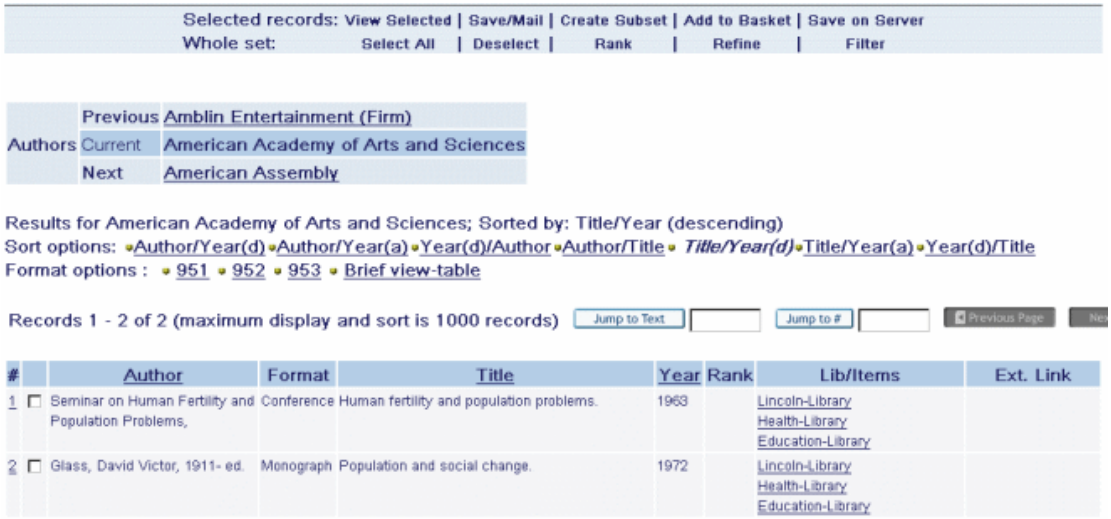
When a Web OPAC user signs in, he is assigned his interface language default as defined by the System Librarian in the Profile Information window in the Circulation GUI (See the Patron chapter in Circulation, Patron Profiles).

If a Web OPAC user does not sign in, he is assigned the default profile (including the default language) that has been defined for guest users by the System Librarian in the Profile Information window in the Circulation GUI.

Any Web OPAC user can change the interface language, but only a signed-in user can save his change of the interface language default.

3.4 Brief View of Records

The Brief View is available in a number of different formats. The table format is the most common:



You define which format is displayed by default and which fields are displayed by entering the appropriate setting in the Profile Information window in the Circulation GUI for the default profile (ALEPH). (See the Patron chapter in Circulation, Patron Profiles).

Table Format

If you want the **table format** to be displayed, enter "000" in the Brief Format field from the Profile Information window in the Circulation GUI for the default profile (ALEPH). (See the Patron chapter in Circulation, Patron Profiles).

To determine which fields are displayed in the brief view table format, edit www_tab_short.

Customizing Other Formats

You can set up other custom formats by editing the `www_f_tab_short_999` table.

If you want a custom format to be displayed by default, enter its format number in the Brief Format text box in the Profile Information window in the Circulation GUI for the default profile (ALEPH). (See the Patron chapter in Circulation, Patron Profiles).

Brief Records
To define the number of Brief records displayed on one page, enter a value in the **Records per file** text box, for example 20, from the Profile Information window in the Circulation GUI for the default profile (ALEPH). (See the Patron chapter in Circulation, Patron Profiles).

In this example, up to 20 records can be displayed on one screen (20 being the maximum number of records possible). The Web OPAC user can go to the next 20 records by clicking **Next**.

3.5 Full View of Records

You can set up the system to automatically display records in Full View, if the number of records in a result set does not exceed or equal a specified number. You set this number in the Auto Full box in the Profile Information window in the Circulation GUI for the default profile (ALEPH). (See the Patron chapter in Circulation, Patron Profiles).

For example, if you enter "05", records are displayed in Full View if a query results in less than five records.

Note that if you enter "00", all records will automatically display in Full View.

3.6 Additional Optional Fields

The following fields can also be configured in the bor-new file:

Z304_ADDRESS1
Z305_PHOTO_CHARGE
Z305_HOLD_ON_SHELF
Z305_BOR_TYPE
Z303_ILL_ACTIVE_LIMIT
Z303_ILL_TOTAL_LIMIT
Z303_ILL_LIB

4 HTML Files and the Interface

ALEPH Web OPAC screens are based on HTML (HyperText Markup Language) files. The design of these screens is based on Cascading Style Sheets (CSS). The HTML and CSS files are located in the `/alephe/www_f_` directory.

You can design Web OPAC files in accordance with your library's needs. Files must be written in accordance with the rules of HTML. This section of the ALEPH User Guide shows you which parameters can be modified and which parameters must not be changed. This section assumes that you have a working knowledge of HTML.

4.1 ALEPH and CSS

ALEPH uses Cascading Style Sheets (CSS) for adding style definitions to Web files. CSS gives more control over the look and layout of HTML files. The use of CSS controls:

- Fonts, font sizes, and font colors
- Line spacing and length
- Margins and indents
- Background colors

... and more.

ALEPH uses a single external style sheet, **exlibris.css**, located in the alephe/www_f_<lng> directory. The external style sheet is linked to HTML files through the HTML <link> tag:

```
<link rel="stylesheet"
href="&server_css/exlibris.css"TYPE="text/css">
```

The <link> tag is placed in the **meta-tags** HTML file located in the `alephe/www_f_<lng>` directory. The external style sheets must not contain any HTML tags such as <head> or <style>. The style sheet must consist of style rules or statements only. Among the benefits of using a single CSS are:

- The ability to ensure a consistent style.
- The ability to update the look of the site with a single change.
- The HTML code becomes much simpler and more manageable.

Font definitions in the `exlibris.css` file control more than just the displayed text in Web files. Font definitions also control the display for the retrieved data from the catalog. If the catalog contains special scripts such as CJK, Hebrew, Arabic, and so on, you must ensure that appropriate fonts have been defined. It is important that lists of font names have the fonts listed in the same order, since the browser picks the first one available. If the catalog does not have special scripts, you do not need to define fonts for them in the `exlibris.css` file.

Note that font definitions in the `exlibris.css` file are not used if the user's browser has been set up to ignore font styles. In MS Internet Explorer this is set up under Internet options -> General -> Accessibility -> "Ignore font styles specified on Web files". In Netscape this is set up under Edit -> Preferences -> Fonts -> "Use my default fonts overriding document specified font".

Note: Each browser supports Cascading Style Sheets in different ways and triggers different display behavior.

4.2 Include Files

Include files are files that contain parts which are repeated in many screens within Web OPAC, such as the menu bar that appears at the head of each file. Include files save repeated writing of HTML code and help keep a uniform look for the interface.

The include file is invoked in the HTML file through the `<include>` tag. This tag is not a standard HTML tag and is recognized only by ALEPH programs. The rules for use of this tag are as follows:

- The format for the code is:
  `<include>[file name]`
- The <include> tag must be located on the left side of the file. ALEPH programs ignore an `<include>` tag that is located in any other part of the file.
- The include files are located together with the other Web files in the `/alephe/www_f_<lng>` directory.

4.3 File Extensions

Web files can be base-sensitive. This allows the use of different files for different libraries when needed. For example, the pull-down menu of possible search lists (find-code-include) might differ from library to library. Let us say that USM01 and USM10 have different search codes. To use different search lists, you must add the library code as an extension to the "find-code-include" file. For example, find-code-include-usm10, find-code-include-usm30, and so on.

The base-sensitive functionality works in conjunction with the tab_base.lng columns 1, 4,5 and 6. Web programs search for Web files in the following order:

- Look for the file with the base name extension (Col. 1).
- If not found, look for the file with Col. 4 extension.
- If not found, look for the file with Col. 5 extension.
- If not found, look for the file with Col. 6 extension.
- If not found, look for the file with no extension.

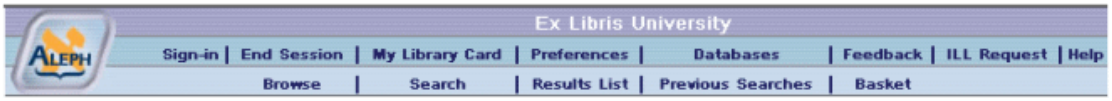
The file extension mechanism also allows you to create a different look for each base or logical base in a multi-base system. In addition, the extension code mechanism described above is also used for the following tables that govern Full record display:

- edit_doc.lng
- edit_doc_999.lng
- edit_doc_999_<aut_library>.lng.

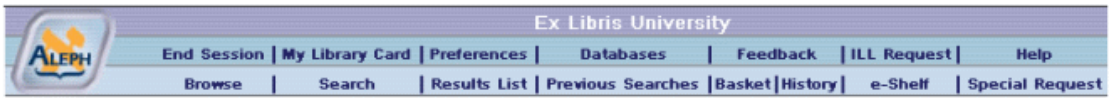
The use of the extension code with these tables lets you customize different Full record displays for different bases.

4.4 Menu Bar

The menu bar appears at the top of the Web OPAC screen. The options shown on the menu bar vary according to whether or not the user has signed in. If the user has not signed in, several options are unavailable and the menu bar looks like this:



When a user has completed sign-in, the display of the menu bar is as follows:



There are two relevant HTML files: head-1-bor and head-1-no- bor. The difference is that if a user is signed in, the head-1-bor file is displayed, and if the user has not signed in, the head-1-no-bor file is displayed.

This has been designed so that the "Sign-In" link can be changed to an "End Session" link once a user has signed in.

4.5 Encryption of patron connection information using SSL

Patron connection information is considered sensitive information. You can encrypt it using the SSL encryption protocol.

In order to send the information through HTTPS (SSL) instead of regular HTTP, carry out the following steps in all login screens:

1. In the form method definition, replace action=”&server_f” with action=”&server_f_ssl”.
2. Immediately after the form definition, add the following line:

```
<input type=”hidden” name=”ssl_flag” value=”Y”>
Here is an example:
<form method=post
      name="form1"
      action="&server_f_ssl">
      <input type="hidden" name="ssl_flag" value="Y">
```

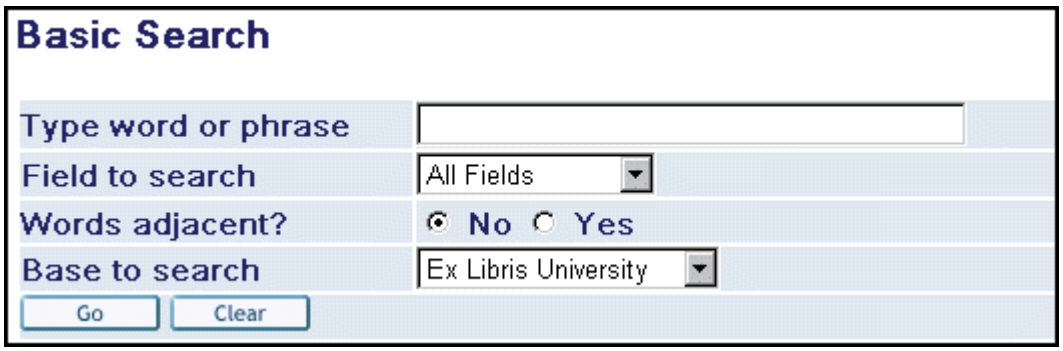
Note that in order for this setup to work, Apache and SSL must be installed, and a valid certificate placed in the system.

4.6 Setting up the Browse and Search Screens

The Browse and Search screens have similar elements as they are all made up of a combination of a number of HTML files. The HTML files are combined by using the "include" command.

Pull-Down Menus

The pull-down menus that are displayed in the Browse and Search screens are defined by a number of files. For example, the codes in the following pull-down list from the Browse screen:



are defined by the scan-include-2 HTML file, located in the `alephe/www_f_lng` directory.

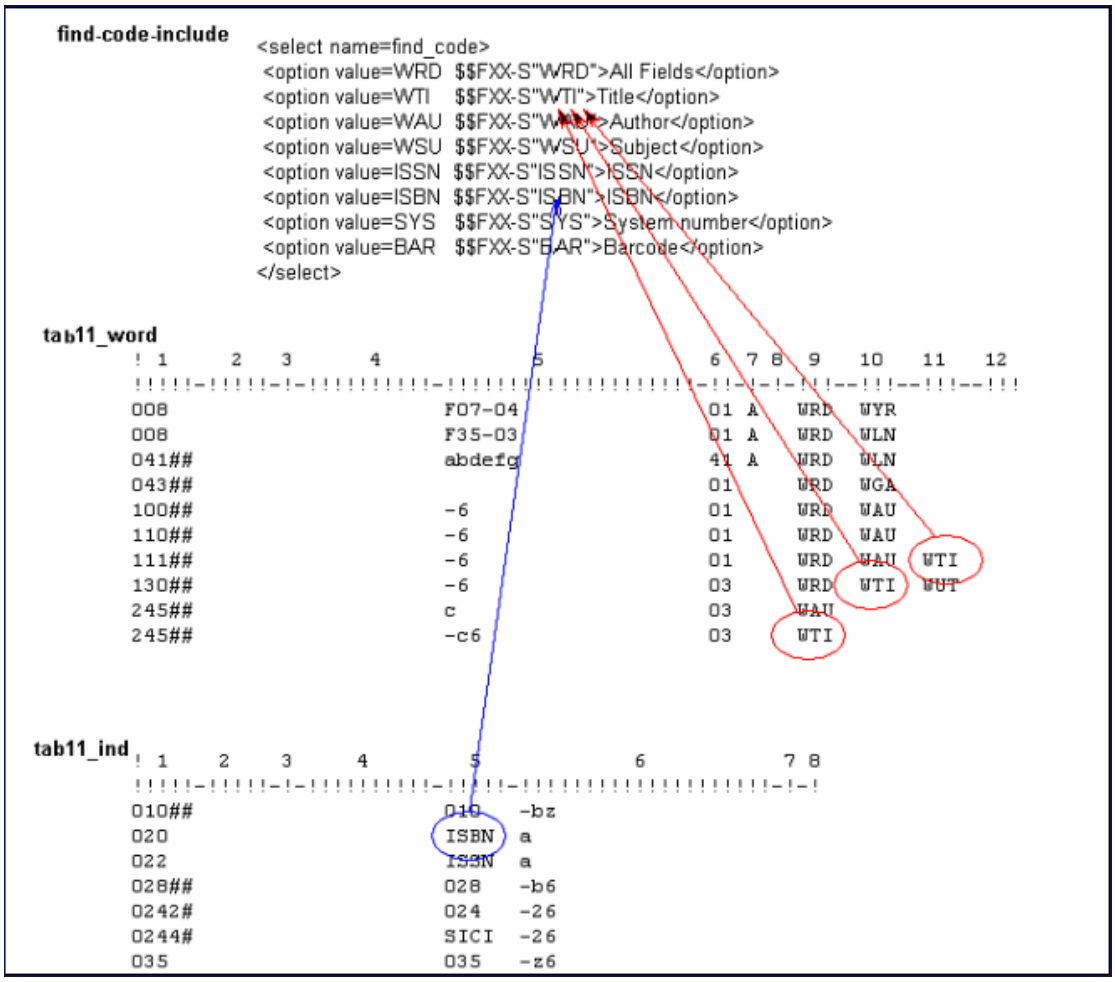
```
<select name=scan_code>
<option value=TIT $$5000-S"TIT">Title (omit initial
articles)</option>
<option value=AUT $$5000-S"AUT">Author</option>
<option value=SUB $$5000-S"SUB">Subject</option>
<option value=SUL $$5000-S"SUL">Subject - L.C.</option>
<option value=SUM $$5000-S"SUM">Subject - MESH</option>
<option value=SYS $$5000-S"SYS">System number</option>
<option value=PLA $$5000-S"PLA">Place of Publication</option>
<option value=PUB $$5000-S"PUB">Publisher</option>
<option value=SRS $$5000-S"SRS">Series</option>
<option value=SHL $$5000-S"SHL">Location</option>
<option value=ISSN $$5000-S"ISSN">ISSN</option>
```

The scan-include-2 HTML file includes a special command that causes the user's last choice to be displayed as a default choice the next time the same type of search is made.

For example, if the user chooses the "Subject" option from the Browse screen, the $$5000_S "SUB" portion of the following line causes the system to recall the last option chosen, in this case, subject.

```
<$$5000-S"SUB">Subject
```

The following figure shows the relationship between the `tab11_acc` and `tab11_ind` tables which assign fields to headings indexes and the HTML code:



There are other pull-down menus that are set by the system. For example, a similar menu exists in the Holdings screen for the Select Sublibrary combo-box.

Additional Search Options

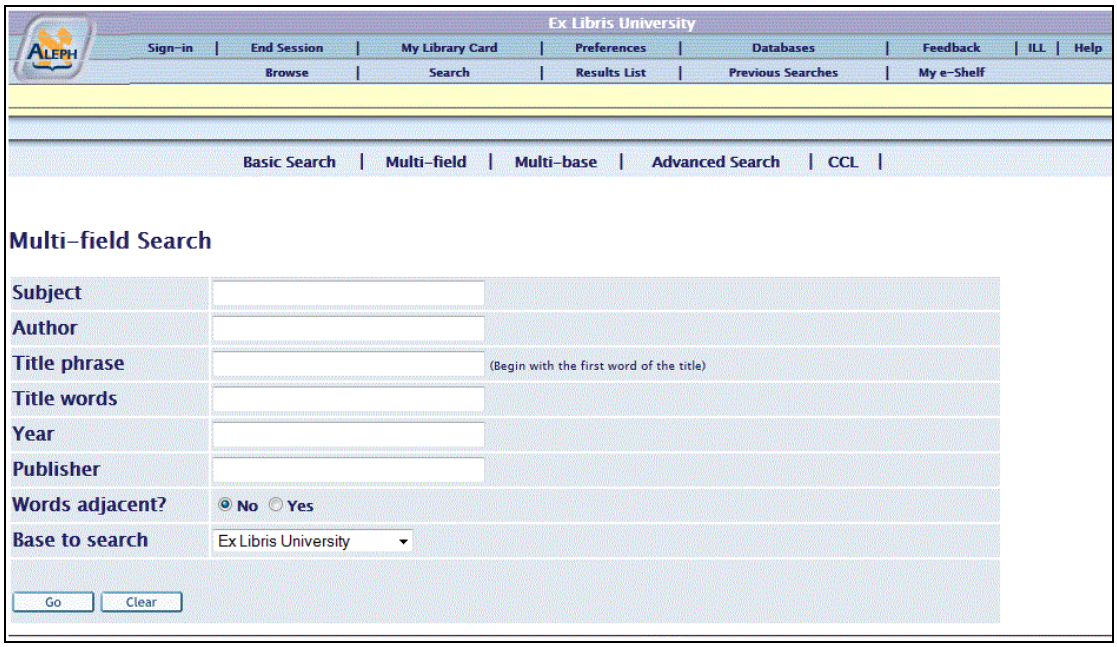
This section is intended to show you how search windows are set up in ALEPH. All the screens and codes shown in this section are pre-defined and should not be changed without consultation with ALEPH Customer Support.

In addition to basic Browse, Search and Direct Access functions, ALEPH offers a variety of more sophisticated search functions. From the Search window, the following are available:

- Multi-field Search
- Multi-base Search
- Common Command Language Search

(See Search for detailed explanations of all of these functions.)

The following *Multi-field Search* screen (filename "find-a") shows how the more sophisticated search features are incorporated into the Web OPAC:



This screen has a number of different fields for the user to fill in and each field is used for a pre-programmed purpose. For example, the ability to search the Subject index is defined in the HTML file by the following:

```
<tr>
  <th class=td2 align=left width=10% nowrap>Subject</th>
  <td class=td2 id="small">
      <input type="hidden" size=13 name=find_code value="WSU">
      <input size=35 name=request value="$0100">
  </td>
</tr>
```

Note the following points about this example:

- The word "Subject" in the first line may be changed or translated.
- The fact that the purpose of this field is pre-programmed and may not be affected by the user, is shown in line 3 by **input type="hidden"**.
- The name of the hidden field is the parameter **find_code**. Each hidden field on the screen has the same name.
- The index that will be searched is the "Words from Subject" index, indicated by **value="WSU"**. The codes for the indexes used in the screen must be ones that are defined in tab00.lng.

```
H WRD    W-001       00        01        Words
H WTI    W-002       00        02        W-titles
H WAU    W-003       00        02        W-authors
H WSU    W-004       00        03        W-subjects
```

- The name of the input field (the field that users can actually type in) is the parameter request. Each input field on the screen has the same name.

- The value of the input field is "$0100". Each input field on the screen must be given a unique value. The objective is to stores the values in a cache in the system's memory for future use.

The HTML code for the Author field is similar to the Subject field, with the addition of a hidden Boolean operator, **AND**, as shown below:

```
<tr>
  <th class=td2 align=left>Author</th>
  <td class=td2 id="small">
     <input type="hidden" size=3 name=request_op value="AND">
     <input type="hidden" size=6 name=find_code value="WAU">
     <input size=35 name=request value="$0200">
  </td>
 </tr>
```

Note that every Boolean operator is assigned a name. The name is the parameter **request_op**. Also note that in this example, the index being searched is the "Words from Author" index, represented by the code "WAU".

In the next example, for the Title field, the index being searched is the "Title Headings" index, represented by the code "TIT". Note that in this case, the search is on a headings list, and that the user must enter the title from the beginning text, although not all text.

```
<tr>
  <th class=td2 align=left>Title phrase</th>
  <td class=td2 id="small">
     <input type="hidden" size=3 name=request_op value="AND">
     <input type="hidden" size=5 name=find_code value="TIT">
     <input size=35 name=request value="$0300">
       <span class="text1">(Begin with the first word of the title)
</span>
  </td>
 </tr>
```

The ability to choose a base to search in is defined in the HTML file by the following:

```
<include>base-list-include\FXX\9000\
```

This line calls the base-list-include Web file, which is responsible for presenting the drop-down list containing the different bases. The content of the drop-down list is taken from tab_base.lng.

The drop-down list automatically selects the current base entry by using the placeholder $9000 when calling the base-list-include file.

This line also appears in the following Web files:

find-b
find-b-list-head
find-d
find-c
short-tail

The operation of the Boolean operators, AND, OR and NOT, in Web OPAC is as follows:

- Operations in parentheses are evaluated first.
- When there are no parentheses, the execution order (precedence) of the Boolean operators is as follows: NOT, AND, and OR.
- Search statements are evaluated left to right.
- The system recognizes the operators both in lowercase and uppercase.
- When the Search query is performed on a headings index (for example, TIT= gone but not forgotten), AND/OR/NOT are treated as real words, that is, not as Boolean operators.

In search statements with only AND and OR, operations in parentheses are evaluated first. Within any group of operations, NOT operations are done first from left to right followed by AND operations, and then by OR operations executed from left to right.

Search Statement	Executed as
A and B not C	A and (B not C)
A or B not C	A or (B not C)
A not B and C	(A not B) and C
A and B or C	(A and B) or C
A or B and C	A or (B and C)
A and (B or C)	A and (B or C)

Note that the operation of the Boolean operators is predefined and cannot be changed by the System Librarian.

For translation to other languages, see `tab_ccl_boolean_operators`.

Set Up the Search/Browse Page

The Search/Browse screen allows the user to make a search query and a browse query in the same page.

To implement this functionality, perform the following steps:

1. Replace the include file `scan-include-2` with `find-e-head`, in the following files:

   ```
   /alephe/www_f_eng/scan
   /alephe/www_f_eng/scan-acc-head
   /alephe/www_f_eng/scan-acc-head-hesci
   /alephe/www_f_eng/scan-acc-head-liedu
   /alephe/www_f_eng/scan-acc-head-usm10
   /alephe/www_f_eng/scan-acc-head-usm30
   /alephe/www_f_eng/scan-acc-head-usm90
   /alephe/www_f_eng/scan-acc-ug-head
   /alephe/www_f_eng/scan-ind-head
   /alephe/www_f_eng/scan-sys-head
   /alephe/www_f_eng/scan-word-head
   ```

2. Replace the include file `find-b-head` with `find-e-head`, in the following files:

```
/alephe/www_f_eng/find-b
/alephe/www_f_eng/find-b-list-head
```

3. Replace `find-b-0` with `find-e-0` in the following files:
```
/alephe/www_f_eng/base-list
/alephe/www_f_eng/base-list-usm90
/alephe/www_f_eng/find-b-head
/alephe/www_f_eng/find-b-top
/alephe/www_f_eng/find-include-buttons
/alephe/www_f_eng/find-include-buttons-usm90
/alephe/www_f_eng/head-1-bor
/alephe/www_f_eng/head-1-bor-ext-ill
/alephe/www_f_eng/head-1-nobor
/alephe/www_f_eng/head-1-nobor-ext-ill
/alephe/www_f_eng/head-1-nobor-ext-ill-pds
/alephe/www_f_eng/head-1-nobor-pds
/alephe/www_f_eng/head-replace
```

4. Remove the Browse link from the top control buttons.

Note that you can either use the browse/search functionality or the separate Browse and Search functionalities.

Floating Keyboard

Multi-script floating keyboard functionality is available in Web OPAC in the following screens:

- find-a (multi field search)
- find-m (multi base search)
- find-d (advanced search)
- short-tail

The floating keyboard can contain different writing scripts. The content of the floating keyboard can be edited in the floating-keyboard.xml file under ./alephe/www_f_lng.

A link such as the following should come after the **Go** and **Clear** buttons in the above Web pages:

For other languages click here

4.7 Result Screens

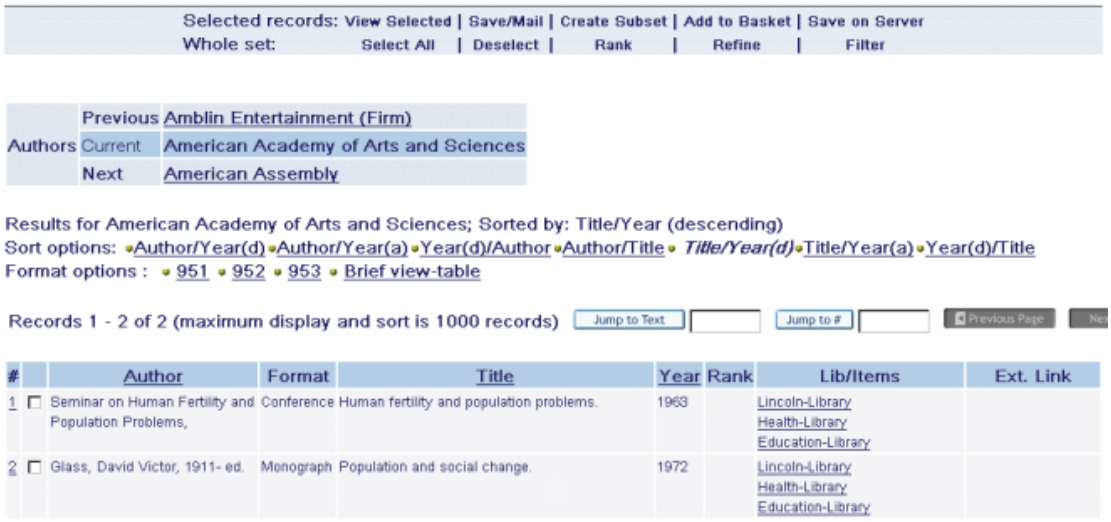
The following screens are discussed in this section:

- Brief View (Table Format)
- Other Brief Format Options
- Full View

- Sort
- Filter

4.7.1 Brief View (Table Format)

There are two brief views available in the Web OPAC, Brief View-Table and Brief View-List. This section is intended to show the reader how the Brief View-Table is set up. A basic knowledge of HTML is assumed. The Brief View, shown below, displays a list of records in a table format.



The Brief View-Table will be displayed when:

- The user chooses Brief View-Table format from the Preferences screen;

  or if

- The library has "000" set as the default display format in the Brief Format field for the profile that is in effect. The default Brief format is set in the Profile Information window in the Circulation GUI for the default profile (ALEPH). (See the Patron chapter in Circulation, Patron Profiles).

The **www_tab_short.lng** table defines which information will be displayed in the columns of the short display in table format in Web OPAC. The `www_tab_short.lng` table is located in the library's tab/ directory. You can edit `www_tab_short.lng`.

The order of the columns defined in `www_tab_short.lng` in the `alephe/tab` directory determines the order of the display, in conjunction with the short-a-head and short-a-body HTML files in `www_f_lng`. These files should have placeholders in sequential order, and should have as many placeholders (from 0300) as the number of columns defined in the `www_tab_short.lng` table.

The field you define for a column can be dependent on the bibliographic record format (FMT). You can also set up alternative definitions by assigning them the same identifying number in column 1 of this table.

Example:

```
!-!-!!!!!!!!!!-!!!!!!!!!!!!-!!-!!-!!!!-!-!-!!!!!!!!!!-!!-!!!!!-!!!!!-
>
1 L Author     02---A01---A 00 00 0100 S 9 -          BK 1####  700##
1                                                     SE 260##
1                                                     ## 1###
```

In the above example the "Author" column in the Web OPAC will be taken from fields 1#### or 700## if the document is a book (BK), from field 260## if the (SE) document is a journal, and from field 1#### in any other instance.

TYP is a virtual field defined in the tab_type_config.lng table. The TYP field indicates the type of record - (Electronic Resource, Web Resource, Map, Kit, and so on), based on coding in the LDR (pos. 06,07), 006 (pos. 00), 007 (pos. 00,01) and 008 (pos. 23) fields.

You can set a single column to display more than one field of information. In the following example, a single column is set to display more than one field of information, by repeating the column number (in this example the column number is 2) in column 1, entering the name of the field in column 3 (in this example the field name is title), and entering the field codes in columns 12 and 13 in each of the column number lines.

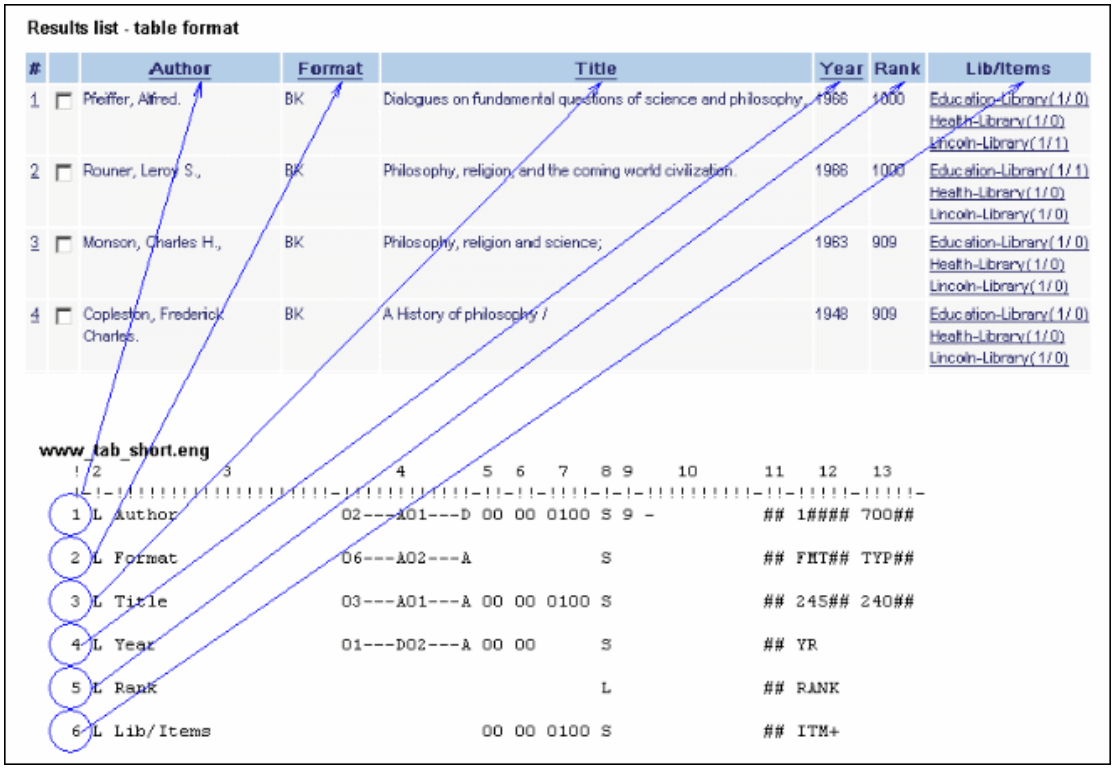
Example:

```
2 L Title      03---A01---A 00 00 0100 S              BK 245## 240##
2                                                     ## 245## 240##
```

In the above example, both column 12 and column 13 have field codes. If the record does not contain the field code in column 12, the system searches for an alternative using the field code entered in column 13.

You can display more than one field name in a column by defining a second field name by repeating the column number in col. 1, entering a name in col. 3 (this name will not display), and entering the data element code in col. 12, and (optional) alternative codes in col. 13 and on.

The following figure shows the relationship between the `www_tab_short.lng` table and the Brief View in Table format (`short-a-head` and `short-a-body`):



short-a-head and short-a-body

short-a-head and **short-a-body**, together with `www_tab_short.lng`, determine the order in which columns are displayed in the Brief View Table. short-a-head and short-a-body are HTML files located in the `alephe/www_f_lng` directory. These files should have placeholders in sequential order, and should have as many placeholders (starting from 0300) as the number of columns defined in the `www_tab_short.lng` table.

The short-a-head file defines the display of the headers according to `www_tab_short.lng`. To remove a column, for example, the "Year" column, you must delete a placeholder line from "`short-a-head`" and "`short-a-body`". If you want to add a column from `www_tab_short.lng`, you must add the relevant parameter to `short-a-head` and `short-a-body`.

The `short-a-body` file defines the display of the content of each column.

Here is an example of a typical definition for `short-a-head`. The placeholders following are used for column captions, except for 0100 and 0200. 0100 is a placeholder for a number link to the full record display, and 0200 is a placeholder for a check box. The other placeholder numbers relate to `www_tab_short.lng` where 0300 is the first column defined, 0400 is the second column defined, and so on.

Example:

```
<tr class=tr1>
  <th class=text3>$0100</th>
  <th class=text3>$0200</th>
  <th class=text3>$0300</th>
  <th class=text3>$0400</th>
```

```
  <th class=text3>$0500</th>
  <th class=text3>$0600</th>
  <th class=text3>$0700</th>
  <th class=text3>$0800</th>
</tr>
```
Note that the column headers here must match with the corresponding contents columns defined in short-a-body and must have the same order.

Here is an example of a definition for a corresponding short-a-body file:

Example:
```
<tr valign=baseline>
  <td class=td1 id=centered width="1%" valign=top>$0100</td>
  <td class=td1 id=centered width="1%" valign=top>$0200</td>
  <td class=td1 $2300 width="20%" valign=top>$0300  </td>
  <td class=td1 $2400 width="5%" valign=top>$0400  </td>
  <td class=td1 $2500 width="30%" valign=top>$0500  </td>
  <td class=td1 $2600 width="5%" valign=top>$0600  </td>
  <td class=td1 $2700 width="5%" valign=top>$0700  </td>
  <td class=td1 $2800 width="20%" valign=top>$0800  </td>
</tr>
```
The **width** parameter specifies the width of each column as a percentage of the width of the whole line.

In our examples, the placeholders refer to the following column headings:

```
Column heading   Placeholders
Author           300
Format           400
Title            500
Year             600
Rank             700
Lib/Items        800
```

If you want to add another column, for example, *Collection*, follow these steps:

1. Add a line number, 9, to column 1, and a new field name (collection) to www_tab_short.lng, (located in the library's tab directory).

2. Add a new line to short-a-head (located in the www_f_lng directory) with a sequenced placeholder.

   For example:

   ```
   <th class=text3>$0900</th>
   ```

3. Add a new line to short-a-body (located in the www_f_lng directory) with sequenced placeholders.

   For example:

   ```
   <td class=td1 $2900 nowrap valign=top>$0900  </td>
   ```

4.7.2 Other Brief Format Options

Apart from the Table format, the Brief View can also be set up to display in a List format which is customizable.

To customize Brief View List formats, enter the xxx01/tab/www_f_tab_short_999 file and modify format as required:

```
!1  2     3     4     5     6     7     8     9     10    11    12
13    14
!   15   16    17    18    19    20    21    22    23    24    25
26    27
!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-
!!!!!-!!!!!-!!!!
!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-
!!!!!-!!!!!
951 1#### 24### ITM5# DIS
952 24### DIS
999 1#### 24### ITM5# DIS
```

Note that the 999 format is reserved for the default view, and that the 950 format is reserved for the Full View. Custom formats must be numbered from 951-998. When adding a field to a custom format, make sure that the field is also present in the `xxx01/tab/edit_doc_999.lng` file. The fields are displayed according to the rules set in `edit_doc_999.lng`.

Regardless of the default, patrons using the OPAC can choose their own preferred display formats from the Preferences screen.

To edit the brief view display formats available from the Preferences screen:

1. Modify the file `alephe/www_f_eng/option-display` (for signed-in patrons) and the file `alephe/www_f_eng/option-display- session` (for guests).

2. Find the following lines in the Web file which contain the display format options available from the Preferences screen:

   ```
   <td class=td2 align="left" valign=top>Format:
     </td>
     <td class=td2 valign=top>
      <span class=text9>
      <input type="radio" name=SHORT_FORMAT
        value="951"
        $$0300-C"951">Option 1(951)
      <input type="radio" name=SHORT_FORMAT
        value="952"
        $$0300-C"952">Title only  (952)
      </span>
     </td>
   ```

3. Edit them as required.

   In addition, a library patron who has signed in can update his Personal Profile. To update the brief display format options available from this screen:

Open `alephe/www_f_eng/bor-profile-display`, and find the following lines:

```
<td class=td2 valign=top>
      <span class=text9>
      <input type="radio" name=SHORT_FORMAT
             value="951"
       $$0400-C"951">Option 1(951)
<input type="radio" name=SHORT_FORMAT
       value="952"
       $$0400-C"952">Option 2 (952)
</span>
</td>
```

These are the lines that govern the display of the brief format options available. To add options, edit these lines as required.

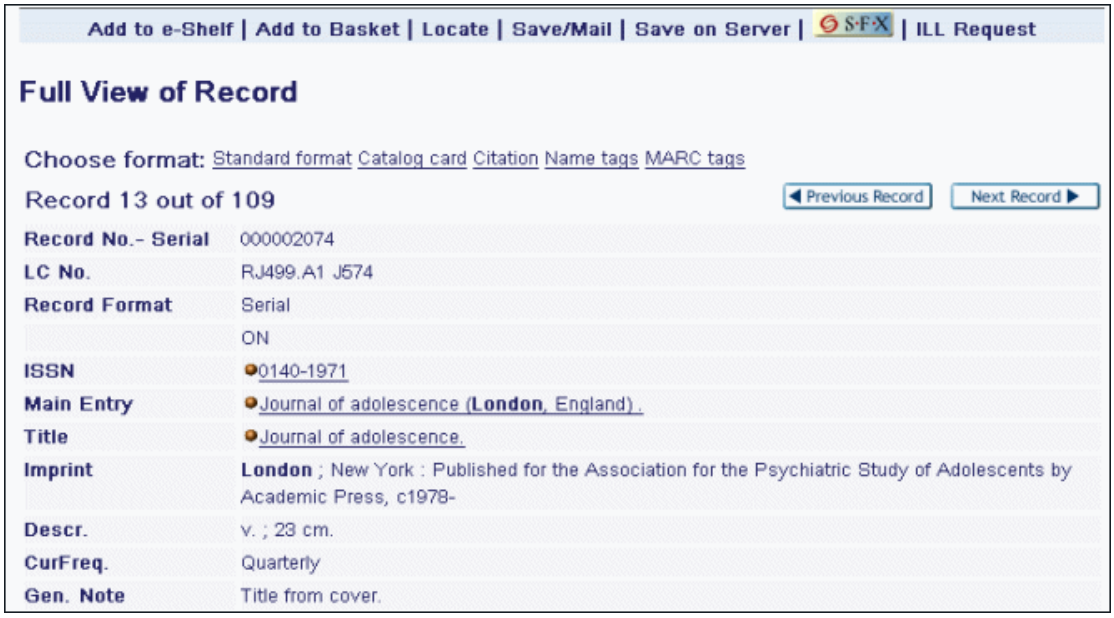
4.8 Full View

There are five Full view formats that the user can choose from:

- Standard
- Name and MARC Tags
- Catalog Card
- Citation

In addition, you can define your own Full view (which is a subset of Standard view) and Save/Mail options.

Full View - Standard



This screen is made up of full-set-head (which displays the information above the table and the top border of the table), full-999-body (which determines the

characteristics of the font and the width of the columns) and full-set-tail (which closes the table and displays the information below the table, in this case, the icons for Previous Record/Next Record).

The following is an example of the HTML code for the `alephe/www_f_eng/full-999-body` file:

```
<tr>
  <td class=td1
      id=bold
      width=15%
      valign=top
      nowrap>
    $0100
  </td>

  <td class=td1 $AL02>
    $0200
  </td>
 </tr>
```

The first <td> section shown defines the field tag; the second <td> section shown defines data display.

The values that begin with dollar signs, $0100 and $0200, are placeholders for data. The system fills in the actual data when the screen is displayed. Do not change these placeholders - otherwise the program will not recognize the data to display.

You can determine exactly which fields are displayed in the Full view by editing the edit_doc_999.lng table. Both the numeric code and the field name are set in this table.

Interaction with Syndetics

If your library works with Syndetics, you can enrich your Web OPAC by adding a cover image to the full record display. Cover images are derived from the record's ISBN and an interaction with the Syndetics Internet server. You can initialize this service by adding the value `ICON` to the `edit_doc_999.lng` table. For example:

```
## ICON                 D LCover Image          Y       E       W
```

Cover images are hyperlinked to additional catalog enrichment elements, such as summaries, author information, excerpts, book reviews, and so on. When the cover image is clicked, a pop-up window appears, containing the available enrichment options.

To indicate that your library has a license to use Syndetics, add the following line to `$alephe_tab/license`:

```
SYNDETICS           Y
```

The default is N.

The client code assigned by Syndetics to a customer must be defined in $alephe_root/www_server.conf under the variable name syndetics_client_code.

For Example:

setenv syndetics_client_code your_code

Note that the Syndetics link will not work if this code is not defined.

In the Full View of Record window, there are services which are available for a particular field.

The services currently available are scan (acc or ind), find (acc), aut, aut-bib, engine, and find-m-acc. These services are defined in `xxx01/tab/tab_service`.

- **scan** activates the display of a browse list. All the browse lists defined for the field in the tab11_acc table are displayed for you to choose from.
- **find** activates the Find command. All the word lists defined for the field in the tab11_word table are searched.
- **aut** activates the display of the authority record that is linked to the field's heading.
- **aut-bib** is used only for an Authorities library (XXX10). This service allows the user to initiate a search in an Authorities database to find a relevant record. From this record, the aut_bib service can be used in order to create a set of the relevant Bibliographic records, that are indexed using the same text as one of the AUT fields. This service invokes a FIND query from the AUT record on a SCAN (Z01) list in the BIB library. The parameters for the query (that is, which BIB library and BIB SCAN list) are defined in the www_f_<lng>/service-aut-bib HTML file. The FIND command is truncated at 50 characters and normalized using filing procedure **98**. You can strip unwanted subfields from the AUT record field. Do not include a character conversion in this procedure; the BIB library's relevant character_conversion program is used. This is particularly useful for libraries which use UDC/DDC classification for subject indexing of BIB records, and for an authority database in which each UDC/DDC is an authority record with textual descriptor.
- **engine** activates an external search engine, using parameters stored in the xxx01/tab/tab_z121 table. The parameters include the search engine name, the URL and the search query string. For example

  ```
  !!!!!!!!!!!!!!!!!!!-
  !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!>
  GOOGLE
  http://www.google.com/search?q=$0100&meta=lr%3D%26hl%3Den
  ```

  You can add a new search engine, using the same syntax as above. Note that the search engine name is the same as the link in the www_f_lng/service-engine HTML file, except that in tab_z121, it must be uppercase, and in service-engine, it can be in uppercase or in lowercase. For example:

```
<select name="engine">
          <option value=google>Google</option>
                .../...
       </select>
```

If you do not want the search engine to search the whole field but only specific subfields, you can define the subfields in column 3 of `tab_service`. If you leave column 3 blank, the search is made on the whole field.

Here is an example:

```
! 1   2                                3
!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
100##  www_f_service_engine           a
```

- **find-m-acc** activates a parallel find command on a number of databases simultaneously. The databases to search are defined in the www_server.conf table in alephe, in the environment variable www_parallel_search_base. The base codes are entered in one string, with a hyphen separating each one. Here is an example:

  ```
  setenv www_parallel_search_base       USM01-UNI01
  ```

  The creation of a link is defined by:

  The presence of *M* in Col. 10 of the field defined in edit_doc_999, and the presence of the field tag in tab_buf_Z403, or

  The presence of *S* in column 10 of edit_doc_999.lng as shown in the following example:

  ```
  !1   2   3      4     5 6 7         8          9 0 11  2  13
  4 15 16
  !!-!!!!!-!-!!!!!!!!!!-!-!-!!!!!!!!!!!!!!!!!!!-!-!-!!!-!-!!!!!-
  !-!!-!!
  ## 500##                D LGen. Note           Y S     E
  ```

  The *S* link type modifies a URL with HTML tags so that no copyright warning is shown and no patron authorization is checked. The *S* link type is designed to be used in free-text fields.

Full - Name & MARC Tags

These screens are made up of the following files in the `alephe/www_f_lng` directory: `full-set-head` (which displays the information above the table and the top border of the table), `full-000-body` (which determines the characteristics of the font and the width of the columns) and full-set-tail (which closes the table and displays the information below the table, such as the icons for Previous/Next).

In the `full-000-body` file, do **not** change the placeholders for data (placeholders are the values that begin with a dollar sign ($)).

These formats display all of the fields that are included in the Catalog record. The names of the fields are derived from `tab01.lng`: MARC tags are taken from Col. 2; name tags are taken from Col. 9.

Services (aut, aut-bib, and so on) cannot be displayed in these formats, and editing is not possible.

Full - Catalog Card

This screen is made up of the following files: `full-set-head` (which displays the information above the Catalog Card and the top border of the Card) and `full-xxx-body` (which determines the font characteristics of the text displayed in the Catalog Card) and full-set-tail (which displays the bottom border of the Card and displays the information below the table, such as the icons for Previous Record/Next Record).

In the `full-xxx-body` file, do **not** change the placeholders for data (placeholders are the values that begin with a dollar sign ($)).

You can control the fields that are displayed in the Catalog Card by editing the `edit_doc.lng` table, and changing the format numbered "037", and/or editing the lines in `edit_paragraph.lng` that are used by the `edit_doc.lng 037` format.

Full - Citation

This screen is made up in the same manner as Full - Catalog Card. The only difference is that Citation uses the "038" format in the `edit_doc.lng` table.

Define Your Own Full View

You can define other paragraph-style formats in addition to this Catalog Card format supplied with the system. To do so:

1. Add a new format (with its own format number) to the edit_doc.lng table, such as, for example, format number 012.
2. Add the fields you want to be included in the new paragraph to edit_paragraph.lng table. For example:

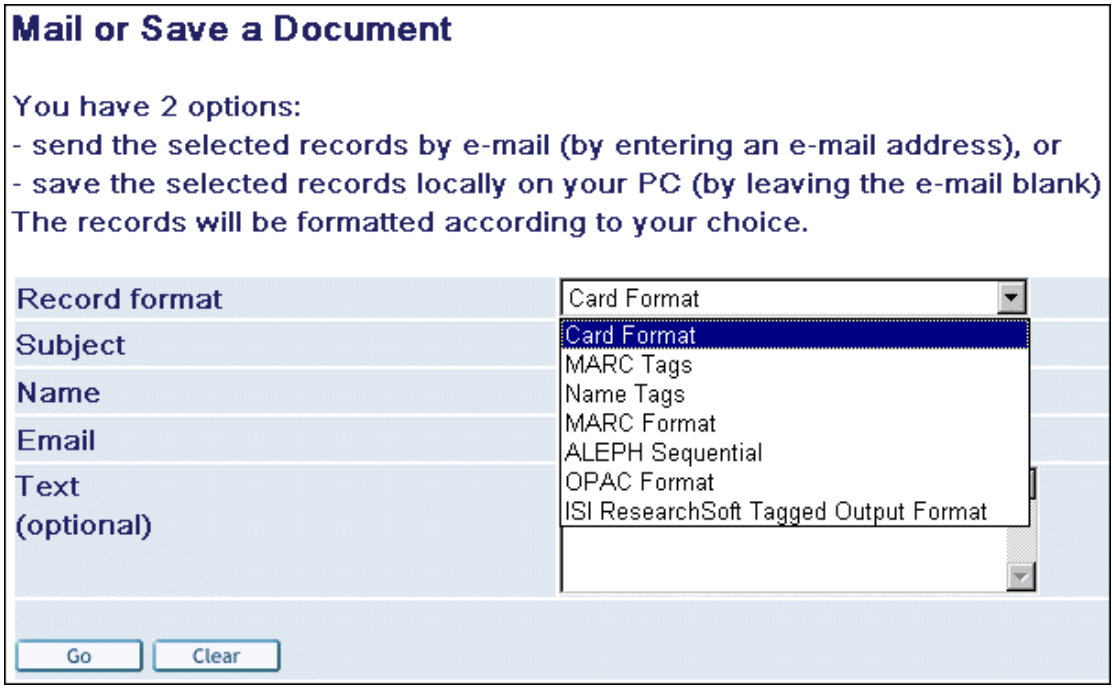
```
012 LOC## D    ##
012 1#### D    ^.^
012 245## 9            .
012 260## W ^          .
012 300## D ^          .
```

3. In the full-set-head HTML file, the new format option might look like this:

```
<a href=&server_f?func=full-set-
set&set_number=$0800&set_entry=$0500&format =012>
```

Mail or Save a Document

This screen allows the user to save documents to disk or to forward them via e-mail. This screen is based on the full-mail HTML file. There are several record format options available for saving and sending records.



The format options reside in the **include-select-format** include file.

Here is an example of a file definition:

```
<select name=format>
          <option value=037 $$FXX-S"037">Card Format</option>
          <option value=001 $$FXX-S"001">MARC Tags</option>
          <option value=002 $$FXX-S"002">Name Tags</option>
          <option value=997 $$FXX-S"997">MARC Format</option>
          <option value=998 $$FXX-S"998">ALEPH Sequential</option>
          <option value=999 $$FXX-S"999">OPAC Format</option>
          <option value=777 $$FXX-S"777">ISI ResearchSoft Tagged
Output Format</
option>
   </select>
```

You can use formats from `edit_doc.lng`, or the 777 format, which is defined in the `edit_doc_777` table. This is the format used to import records into reference managers such as Endnote. The conversion is driven by the `edit_doc_777` table in the tab directory of the bibliographic library. In this table, the ALEPH tags that should be exported are defined. When a record is saved or mailed in the 777 format, the program uses the **www-set-mail-777-00** form. More formats are described above in this Full view section.

4.9 Holdings Screens

The Holdings screen contains three parts:

- Top - BIB record information
- Middle - HOL record information
- Bottom - Item information.

This section explains how you can set up the display of the BIB and HOL information.

The user can click a link on the screen to place an expand, photocopy or hold request. The actual text that is displayed for the link is determined by message numbers 9058, 9059 and 9060 in the `www_f_heading` file.

The Expand link, displays complete information about the item. The actual word that is displayed for the link is determined by message 9058 (for example, 9058 L expand) in the `www_f_heading` file. This "Item Record Expand View" is useful because it enables the System Librarian to display just a few columns of item information in the main Holdings screen, and to save the details for the expanded view. The file that controls the expanded view is `item-global-exp`.

The display of the *Due date* column of the Holdings window, is taken from `$alephe_root/error_lng/circ_status_holding`, error codes 8000 - 8013.

An item that has a hold or a booking request is reported as Requested in the Due Date column. To also report items that only have photocopy requests as Requested:

In $alephe_root/www_f_lng/item-global-body, replace the following line:

<td class=td1>$0400</td>

With:

<td class=td1>$1800</td>

The Heading code 8013 is used when the item has a Z30-ITEM-PROCESS-STATUS. This message appears with the process status description taken from the ADM library's `tab15.lng`. If there is no Z30-ITEM-PROCESS-STATUS, message number 8012 (*On Shelf*) appears.

The display of the BIB record at the head of the list of items is defined by the 012 format, in `xxx01/tab/`**`edit_doc.lng`**.

The display of the HOL record at the head of the list of items is defined by the `xxx60/tab/edit_doc_99.lng` table.

To implement the functionality of the display in the No. of Requests column, the following placeholder ($1600), must appear in `./alephe/www_f_eng/item-global-body` as follows:

```
<!--No Requests-->
<td class=td1>$0900 $1600</td>
```

You can set up the Holdings screen to include links to external locations, such as HTML files or addresses. The links themselves are defined in fields (such as 856) in the BIB and / or the HOL record. The link can be to any machine that is served by a HTTP daemon.

Normally, multiple 856 links can be seen displayed in the Holdings information section of the record, just above Item information.

To include fields from the bibliographic record with electronic contents into the section for the holdings record, for example the 856 field, ensure the following:

- That the `tab/tab_expand` table of the holdings library includes the following line:

  ```
  BUF-Z403    expand_doc_hol_bib
  ```

- That the selected fields (for example, 856) are defined in the `edit_doc_999.lng` (with M in column 10) table of the holdings library as in the following sample:

  ```
  ## 856##                D LElectronic Location Y M     E
  ```

- That the selected fields (for example, 856) are defined in the `tab_buf_z403` table of the holdings library as in the following example:

  ```
  ########## get_buf_z403_bib                856##
  ```

A staff user can see the details of a patron who has borrowed an item by clicking on an underlined due date. In order that the due date link is active, Col.17 in tab31 of the ADM library, must be set to Y for the specific patron's status.

4.10 User Screens

Library patrons can view and carry out their library activities via their online **Library Card**. The My Library Card screen is derived from the `alephe/www_f_lng/bor-info` HTML file.

The format of the bibliographic information is set to 015 in edit_doc.lng configuration file of the BIB and ILL libraries.

You can customize the display on the Library Card in accordance with library policy and setup. For example, your library does not want to allow Advance Booking. To remove the Advance Booking option, open `alephe/www_f_lng/bor-info` and comment out the following lines (that is, by placing **<!--** before the HTML code that controls the information you do not want to be displayed, and by placing **-->** after the HTML code that controls the unwanted information):

```
<tr>
  <td class=td1 align=left>
    <a
href="javascript:replacePage('http://ram19:8995/F/GJ91IA1283R2R62I37H
3NPND8HHTU97IL1A9QIN3N816H5SRJQ-04033?func=bor-item-schedule');">
       Advance bookings</a>
  </td>
  <td class=td1 align=left>
     $2000
  </td>
 </tr>
```

Note that you must **not** change the placeholders for data (placeholders are the values that begin with a dollar sign ($)), although you can comment them out.

All of the other HTML files accessed from `bor-info` (`bor-item-schedule`, `bor-ill`, `bor-photo`, `bor-cash`, and so on) can be customized in the same way. Note that these files have a different structure - they include head, body, and tail files.
Every bor-xxx file contains a link to bor-xxxxx-exp. Some contain a link to bor-xxxxx-exp-head files. These files contain expanded activity information. The bor-xxxxx-exp-head files contain relevant bibliographic information.

4.11 Submitting a Photocopy Form

4.11.1 General

Library patrons can place photocopy requests on items. The photocopy submission window is derived from the `alephe/www_f_lng/item-photo-request` HTML file.

In order to display BIB information on the top of photocopy submission form, insert the following lines in alephe/www_f_lng/item-photo-request-head:

- <input type=hidden name="bib_doc_num" value=$3400>
- <input type=hidden name="bib_library" value=$3500>

4.11.2 Photocopy Submission for Article Document

Libraries can decide to use a pre-filled photocopy submission form when requesting documents that are articles (that is, the document's TYP field = "AN").
When the regular photocopy submission form is used, all fields are displayed empty on the form. A special photocopy submission form can be used for displaying pre-filled information: alephe/www_f_lng/item-photo-request-tail-***an***. This form is optional. In order for this form to be used for article documents, define the following in $alephe_root/www_server.conf:

www_photo_article_prefill Y

According to this setup, when you request documents that are articles (the document's TYP field = AN), the following photocopy form fields are filled with bibliographic information:

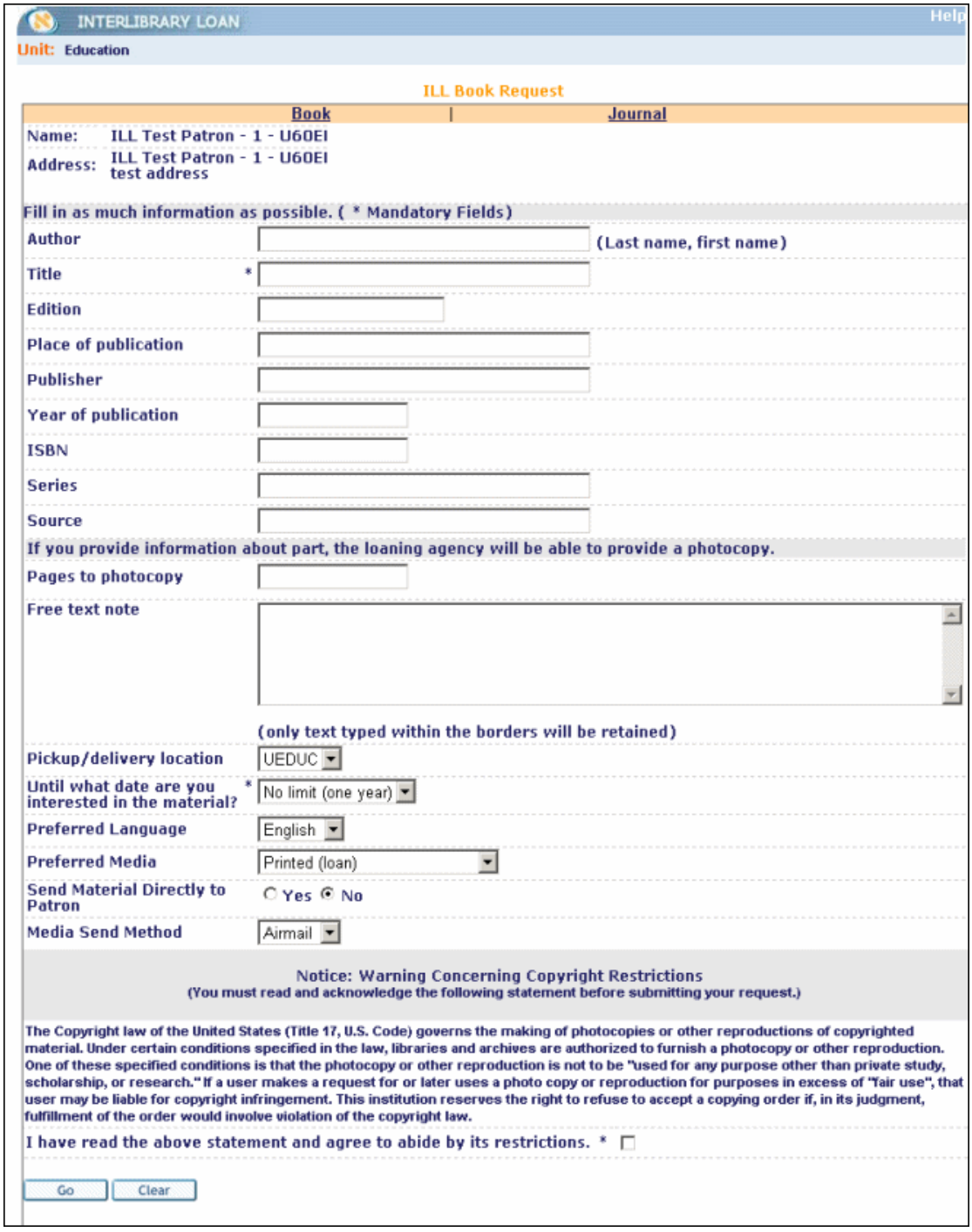
- Author of title (Z38-AUTHOR) –720##, subfield a. (All occurrences of 720, up to the limit of the field size).
- Title of article (Z38-TITLE) – 245##, subfield a + 246, subfield a
- Pages (Z38-PAGES) – 773##, subfield **g** + 300##, subfield a (there is a one byte space delimiter between 773##, subfield g and 300##, subfield a.
- Z38-NOTE-2 will be populated with 001##

For all these fields, fields that do not have $$9 subfields are used. Only for Author of title (Z38-AUTHOR), is a field that has $$9A used.
If the field does not exist in the document, the photocopy form field is left empty.

Note: The TYP field is populated using the `expand_doc_type` routine. For more information, refer to the *Aleph 21 System Librarian's Guide – Indexing*).

4.12 ILL Screens

There are several kinds of ILL screens. Each one is tailored to a specific type of item. Following is part of the screen for a Book/Thesis ILL Request:



The HTML code for the Author field of the ill-book-request file is given below:

```
<tr>
  <td class=td2 width=10% nowrap>Author</td>
<td class=td2> </td>
  <td class=td2>
      <input size=40 maxlength=180 name="BIB___1001_a_1500"
value="$1500">
            <span class="text1">(Last name, first name) </span>
  </td>
 </tr>
```

Pay special attention to "BIB___1001_a_1500".

100 is the tag for author;
1 is the first indicator
a is the subfield.
1500 is the placeholder.

All of the fields follow this same format. Make sure that the tags used in any forms you create are ones that are listed in tab01.lng in the ILL library.

If you want a particular field to be mandatory, make sure that it has an "M" in the "BIB...." code, as in the following example:

```
<tr>
  <td class=td2>Title</td>
  <td class=td2 width=1%>
      <span class=text2>*</span>
  </td>
  <td class=td2>
      <input size=40 maxlength=120 name="BIB_M_2451_a_1700"
value="$1700">
  </td>
 </tr>
```

4.13 Setting up the Special Request Function

The Special Request function allows a user to place Special Requests for items that are in the library's collection, although they are not listed in the computerized catalog. The Special Request form lets the user create a request for these items. The link for making a special request only displays in the menu bar if the patron's profile includes permission for placing a special request. This procedure creates three ALEPH records for an item:

- Item Record
- BIB Record
- Hold Request

To facilitate the Special Request function, the following must be defined in the new-record HTML file:

For an Item Record

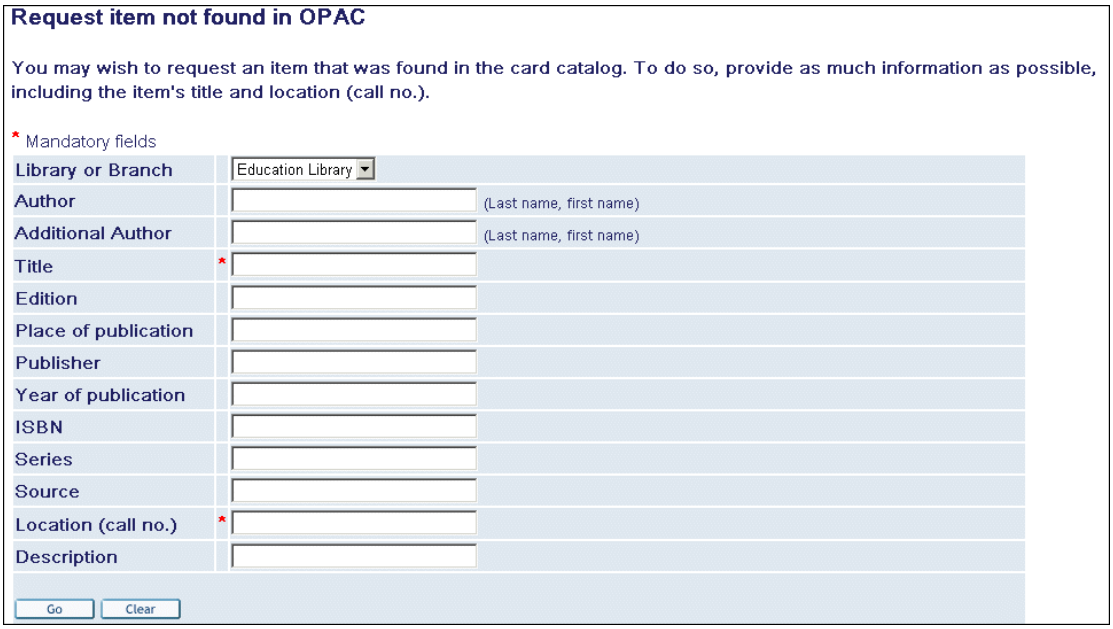
- The sublibrary the item record will be written in.
- The ADM library that the sublibrary is part of.
- The BIB library that the ADM library is related to,
- The item collection (optional), status, material type, and location type (optional).

For a BIB Record

- The BIB database that the ADM database and sublibrary are related to. This is the definition which determines where the BIB records will be written to.
- Field tags for BIB information.

To set up the Special Request function, follow these steps:

1. Define the sublibrary in which the item record will be written. To do so, edit the **new-record** file from the `alephe/www_f_lng` directory which defines the Special Request Form.



The following is the HTML of the Library or Branch field, which must be the uppermost field.

```
<table cellspacing=2 border=0 width="90%">
<tr>
  <td class=td2 width=20% nowrap>Library or Branch</td>
  <td class=td2 width=1%> </td>
  <td class=td2>
     <select name="LIB">
        <option value=USM01-USM50-UEDUC $$3200-S"USM01-
USM50-UEDUC">Education Library</option>
        <option value=USM01-USM50-UHLTH $$3200-S"USM01-
USM50-UHLTH">Health Library</option>
        <option value=USM01-USM50-ULAW $$3200-S"USM01-USM50-
ULAW">Law Library</option>
        <option value=USM01-USM50-ULINC $$3200-S"USM01-
USM50-ULINC">Lincoln Library</option>
      </select>
```

The default data for an item is as follows:

- Name of sublibrary - five characters in "" for example, UEDUC (mandatory)
- Name of collection - up to five characters, for example, AB-D (optional)
- Item Status - two digits, such as 70 (CLOSED STACK) (mandatory, must be in tab15.lng)
- Material type - up to five characters, must be in tab25, for example, BOOK (mandatory)
- Location type, one digit, for example, 8 (optional)

Use hyphens to fill unused positions. Let's look at the following example:

```
<input type=hidden name="UEDUC" value="GEN---70-BOOK--8">
```

The following is an example of a sublibrary definition in the new-record file from the alephe /www_f_lng directory. Each section is separated by a hyphen:

```
<option value=USM01-USM50-UEDUC $$3200-S"USM01-USM50-
UEDUC">Education Library</option>

<option value=USM01-USM50-UHLTH $$3200-S"USM01-USM50-
UHLTH">Health Library</option>
```

If these definitions are correct, a valid item is created and written in the ADM library's Z30 listing of library items.

2. Decide which bibliographic fields appear in the Special Request form and which of these fields are mandatory.

   The following is a example of definitions that appear in the new-record file from the `alephe /www_f_lng` directory.

```
<tr>
  <td class=td2>Author</td>
  <td class=td2> </td>
  <td class=td2>
      <input size=30 maxlength=180 name="BIB___1001_a_1000"
value="$1000">
       <span class="text1">(Last name, first name)  </span>
  </td>
 </tr>
```

   The parts of the following line, which is taken from the example above, are separated by an underline. The first three characters are always BIB. The fifth character can be M, meaning that data entry is mandatory. Characters 7-11 contain field tag + indicators. The last four digits are the same as the placeholder.

   `BIB___1001_a_1000`

   The following is the HTML of the mandatory Title field:

```
<tr>
  <td class=td2>Title</td>
```

```
  <td class=td2>
      <span class=text2>*</span>
  </td>
  <td class=td2>
      <input size=30 maxlength=120 name="BIB_M_2451_a_1200"
value="$1200">
  </td>
 </tr>
```

3. Define an item status for special requests. From the ADM library, use UTIL G/5/15 to edit tab15.lng which defines item statuses and various parameters connected to these statuses. Following is a sample from the table:

```
! 1   2  3  4        5                       6 7 8 9 1 1 1 1 1 1
!                                              0 1 2 3 4 5
!!!!!-!!-!!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!-!-!-!-!-!-!-!-!!
.../...
15A   70 ## L Closed Stack-1                  Y N C N Y N N Y Y 00
15A   71 ## L Closed Stack-2                  Y N C N Y N N Y Y 00
.../...
```

4. In addition to creating the hold request for the user who fills in the Special Request, the system can automatically create a second hold request for a "CATALOGER" user. This feature is available for libraries that want to trap the item when it is returned, in order to send the item to cataloging. The request is generated for the user ID "CATALOGER". Therefore, there must be a user defined whose ID is "CATALOGER" in order that this feature be activated.

   Column 8 refers to Hold requests. To enable Special Requests, you must set this column to C (closed stack). If the item is in a closed stack, hold requests will always be allowed.

4.14 Support For Right-To-Left Text Display In All Formats

ALEPH contains language support for all major written languages as Unicode has been fully implemented throughout the system. This language support includes languages that are written from right to left, such as Arabic or Hebrew. In order to display data that is right-aligned and right-oriented in Arabic or Hebrew characters, follow these steps:

1. Set the environment variable "direction_support" to "Y" (in the file ./alephe/www_server.conf).

2. Add the $AL<NN> syntax, as seen in the following example:

```
<tr>
  <td class=td1
      id=bold
      width=15%
      valign=top
      nowrap>
    $0100
  </td>
  <td class=td1 $AL02>
```

```
    $0200
  </td>
 </tr>
```

Here is an example of a full format display of a bibliographic record containing Latin, Arabic and Hebrew characters:

4.15 Adding an SFX Link Button

The `tab_sfx` table, found in the BIB library's `data_tab`, defines which icon is displayed for the SFX link. The table determines what SFX services are available when a record is found in a specific base.

If one of the services is available, the table also controls which icon will be displayed. This control is done by defining, in column 3, into which index on the HTML page, the icon set in column 2 will be inserted. The index has to be used on the HTML page for it to be displayed. For example, if the `tab_sfx` rows state
:

```
USM01                www_f_sfx           11 N getAuthorEmail
USM01                www_f_sfx_em        12 N getCitedJournal
```

then if the SFX service *getAuthorEmail* is available for a record found in USM01, the `www_f_sfx` icon will be inserted into index 11. For the full-set-head HTML page to display this icon, use the following definition:

```
<td class="bar" align="center" valign="bottom" nowrap>
      $1100
```

```
      $1200
  </td>
```

instead of :

```
<td class="bar" align="center" valign="bottom" nowrap>
      <a href='javascript:open_window("&server_f?func=service-
sfx&doc_number=$0600&line_number=0000&service_type=RECORD");'>
        <img src="&icon_path/f-sfx.gif"
             border=0
             alt="SFX services"></a>
  </td>
```

If none of the services is available, a dimmed and unavailable SFX link button will be inserted into index 11 of the HTML page.

If the searched base is not found in `tab_sfx`, the `www_f_sfx` icon will be inserted into index 11 of the HTML page.

If more than one SFX service is required, columns 1-4 do not have to be repeated. For example:

```
USM01                www_f_sfx            11 N getAuthorEmail

                                               getCitedJournal
```

means that either *getAuthorEmail* or *getCitedJournal* have to be available for the `www_f_sfx` icon to be displayed.

To prevent problems of loading a Record Full View page when the SFX server is busy, make sure that the file sfx-button-data exists under $alephe_root/www_f_lng. and that the "full-set-tail" and "direct-tail" files contain the following lines at the end of the file:

```
<script>
// Function to process SFX buttons
function ProcessSFX(SFX_String) {
    var sfx_elem;
    sfx_elem = document.getElementById("sfx_button");
    sfx_elem.innerHTML = SFX_String
}
</script>
<script src="&server_f?func=load-sfx-
data&doc_number=$0900"></script>
```

You may set up tab_sfx so that the conditional SFX button is linkable if one of the following SFX services is available for the BIB title:

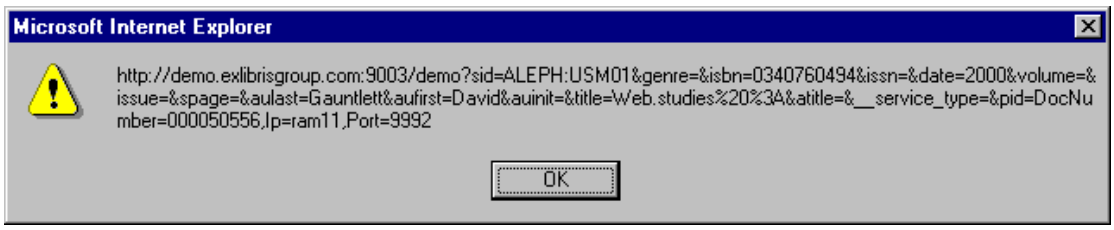
Set the following in $alephe_root/www_server.conf:

setenv sfx_kev Y

The list below is the full set of the services supported only when the environment variable sfx_kev = Y. Otherwise, only the services getFullTxt, getAbstract, getReference, getHolding and getTOC are supported.

- getWebService
- getWebSearch
- getTOC
- getSubject
- getReference
- getPayPerView
- getMessageNoFullTxt
- getHolding
- getFullTxt
- getDOI
- getDocumentDelivery
- getCitedReference
- getCitedRecord
- getCitedJournal
- getCitedGenome
- getCitedBook
- getCitedAuthor
- getBookReview
- getAuthorEmail
- getAuthor
- getAlikeRecord
- getAbstract
- getMsgNoRecomm
- getRecommendation
- getSelectedFullTxt

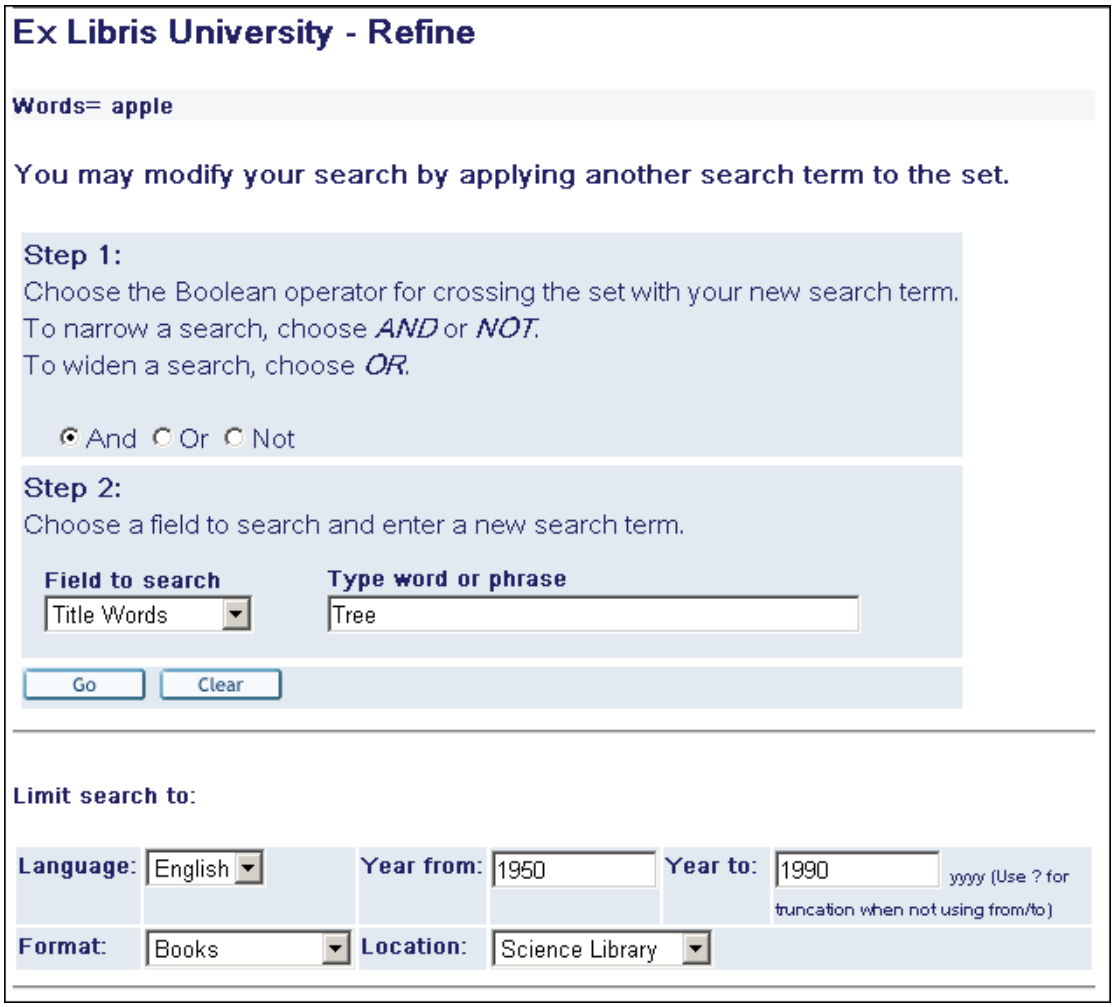
The following message, which pops up before the SFX window appears, presents the OpenURL link:



If you do not want this message to appear, change the following variable in sfx-record (in the www_f_lng directory) from `var debug =0` to `var debug =1`.

4.16 Setting Up the Refine Window

The Refine function allows users to produce a new results set that is narrower than the original one. The Refine window appears when users click **Refine** button from a Results List:



It is divided into the following sections:

- Step 1 - The user can choose one of the Boolean operators AND, OR, NOT.
- Step 2 - The user can select a field to search, such as Author, Title, Subject and so on, and enter a new search term.
- Limit search to - The user can further limit his search by filling in the fields in the Limit search to: section. For search limits to be included in the Refine window, the

following lines must be present in the `./alephe/www_f_eng/short-refine` file:

```
<hr>
<include>find-include-filter
<hr>
```

These filters include filtering by language, year, format, and location. To modify these options, edit the ./alephe/www_f_eng/find-include-filter file.

4.17 Photocopy Information

When viewing patrons photocopy requests details (in my library card), the note-1 and note-2 are concatenated into a single note field. To view the notes separately, add the following in ./alephe/www_f_lng/bor-photo-exp:

```
<tr>
<td class=td1 id=bold align=left>Note 1:</td>
<td class=td1 align=left> $1600</td>
</tr>
<tr>
<td class=td1 id=bold align=left>Note 2:</td>
<td class=td1 align=left> $1700</td>
```

5 Sort

A user can sort a results list by clicking on a link in all of the brief views in the Web OPAC. ALEPH sorts the records in the table according to the sort rules set for the column. This is defined by the predefined sort options set up in the `www_f_sort_heading.lng` table in the library's tab directory. The set remains in the last order selected, and the sort factor is displayed at the top of the screen. If the set contains more records than the sort threshold allows, the set cannot be sorted. The maximum sort limit is 1000.

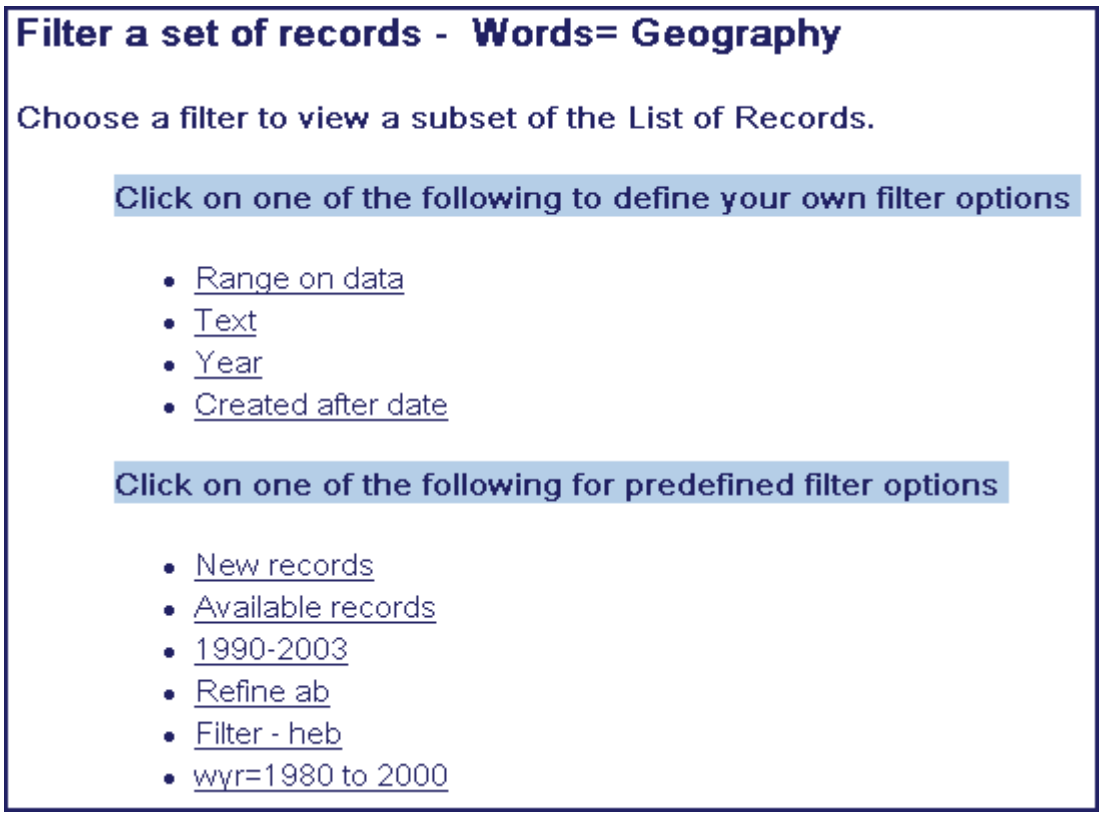
Example of the www_sort_heading.lng table:

```
!1   2 3   4       5                                  6
!!!!!!!!!!!!-!!!!!!!!!!!!
```

```
!!-!!-!!!!!-!!!!!-!-!!!!!-!!!!!-!-!!!!!-!!!!!-!-!!!!!-!!!!!-!-!!!!!-
!!!!!-!-!!-!
!
01 95 008            260## c
08 0
4
02 01 1#### a
00 0
0
03 11 245## a     2 240##       2
00 0
0
04 22 050##          LOC## hjl
00 0
0
05 11 440## av    2
00 0
0
06    TYP##          FMT##
00 0
```

6 Filters

The Filter feature lets users define filter options after carrying out searches, creating their own filters or using predefined filter options.



When the user chooses one of the first four choices on the list, the relevant screen is displayed, as detailed below.

- Range on data - defined in short-filter-r
- Text - defined in short-filter-s
- Year - defined in short-filter-y
- Created after date - defined in short-filter-a

The other choices on the list, which appear below the line, are derived from the various files in `www_f_lng`, and further refine them. Each predefined filter option on the above screen is explained below so that you can understand the general concepts. Editing takes place in the short-filter file.

Range on data
The Range on Data filter displays records containing terms relating to the text in the range specified by the user.

As well as allowing the user to filter by an entire field's content, you can also specify a subfield by setting up the HTML page accordingly (`./alephe/www_f_eng/short-filter-r`) as shown in the following example:

```
<select name=field_code>
         <option value="245__a" $$0500-S"245__a">Title</option>
         <option value="100" $$0500-S"100">Author </option>
         <option value="110" $$0500-S"110">Corporate Author</option>
         <option value="111" $$0500-S"111">Meeting</option>
         <option value="260__a" $$0500-S"260__a">Place of
publication</option>
         <option value="260__b" $$0500-S"260__b">Publisher</option>
      </select>
```

- Use underscore ("_") instead of hash ("#") to specify "any indicator".
- If no indicators are specified (as in "100", "110", and "111" in the example above), this implies - "##" - any indicators. Note that "100__" is equivalent to "100", and may be used for improved clarity).
- If no subfield is specified, the entire field's content is checked.
- Only one subfield code is allowed in each line.

New records
This filter uses short-filter-a as its basis and specifies that records updated within the last seven days should be displayed. (The system checks the CAT field in the Catalog record to determine the date.)

```
<li><a href="&server_f?func=short-filter-a&start_date=&DATE-D007">
New records</a>
```

If you wanted to specify records for the past year, you would change "DATE-D007" to "DATE-Y001". In these examples, "D" stands for day and "Y" stands for year.

Available Records
This filter displays records containing items located on the shelf.

```
<li><a href="&server_f?func=short-filter-3">
Available records</a>
```

This option uses the short-filter file as its basis.

1980-2000
This filter displays records that were published between 1980 and 2000.

```
<li><a href="&server_f?func=short-filter-
y&from_date=1980&to_date=2000">
1980-2000</a>
```

This option uses the short-filter-y file as its basis, and further specifies that the "from" date should be 1980 and the "to" date should be 2000.

Refine ab
This filter displays all records of the current set that have words beginning with the letters *ab*.

```
<li><a href="&server_f?func=find-
c&CCL_TERM=wrd=ab?%20and%20SET=$0100">
Refine ab</a>
```

This option uses the find-c files as a basis and specifies that the system should search the "wrd" index for words that start with *ab* (The codes for the indexes are defined in `tab00.lng`).

By including **%20and%20SET=$0100"** , the Refine ab option specifies that the current set of records should be used as the basis for the filter.

wyr=1980 to 2000
The following filter displays all records of the current set that have the years 1980 to 2000 in the "wyr" (words in year field) index. (The codes for the indexes are defined in `tab00.lng`). The code for word index years used by the filter is hard coded, therefore it must be defined as WYR.

```
<li><a href="&server_f?func=find-c&CCL_TERM=wyr=1980-
>2000%20and%20SET=$0100">
wyr=1980 to 2000</a>
```

By including **%20and%20SET=$0100"** , this option specifies that the current set of records should be used as the basis for the filter.

Filter - eng

The following filter will display all records of the current set that have "eng" (for English) in the "wln" (words in language field) index.

```
<li><a href="&server_f?func=find-
c&CCL_TERM=SET=$0100%20and%20wln=eng">
Filter - eng</a>
```

6.1 Setting Up Bases and Filters

Instead of retrieving all the records in a particular physical database (for example), you may want to limit/filter the records retrieved to those in a particular location, in a particular format, in a particular language, and so on.

Such limiting can be either a pre-filter or a post-filter. With a pre-filter, the user selects a particular database before starting the search; with a post-filter, the user takes the results of the search then selects a location, format, and so on, by which to limit them.

You set up post-filters by using `alephe/www_f_lng/short-filter`. (For English, `www_f_eng/short-filter`). All the examples below can be set up in `short-filter`.

Location as Post-Filter
For example, you might have locations as post-filters (with this definition in `alephe/www_f_lng/short-filter`):

```
<tr bgcolor="#eeeeee">
            <td colspan=2>
<div class="text3">
Limit by location
</div>
<p>
<form>
<div class="text4">
Stanhope University Libraries
<br>
<select name="variable name">
<option>
<option value= "&server_opac/find-c?CCL_TERM=wcl=birk^and^SET=$0100"
target="err
or">Birks Religious Studies Reading Room
<option value= "&server_opac/find-c?CCL_TERM=wcl=bldr^and^SET=$0100"
target="err
or">Blackader-Lauterman Architecture and Art
Â…Â…Â…<and so on.>
```

The index code you use depends on the indexes you define in the tab11_word table.

Formats as Post-Filters

Alternately, you can have formats as post-filters (with this type of definition in `alephe/www_f_lng/short-filter`):

```
.../...
Material Type
<br>
<select name="variable name">
<option>
<option value="&server_opac/find-
c?CCL_TERM=SET=$0100^and^wtp=Archive" target="e
rror"> Archive
<option value="&server_opac/find-
c?CCL_TERM=SET=$0100^and^wtp=Graphic" target="e
rror"> Graphic
<option value="&server_opac/find-c?CCL_TERM=SET=$0100^and^wtp=Kit"
target="error
"> Kit
<option> - - - - - - - - - - - - - - -
<option value="&server_opac/find-c?CCL_TERM=SET=$0100^and^wtp=Map"
target="error
"> Map
Â…<and so on.>
```

The index code (WFT vs. WTP) you use depends on the indexes you define in the tab11_word table.

Language as Post-Filter

You can also set languages as post-filters (with this definition in `alephe/www_f_lng/short-filter`):

```
<form>
<div class="text4">
Language
<br>
<select name="variable name">
<option>
<option value="&server_opac/find-c?CCL_TERM=SET=$0100^and^wln=ara"
target="error
"> Arabic
<option value="&server_opac/find-c?CCL_TERM=SET=$0100^and^wln=chi"
target="error
"> Chinese
<option value="&server_opac/find-c?CCL_TERM=SET=$0100^and^wln=cze"
target="error
"> Czech
```

Other Post-Filters

The `alephe/www_f_lng/short_filter` comes with some built-in filters:

```
<form>
<div class="text4">
Define your own limit options
<br>
<select name="variable name">
<option>
<option value="&server_opac/short-filter-r">Range on data
<option value="&server_opac/short-filter-s">Text
<option value="&server_opac/short-filter-y">Year
<option value="&server_opac/short-filter-a">Recent acquisitions
<option>
```

Pre-Filters (Databases)

With a pre-filter, the user selects a particular database before starting the search.

The `alephe/tab/tab_base.lng` table is used to specify pre-filtering. Any keyword/Boolean search which can be performed through the Find/Search function can be the basis of a database. The filtering criteria are specified in column 9 of the `tab_base.lng` entry. Additional information on `tab_base.lng` can be found in the ALEPH UTIL Menus - 16.02 document, section UTIL Y/3.

The `tab_base.lng` entry for a physical database looks like this:

```
!         1          2           3            4     5     6     7   8
9
!!!!!!!!!!!!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-
!-!!!!!!!!!>
USM01                  Ex Libris University                    USM01
N
```

Columns 4, 5 and 6 are used for file extensions. Column 8 defines whether or not to use Z0102 to create a logical base.

Excluding Suppressed, Deleted Records

Almost all sites will want to have the database(s) which the public uses exclude suppressed records, logically-deleted records, and possibly substandard, "fast-cat" records. This is accomplished with a specification like this:

```
!         1          2           3            4     5     6     7   8     9
!!!!!!!!!!!!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!-
!!!!!!!!!>
USM01_PAC              USM01 PAC                                USM01 N
wrd=alldocuments not (wst=suppressed or wst=deleted or wst=circ-created)
```

wst is a word code which is specified in `xxx01/tab/tab11_word` as a word index on the STA field:

```
STA##                                          01     WRD  WST
```

(The STA field looks like this: "`STA $$aSUPPRESSED`"; "`STA $$aDELETED;`" and so on, and is found in the bibliographic record.).

Note

Bases apply only to Words and Browse searches. Direct index (number) searches are always done on the entire bibliographic file. You may want to remove or change the tag of OCLC numbers or other overlay-matching numbers if records are being logically deleted.

Locations as Pre-Filters

Here is a sample definition for `tab_base.lng` entries for databases that are filtered by sublibrary:

```
!         1          2           3            4     5     6     7   8     9
!!!!!!!!!!!!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!-
!!!!!!!!!>
LAW                    Law Library                              USM01 Y
wsl=ulaw
```

What you specify for the code depends on which indexes you have defined in the tab11_word table. If you have defined a single WCL with both the sublibrary and collection then you can use that:

```
LOC##                      bc                    01     WCL
```

If you have defined separate sublibrary and collection indexes:

```
LOC##                      bc                    01     WSC
LOC##                      b                     01     WSL
LOC##                      c                     01     WCL
```

then you can use the WSC and/or WSL.

Note

If none of your collection codes are used as sublibrary codes, then you only need to define the single WSC (or WCL) index with LOC (or PST) subfields b and c.

You might think of specifying "EXUCAT" in column 7, that is, having LAW be a database built on EXUCAT rather than EXU01, thus making it unnecessary to repeat the EXUCAT filtering criteria. You cannot do this. The value in column 7 must be a physical library; it cannot be just another base.

Formats for Pre-Filters

The `tab_base.lng` entries for databases which are filtered by format might look like this (using the wfm index, built on the FMT field):

```
!         1          2           3            4     5     6     7   8
9
!!!!!!!!!!!!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-
!-!!!!!!!!!>
SERIALS                Serial (USM01)                          USM01 N
wfm=SE
MONOGRAPHS             Monographs (USM01)                      USM01 Y
wfm=BK
```

Or like this (using the wtp index, built on the TYP field):

```
!         1          2           3            4     5     6     7   8
9
!!!!!!!!!!!!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-
!-!!!!!!!!!>
 TPE                    Periodicals          USM90             USM90
Y wtp=se
```

Note

The FMT field is a physical field which is stored in the BIB record. The TYP field is a virtual field generated by `expand_doc_type`. It uses the configuration setup in a `tab_type_config.lng` table, for example using LDR pos. 06,07; 006 field pos. 00; 007 field pos. 00, 01; and 008 pos. 23 to arrive at more detailed formats than those specified in the FMT field. To index it, you need to have a:

```
WORD       expand_doc_type         tab_type_config
```

entry in the `xxx01/tab/tab_expand` table.

7 Error Messages

You can edit and translate the error messages that are displayed to users by editing the `www_f_heading` file in the `$alephe_root/error_lng` directory. Following is an example from that file:

```
9031 L There is a copy available; request not accepted.
9032 L Patron is missing an ILL library. Unable to process request.
9033 L Error in patron ILL library. Unable to process request.
9035 L Too many hits. Refine request.
```

```
9036 L Error parsing find request.
9037 L Set is too large for adjacency check. Refine request or skip
adjacency check
```

Key to table:

- **Column 1** is the message number. Do not change this number.
- **Column 2** was the code for the character set of the column heading and is now irrelevant.
- **Column 3** is the text of the message that is displayed to the user.

Note

When writing error messages with quotation marks ("), use only the HTML definition for quotation marks - *"* - to ensure that they display correctly.

8 Default Text for Links

The cataloger uses the LKR field to create links from one bibliographic record to another. The following table shows the possible types of LKR links for BIB records.

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 (that is, the Z103-LKR-TEXT-N field, which is the test of the up link, 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                              3
!!!!!-----!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
NONE       Click to view related record
```

NONE is reserved for both the Search functionality in GUIs and the Web OPAC.

The LKR fields can be displayed according to various sorts.

9 Rank

Web OPAC users can rank sets, using the words that appear in the Rank Set window. Web OPAC ranks the sets, using the words of the search query.

There are three factors involved in setting the ranking weight of a word. The weight is calculated by:

- the number of occurrences of the word in the record (a document gets a higher ranking if a significant word occurs several times).
- the weighting factor defined in column 9 of `tab00.lng` (adds weight according to the particular word list).
- an initial factor attributed to each of the words used for ranking (in case more than one word is used), based on the frequency of the word in the whole database. The more frequent the word is (the more results its search obtains), the lower its initial weight.

To define a weighting factor, set the last two digits in the W section of `tab00.lng` (Col.9) as appropriate. Other sections (ACC and IND) should have 00 in column 9. Here is an example of a definition:

```
!   2     3   4 5  6  7 8   9   10            11
!-!!!!!-!!!!!-!-!!-!!-!-!---!!-!!!!!-!!!!!!!!!!!!!!!!!!!
H WRD   W-001      00       01       Words
H WTI   W-002      00       02       W-titles
H WAU   W-003      00       02       W-authors
H WSU   W-004      00       03       W-subjects
H WUT   W-005      00       02       W-Unif .Titles
H WSM   W-006      00       03       W-MeSH Subj.
H WNO   W-007      00       01       W-Notes
H WNC   W-008      00       02       W-Table of contents
H WPU   W-009      00       02       W-publishers
H WYR   W-010      00       02       W-year
```

Note that the system only uses the WRD index.

10 Adding a New Index

To add a new index, follow these steps:

Step 1: Define the Index Type
Decide which type of index you want. There are three types: Direct, Words (Find), or Headings (Browse).

Step 2: Add New Index to the Indexing Tables

The indexing tables are:

- tab11_acc (for headings indexes)
- tab11_ind (for direct indexes)
- tab11_word (for words indexes)

Include in the relevant indexing table the code you specified in tab00.lng for each field which you want to have sent to your new index.

If you specified "IND" in tab00.lng, put the entry in the tab11_ind table for direct indexes. If you specified "ACC", put the entry in the tab11_acc table for headings indexes, and if you specified "W-nnn", put it in the tab11_word table for word indexes. Consult the ALEPH Configuration Guide - 16.02 document for other parameters.

The following is a sample of the tab11_ind table:

```
! 1    2    3     4        5            6           7 8
!!!!!-!!!!!-!-!!!!!!!!!!-!!!!!-!!!!!!!!!!!!!!!!!!!!-!-!
010##                    010   -bz
010##                    STIDN -bz
020                      ISBN  az
```

The following is a sample of the tab11_acc table:

```
! 1    2    3     4        5            6           7 8
!!!!!-!!!!!-!-!!!!!!!!!!-!!!!!-!!!!!!!!!!!!!!!!!!!!-!-!
100##                    AUT   -e468
245##                    TIT   -ch68                  2
```

The following is a sample of the tab11_word table:

```
! 1     2   3      4            5             6  7 8  9   10   11
12
!!!!!-!!!!!-!-!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-!!-!-!-!!!--!!!--!!!--
!!!->
100##                    -6                    01     WRD  WAU
245##                    c                     03     WAU
245##                    -c6                   03     WTI
```

The index codes can be up to 5 characters (in tab00.lng, tab11_acc and tab11_ind).

Step 3: Add New Index to the tab00.lng Table
Go to xxx01/tab/tab00.lng and define a code for the new index. Make sure the code is not in use already.

Note

Although the code can be up to five characters, by convention, three characters are used. For a Direct index, specify "IND" for the index type. For a Headings index,

specify "ACC" for the index type. For a Words index, find the next "W-nnn" value and use that for the index type. Consult the Database Management Guide for other parameters.

The system automatically creates the SYS index for system numbers and the BAR index for barcodes. Therefore, although they do not need to be defined in the indexing table (`tab11_ind`, `tab11_acc`, `tab11_word`), they must be defined in `tab00.lng` in order to define the index name in column 11.

Example:

```
!   2     3   4 5  6  7 8   9   10            11
!-!!!!!-!!!!!-!-!!-!!-!-!---!!-!!!!!-!!!!!!!!!!!!!!!!!!!
H SYS   IND     21 00       00       System Number
H BAR   IND     21 00       00       Barcode
```

Step 4: Run the Index Job

Open the Cataloging GUI and click Services to access the **Build Indexes to the Catalog** submenu options.

- For a Direct index, run p_manage_05.
- For a Headings index, run p_manage_02,
- For a Words index, run p_manage_01.

Note

On large databases, these indexes take a number of hours to run - the larger the database, the longer it takes to run the index. This job locks the library.

Step 5: Adding New Indexes to the Browse and Search Screens

- A Direct index can be added to either the Browse or the Search screen. Add it to the Web Include files
- If you want an index to be searchable on the Web, it has to be added to an HTML file. For a Headings index, add it to `www_f_lng/scan-include-2-xxx01`.

Example:

```
!   2     3   4 5  6  7 8   9   10            11
!-!!!!!-!!!!!-!-!!-!!-!-!---!!-!!!!!-!!!!!!!!!!!!!!!!!!!
H SYS   IND     21 00       00       System Number
H BAR   IND     21 00       00       Barcode
```

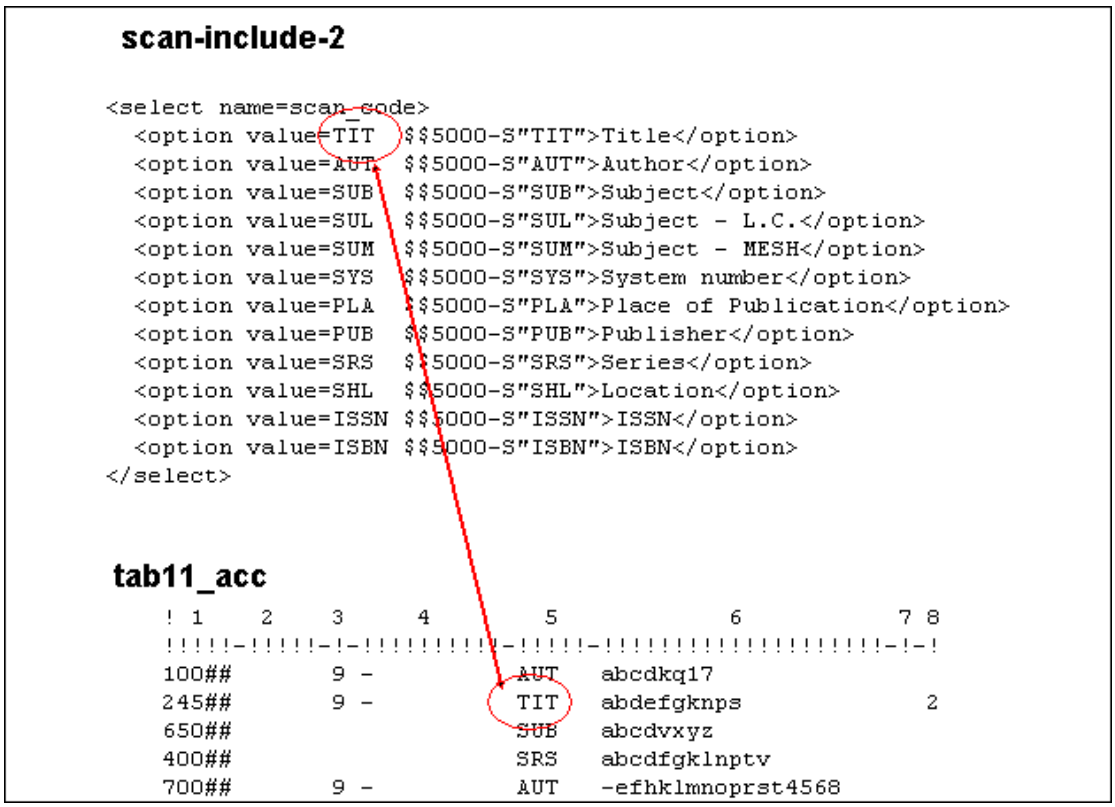
For a Words index (search), add it to `www_f_lng/find-code-include`.

Example:

```
<select name=find_code>
     <option value=WRD $$FXX-S"WRD">All Fields</option>
     <option value=WTI $$FXX-S"WTI">Title</option>
```

```
 <option value=WAU $$FXX-S"WAU">Author</option>
 <option value=WSU $$FXX-S"WSU">Subject</option>
 <option value=WSU $$FXX-S"WSM">Medical Subjects</option>
</select>
```

The following figure shows the relationship between the tables and the HTML code:



11 Setting Up the SDI Function

The SDI interface allows patrons with SDI permission to define SDI (Selective Dissemination of Information) profiles for themselves, based on their areas of interest. Patrons can access and modify their SDI profiles from the patron information (My Library Card) screen by clicking on the SDI Profile link from the menu bar. SDI profiles are stored in a special Oracle table, Z325, which is located in the USR00 library.

An SDI request defines a search query, a sublibrary to search (a patron can also set his SDI request regardless of location), a list of bases to search, a frequency with which the SDI query will be run, expiry date for the SDI request, the suspend period, and the character set. When the system runs the query (p_sdi_01 service), the records that have been added or updated in the database since the last time the query was run are retrieved. The resulting set of records is mailed to the patron. The format of the records is defined in the patron's SDI profile. If a patron clicks **Online Search** from his SDI profile window, then the Z325-last-action-date is updated and no SDI letter will be sent to the patron regarding items found in the online search.

SDI functionality in ALEPH is controlled by the setup of the `p_sdi_01` service. By using the job scheduling available in ALEPH (`job_list`), you can set this service to

run at regular intervals. The only parameter the process requires is the library in which the Z325 is stored. For example:

```
W1 23:30 Y USR00 p_sdi_01 USR00
```

If the ADM library defines USER-SHARING=Y (in tab100), then p_sdi_01 has to be run in the usr_library.

If the ADM library does NOT define "User Sharing" or defines USER-SHARING=N (in tab100), p_sdi_01 cannot be run in the usr_library. It has to be run in the specific ADM library.

For example:

```
W1 23:30 Y USR00 p_sdi_01 USM50
```

The format for the e-mail message sent to patrons is stored in `/xxx01/form_lng/bor-sdi-00`.

Patrons can also create a new SDI request via the current search results list and previous searches.

When a new record is added to the database, the system checks if an SDI-ready record (Z324) should be created for it. Part of this mechanism is sending a Z105 message to the USM01 library where Z324 records are stored. The following setup should be defined in `/usm50/tab/tab_z105`:

```
!      1        2   3     4     5     6     7     8     9    10    11    12
!!!!!!!!!!!!!!!-!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!
UPDATE-SDI      8 USM01
```

12 Logical Bases

A logical base is a collection of records which share a common denominator, such as a particular library or sublibrary, material type, works published before 1800, and so on. Logical bases are organized in a manner that facilitates regular library activities, such as the retrieval of information. A logical base consists of a subset of the records available in an entire database (for example, USM01) but it is presented to the user as though it were a separate database. In effect, logical base act as pre-filters for a Find query. The Browse (headings) lists are also filtered, so that a list includes only headings that are relevant to the logical base.

Logical bases are defined in the `tab_base.lng` table in the `alephe/tab` directory. The following is an extract from `tab_base.lng`:

```
!         1          2           3            4     5     6     7   8
9
!!!!!!!!!!!!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-
!-!!!!!!!!!>
USM01                   Demo Library         USM01 USM01       USM01
EDUC                    Education Library    USM01             USM01 N
wsl=ueduc
SERIALS                 Serials              USM01             USM01 Y
wfm=SE
```

```
MUSIC                   Music                   USM01               USM01 Y
wfm=MU
```

Column 8 defines whether or not a logical base should be created.

12.1 Logical Base Counter

The Z0102 table (logical base counter) is an option which can be used to make browsing from the Web OPAC more efficient when scanning logical bases which are less than 10% of the total database.

When a logical base is being browsed, the system uses the Z0102 table to determine whether or not to display the heading (Z01) without having to retrieve the documents attached to the heading and read them.

The Z0102 table is built for each heading and for each logical base. The table includes the filing text, the access sequence of the heading and a counter of the relevant documents attached to the heading. The pointer to the documents is stored in the Z02 (ACCDOC) table.

Conversion is not applicable; the Z0102 records are generated by running the Build Counters for Logical Bases (manage-32) service from the Services menu of the Cataloging GUI. You should always run this service after building the Headings index (manage-02).

When a heading is updated or when a new heading is created, the logical base counter is automatically updated by the UE_01 (specifically by the ue_01_z0102_index sub-process) daemon process. When a heading (Z01) is updated or created, the UE_01 process updates the Z0102 table.

It is recommended to set large bases to work with the regular Z01 mechanism instead of the Z0102 mechanism. This saves space and it increases the performance of the building process. The bases that work with the Z0102 must be set to Y(es) in column 8 of `/alephe/tab/tab_base.lng`.

You can view and analyze the setup of the bases by running UTIL H/1/10. Following is an example of this type of report which includes recommended and current setup of the Z0102 column (column 8) of the `tab_base.lng` table and a column indicating that there is a conflict between them:

```
Base            | # Docs       | Recommended | Current | Conflicts
--------------------+---------+-------------+----------+-------
BOOKS           |     2700     | No          | No       |
FAIRYTALES      |NEGATIVE BASE | No          | Yes      | *** Err ***
HISTORY         |      166     | Yes         | Yes      |
INTROD          |       43     | Yes         | Yes      |
HEALTH          |        3     | No          | Yes      | *** Err ***
NOTCLASSIFIED   |NEGATIVE BASE | No          | No       |
SERIALS         |      306     | Yes         | Yes      |
USM01           | FULL BASE    | No          | No       |
```

If a base has the `wrd=alldocuments not...` command in Column 9, or if the base has no documents, then the caption in the # Docs. column in the report will be: NEGATIVE BASE.

If a base has no command in Column 9, then the caption in the *# Docs.* column in the report will be: `FULL BASE`.

Here is an example. `tab_base.lng` has the following lines:

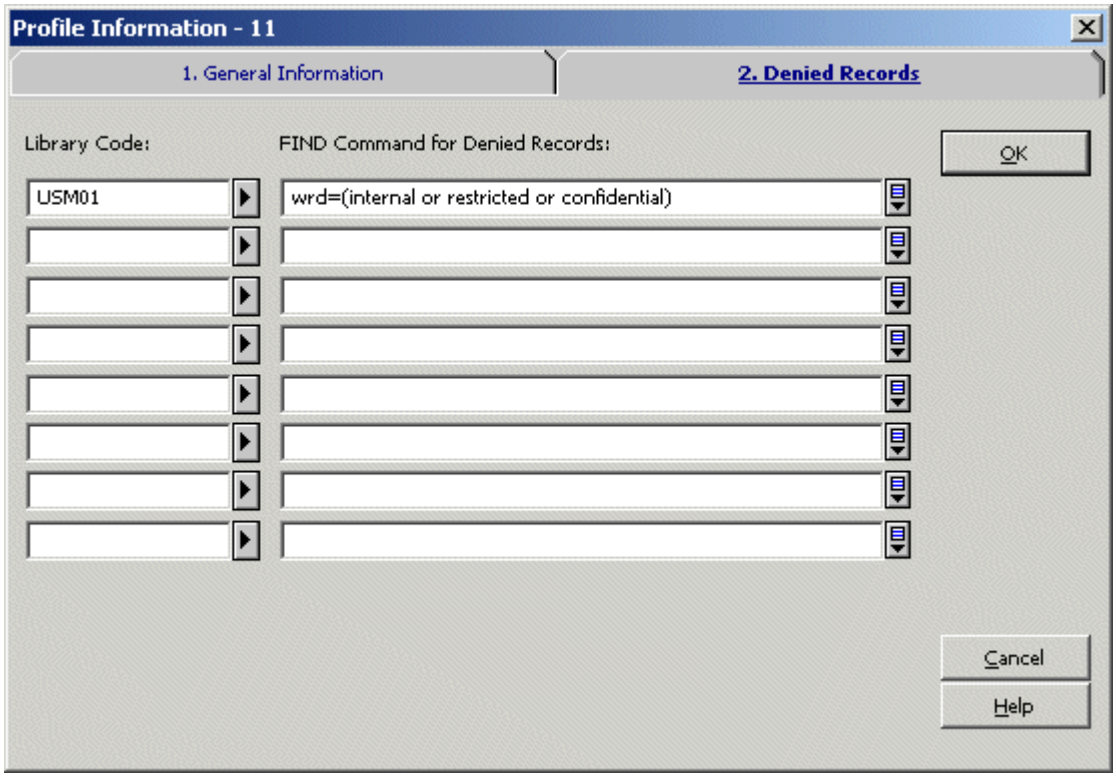
```
!         1          2           3            4     5     6     7   8
9
!!!!!!!!!!!!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-
!-!!!!!!!!!>
NOT_ART                Not Art              USM01             USM01 Y
wrd=alldoc
uments not wrd=art
USM01                  Ex Libris University                   USM01 Y
```

The report will present the bases-related information as follows:

```
Base            | # Docs       | Recommended | Current | Conflicts
----------------+--------------+-------------+---------+----------
NOT_ART         |NEGATIVE BASE | No          | No
USM01           |FULL BASE     | No          | No
```

12.2 Denied Libraries

You can define the logical base (`tab_base.lng`) accessed by guest users. To prevent a guest user from entering a particular base, you can limit access by putting an asterisk (*) in col. 2 in `tab_base.lng` in the line where that base is defined. In addition, you can prevent a user from seeing records (in a logical base) by setting Denied Records in the Access rights tab of the Profile information window in the Circulation GUI. You do this by entering queries in the **FIND command for denied records** fields.



(See the Patron chapter in Circulation, Patron Profiles).

13 Patron ADM Library (Single/Multiple)

When patrons sign in to a Web OPAC that has multiple ADM libraries, the default library that is displayed when the patron clicks **My Library Card** is the ADM library that was chosen during sign-in. A patron can also choose *Shared Pool* from the library branch drop-down menu. In this case, the user's HOME-LIBRARY (primary sublibrary from his Global Patron Information – Z303 record) is taken to indicate to which ADM library the user belongs. If the user does not have a HOME-LIBRARY, the ADM library is determined by the first local patron information record (Z305). If no Z305 record exists for the user then TAB100-DEFAULT-BOR-LIBRARY in the BIB library is used. Here is an example:

```
DEFAULT-BOR-LIBRARY=USM50
```

In this case, the ADM environment of the user will be USM50.

If none of the abovementioned options supplies an ADM library, then error 9849 will be displayed, and the log-in action will fail.

On the *My Library Card* screen, the ADM libraries in which the patron is registered display, and he can click the link to switch to another library.

Information regarding loans and holds is managed and displayed separately for each ADM library. Information about the patron such as proxies, SDI profile, patrons. address and more, is managed by the USR00 library.

When the user places a request on an item in the OPAC, the system automatically checks the ADM library of the item, and chooses the most relevant Z305 record.

14 Synonyms Dictionary for Word Searching (Optional)

ALEPH includes a feature whereby groups of words can be set as synonyms (words having the same or nearly the same meaning as another in the language). In this case, a search for any one of the words automatically includes a retrieval of all the words that have been defined as synonyms in the group. For example, if karel, karl, carol, charles are set as synonyms, a FIND on any one of these words retrieves the records which contain any of the other words.

Synonym groups are created and maintained through the UTIL B menu of the bibliographic library, or through the Web Utilities. Web utilities are available from the Web Staff Menu; click Utilities, and choose the bibliographic library.

The following options are available for synonyms management:

- Add a group of synonyms
- Remove word(s) from a group of synonyms
- Unlink an entire group of synonyms
- View synonym groups

15 Setting up Save/Mail Encoding Options

You can set up the Save/Mail form invoked when Save/Mail is clicked to allow users to select different encoding options for records sent by e-mail.

To enable this functionality:

2. Remove the line CHARACTER_CONVERSION=NONE from all Web OPAC-related forms in the usm01/form_lng directory.

3. Set up character conversion for printouts for both Data and Form in mail-related HTML pages.

   Example (full-mail):

```
<td class=td2 id=small >
  <input type="radio"
         name="encoding"
         value="UTF_TO_WEB_MAIL_ASCI"
         $$1100-C"UTF_TO_WEB_MAIL_ASCI">ASCII (recommended /
default)
</td>
<td class=td2 id=small >
  <input type="radio"
         name="encoding"
         value="NONE"
         $$1100-C"NONE">Unicode / UTF-8 (non-Roman character
sets)
</td>
<td class=td2 id=small >
  <input type="radio"
         name="encoding"
         value="UTF_TO_WEB_MAIL"
         $$1100-C"UTF_TO_WEB_MAIL     ">ISO 8859-1 (Roman
character sets)
</td>
```

   You can define as many options as you like as long as they are also defined in the `$alephe_unicode/tab_character_conversion_line` configuration table.

   You can modify the encoding of the subject line of the e-mail message sent by the user by editing the UTF_TO_WEB_SUBJECT section in the `$alephe_unicode/tab_character_conversion_line` configuration table.

   Example:

```
UTF_TO_WEB_SUBJECT   ##### # line_utf2line_sb
unicode_to_8859_1
```

The above setting will save or send records by e-mail in the appropriate encoding but most mail programs will not show it without manual selection of the correct encoding by the recipient.

In order to avoid this, your ALEPH installation includes an example script, $alephe_tab/send_mail.example, that shows how this can also be solved.

The following section controls the encoding passed from the Web page:

```
switch ($4)
    case "NONE":
        echo "Content-Type: text/plain; charset=UTF-8" >>
$TMPDIR/send_mail.$$
        breaksw
    case "UTF_TO_WEB_MAIL_ASCI":
        echo "Content-Type: text/plain; charset=US-ASCII" >>
$TMPDIR/send_mail.$$
        breaksw
    case "UTF_TO_WEB_MAIL":
        echo "Content-Type: text/plain; charset=ISO-8859-1" >>
$TMPDIR/send_mail.$$
        breaksw
    case default:
        echo "Content-Type: text/plain; charset=UTF-8" >>
$TMPDIR/send_mail.$$
        breaksw
endsw
```

You can use this example script to create an appropriate *charset=* section inside the SMTP header. The same applies to the subject line though it can only hold one pre-defined value for charset.

If you are using `send_mail.example` as your mailing program, rename it as `$alephe_tab/send_mail` and make sure it has execution permission as follows:

```
chmod +x $alephe_tab/send_mail
```

Note

Any change in the `tab_character_conversion_line` conversion table relating to the WEB sections must also be reflected in this script.

16 Save/Mail on Multi-Apache Topology

When using a multi-Apache topology, clicking **save/mail** displays a File Download dialog box that allows the user to save the file.

To support the save/mail feature in such topology, the following setup is required:

1. Set the preview_required attribute to `Y` in all forms that use the Save/Mail option by adding the following line:

   ```
   <input type=hidden name=preview_required value="Y">
   ```

   After the line:

   ```
   <input type=hidden name=func value=myshelf-mail>
   ```

   In the following files:

   ```
   ./alephe/www_f_lng/myshelf-mail
   ./alephe/www_f_lng/myshelf-course
   ./alephe/www_f_lng/user-info-mail
   ./alephe/www_f_lng/short-mail
   ./alephe/www_f_lng/full-mail
   ```

2. The number of docs that can be downloaded is limited at user profile level by the variable Z61-MAX-SAVE-MAIL-FILES. Define for all profiles (Z61) an appropriate maximum limit of the number of save/mail files for all profiles (the default is 999999). It is recommended not to exceed the value of 20,000 for this limit, thus enabling a reasonable response time to the end user.

For more information, refer to the *Aleph Topologies* document.

Note:
With this topology, the save/mail functionality is not based on the search result, so an additional search is performed. Sort is not applied.

17 Local Own Check Procedure

When an ACQ request form is submitted, the system activates the “locate similar record in local database” function for blank ACQ requests and for titles found in an external base. This function is inactive for titles found in a local base. The base that is used for this locate action is the BIB library where the ACQ request's BIB document is about to be created.

The base used for this locate action is defined in the "www_base" environment variable of aleph_start.
The locate query is taken from ./xxx01/tab/tab_locate.

The ACQ request BIB document is created in the BIB library to which the patron’s ADM library is attached.

If an ACQ request is created based on a BIB record that is not related to the ADM library of the patron's home library, a new BIB record is created. It is possible to set a value in the OWN field of the new BIB record that indicates its connection to the ADM library. To do this, populate the ACQ-REQUEST-OWN variable in tab100 of the ADM library with a string of up to 10 characters. For example

./[ADM_library]/tab/tab100

ACQ-REQUEST-OWN=USM50

18 ACQ Request for Library with more than One BIB Library

This chapter is relevant only for library which its ADM library is set to work with more than one BIB library.

For example:

./alephe/tab/library_relation is set as in the following example:

BIB USM50 XXX01 YYY01

Such setup; relates the ADM library (USM50) to two BIB libraries (XXX01 and YYY01).
When an ACQ request is created via the Web OPAC, there is an option to configure the OPAC ACQ Confirmation page so that the patron can select the BIB library in which the BIB record will be created.
The locate procedure, which is activated during submitting ACQ Request, will be performed at the selected BIB library.
To support this, update the following OPAC HTML pages:

./alephe/www_f_eng/acq-request
./alephe/www_f_eng/acq-request-doc-tail
./alephe/www_f_eng/acq-request-confirm
./alephe/www_f_eng/acq-request-doc-tail-conf

For two BIB libraries, XXX01 and YYY01, update the above HTML pages as in the following example:

```
<tr>
  <td class=td2>BIB Library<br></td>
  <td class=td2> </td>
  <td class=td2>
    <select name="bib_lib">
      <option value="XXX01" $$3700-S"XXX01">BIB library XXX01</option>
      <option value="YYY01" $$3700-S"YYY01">BIB Library YYY01</option>
    </select>
  </td>
 </tr>
```

Note that if the HTML pages are not updated, the system will use the first BIB library as set in library_relation table (XXX01 in above example).

19 Applying ADM Filter in OPAC-ACQ Requests

The ACQ request flow appends the order that is created as a result of an ACQ request on a BIB and an ADM record.

Usually, when the system points to a BIB record for attaching an ACQ request, the ADM library that owns the title is not taken into account. In certain library structures, this might causes the system to attach an ACQ request belonging to a certain library to a BIB record owned by a different library. Such a situation can be avoided by applying a filter routine within the ACQ request flow: the ADM library filter.

There is an option to apply the ADM library filter when the system points to a BIB record for attaching an ACQ request.

The ADM library ensures that the ACQ request relies on a BIB that has an ADM record belonging to the patron's home-library. This BIB is used for the ACQ request only if the BIB has an ADM record that is related to the library that matches the patron's home library. Otherwise, a new BIB records is created.

To use the ADM library filter mechanism, the library must configure a hidden parameter in the ACQ Request confirmation HTML pages.

This parameter should be set for libraries whose structure requires activating the ADM filter, for example, a multi BIB/ADM library architecture in which there are several BIB libraries and each of the ADM libraries is connected to more than a single BIB library.

19.1 Activating the ADM Library Filter

Once the candidate BIB record is detected by the locate action, the system conducts an additional check to see if the ADM library filter action should be applied.

In order to turn-on the ADM library filter, set the following hidden parameter with:

```
<input type=hidden name="APPLY_ADM_FILTER" value="Y">
```

The value of the parameter can be:

- Y – Activate the ADM library filter

or

- N – Default value. Do not activate the ADM library filter (append the ACQ request to the candidate BIB record).

The parameter should be set in the following OPAC HTML pages:

- ./alephe/www_f_eng/acq-request
- ./alephe/www_f_eng/acq-request-doc-tail
- ./alephe/www_f_eng/acq-request-confirm
- ./alephe/www_f_eng/acq-request-doc-tail-conf

If the new hidden parameter is not set or set to "N", the located BIB record is used for ACQ requests without applying the ADM filter to the detected BIB record.

The ADM library filter conducts the following:

- The system detects the BIB library of the located BIB record. This BIB record is considered as a ‘candidate BIB record’.
- ADM library filter setup check - The system consults the parameter set in the HTML page to see if the ADM Library filter should be activated.

- If the ADM library filter parameter is turned on in the HTML page, the ADM filter is applied so that the ACQ request is appended on a BIB record that is related to the patron’s home library. If the located BIB record dose not relate to an ADM record that belongs to the patron’s ADM library, a new BIB record is created to which the ACQ request is attached.

19.2 ADM Library Filter Routine

The ADM Library filter takes place only when the hidden parameter ‘APPLY_ADM_FILTER’ is set to “Y”.

The routine works as follows:

The ACQ request flow traces a candidate BIB record for assigning the ACQ request and the ADM library filter checks to see if the BIB record has an ADM record to which the patron belongs.

- If yes – the BIB record is used for attaching an ACQ request.
- If no – a new BIB record is created by duplicating the traced non-relevant BIB record. The newly created BIB record contains bibliographic information similar to the one from which it is driven. Fix, expand fields, and OWN values are stripped. A new ADM record is attached to the new BIB record, to which the ACQ request is appended.

If the candidate BIB has no attached ADM record at all, a new BIB record is created (rather than using the existing BIB record with no ADM record).

20 Log File Expansion

The `www_server` log file can be expanded to include information on the status of the checks that were performed on a hold request. This means that the exact number of errors caused by each check configured in `tab_hold_request` can be logged. This is done by setting the environment variable DEBUG_CHECK_HOLD to Y prior to `www_server` initialization in the following manner:

```
setenv DEBUG_CHECK_HOLD Y
```

If this configuration is set, the log file will include information such as :

```
check_hold_request "CIRC      " :

After check_hold_request_a BUF-ERROR-NO-ERRORS=01

After check_hold_request_b BUF-ERROR-NO-ERRORS=02

After check_hold_request_c BUF-ERROR-NO-ERRORS=02
```

21 Setting Up BIB Request Policy

21.1 Group Definition and Availability

The following two steps are performed when a BIB request is initiated:

- Any ADM library, which has items that are connected to the BIB record, is checked with the tests defined in the HL-GRP-PRE section of the tab_hold_request configuration table. If the ADM library does not pass the tests, it is ignored and its items are not considered for placing the hold request.

- The items that are connected to the requested title are split into groups of *like items*, so that each group can be selected separately. A group of *like items* is defined as a set of items that fulfill the following conditions:

    - They belong to the same ADM library.
    - They are similar in their enumerations (item enumeration level 1, 2 and 3) and in their chronological information (item chron level 1, 2, and 3).
    - They are similar in their sublibrary, item status, item process status and collection information, depending on the `tab100` setup of
        - SUB-LIBRARY-DIVISION
        - HOLD-REQUEST-ITM-STATUS
        - HOLD-REQ-PROCESS-STATUS
        - HOLD-REQUEST-COLLECTION

Items in a group are considered requestable only if they pass the checks that are defined in `tab_hold_request` under the HL-GRP section. Therefore, if all of a group's items do not pass the HL-GRP checks, the group is considered not requestable and it will be shown in the groups list with the *Items* link instead of the *Request* link.

Note that items that are not displayable in the OPAC according to the `tab15.lng` setup are ignored when the *like item* groups are built. Also note that item request exactness (according to column 11 of `tab15.lng`) is also ignored when requests are created from the BIB level.

21.2 Pickup Location List

The available pickup locations are built according to the setup of `tab37`. The list will include all of the allowed pickup locations of all of the items in the group. You can use Column 5 (Availability of item) to filter out a sublibrary from the drop-down list of pickup locations. If Column 5 is set to Y, the line only applies to available items (that is, on the shelf). If Column 5 is set to N, the pickup location will display only if the item is not available (that is, the item is on loan or on the hold shelf).

For example, consider the following tab37 setup:

```
MUS   ## ## ## Y MUS  HIL  WID
MUS   ## ## ## N MUS  HIL  WID
WID   ## ## ## Y HIL  MED
WID   ## ## ## N WID  HIL   MED
```

If the group has items that belong to sublibrary MUS as well as items that belong to sublibrary WID, the pickup list displays all locations, meaning: MUS,HIL,WID and MED.

However, if a WID item is available (according to the availability routine that is defined in the tab100 variable AVAILABILITY-ROUTINE) the WID sublibrary will be removed from the pickup list. This is because WID items allow WID to be a pickup location only if a WID item is not available. The pickup list will therefore be MUS, HIL, MED. On the hold request form, a message will indicate that WID has been removed from the pickup list.

In addition, the selected pickup location sets which items are considered requested. In the example above, if MUS was selected as a pickup location, the WID item is not considered requested even though it is a *like item*. This is because MUS is not a valid pickup location for WID items.

21.3 Item Level Request

The tab100 variable, HOLD-BY-ITEM-GROUP, is used to set the system behavior when items are requested from the item level. If the variable is set to Y, then the list of pickup locations will be built according to the tab37 pickup locations of all items that are *like copies* of the requested item. If the variable is set to N, then the list of pickup locations is built only according to the tab37 line of the requested item.

22 Setting Up My e-Shelf

After performing a search, patrons or guest users can select items to add to their e-Shelf (My e-Shelf) from the results list or from the full view of a record. The contents of the e-Shelf can than be sent by e-mail, printed or saved.

To delete from the database folders and records (Z129 records) that have been generated by guest users, run UTIL/A/18.

The e-Shelf is made up of the following files:

```
myyshelf-clear-basket
myshelf-course
myshelf-delete-folder
myshelf-folder-list
myshelf-folder-list-body-no-doc
myshelf-folder-list-left-body-1
myshelf-folder-list-left-body-2
```

```
myshelf-folder-list-left-body-3
myshelf-folder-list-left-body-no-folder
myshelf-folder-list-left-head
myshelf-folder-list-left-tail
myshelf-folder-list-right-body-1
myshelf-folder-list-right-head
myshelf-folder-list-right-tail
myshelf-full-head
myshelf-full-tail
myshelf-mail
myshelf-no-doc-basket
myshelf-no-fol-for-delete
myshelf-note-full
myshelf-note-short
myshelf-refresh-page
myshelf-save-as-fol
myshelf-short-body
myshelf-short-folder-list
myshelf-short-head
myshelf-short-head-2
myshelf-short-head-2-no-course
myshelf-short-select-include
```

The `www_tab_myshelf_short` table (located in the tab directory of the USR library.), defines which information will be displayed in the columns of the short display in the My e-Shelf function.

The order of the columns defined in this table determines the order of the display, in conjunction with the `myshelf-short-head`, `myshelf-short-head-2`, `myshelf-short-head-2-no-course` and `myshelf-short-body` HTML files in the `www_f_eng` directory.

22.1 My e-Shelf and Course Reading Addresses

There is an additional option for saving the e-Shelf contents onto the server (usually in `alephe_scratch` unless defined otherwise in `aleph_start` by using the `SAVE_FILE_TMPDIR` parameter) and notifying the Course Reading administration. This option is contained in the `myshelf-course` HTML file. The patron must have course permission in his profile to be able to see this option.

You can choose to specify the e-mail address of the Course Reading administration inside this HTML file. You can also use this option to build a drop-down menu of e-mail addresses where there are several course reading administrations to choose from. For example, if the following definition is present in the SEND_MAIL setting in the `myshelf-course` HTML file:

```
<input type=checkbox name="SEND_MAIL" $$0200-C"on" value="
mdewey@exlibris-usa.com ">
```

the system sends the e-mail notification message to mdewey@exlibris-usa.com:

Select this option to inform the Course Reading administration that you want these records added.

If the value in the `myshelf-course` HTML file SEND_MAIL setting is not an e-mail address, the system chooses the e-mail address defined in the `www_course_email_address` environment variable present in `www_server.conf`. For example, the following setting sends the e-mail notification message to the value defined in `www_course_email_address`:

```
<input type=checkbox name="SEND_MAIL"> Check here in order to inform
the Course Reading admini...
```

22.2 Setting the Maximum Number of Records in a Folder

Each folder (including the BASKET folder) can contain a maximum of 500 records.

You can define the maximum number of records allowed in a folder by setting the `www_myeshelf_limit` environment variable in the `www_server.conf` file.

If the variable is not defined, the default for the maximum records in each folder is 500 records.

If the number of records to be stored in a folder exceeds the maximum number as defined in the `www_myeshelf_limit` variable, the user is prompted with the following message:

There is no limit on the number of folders that a patron or guest can create and manage.

My e-Shelf and Union View

An availability column can be added to ‘My e-shelf’ screen of a Union View OPAC Interface. The column displays an indication whether the contributing library to which the user belongs already has the record.

The column must be added to www_tab_myshelf_short.lng table, for example:

```
! 2        3                4        5  6   7     8 9    10        11    12     13
!--!-!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!-!!-!!-!!!!-!-!-!!!!!!!!!!-!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!
9 L  Availability                                  L                ##  AVAIL
```

In addition, the e-shelf HTML pages must be configured to include this column, just like any column that is setup in www_tab_myshelf_short.lng.

- myshelf-short-body
- myshelf-short-head-2
- myshelf-short-head-2-no-course

If the logged in user’s library already has a record that belongs to the set of the record that is in the e-shelf, a message such as “Record already exists in your library” will be displayed in this column

The message can be configured by the library.

The system must be able to recognize to which contributing library the logged in user belongs, based on the user’s z66_user_own_check field. The message will be

displayed if the SID $b of one of the merged records matches the content of z66_user_own_check.

Note that the following line must exist in tab_expand of the union library:

WEB-BRIEF expand_doc_merge_unio

22.3 Setting Up the Note Column in the Records Which Were Not Requested Table

To add a note that specifies the reason for a rejected hold request, add the following to ./alephe/www_f_lng/group-request-exec-head:

After:

<th class=text3>Title</th>

Add:

<th class=text3>Note</th>

In /alephe/www_f_lng/group-request-exec-body

After:

<td class="td2" nowrap>$0200</td>

Add:

<td class="td2" nowrap>$0300</td>

22.4 Setting Up the Submitted a Hold Request Column in My e-Shelf

To add a column that indicates if a record has a hold request, add the following to /alephe/www_f_lng/myshelf-short-body:

After:

<td class=td1 id=centered width="1%" valign=top>$0400</td>

Add:

<td class=td1 id=centered width="1%" valign=top>$1600</td>

In /alephe/www_f_lng/myshelf-short-head-2-no-course,

After:

Documents in My e-Shelf

Add:

<img src="&icon_path/f-check-mark-1.gif"

border=0 alt="Submitted a hold request">

submitted a hold request

After:

<th class="text3">$0300</th>

Add:

<th class="text3">$1600</th>

23 Search Keyword in Headings

A keyword search in the heading list can be implemented in the Aleph search mechanism. This chapter describes the keyword search in heading list functionality and its setup.
The Keyword search can be activated by the user via:

- WEB OPAC - Browse (./www_f_lng/scan-include-2)
- WEB OPAC – Search/Browse (./www_f_lng/find-e-head)
- GUI-Search-Browse

In order to activate keyword search in heading list functionality, the Z01x mechanism must be activated. (See details in the Setting-Up Keyword Search in headings section).

Keyword search in heading list functionality cannot be applied for external libraries (EXT libraries).

23.1 Keyword Search in Heading List Functionality Via Web OPAC-Browse

Upon implementing the keyword search in heading list functionality (see details in section 23.5 Setting Up Keyword Search in Heading List Functionality), the headings list can be searched for keywords that is contained within the heading text.

The user submits a browse query, selects an index to browse, and selects the **Search Keyword in Headings** checkbox.

In the Browse List, the system displays all headings which include the browse query.

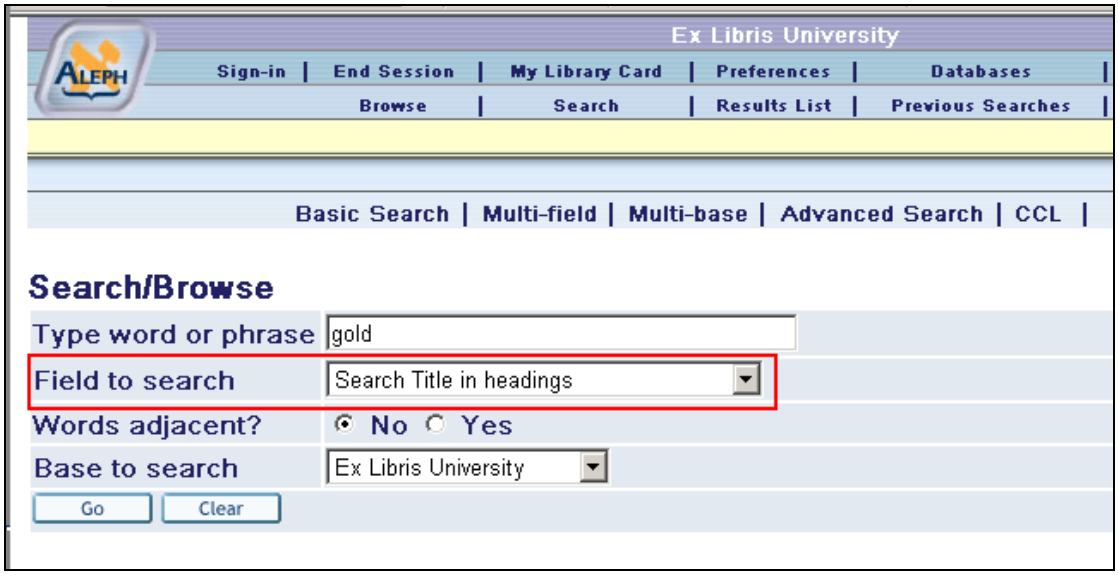
The above example shows a browse list of headings that include the word “gold” in their title index.

User may also type “gold*”. In this case, the browse list displays headings that include words that begin with “gold”, for example, “Broadway golden show”, “The Goldenberg's book”, “Silver and gold”, etc.

23.2 Search Keyword in Heading via Web OPAC-Search/Browse

For systems which are set to apply the Search/Browse mechanism (fine-e-head, see details in 23.5 Setting Up Keyword Search in Heading List Functionality); the Search Keyword in Headings can be applied by selecting heading search in the **Field to search** dialog box.

The following example shows a heading keyword search for the word “gold”.



The search results are the same as described above in the section "Search Keyword in Headings via WEB OPAC-Browse".

See setup details in 23.5 Setting Up Keyword Search in Heading List Functionality section.

23.3 Search Keyword in Heading via GUI-Search-Browse

The GUI-Search-browse screen includes a checkbox **Search Keyword in Headings** that indicate to the system to perform a keyword search in the heading list.

The search results are the same as described above in the section 23.2 Search Keyword in Heading via Web OPAC-Search/Browse.

23.4 Search Rules and Logical Operators for Keyword Search in Headings

The following section explains how the keyword search in heading list functionality works for various user's input.

Keyword search in heading list functionality supports "and", "or", and "not" Boolean operators.

"And" is assumed if no operator is present.

Operators can be typed in English or expressed by using any of the following conventions:

- AND = + (plus sign)
- AND = & (ampersand)
- OR = | (pipe sign)
- NOT = ~ (tilde)
- - (minus sign)

Note that the phrases OR, AND, and NOT are not supported as a text string but only as Boolean search operators.

When entering keywords, the user can use truncation, masking, and logical operators, as explained below.

23.4.1 Truncation

The user can type a portion of a word and add an asterisk (*)at the end. This indicates that the user wants all words that begin with that portion of the word.

Note: A truncation symbol cannot be used at the beginning or a middle of a word. The portion of the word must contain at least three characters before the truncation symbol.

For example:

```
Chloro*
```

retrieves all records containing words beginning with *chloro* in an indexed field.

23.4.2 Logical (Boolean) Operators

The user can narrow or broaden a search by using the Boolean operators: &,+, |, ~, AND, OR and NOT. The user can use these operators in a search string.

For example:

```
government and labor
```

retrieves all records containing *government* and *labor*.

```
government and lab*
```

Retrieves all records containing *government* and (*labor* or *labour* or *laboratory or* any other word beginning with *lab*).

```
government or cabinet
```

retrieves either *government* or *cabinet*.

```
government not parliament
```

retrieves all records containing the word *government* but not including the word *parliament*.

23.4.3 Parentheses

Parentheses must be used in a retrieval request, when required for correct query syntax.

For example:

```
(strategy or tactics) and (dunnigan and churchill)
```

23.4.4 Phrase Searches

In order to search for an exact phrase, the user can type in the phrase with or without double quotes.

For example, to find the exact phrase **Olympic Games**, enter: *"olympic games"* or *olympic games*.
Each of those queries retrieves all records containing the exact phrase *olympic games.*

To retrieve headings that include these words in any order not necessarily in adjacency, use the AND expression (for example, *olympic and games).* To search for a phrase which include “and”, “or” or other operators as regular words, surround the whole phrase with double quotes.

23.4.5 Using the "-" Symbol

The – (minus sign) can be used to lower the score of documents that contain unwanted terms. The ,minus sign is useful when you want to search for documents that contain one query term but want the presence of a second term to cause a document to be ranked lower.

term1-term2 returns documents that contain term 1. The system calculates the score by subtracting the score of term 2 from the score of term 1. Only documents with positive scores are displayed.

Example:

Suppose a query for the term `cars` returns documents about *Ford cars*. You can lower the scoring of the *Ford* documents by using the following expression:

```
cars – Ford
```

This expression returns documents that contain the term `cars` and possibly `Ford`. However, the score for a returned document is the score of `cars` minus the score of `Ford`.

23.5 Setting Up Keyword Search in Heading List Functionality

In order to enable keyword search in heading list functionality, Z01x must be created for Z01- Access Heading records. Enabling the keyword search in headings functionality consumes extra Oracle database space. The extra database space needed is about 70% of the Z01 table size.

Note that indexing Z01x requires an oracle user with “JOB CREATE” enabled.

The following are the required steps for implementing Z01x records:

Step 1:

Define Z01x in the `file_list` of the BIB Library (XXX01) as in the following example:

```
Sample of . /usm01/file_list:
-1-|--2---------|------3-----|--------4------------|---5--|---6--→
TAB z01x              2M              0K              ts0
IND z01x_id           1M              0K              ts1
INX z01x_id1          1M              0K              ts1
```

After Z01x is defined in `file_list`, create the Z01x table by running util/a/17/1 in the BIB library.

Step 2:

In order to create Z01x for current Z01 heading records, in BIB library run the script `p_fill_z01x`.

The following is an example of how to run the script in the USM01 library:

```
>ap

> csh -f p_fill_z01x USM01
```

This script performs the following actions:

- Lock library
- Drop indexes on z01x (if exist)
- For each Z01 record, create z01x records
- Create indexes for z01x
- Unlock library

At this stage Z01x records are created and indexed and keyword search in heading is enabled via OPAC and GUI.

Note that after updating/loading headings using a bulk mode (batch services), the Z01x script should be run.

This means that after submitting the below batch services, the p_fill_z01x script should be run so that z01X records will be updated/created and indexed.

- Update Headings Index (manage-02)
- Pre-Enrich Bibliographic Headings Based on the Authority Database (manage-102)
- Alphabetize Long Headings (manage-17)

Step 3

To enable Search By Keyword in headings, add the following lines to alephe/unicode tab_character_conversion_line:

```
!!!!!!!!!!!!!!!!!!!-!!!!!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!-!
WORD-FIX         ##### # line_utf2line_utf       unicode_to_word_gen
```

```
HEADING-KEYWORD      ##### # line_utf2line_utf    unicode_to_word_gen
```

Note that the HEADING-KEYWORD definition must be the same as the WORD-FIX definition unless the naco_diacritics table (FILING-KEY-10) is not used for normalized text in headings. In this case the HEADING-KEYWORD definition must correspond to the character conversion procedure used for normalized text in headings.

Step 4:

To ensure that the Z01x mechanism is optimized, the optimize procedure should be set in `job_list`.

Following is a sample of setting Z01x optimizes procedure for USM01 BIB library:

```
Sample of ./alephe/tab/job_list:


W0 00:00:00 Y              USM01 optimize_z01x_context_index    USM01
```

This procedure removes duplicate entries from the "z01x_id1" index in order to reduce its size and help the Z01x work faster.

It works similar to UTIL A/17/17/6 "Optimize Context Index".

Step 5:
In the Web OPAC-Browse HTML page, add the checkbox for **Search keyword in headings**:

The checkbox should be located next to the **Type word or phrase** edit field:

```
Relevant lines in ./alephe/www_f_lng/scan-include-2:

  <td class=td2 valign=top>
        <input size=29
               name="scan_start"
               maxlength=100
               value="$0100">
        <input type="checkbox"
               name="scan_include"
               value= "Y"
               $$0800-C"Y">
        Search keyword in headings
  </td>
```

Step 6 - Relevant only for libraries that applied the Web OPAC-Search/Browse option (find-e-head)

The **Field to Search** drop-down list in `./www_f_lng/find-e-head` should be updated with new lines that point to the Search in Keyword functionality.

The list of **Field to Search** is defined in `./www_f_lng/find-e-code-include`.

Add the desired index to `find-e-code-include`. The string "`FACC_`" must precede the index code as in the following example.

```
For example, to set keyword search for title (TIT index) in headings
list, set the following line in ./www_f_eng/find-e-code-include :

       <option  value=FACC_TIT    $$FXX-S"FACC_TIT">Search  Title  in
headings</option>
```

24 Web OPAC and OpenSearch™ Compatibility

If you want to make your site available for searches using the A9 search interface (the search is initiated from http://a9.com/) you need to register your site in A9 and also prepare a number of configuration files required by the OpenSearch protocol. When registering your ALEPH database at the A9 site, you have to define a number of keywords that describe the areas of knowledge covered in your database, institutional affiliations - or any other keywords that would allow an end user to make a decision as to whether include your site in the list of sites s/he wants to search simultaneously.

OpenSearch requires the following mandatory information:

- The search query syntax to be used when searching your database
- The search query itself.

You should provide some of this information when you register at the A9 site, and there are also three configuration files that have to be created and stored on the machine hosting your ALEPH database. These files should reside under `alephe/jboss_conf` directory. These files include:

- Query syntax rules and keywords as mentioned above
- Parameters for the JBoss to construct the X-service
- The XSL file with definitions controlling the output display at A9.

The `QuerySyntax.xml` file includes XML tags required by A9 and information about the server to be searched. The full syntax is described in www.A9.com. A valid example might look like:

```
<?xml version="1.0" encoding="UTF-8"?>
<OpenSearchDescription xmlns="http://a9.com/-/spec/opensearch/1.1/">
<ShortName>ALEPH Search</ShortName>
<LongName>ALEPH Search</LongName>
<Developer>ALEPH Development Team</Developer>
<Description>Search ALEPH Catalog</Description>
<Tags>exlibris aleph catalog library</Tags>
<Contact>contact.us@my.library</Contact>
<Url type="application/rss+xml"
template="http://alephex.exlibris.co.il:1891/AlephWebApi/OpenSearch?r
equest={searchTerms}&amp;start_page={startPage?}"/>
</OpenSearchDescription>
```

The following URL points to the above XML, and it is also the one to be registered at A9:

```
http://alephex.exlibris.co.il:1891/AlephWebApi/OpenSearch?action=QuerySyntax
```

Note that the port number specified in the template should correspond to the number defined for the JBoss.

The `QuerySyntax.xml.tml` template file is available in the `alephe/jboss_conf` directory.

The `present2rss.xsl` file defines the fields of the data that will be presented once the search results are fetched and defines the display features - how many hits will be displayed per page, which fields of the resulting records will be clickable, and so on. The present2rss.xsl.tml template file is available in the `alephe/jboss_conf/xsl` directory. The current setting turns the title section of the `present2rss.xsl` file into a clickable link to the document.

The `main.properties` file resides under the `alephe/jboss_conf` directory. This file defines the relation between the JBoss and the ALEPH X-server. It also defines the search base that is searched when a query is submitted in A9. Any physical or logical base defined in `alephe/tab/tab_base.lng` can be specified here. The `main.properties.tml` template file is available in the `alephe/jboss_conf` directory.

25 Web OPAC Security

25.1 XSS Validation

The Web OPAC contains free text fields that allow the insertion of HTML text that can modify the page displayed. Without proper security care, the site may be abused. An attacker can create a program that pretends to be a real browser and submit JavaScript that may open security holes.

If you have encountered this problem, activate the HTML validation variable (XSS-VALIDATION).

Note that even without the activation of this flag, the malicious insertion of HTML scripts does not corrupt the database.

25.2 CSRF Valuation

CSRF (Cross Site Request Forgery) is a type of internet identity theft where a malicious Web site attempts to access information from a second Web site open on a user's computer. If the user has logged on to the second Web site, the malicious Web site may be able to access sensitive information.

In order to prevent this violation from occurring in Aleph, when a user logs on to the Aleph Web OPAC, a token ID is created and saved in the Aleph database as part of the session entity. This token ID is validated by the OPAC before any actions can be performed on the database. Any actions without the token ID are blocked.

In order to apply the token validation, set $alephe_tab/tab100:
TOKEN-VALIDATION = Y

In order to support the token parameter, configure the $9900 place holder as hidden in the following HTML files:

- item-photo-request-head
- bor-include-4
- collect-data-js
- bor-update
- bor-update-without-date
- new-ill-book-request
- new-ill-journal-request
- item-hold-request-head
- booking-page-form-head
- booking-req-conf-2
- title-request-form-head
- myshelf-note-short
- myshelf-note-full
- acq-request-doc-head
- acq-request-doc-head-conf
- acq-request
- new-ill-r-head
- new-ill-req-conf-tail
- group-request-head
- myshelf-short-head-2
- myshelf-short-head-2-no-course
- new-record
- bor-note-new
- item-hold-request-details
- item-photo-request-details

In the./alephe/www_f_lng/bor_note_new file, change:

<a href="&server_f?func=bor-note-new&clear_level=2">

to:

<a href="&server_f?func=bor-note-new&clear_level=2&token_id=$9900">

25.3 Session Fixation

Authenticating a user without invalidating any existing session identifier gives an attacker the opportunity to steal authenticated sessions: An attacker is able to force a known session ID on a user so that after the user authenticates his identity, the attacker has access to the authenticated session.

In a typical example of session fixation vulnerabilities exploitation, an attacker creates a new session on a Web application and records the associated session identifier. The attacker then causes the victim to authenticate his identity on the server using the

same session identifier, giving the attacker access to the user's account through the active session.

In order to prevent this violation from occurring in Aleph, the session ID is changed when the user logs on to Aleph.

To have the session ID change at log on, set the following in alephe_tab_tab100:

SESSION-FIXATION = Y

25.4 Apache Security Vulnerability Patch

The Apache server contains a security vulnerability. To avoid the security vulnerability:

1. Add the following line to the./alephe/apache/conf/httpd.conf file, just before the Section 2 line:

```
RequestHeader unset Range
```

2. Restart the Apache server.

3. Restart the Web server.

26 Web OPAC Accessibility

Web accessibility allows people with disabilities to use the World Wide Web.

This is important because the World Wide Web is an important resource in many aspects of life, such as education, employment, government, commerce, and health care. It is essential that the Web be accessible in order to provide equal access and equal opportunity to people with disabilities. Having the Web accessible can also help people with disabilities more actively participate in society.

This chapter covers a set of accessibility recommendations. Complying with them assures that the HTML code complies with international standards of code writing, accessibility, and usability to make your content more accessible to a wider range of people including users with disabilities.

The goal of this chapter is to enhance the Aleph Web OPAC to comply with the W3C Web Content Accessibility Guidelines 1.0, level A Section 508 of the Rehabilitation Act (29 U.S.C. 794d) – the leading international accessibility and industry standard. (For exceptions, see Exceptions on page 111).

26.1 Adding Descriptions to Images

Use the alt tag to provide a brief description of the content and function of images, so that a screen reader can describe the image. It should be as short as possible - 1024 characters maximum. If the image is used for cosmetic purposes (a blank space, dot, etc.) leave the alt attribute empty.

For example, change:

```
<img          style="display:none;" id="google_book"/ >
```

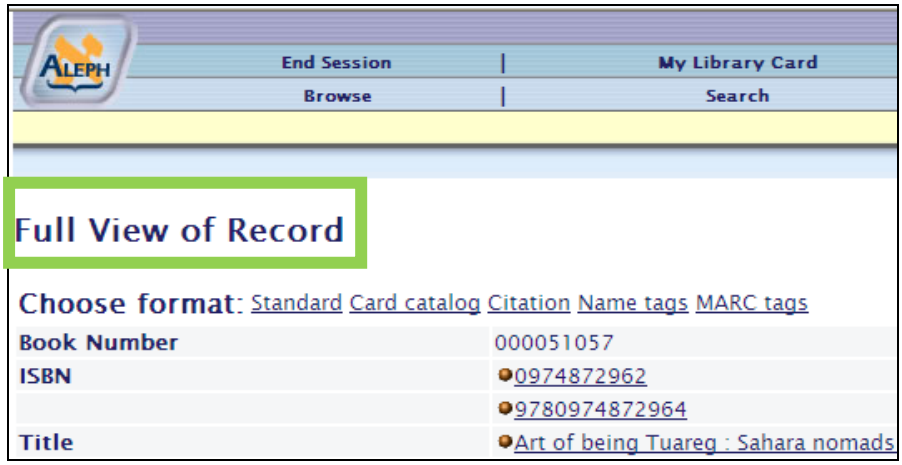
To:

```
<img           style="display:none;" alt="Book Front Cover"
id="google_book"/ >
```

26.2 Defining Titles with the Heading Attribute

Define the main title of a page with the <h1> attribute.

For example:



Change:

```
<div class="title">
  Full View of Record
</div>
```

to:

```
<div><h1>
  Full View of Record
</h1></div>
```

In order to keep the title display consistent with the previous title view, define the <h1> attribute in the Web OPAC CSS file. See the following example:

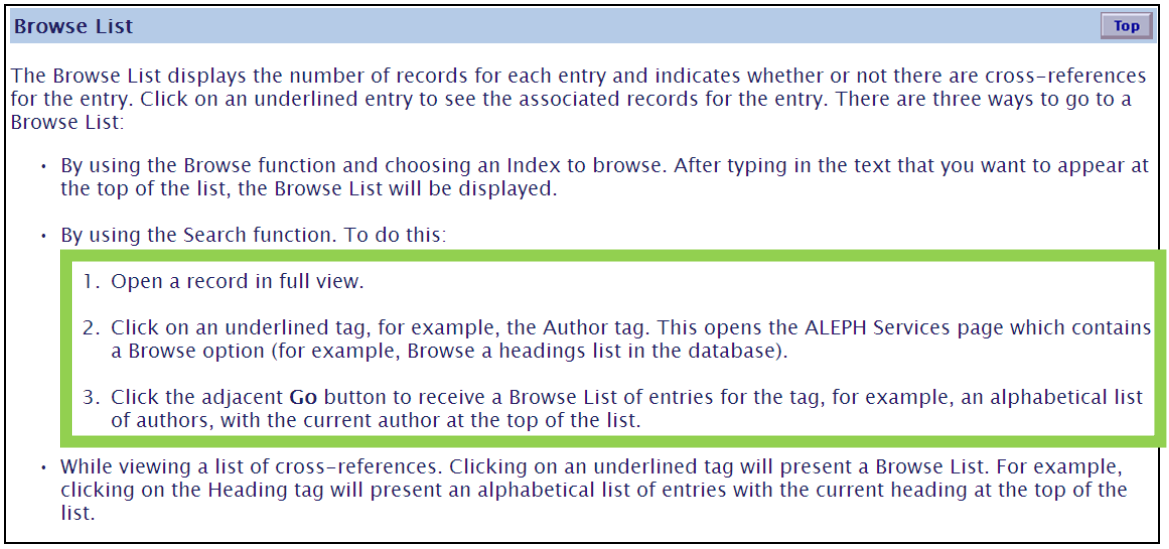
```
h1
{
font-size:120%;
font-family:'Lucida Sans Unicode','Arial Unicode MS',Arial,
Helvetica, serif;
font-weight:bold;
color:#212063;
}
```

26.3 Bulleted and Numbered Lists

HTML offers several mechanisms for specifying lists of information. All lists must contain one or more list elements such as bullets and numbers.

Update all lists in the Web OPAC to have open and close <ol> or <ul> tags and each item of the list to have an <li> tag.

For example:



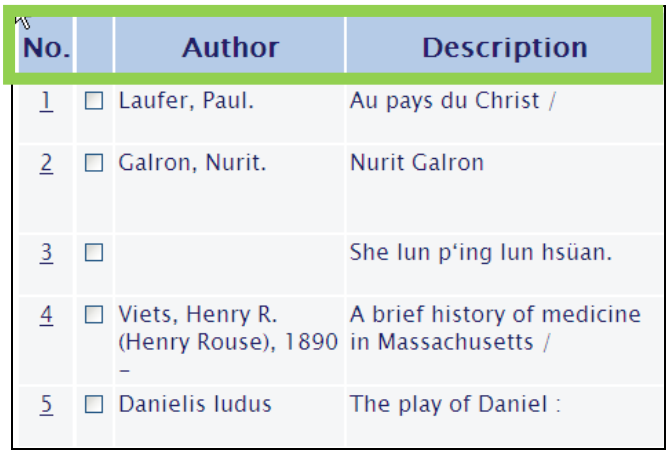
The following is an example of the code:

```
<ol>
<li>Open a record in full view.
<br><br>
<li>Click on an underlined tag, for example, the Author tag. This opens the ALEPH Services page which contains a Browse option (for example, Browse a headings list in the database).
<br><br>
<li>Click the adjacent <b>Go</b> button to receive a Browse List of entries for the tag,for example, an alphabetical list of authors, with the current author at the top of the list.
</ol>
```

26.4 Indicating Table Headers

In order for a data table to be accessible, indicate row and column headers with the <th> tag, so that screen readers can navigate through data table one cell at a time. Do this only for data tables and not for visual layout tables.

For example:



No.		Author	Description
1	☐	Laufer, Paul.	Au pays du Christ /
2	☐	Galron, Nurit.	Nurit Galron
3	☐		She lun p'ing lun hsüan.
4	☐	Viets, Henry R. (Henry Rouse), 1890 –	A brief history of medicine in Massachusetts /
5	☐	Danielis ludus	The play of Daniel :

The code is:

```
  <th class="text3">No.</th>
  <th class=text3><br></th>
  <th class="text3">Author</th>
```

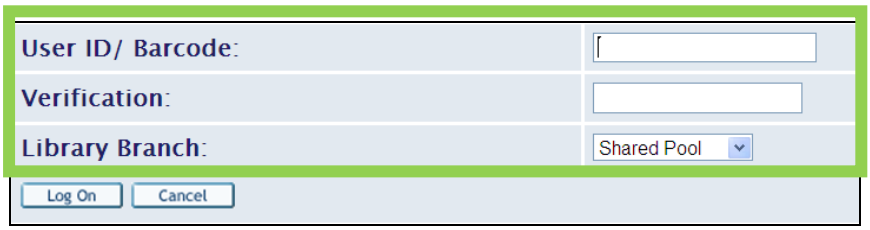
```
  <th class="text3">Description</th>
```

26.5 Assigning Labels to Input Form Elements

HTML forms are used to pass data to a server. A form can contain input elements such as: text fields, checkboxes, radio-buttons, and submit buttons. Assign all input elements a label with the <label> tag.

26.5.1 Text Fields

The following is an example of a text field:



Change the code from:

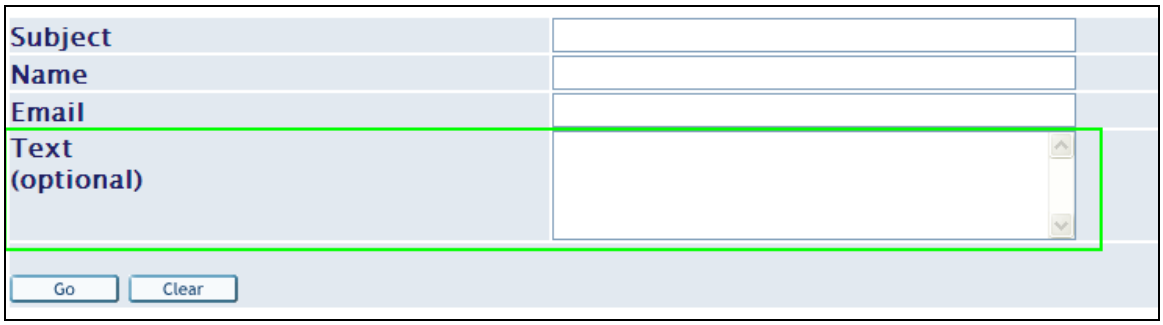
```
<Tr>
<td class=td2> Verification:</td>
  <td class=td2>
  <input type=password
    size=21
    maxlength=20
    name=bor_verification
    value ""=>
 </td>
 </tr>
```

To:

```
<Tr>
<td class=td2><label for="pat_password"> Verification:</label></td>
  <td class=td2>
  <input id="pat_password"
    type=password
    size=21
    maxlength=20
    name=bor_verification
    value ""=>
 </td>
 </tr>
```

26.5.2 Text Area Fields

The following is an example of a text area field:



Change the code from:

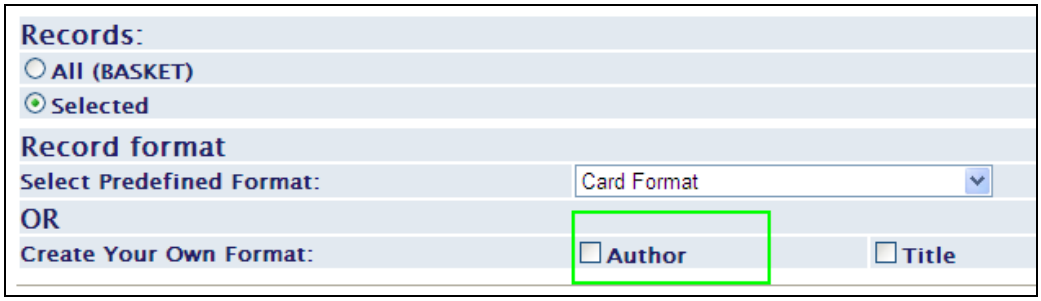
```
>tr<
                <td valign=top class=td2>Text
>  br>(optional)</td<
> td class=td2<
>  textarea name="text" cols=39 rows=4>$0800</textarea<
/> td<
              </td>
             </tr>
```

To:

```
>tr<
                <td valign=top class=td2><label for="text">Text
>  br>(optional)</label></td<
> td class=td2<
>  textarea id="text" name="text" cols=39 rows=4>$0800</textarea<
/> td<
              </td>
             </tr>
```

26.5.3 Checkbox Fields

The following is an example of a checkbox field:



Change the code from:

```
              <td class=td2 id=small >
                <input type="checkbox"
                  name="own_format"
                  value="1%23%23%23%23">Author
              </td>
```

To:

```
              <td class=td2 id=small >
                <input id="author" type="checkbox"
                  name="own_format"
                  value="1%23%23%23%23"><label
for="author">Author</label>
              </td>
```

26.5.4 Radio Button Fields

The following is an example of a radio button field:

Change the code from:

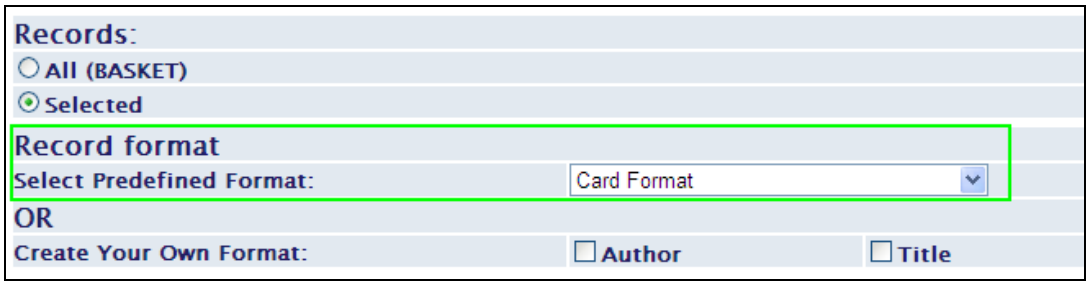
```
    <td class=td2 id=small >
      <input type="radio"
             name="encoding"
             value="UTF_TO_WEB_MAIL_ASCI"
             checked>ASCII (recommended / default)
    <td>
```

To:

```
    <td class=td2 id=small >
      <input id="ascii" type="radio"
             name="encoding"
             value="UTF_TO_WEB_MAIL_ASCI"
             checked><label for="ascii">ASCII (recommended / default)
    </label></td>
```

26.5.5 Drop-Down List Fields

The following is an example of a drop-down list field:



Change the code from:

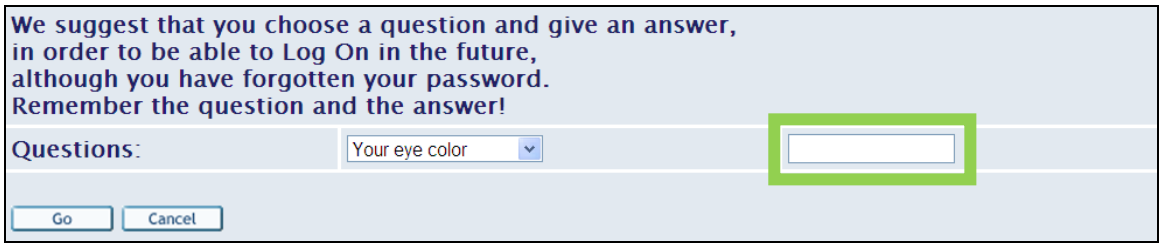
```
<tr>
  <td class=td2 id=small width=20% nowrap>
      Select Predefined Format:
  <td>
  <td class=td2 id=small colspan=7>
   <select name=format>
          <option value=037              >Card Format</option>
          <option value=001              >MARC Tags</option>
          <option value=002              >Name Tags</option>
          <option value=997              >MARC Format</option>
```

```
          <option value=998              >ALEPH Sequential</option>
          <option value=999              >OPAC Format</option>
          <option value=777                  >ISI ResearchSoft Tagged
Output Format</option>

   </select>
  </td>
 </tr>
```

To:

```
<tr>
  <td class=td2 id=small width=20% nowrap><label for="format">
      Select Predefined Format:
  </label></td>
  <td class=td2 id=small colspan=7>
   <select id="format" name=format>
          <option value=037              >Card Format</option>
          <option value=001              >MARC Tags</option>
          <option value=002              >Name Tags</option>
          <option value=997              >MARC Format</option>
          <option value=998              >ALEPH Sequential</option>
          <option value=999              >OPAC Format</option>
          <option value=777              >ISI ResearchSoft Tagged
Output Format</option>

   </select>
  </td>
 </tr>
```

If the input field does not have an associated caption, for example:



Create a hidden label. For example:

```
<td  class=td2><label  for="question_answer"  class=hidden>  Question
answer</label>
      <input type=password
            id="question_answer"
            value=""
            name=PASSWORD_VALUE
            size=16
            maxlength=15>
  </td>
```

Define the hidden class in the Web OPAC CSS file as:

```
.hidden
{
    position:absolute;
    left:-10000px;
    top:auto;
    width:1px;
    height:1px;
```

```
    overflow:hidden;
}
```

26.6 Color Indications

Information or restrictions indicated using colors are not detectable to screen readers. Therefore, make sure that the pages do not indicate information or restrictions with color. For example, red asterisks to indicate mandatory fields are not detected by screen readers.

Use the title attribute for such items.

For example:

```
<table cellspacing=2 border=0 width="60%">
 <tr>
  <td width=15% class=td2 nowrap><label for="patron_id">
    Borrower ID/Barcode :
  </label></td class=td2>
  <td class=td2>
      <span class=text2 title="Mandatory">*</span>
  </td>
  <td class=td2>
    <input id="patron_id" size=30 name=id maxlength=12 value="">
  </td>
 </tr>
 <tr>
```

26.7 Additional Changes for Aleph Web Files

Change the following HTML files, as follows, to support accessibility:

- **bor-loan-body**

Line 4 – change from:

```
    <td  class=td1  valign=top  align="center"><input  type="checkbox"
name="$1100"></td>
```

To:

```
    <td       class=td1       valign=top       align="center"><label
for="select_row_$2200" class=hidden>Select     Row</label>
    <input id="select_row_$2200" type="checkbox" name="$1100"></td>
```

- **booking-matrix-body**

Change from:

```
                <td class="$3100" align="center" nowrap><br></td>
                <td class="$3200" align="center" nowrap><br></td>
          <td class="$3300" align="center" nowrap><br></td>
                <td class="$3400" align="center" nowrap><br></td>
                <td class="$3500" align="center" nowrap><br></td>
          <td class="$3600" align="center" nowrap><br></td>
```

```
                <td class="$3700" align="center" nowrap><br></td>
                <td class="$3800" align="center" nowrap><br></td>
          <td class="$3900" align="center" nowrap><br></td>
                <td class="$4000" align="center" nowrap><br></td>
          <td class="$4100" align="center" nowrap><br></td>
          <td class="$4200" align="center" nowrap><br></td>
          <td class="$4300" align="center" nowrap><br></td>
         <td class="$4400" align="center" nowrap><br></td>
         <td class="$4500" align="center" nowrap><br></td>
         <td class="$4600" align="center" nowrap><br></td>
         <td class="$4700" align="center" nowrap><br></td>
         <td class="$4800" align="center" nowrap><br></td>
         <td class="$4900" align="center" nowrap><br></td>
         <td class="$5000" align="center" nowrap><br></td>
         <td class="$5100" align="center" nowrap><br></td>
         <td class="$5200" align="center" nowrap><br></td>
         <td class="$5300" align="center" nowrap><br></td>
         <td class="$5400" align="center" nowrap><br></td>
```

To:

```
          <td     class="$3100"     title="$5500"     align="center"
nowrap><br></td>
         <td      class="$3200"      title="$5600"      align="center"
nowrap><br></td>
         <td      class="$3300"      title="$5700"      align="center"
nowrap><br></td>
         <td      class="$3400"      title="$5800"      align="center"
nowrap><br></td>
         <td      class="$3500"      title="$5900"      align="center"
nowrap><br></td>
         <td      class="$3600"      title="$6000"      align="center"
nowrap><br></td>
         <td      class="$3700"      title="$6100"      align="center"
nowrap><br></td>
         <td      class="$3800"      title="$6200"      align="center"
nowrap><br></td>
         <td      class="$3900"      title="$6300"      align="center"
nowrap><br></td>
         <td      class="$4000"      title="$6400"      align="center"
nowrap><br></td>
         <td      class="$4100"      title="$6500"      align="center"
nowrap><br></td>
         <td      class="$4200"      title="$6600"      align="center"
nowrap><br></td>
         <td      class="$4300"      title="$6700"      align="center"
nowrap><br></td>
         <td      class="$4400"      title="$6800"      align="center"
nowrap><br></td>
         <td      class="$4500"      title="$6900"      align="center"
nowrap><br></td>
         <td      class="$4600"      title="$7000"      align="center"
nowrap><br></td>
         <td      class="$4700"      title="$7100"      align="center"
nowrap><br></td>
         <td      class="$4800"      title="$7200"      align="center"
nowrap><br></td>
              <td    class="$4900"    title="$7300"    align="center"
nowrap><br></td>
         <td      class="$5000"      title="$7400"      align="center"
nowrap><br></td>
```

```
        <td      class="$5100"      title="$7500"      align="center"
nowrap><br></td>
        <td      class="$5200"      title="$7600"      align="center"
nowrap><br></td>
        <td      class="$5300"      title="$7700"      align="center"
nowrap><br></td>
        <td      class="$5400"      title="$7800"      align="center"
nowrap><br></td>
```

- **item-hold-request-head**

The pickup options list comes from the code along with the ID of the label (id=”pickup”). Therefore, define the label in the HTML page as “pickup”.

Change from:

```
<td class="td2" width=20% nowrap>Pickup/delivery location :</td>
```

To:

```
<td class="td2" width=20% nowrap><label for="pickup">Pickup/delivery
location :</label></td>
```

- **group-request-head**

The pickup options list comes from the code along with the ID of the label (id=”pickup”). Therefore, define the label in the HTML page as “pickup”.

Change from:

```
<td class="td2" width=20% nowrap>Pickup location :</td>
```

To:

```
<td class="td2" width=20% nowrap><label for="pickup">Pickup location
:</label></td>
```

26.8 Skipping Repetitive Links

Give pages an option to skip repetitive links in order to allow users with a screen reader to avoid hearing the same links in each page to which they browse.

Add the following lines in each place you want to skip.

Add the highlighted line at the beginning of the code:

```
<A HREF="#skip-section" class="hidden">Skip Section</A>
<include>include-window
<table   width="100%"   border="0"   cellspacing="0"   cellpadding="0"
align="center">
  <tr class="topbar">
    <td valign="middle" height="20" width="103">
      <a   href="http://www.exlibris.co.il"   class="blue"   target=new
title="Ex Libris, developer of the ALEPH Integrated Library System">
      <img       src="&icon_path/f-aleph-logo-1.jpg"       width="103"
height="20" border="0" alt="Aleph logo"></a></td>
    <td   class="topbar"   valign="middle"   height="20"   width="*"
align="center" colspan="15">
       &f-base
    </td>
  </tr>
```

And the following line at the end:

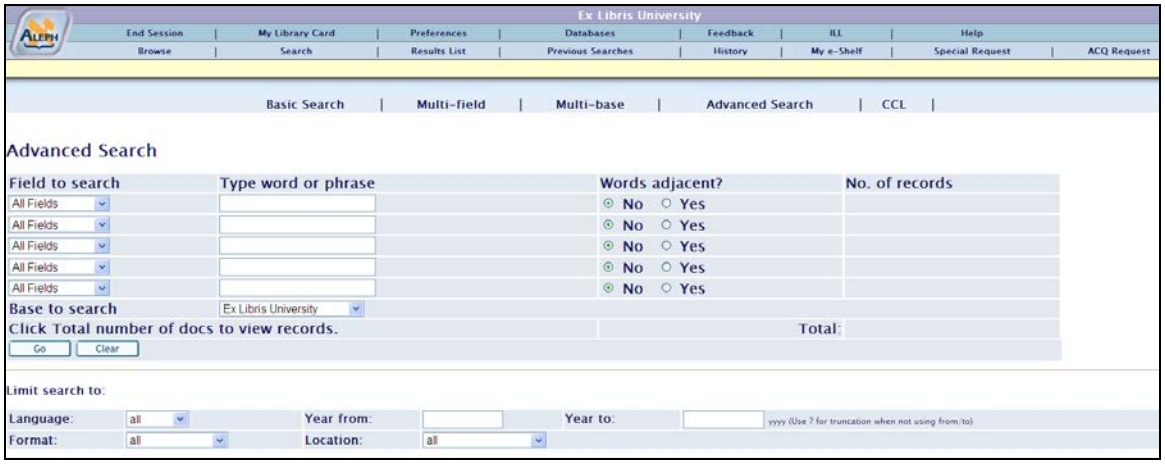
```
  <tr>
```

```
    <td    valign="middle"    height="20"    background="&icon_path/f-
spacer.jpg" colspan="17" nowrap>
       
    </td>
  </tr>
</table>
<A NAME="skip-section"></A>
```

26.9 Exceptions

The following pages have a limited level of accessibility:

- Advanced search



- All the screens that have additional filter option after the Go button. For example:

Auto refresh is not accessible according to section 508; however, in the Aleph Web OPAC, auto refresh can be made accessible using the following tag in:

```
<META HTTP-EQUIV="REFRESH" CONTENT="1200; URL=http://il-
aleph09.corp.exlibrisgroup.com:8991/F/Y3Q77Y6GMASJVXBGJ7FPLE88YVIX5DH
T6FU6T3NK627BFY2SCD-00001?func=logout">
```

Note: Since most screens reader support java script functionality, the <non script> tag has not been added to the Aleph Web OPAC.

27 Link to Google Books Service

It is possible to link a record's full view, which is displayed after a search query, to the Google Books Service. To implement this option, add the following to the full-set-tail HTML page after the "Choose format" table:

```
<a  id="$0800" style="display:none;">
<img           style="display:none;" id="google_book"/>
About this book
</a>
<script>
// Function to process GBS info and update the dom.
function ProcessGBSBookInfo(booksInfo) {
  var url_elem = document.getElementById("$0800");
  var pic_elem = document.getElementById("google_book");
  for  (var  items_idx  =  0;  items_idx  <  booksInfo.totalItems  ;
items_idx++)
  {
        var item = booksInfo.items[items_idx];
        if(!item.volumeInfo  ||  !item.volumeInfo.industryIdentifiers)
{
                continue;
        }
        url_elem.href = item.volumeInfo.infoLink;
        url_elem.style.display = '';
        if (item.volumeInfo.imageLinks.smallThumbnail)
        {
           pic_elem.src = item.volumeInfo.imageLinks.smallThumbnail;
           pic_elem.style.display = '';
        }
   }
</script>
<script src="$0700"></script>
```

28 Forgot Password

Patrons can reset their password to the OPAC by activating the Forgot Password? link in the Web OPAC login page. Based on the library setup, the link can provide two options for resetting a forgotten password:

- Answering a security question whose answer was previously defined

or

- Sending an email to the patron's email account that provides the ability to reset the password.

The functionality of the Forgot Password flow is set in the following login pages:

./alephe/www_f_lng/login

./alephe/www_f_lng/login-acq-session

./alephe/www_f_lng/login-ill

./alephe/www_f_lng/login-session

28.1 Reset Password by Answering a Predefined Question

In order to implement the question-answer option, set the following line in the login pages (listed above).

```
<a href="&server_f?func=file&file_name=bypass-passwd">
```

28.2 Reset Password by Getting an Email

In order to implement the reset password option by getting an email, set the following line in the login pages (listed above):

```
<a href="&server_f?func=file&file_name=forgot-password">
```

- The following files should be set in to ./alephe/www_f_lng:

```
forgot-password
```

```
forgot-password-rst
```

- Make sure the following lines <u>are not</u> set in ./alephe/www_f_lng/bor-new:

```
           <td colspan=3>
             <hr>
           </td>
          </tr>
          <tr>
           <td class="text9" colspan=3>
              If you forgot your password, you can use the "Password
              Bypass"  <br>  mechanism  by  selecting  one  of  the
questions and answering it.
           </td>
          </tr>
          <tr>
           <td        class=td2        width=20%        nowrap><label
for="question">Question Database:</label></td>
           <td class=td2>
               <select id="question" name=N_PASSWORD_CODE>
                 <option      value=01      $$2500-S"01">Your      eye
color</option>
                 <option     value=02     $$2500-S"02">Your     lucky
number</option>
                 <option      value=03      $$2500-S"03">Size      of
shoes</option>
                 <option    value=04    $$2500-S"04">Your    father's
name</option>
                 <option    value=05    $$2500-S"05">Your    mother's
name</option>
               </select>
           </td>
           <td  class=td2><label  for="question_answer"  class=hidden>
Question answer</label>
               <input       id="question_answer"       type=password
name=N_PASSWORD_VALUE size=16 maxlength=15 value="$2600">
           </td>
          </tr>
          <tr>
```

- Make sure the following lines <u>are not</u> set in./alephe/www_f_lng/ bor-update-password

```
  <td colspan=3>
    <hr>
  </td>
 </tr>
 <tr>
  <td class=td2 colspan=3>
     We suggest that you choose a question and give an answer,
     <br> in order to be able to Log On in the future,
```

```
     <br> although you have forgotten your password.
     <br> Remember the question and the answer!
  </td>
 </tr>
 <tr>
  <td class=td2 width=20% nowrap><label
for="questions">Questions:</label></td>
  <td class=td2>
      <select id="questions" name=PASSWORD_CODE>
        <option value=01 $$0100-S"01">Your eye color</option>
        <option value=02 $$0100-S"02">Your lucky number</option>
        <option value=03 $$0100-S"03">Shoe size</option>
        <option value=04 $$0100-S"04">Your father's name</option>
        <option value=05 $$0100-S"05">Your mother's name</option>
      </select>
  </td>
  <td class=td2><label for="question_answer" class=hidden> Question
answer</label>
      <input type=password
            id="question_answer"
            value=""
          name=PASSWORD_VALUE
          size=16
          maxlength=15>
  </td>
 </tr>
 <tr>
```

- Define the varible `www_max_hours_reset_pw` in ./alephe/www_server.conf

This variable defines the maximum number of hours in which the URL that is sent by mail is active.

The variable can be from 01 to 99 (two-digits)

Default value: 24

For example:

```
setenv www_max_hours_reset_pw       12
```

- Set the following files in /aleph/error_lng/www_f_heading:

```
0390 L Fail to submit the Reset Password form. Please apply to
library Staff!

0391 L The Password verification fields do not match. Please retype
again

0392 L Your request to reset password has been received. A message
has been sent to your Email address.
```

- Set the following files in ./[library]/tab/form_description.lng

```
reset-password-mail               Resetting Library account password

reset-password-confirm          Successful reset of your library
account password
```

- Set the following files in ./[library]/form_lng/ :

```
reset-password-mail-00

reset-password-confirm-00
```

29 Web OPAC-Related Tables

General

Table Extensions

You can use different tables for Web OPAC by using the file extensions: *filename.sublibrary, base* and so on.

Example:

edit_doc_999.lng.*ulinc*

Library Tables

The following is a list of some of the configuration tables that are related to Web OPAC. They are located in the bibliographic library's tab directory unless otherwise stated (for example `xxx01/tab/tab05.lng`).

DISPLAY TABLES

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_field.lng` and `edit_paragraph.lng`.

edit_doc_777

This table defines tag conversion routines and tags to be included as the output of the 777 format. This format is intended for the export of bibliographic records in "ISI ResearchSoft Tagged Output Format". This format can be used to import records into reference managers such as Endnote.

edit_doc_999.lng

`edit_doc_999.lng` defines the linked display format for documents for viewing in the Web OPAC. The display can include "expanded" fields that might be added from HOLdings, items or ACC (Z01) fields.

In addition, there are definitions that determine how links to items display (ITM...), and whether or not links to other records are displayed.

edit_doc_999_aut_xxx10.lng

edit_doc_999_aut_xxx10.lng defines the display of AUT records for cross- reference display. In the Web OPAC, clicking on a reference allows for the display of the linked Authority Record. The display of the record is defined in the `edit_doc_999_aut_xxx10.lng` table where "xxx10" is the code of the AUT library (for example, USM10).

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. Therefore, 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.

edit_paragraph.lng
edit_paragraph.lng defines the display of a paragraph of data from the bibliographic record. The `edit_paragraph.lng` 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 bibliographic information (for example, on User List of Loans, OPAC List of Items, and so on).

tab05.lng
This table defines the caption to be displayed in the Web OPAC before $$n and $$m.

tab06
Defines non-highlighted words in the Full View format in the Web OPAC.

tab22
This table defines the fields that are included in the "short-doc" (z13) - and also defines the fields in the My e-Shelf Brief format.

tab_buf_z403
This table defines URI links and the way in which they are displayed.

tab_events.lng
`tab_events.lng` defines which transactions will write a record to the Z35 or Z69 Oracle tables. While `tab_event.lng` in the administrative library defines which transactions write a record to the Z35 Oracle table, the `tab_event.lng` table in the bibliographic library defines which transactions write a record to the Z35 Oracle table (types 10,11,12) or Z69 oracle table (types 20-35). While the transactions in the Administrative library relate to Circulation, the transactions in the bibliographic library relate to the Web OPAC and to the z39 server.

If the `tab_event.lng` table in the bibliographic library does not exist in the database, then the events will not be created. This does not cause a problem.

tab_item_list_order
The table defines the display order of the items in the Web and in the Search functions in the ALEPH GUIs according to the sublibraries for each base.

tab_service
This table defines the Web OPAC services that are available for a particular field. The services currently available are:

- SCAN (ACC or IND) which activates display of a browse list.
- FIND (ACC) which activates the find command.
- AUT which activates display of the authority record to which the field heading is linked.

- ENGINE which activates an external search engine, using details defined in Z121.

tab_sfx
This table is used to set the icon that will be displayed for the SFX link in the Web OPAC. The table can be used to define what SFX services are available for a record that was found on a specific base, and which icon is displayed to signify that the services are available.

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

- the `option-sort` HTML screen
- `www_server.conf`
- `tab01.lng` - document record fields (for correct filing of z101 sort keys).
- `www_f_sort_heading.lng`

tab_sub_lib_sort
This table contains definitions of the sort order of a sublibraries list in various ALEPH functions.

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.

tab_z121
This table contains definitions for the Web service search engine. These definitions include the search engine name, and the URL.

www_f_tab_short_999
`www_f_tab_short_999` defines the short display of bibliographic data in custom formats (but not in table format) in the Web OPAC. You can set up different formats, varying the field codes used, and the order in which they appear. You must use format numbers from 951-998 for your custom formats.

Note that a format with the format number 999 must be present in the table. The 999 format is used by the system if a format that does not exist in the table is used. Format number 950 is reserved for the full `edit_doc_999.lng` view.

www_f_sort_heading.lng
The brief display of bibliographic data in the Web OPAC allows for sorting of the different display elements as defined in `www_tab_short.lng`. `www_f_sort_heading.lng` defines the sort options and can be edited using UTIL J/2.

www_tab_myshelf_short.lng
This table defines which information is displayed in the columns of the short display in the My e-Shelf function.

The order of the columns defined in this table determines the order of the display, in conjunction with the `myshelf-short-head-2`, `myshelf-short-head-2-no-course` and `myshelf-short-body` HTML files in the www_f_lng directory.

www_tab_short.lng
`www_tab_short.lng` defines the short display of bibliographic data in the Web OPAC. The short display of bibliographic information in table format will be the default in the online Web OPAC when:

- "option-display-table" is chosen (From the Preferences screen, option display format, option brief view in table);

  or

- If the library has "000" set as the default display format in www_server.conf.

INDEXING TABLES

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

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

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

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

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

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.

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`.

tab_word_breaking
This table defines routines for word breaking procedures.

path_convert

The path_convert tables under the libraries tab directory redirect files and tables. Every library has its own set of tables and files that can be maintained separately. However, it is possible for libraries to share the same table or direction by setting up redirection in the path_convert tables. For example, two libraries can use the same Web screens, forms, tab directories and so on.

Note that you can define general Web directories under alephe.

Other Tables

library_relation
alephe/tab/library_relation defines the connections between libraries.

Note

If changes are made to the table, the servers must be reactivated in order to have the changes reflected in the clients.

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

There are a number of Web OPAC-related tab100 variables. They are:

1. ACC-COUNT-LIMIT. This variable controls the maximum number of records shown in the "Number of records" column in the headings list in Web OPAC after a browse.

   The default value is 50. If there are more records than the number set in this variable, the system leaves off counting, and ++ displays, rather than the number itself. The upper limit for this counter is 998. Note that a high number affects system performance so it is suggested to set this number no higher than 50. When the user clicks on the heading to display the list of records, the actual count is clearly displayed.

2. ACC-COUNT-STYLE This variable determines how the counter of a number of related BIB records will be displayed in a browse list when the heading is a "see" reference.

   Possible values are: S or R.

   R = counts both the BIB records linked to this heading and also the BIB records linked to the heading pointed to. Normally, there will be no records under the "see" heading, since it refers to another heading. However, this can occur in libraries that do not automatically update BIB records because of AUThority record "see" references (these libraries use UPD=N in the AUThority record).

   S = count only the BIB records that are linked to this "see" heading. In libraries that do not use UPD=N, and do automatic correction from cross references, this will normally be zero.

   The default value is S.

3. AUTO-TRUNCATE-Z01-FIND. This variable controls the automatic truncation option in Find.

   Possible values are: Y or N.

   Y = when a FIND search is performed, the search term is automatically set as truncated. For example, the search term *TIT=Introduction to demography* retrieves all titles beginning with *Introduction to demography.....*

   The default value is N. If the value is N, then the same query will only retrieve titles containing the exact wording *Introduction to demography* and will not retrieve titles containing any additional text.

4. BASE-DEMO-LIB This variable is used for the tables' synchronization report (UTIL H/1), and for UTIL G/2 values.

   Possible values are: 1 or 2.

   1 = the system's demo database is USMXX.

   2 = the system's demo database is MABXX.

   The default value is 1.

5. BROWSE-NO-MATCH This variable determines whether or not a message is displayed in the browse results list. This message states that an exact match for the browsed string should have appeared here and was not found.

   If the variable is set to Y, code 9982 that resides in $alephe_root/error_lng directory under www_f_heading will appear immediately preceding the next best match in the list.

   If the parameter is set to N, the message will not appear.

   The default value is N.

6. HOLD-BY-ITEM-GROUP This variable controls the pickup location list policy when hold requests are placed from the full view.

   Possible values are: Y or N.

   Y = When a hold request is created from the Web OPAC, the list of pickup locations will be built according to the tab37 pickup locations of all items that are like copies of the requested item.

   N = The list of pickup locations is built only according to the tab37 line on the requested item.

7. FILTER-LIBRARY-CARD This variable controls the behavior of the library card with regard to shared ADM libraries in which the patron is not active.

   Possible values are: Y or N.

   Default : N

   Y = When showing the library card, show only ADM libraries which share this patron and in which the patron is active, meaning that there are loans, hold requests, booking requests, photocopy requests or cash transactions that belong to this patron.

   N = When showing the library card, show all ADM libraries which share this patron, regardless of whether the patron is active in them or not.

Note that in either case, the ADM to which the patron's home library belongs will be displayed first.

8. ITMG-DISPLAY This variable lets you control the links of full-999, so that links are only displayed for sublibraries that have real z30 and not just HOL records.

   Possible values are: Y or N.

   Y = display the sublibrary in edit_doc_999 only if there are Z30 (item) records for the library.

   N = display the sublibrary in edit_doc_999 if there are either Z30 (item) or HOLdings record for the library. The default value is N.

9. PDS-AWARE

   This variable controls the access method to the Web OPAC, i.e. whether or not PDS will be used to manage the OPAC authentication and authorization process.

   Note that the variable is configurable only in the alephe/tab tab100 table.

   Possible values are Y or N. The default value is N.

   Y = Use PDS for user authentication with the Web OPAC.

   N = Do not use PDS for user authentication with the Web OPAC.

10. PDS-KEY-TYPE

    This variable defines the Z308-KEY-TYPE against which all PDS authentications are verified. Multiple key types may be defined by using a comma separator, for example, 01,02,03.

    Note that the variable is configurable only in the alephe/tab tab100 table.

Note: If you are working with PDS in Oracle mode, the identifiers can only be up to 100 characters in length. This is a PDS limitation.

11. UNION-LIBRARY Determines how the system will display equivalent BIB records.

    Possible values are: 0, 1 or 2.

    0 = BIB database does not use "union" features

    1 = Union View Catalog: BIB database uses Z120 and Z127 to display equivalent BIB records as one record. The single record display is linked to the multiple related ADM records for items/holdings display

    2 = Union Catalog: BIB database can use Z120 and Z127 to display equivalent BIB records as one record. Holdings information is embedded in the BIB record, and there is no item display.

    The default value is 0.

12. X852-HOLDING This variable is used for union catalogs and defines whether or not to use a special program for holdings from the 852 tag.

    Possible values are: Y or N.

    Y = whether to use a special program that presents a list of holdings taken from an 852 tag. For union catalogs.

    The default value is N.

13. Z39-LOGIN-BASE-CHECK This variable restricts the non-signed users connecting via Z39.50 accessing limited bases marked with * in tab_base.lng.

    Possible values are: Y, N

    The default value is N.

    Y= non-signed users connecting via Z39.50 have access to all the bases that have user Z39 defined except access to limited bases marked with * in tab_base.lng

    N= non-signed users connecting via Z39.50 have access to all the bases that have user Z39 defined

14. XSS-VALIDATION This variable determines whether to activate the HTML validation added to prevent an XSS breakthrough (a malicious insertion of HTML script by a user).

    Possible values are: Y, N

    The Default value is N

    Y= Active the HTML validation

    N= Non Active HTML validation

tab_base.lng_count

This table contains definitions of the Web databases displayed in the browse list.

tab_base.lng

tab_base.lng defines the logical and physical databases that can be accessed by the user. Logical bases are defined by setting a FIND command that serves as a pre-filter or scope. In order to set a logical base to include everything **except** a group of records, use "alldocuments" to define "everything", together With "not". for example all not wsp=suppressed table format.

You can set a mandatory sign-in in order to access a base (this can be a library or a logical base). This is done in `alephe/tab/tab_base.lng`, by placing an asterisk in column 2. There is no check for specific user names; any valid user name will gain access to the database.

tab_ccl_boolean_operators

The `alephe/tab/tab_ccl_boolean_operators` table defines which words are treated as Boolean operators in given language. For each operator a translation to the ALEPH 500 standard Boolean operator (AND, OR , NOT) is defined.

tab_ill_unit.lng

The `./alephe/tab/tab_ill_unit.lng` table defines which ILL Centers are displayed in the drop-down list when the ILL Request link is clicked. The server address of each ILL Center is defined in this table.

Tab_z36_sort

The [adm_lib]/tab/tab_z36_sort table defines which sort routine will be present for the loan list. The loan list in the WEB is sorted using the entry WWW-USER.

tab_mime_type

`alephe/tab/tab_mime_type` assigns a special file extension to any document format. This file extension is used automatically when documents are saved on the computer in Web OPAC. If a format is not assigned an extension, the *.sav* extension is used.

www_server.conf

This table defines parameters for the Web server. For more information see Setting Up the System on page 8.

30 Cancelling Hold Requests

Cancel a hold request in the Web OPAC by clicking the **Delete** button in the following Library Card pages:

- The hold request full display
- The title request list
- The title requests full display

The Web OPAC may be configured to allow cancelling a hold request only before the library started processing the hold request (i.e. no call slip has been printed yet). It may also be configured to allow cancelling a hold request after the requested item has already been placed on the hold shelf.

Configuring these options is done by manipulating the following HTML files:

- bor-hold-exp (for the hold request full display)
  - Use the $0700 place-holder to allow cancelling only unprocessed requests.
  - Use the $2300 place-holder to allow cancelling on hold shelf requests.
- bor-title-exp (for the title request list)
  - Use the $0700 place-holder to allow cancelling only unprocessed requests.

- Use the $2100 place-holder to allow cancelling on hold shelf requests.

- bor-title-body (for the title request full display)
  - Use the $2000 place-holder to allow cancelling only unprocessed requests.
  - Use the $2100 place-holder to allow cancelling on hold shelf requests.

Library Card pages, to enable cancelling the request both before it has been processed and after the item is already on the hold shelf, place both place-holders on the relevant HTML page.

31 Staff Request and Requester Name Field

The Central Library is a library that offers services to subscribed libraries. Subscribed libraries may request material from the central library on behalf of their patrons. The subscribed libraries are identified in the central library as "pseudo patrons", but their patrons are not identified in the central library.

When a patron of the subscribed library wants to request material from the central library, the staff of the subscribed library creates the request for the patron. The Requester name field is filled with information about the requesting patron, for tracking purposes.

31.1 Requester Name Field on Request Creation Forms

1. Display the requester name field on hold and photocopy request creation forms:

   In the following files:

   - alephe/www_f_lng/item-hold-request-tail
   - alephe/www_f_lng/item-photo-request-tail
   - alephe/www_f_lng/item-photo-request-tail-an (if you are using this file)

   Insert the following lines:

```
<tr>
 <td class="td2">Requester Name </td>
 <td class="td2">
   <input size=30 maxlength=100 name="requestername"  value="$3800"></td>
</tr>
```

2. Apply the "clear" function also to requester name field:

   In the following files:
   - alephe/www_f_lng/item-hold-request-head
   - alephe/www_f_lng/item-photo-request-head

Insert the bolded line:

```
<script language="Javascript">
<!--
 function my_reset()
 {
  for (var i = 0; i< document.form1.elements.length; i++) {
   if (document.form1.elements[i].name == "author" ||
       document.form1.elements[i].name == "title" ||
       document.form1.elements[i].name == "pages" ||
       document.form1.elements[i].name == "note" ||
       document.form1.elements[i].name == "note2" ||
       document.form1.elements[i].name == "addlinfo" ||
       document.form1.elements[i].name == "requestername") {
      document.form1.elements[i].value = "";
    }

  }
 }
// -->
</script>
```

3. Apply the "back" option also to requester name field:
   In the following files:
   - alephe/www_f_lng/item-hold-request-details
   - alephe/www_f_lng/item-photo-request-details

Insert the following bolded line:

```
<a href="&server_f?func=item-photo-
request&doc_library=$0100&adm_doc_number=$0200&item_sequence=$0300&yea
r=$2500&volume=$2600&sub_library=$2700&type=$2800&no_loaned=$2900&star
t_rec_key=$3000&end_rec_key=$3100&bib_doc_num=$0600&bib_library=$0500&
source_form=PHOTO_DETAILS&author=$4200&title=$4300&pages=$4400&note1
=$4500&note2=$4600&pickup=$4100&addlinfo=$4700&requestername=$3800" >
<img src="&icon_path/f-back.gif" border=0 alt="back to Photocopy Request form">
```

- In order to display the requester name field on request confirmation screen:
  In the following files:
  - alephe/www_f_lng/item-hold-request-details
  - alephe/www_f_lng/item-photo-request-details

Insert the following line:

```
<input type=hidden name="requestername" value="$3800">
```

When the form is submitted, the request is linked to the subscribed library's user with the Requester Name saved in the new Z37-REQUESTER-NAME / Z38-REQUESTER-NAME field.

If cash charges are created, they are charged for the subscribed library's user, and the Z31-REQUESTER-NAME field is populated with the input requester name.

Note: Hold and Photocopy creation forms might be defined with different suffixes for different patrons.

In this case, add the Requester Name field to request forms with the relevant suffix.

For more information, see the **Different Hold Requests forms** and **Different Photocopy Requests forms** sections of the *Aleph 21 System Librarian's Guide-Circulation*.

31.2 Requester Name Information in My Library Card

- In order to display requester name information in cash list:
  Insert the following line to alephe/www_f_lng/bor-cash-head:
  <th class="text3">Requester Name</th>
  Insert the following line to alephe/www_f_lng/bor-cash-body:
  <td class=td1 valign=top>$1100</td>

- In order to display requester name information in requests list:
  Insert the following line to alephe/www_f_lng/bor-hold-head:

```
<th class="text3">Requester Name</th>
```

  Insert the following line to alephe/www_f_lng/bor-hold-body:

```
<td class=td1 valign=top>$1900</td>
```

- In order to display requester name information in requests list:
  Insert the following line to alephe/www_f_lng/bor-photo-head:

```
<th class="text3">
Requester Name
</th>
```

  Insert the following line to alephe/www_f_lng/bor-photo-body:

```
<td class=td1 valign=top>$1100 </td>
```

- In order to display requester name information in request expanded information:
  Insert the following line to alephe/www_f_lng/bor-hold-exp:

```
<tr>
<td class=td1 id=bold align=left>Requester Name:</td>
<td class=td1> $2400</td>
</tr>
```

- In order to display requester name information in request expanded information:
  Insert the following line to alephe/www_f_lng/bor-photo-exp:

```
<tr>
<td class=td1 id=bold align=left>Requester Name:</td>
<td class=td1 align=left> $1500</td>
</tr>
```

31.3 Sort Based on Requester Name Information

- In order to sort my library card → hold requests list according to open date and requester name, set 05 in www_user_z37_sort_routine and environment variable (defined in `www_server.conf`).

- In order to sort my library card → photocopy requests list according to open date and requester name, set 08 in www_user_z38_sort_routine and environment variable (defined in `www_server.conf`).

- In order to sort my library card → cash list according to transfer number, date and requester name, use 09 sort routine in ./<ADM library>/tab/tab_z31_sort, and define matching entries in
./<ADM library>/tab/pc_tab_exp_field.lng under CASH-SORT-TYPE.

- In order to sort my library card → cash list according to requester name, use 10 sort routine in ./<ADM library>/tab/tab_z31_sort, and define matching entries in
./<ADM library>/tab/pc_tab_exp_field.lng under CASH-SORT-TYPE.

Refer to the **tab_z31_sort** chapter of the **Aleph 21 Configuration Guide** for more details.

32 Server Side Scripts in the Aleph OPAC

The OPAC HTML pages can be updated with the tag <program> that enables a call to a server-side script. In order to add content from an external script into an HTML page, add a line in the following format to the desired place in the HTML page:

<program> external script name

Place the external script in a directory specially created for this purpose, under the u tree.

For example: create a new ext directory under `$alephe_root`.

In `aleph_start`, define the environment variable `opac_ext_prog` as the path of this directory. For example:

```
setenv    opac_ext_prog   $alephe_root/ext
```

The application identifies the tag <program> as a request to execute a script and calls the program defined as external script name from the `$opac_ext_prog` directory.

Example from an HTML page:

```
<include>bor-include-4
<program>add_more_bor_info.pl
<div class=title>
```

Place the `script add_more_bor_info.pl` in the `$opac_ext_prog` directory.

The external scripts are written and maintained by the libraries only. The performance and the content of the scripts are the responsibility of the libraries and should be validated before usage.

32.1 Script Input

The script must be based on the data available in the Web session information table: Z63. The Aleph OPAC programs pass all the relevant information to the external script as parameters. The information available from Z63 is detailed in the following table. Note that the parameters must be read by the external script according to the order defined in the table.

The Aleph OPAC programs generate the output file name (specific to the current session) and send to the script this name as a parameter. The external script uses this name if it has HTML lines to write as an output.

Parameter Number	Parameter	Z63 field
1	Output file	---
2	Session ID	Z63-SESSION-ID
3	Base	Z63-BASE
4	Patron ID	Z63-BOR-ID
5	Patron ADM library	Z63-BOR-LIBRARY
6	Current language	Z63-CON-LNG
7	Search result set number	Z63-SET-NUMBER

32.2 Script Output

The script manages the data retrieval and displays it in HTML format. The output of the script (HTML lines) is written to an output file whose name is received as a parameter by the external script.

The Aleph OPAC programs replace in the HTML page the following line with the output of the external script, which must be valid HTML code.

<program>script name

33 List of Web OPAC Files

This section includes the list of Web files for the ALEPH Web OPAC interface that reside in the `../alephe/www_f_lng` directory. The list includes HTML files that support the following functions and subfunctions:

- ADAM on page 134
- Advance Booking Information on page 148

- Advanced Search on page 132
- Basic Search on page 131
- BIB Request on page 138
- Booking Error Page on page 144
- Availability on page 144
- Browse on page 132
- Browse list - ACC on page 132
- Calendar on page 152
- Cash Information on page 149
- CCL Search on page 132
- Confirmation Page on page 144
- Cross references on page 146
- Databases on page 152
- Error on page 152
- Feedback on page 152
- Filter on page 141
- Full on page 134
- Help on page 152
- History on page 139
- History-Loan Information on page 147
- Hold Information on page 147
- History-Hold Request Information on page 148
- ILL on page 142
- Ill Center on page 143
- ILL Information on page 148
- Information display from Holdings Schema records, element set B1 on page 136
- Information display from Holdings Schema records, element set B2 on page 136
- Information display from Holdings Schema records, element set C2 on page 137
- Item Booking on page 143
- Items on page 135
- Loan Information on page 146
- Locate on page 134

- Login on page 145
- Mail on page 151
- Multi-base Search on page 131
- Multi-field Search on page 131
- My e-Shelf on page 139
- Note for borrower on page 151
- Password on page 151
- Photo Information on page 148
- Preferences on page 146
- Previous Searches on page 139
- Profile on page 150
- Proxy Information on page 150
- Rank on page 146
- Refine on page 141
- Results lists on page 133
- Save File on Server on page 141
- SDI on page 150
- Search on page 131
- Services on page 135
- SFX on page 151
- Special Request on page 141
- Update Address on page 150
- User on page 146
- User Information on page 146
- Z39.50 Item on page 136

You can check which files a particular page is made from by using the source viewing option on your browser (for example, right-click and select **View Source** in Internet Explorer 5.0 and later).

The file names are written at the top of the file in a HTML tag. For example:

```
<!-- filename: find-b -->
```

Note

Files that are present in the installed system, but not listed in this section (List of Web OPAC Files), can be ignored, but should not be removed. They are not relevant for the configuration of your Web OPAC.

For information on customizing Web OPAC pages, refer to the document *Web OPAC Screens: Structure and Customization.*

33.1 List of Web OPAC Files

Function	Subfunction	Filename	Include Files
Search			
	Basic Search	Find-b	meta-tags head-1-bor/nobor find-include-buttons find-b-head find-code-include find-include-filter copyrights
		Find-b-permute-head	meta-tags head-1-bor/nobor find-code-include
		Find-b-permute-head-top (for CJK)	meta-tags head-1-bor/nobor find-code-include
		find-b-permute-body	
		find-b-permute-tail	copyrights
		find-b-list-head	meta-tags head-1-bor/nobor find-include-buttons find-b-head
		find-b-list-body	
		find-b-list-tail	copyrights
	Multi-field Search	find-a	meta-tags head-1-bor/nobor find-include-buttons find-include-filter copyrights
	Multi-base Search	find-m	meta-tags head-1-bor/nobor find-include-buttons find-code-include find-include-filter copyrights
		find-m-results-head	meta-tags head-1-bor/nobor
		find-m-results-body	

Function	Subfunction	Filename	Include Files
		find-m-results-tail	copyrights
	Advanced Search	find-d	meta-tags head-1-bor/nobor find-include-buttons find-code-include find-d-code-include (usm90) find-include-filter copyrights
	CCL Search	find-c	meta-tags head-1-bor/nobor find-include-buttons find-include-filter copyrights
	Search/Browse	find-e-head	find-e-code-include base-list-include find-include-filter
	Floating Keyboard	floating-keyboard	floating-keyboard.js meta-tags
	Refined Search	find-r	meta-tags head-1-bor/nobor find-include-buttons find-code-include find-include-filter copyrights
Browse			
	Browse	scan	meta-tags head-1-bor/nobor scan-include-2 copyrights
	Browse list - ACC	scan-acc-head	include-window meta-tags head-1-bor/nobor scan-include-2
		scan-acc-body	
		scan-acc-tail	copyrights
		scan-ind-head	meta-tags head-1-bor/nobor scan-include-2
		scan-ind-body	
		scan-ind-tail	copyrights
		scan-word-head	meta-tags head-1-bor/nobor scan-include-2

Function	Subfunction	Filename	Include Files
		scan-word-body	
		scan-word-tail	copyrights
		scan-sys-head	meta-tags head-1-bor/nobor scan-include-2
		scan-sys-body	
		scan-sys-tail	copyrights
		scan-acc-ug-head	meta-tags head-1-bor/nobor scan-include-2
		scan-acc-ug-body	
		scan-acc-ug-tail	copyrights
	Browse headings list	heading-acc-head	meta-tags head-1-bor/nobor
		heading-acc-head-1	
		heading-acc-head-2	
		heading-acc-body	
		heading-acc-tail	copyrights
Results lists			
		short-1-head	submit-form-include meta-tags include-window short-select-include head-1-bor/nobor short-include-buttons short-include-2
		short-2-head	meta-tags include-window submit-form-include short-select-include short-include-buttons short-include-2
		short-4-head	submit-form-include meta-tags include-window short-select-include head-1-bor/nobor short-include-buttons short-include-2
		short-tail	copyrights
		short-a-head	
		short-a-body	

Function	Subfunction	Filename	Include Files
		short-a-tail	
		short-b-head	
		short-b-body	
		short-b-tail	
		short-c-head	
		short-c-body	
		short-b-body-separator	
		short-c-tail	
Full			
		full-set-head	include-window meta-tags head-1-bor/nobor
		full-set-head-bor	include-window meta-tags head-1-bor/nobor
		full-set-head-nobor	include-window meta-tags head-1-bor/nobor
		full-set-head-bor-ext-ill	include-window meta-tags head-1-bor/nobor
		full-set-head-nobor-ext-ill	include-window meta-tags head-1-bor/nobor
		full-999-body	
		full-xxx-body	
		full-000-body	
		full-999-separator	
		full-set-tail	tail-bar-include copyrights
		direct-head	meta-tags include-window head-1-bor/nobor
		direct-tail	tail-bar-include copyrights
	External Location	media-permission	meta-tags include-window
		media-no-permission	meta-tags copyrights
	Locate	Locate-list	meta-tags copyrights
	ADAM	full-999-body-media	
		full-head-media	
	Down links (view)	dn-link-view-head	meta-tags include-window head-1-bor/nobor
		dn-link-view-body	
		dn-link-view-tail	tail-bar-include copyrights

Function	Subfunction	Filename	Include Files
Services			
		service-head-tag	meta-tags
		service-find-acc-head	
		service-find-acc-tail	
		service-scan-acc-head	
		service-find-m-acc	
		service-find-m-acc-tail	
		service-scan-acc	
		service-scan-acc-tail	
		service-engine	
		service-scan-ind	
		service-aut	
		service-tail	copyrights
		service-aut-bib	
		service-gen-exec	meta-tags
		service-sfx-record	
		service-no-service	meta-tags copyrights
		www-ext-copyrights	meta-tags
		www-ext-copyrights-9	
		media-display-exec	meta-tags
Items			
		item-global-head-1	meta-tags head-1-bor/nobor full-xxx-body
		item-global-head-2	
		item-global-body	
		item-global-body-head	
		item-global-tail	copyrights
		item-global-tail-1	
		item-global-tail-2	
		item-global-body-no-item	
		item-global-body-tail	
		item-global-dropdown-menus-a	

Function	Subfunction	Filename	Include Files
		item-global-dropdown-menus-b	
		item-global-dropdown-menus-b1	
		item-global-dropdown-menus-b2	
		item-global-dropdown-menus-b3	
		item-global-dropdown-menus-b4	
		item-global-dropdown-menus-b5	
		item-global-dropdown-menus-c	
		item-global-prev-next	
		login-item	meta-tags head-1-bor/nobor copyrights
		item-global-exp	include-window meta-tags head-1-bor/nobor copyrights
	Z39.50 Item		
		item-global-ug-head-1	meta-tags include-window head-1-bor/nobor
		item-global-ug-head-2	
		item-loan	meta-tags head-1-bor/nobor copyrights
		item-hold-request-head	meta-tags head-1-bor/nobor
		item-hold-request-body	
		item-hold-request-tail	copyrights
	Information display from Holdings Schema records, element set B1	item-ug-2-head-bb1	
		item-ug-2-body-bb1	
	Information display from Holdings Schema records, element set B2	item-ug-2-head-bb2	
		item-ug-2-body-bb2	

Function	Subfunction	Filename	Include Files
	Information display from Holdings Schema records, element set C2	item-ug-2-head-bc2	
		item-ug-2-body-bc2	
	Information display from Holdings, remote library items, with no circulation, information (ACC central database).	item-cen-head-1	meta-tags head-1-bor/nobor
		full-999-body	
		item-global-tail-2	
		item-cen-head-2-a	
		item-cen-head-2	
		item-cen-body	
		item-cen-body-no-item	
		item-cen-tail	copyrights
		item-cen-tail-2	copyrights
ACQ Request	Acquisition Request - New BIB record	acq-request	meta-tags head-1-bor acq-order-group-include copyrights acq-request-doc-head-conf
	Confirmation for Acquisition Request	acq-request-confirm	meta-tags head-1-bor acq-order-group-include
		acq-request-doc-tail-conf	acq-order-group-include copyrights
	Acquisition Request - existing BIB record	acq-request-doc-head	
		acq-request-doc-tail	acq-order-group-include
Photocopy Request			
	Item Photocopy Request	item-photo-request-head	meta-tags head-1-bor/nobor
		item-photo-request-body	

Function	Subfunction	Filename	Include Files
		item-photo-request-tail	copyrights
BIB Request		bib-request-item-list-head	include-window meta-tags head-1
		bib-request-item-list-head-1	
		bib-request-item-list-head-1-r	
		bib-request-item-list-head-1-v	
		bib-request-item-list-head-2	
		bib-request-item-list-head-3-begin	
		bib-request-item-list-head-3-coll	
		bib-request-item-list-head-3-p-stat	
		bib-request-item-list-head-3-stat	
		bib-request-item-list-head-3-sublib	
		bib-request-item-list-head-3-tr	
		bib-request-item-list-head-filter	
		bib-request-item-list-body-begin	
		bib-request-item-list-body-repeat	
		bib-request-item-list-body-tr	
		bib-request-item-list-tail	
		bib-request-verify	meta-tags head-1 copyrights
		bib-request-no-items	meta-tags head-1 copyrights
		bib-request-1	
		bib-request-2	

Function	Subfunction	Filename	Include Files
		item-hold-request-details	meta-tags head-1 copyrights
Previous Searches			
		history-head	meta-tags submit-form-include head-1-bor
		history-body	
		history-tail	copyrights
		history-cross	meta-tags head-1-bor
		history-cross-body	
		history-cross-tail	copyrights
History			
		save-history-head	meta-tags submit-form-include head-1-bor/nobor
		save-history-body	
		save-history-tail	copyrights
My e-Shelf		myshelf-folder-list	meta-tags head-1
		myshelf-course	replace-textarea-js include-replace-hash replace-textarea-hash-js meta-tags head-1 include-select-format copyrights
		myshelf-mail	replace-textarea-js include-replace-hash replace-textarea-hash-js meta-tags head-1 include-select-format copyrights
		myshelf-short-head	meta-tags myshelf-short-select-include head-1
		myshelf-short-head-2	

Function	Subfunction	Filename	Include Files
		myshelf-short-head-2-no-course	
		myshelf-short-body	
		myshelf-short-tail	copyrights
		myshelf-short-folder-list	
		myshelf-short-select-include	
		myshelf-full-head	meta-tags include-window head-1
		myshelf-full-tail	tail-bar-include copyrights
		myshelf-folder-list-body-no-doc	
		myshelf-folder-list-left-head	include-window myshelf-short-select-include
		myshelf-folder-list-left-body-1	
		myshelf-folder-list-left-body-2	
		myshelf-folder-list-left-body-3	
		myshelf-folder-list-left-body-no-folder	
		myshelf-folder-list-left-tail	
		myshelf-folder-list-right-head	include-window myshelf-short-select-include
		myshelf-folder-list-right-body-1	
		myshelf-folder-list-right-tail	
		myshelf-note-full	meta-tags head-1 copyrights
		myshelf-note-short	meta-tags head-1 copyrights
		myshelf-refresh-page	include-window myshelf-short-select-include
		myshelf-save-as-fol	include-window myshelf-short-select-include
		myshelf-no-fol-for-delete	include-window myshelf-short-select-include
		myshelf-delete-folder	meta-tags myshelf-short-select-include

Function	Subfunction	Filename	Include Files
		myshelf-no-doc-basket	include-window myshelf-short-select-include
		myshelf-clear-basket	meta-tags myshelf-short-select-include
		Save-file	head-1-bor/nobor copyrights
	My e-shelf - Request	group-request-head	meta-tags head-1-bor/nobor
		group-request-body	
		group-request-tail	copyrights
		group-request-exec-head	meta-tags head-1-bor/nobor
		group-request-exec-body	
		group-request-exec-tail	copyrights
Special Request			
		new-record	meta-tags head-1-bor/nobor copyrights
		new-book	meta-tags head-1-bor/nobor copyrights
Save File on Server			
		save-server	meta-tags head-1-bor/nobor copyrights
	save-server-d	meta-tags head-1-bor/nobor copyrights	
Refine			
		short-refine	meta-tags head-1-bor/nobor find-code-include copyrights
Filter			
		short-filter	meta-tags head-1-bor/nobor copyrights
	Text Filter	short-filter-s	meta-tags head-1-bor/nobor copyrights
	Range Filter	short-filter-r	meta-tags head-1-bor/nobor copyrights
		short-filter-y	meta-tags head-1-bor/nobor copyrights

Function	Subfunction	Filename	Include Files
	Filter by date	short-filter-a	meta-tags head-1-bor/nobor copyrights
ILL			
		ill-book-request	replace-textarea-js meta-tags head-1-bor/nobor ill-include-buttons ill-request-main-include copyrights
		new-ill-book-request	popcalendar-js meta-tags head-1-bor/nobor new-ill-r-mn ill-request-main-include copyrights
		ill-journal-request	replace-textarea-js meta-tags head-1-bor/nobor ill-include-buttons ill-request-main-include copyrights
		new-ill-journal-request	popcalendar-js meta-tags head-1-bor/nobor new-ill-r-ser ill-request-main-include copyrights
		ill-conference-request	replace-textarea-js meta-tags head-1-bor/nobor ill-include-buttons ill-request-main-include copyrights
		ill-report-request	replace-textarea-js meta-tags head-1-bor/nobor ill-include-buttons ill-request-main-include copyrights
		ill-request-confirm	meta-tags head-1-bor/nobor copyrights
		ill-request-doc-head	meta-tags head-1-bor/nobor

Function	Subfunction	Filename	Include Files
		ill-request-doc-tail	ill-request-main-include copyrights
	ILL Request - Alternate Unit Selection	ill-alternate-unit-h	meta-tags include-window head-1-bor/nobor
		ill-alternate-unit-b	
		ill-alternate-unit-t	copyrights
	ILL Center	ill-unit-head	meta-tags
		ill-unit-body	
		ill-unit-tail	
		close-window	
	Error message - ILL	message-page	meta-tags head-1-bor/nobor copyrights
Item Booking	Booking Request Form	booking-page-form-head	
		booking-page-form-body	
		booking-page-form-tail	
		booking-page-form-head-no-pickup	
		booking-page-form-body-no-pickup	
		booking-page-form-tail-no-pickup	
		bib-info-table-head	
		booking-return-loc-head	
		booking-page-head	include-window meta-tags js-calendar head-1
		booking-page-tail	
		booking-page-item-info	
		booking-page-delivery-form-head	
		booking-page-form-body	
		booking-page-delivery-form-tail	

Function	Subfunction	Filename	Include Files
		booking-page-pickup-form-head	
		booking-page-form-body	
		booking-page-pickup-form-tail	
		booking-page-return-pickup-head	
		booking-page-return-pickup-tail	
	Availability	booking-matrix-head	
		booking-matrix-body	
		booking-matrix-tail	
		booking-req-list-head	meta-tags replace-page-js head-replace bor-include-3
		booking-req-list-body	
		booking-req-list-tail	copyrights
		booking-req-no-booking	meta-tags replace-page-js head-replace bor-include-1 copyrights
		booking-req-expand-head	meta-tags replace-page-js head-replace
		booking-req-expand	copyrights
		item-avail-page-head	include-window meta-tags
		item-avail-page-tail	copyrights
		item-avail-head	
	Confirmation Page	booking-req-conf-1	meta-tags replace-page-js head-replace
		booking-req-conf-2	copyrights
	Booking Error Page	booking-error-head	include-window meta-tags head-1
		booking-error-body	

Function	Subfunction	Filename	Include Files
		booking-error-tail	copyrights
		booking-error-page	include-window meta-tags head-1
	Login from Booking Request	item-booking-request-1	
Advance Booking	Advance Booking - Time slots	item-schedule-head	meta-tags head-1-bor/nobor
		item-schedule-body	
		item-schedule-tail	
Login			
		login	meta-tags copyrights
		login-session	meta-tags head-1-bor/nobor copyrights
		login-request	meta-tags head-1-bor/nobor copyrights
		login-acq-session	meta-tags head-1-bor/nobor copyrights
		login-ill	meta-tags copyrights
		patron-notice	meta-tags head-1-bor/nobor copyrights
		logout	meta-tags head-1-bor/nobor copyrights
		logout-1	
		bor-new	meta-tags replace-page-js head-replace head-1-bor/nobor sub-library-select-usm50 sub-library-select-usm51 copyrights
		bor-new-message	meta-tags replace-page-js head-replace copyrights
		bypass-passwd	meta-tags copyrights
		pds-login	
		pds-login-first	
		pds-sso	

Function	Subfunction	Filename	Include Files
		pds-down	
		pds-logout	
Preferences			
		option-display	meta-tags head-1-bor copyrights
		option-display-session	meta-tags head-1-bor/nobor copyrights
		option-lng	meta-tags head-1-bor copyrights
		option-lng-session	meta-tags head-1-bor/nobor copyrights
Cross references			
		accref-head	meta-tags
		accref-body	
		accref-tail	copyrights
Rank			
		short-rank	meta-tags head-1-bor/nobor copyrights
User			
	User Information	bor-info Or: bor-info-single-adm (in case of single ADM)	meta-tags replace-page-js head-replace copyrights
	Loan Information	bor-loan-head	meta-tags replace-page-js collect-data-js head-replace bor-include-4
		bor-loan-body	
		bor-loan-tail	Copyrights
		bor-loan-exp-head	meta-tags replace-page-js head-replace
		bor-loan-exp	copyrights
		bor-loan-no-loan	meta-tags replace-page-js head-replace bor-include-1 copyrights
		bor-renew-all-head	meta-tags replace-page-js head-replace
		bor-renew-all-head-2	
		bor-renew-all-head-3	

Function	Subfunction	Filename	Include Files
		bor-renew-all-head-renewed	
		bor-renew-all-head-not-renewed	
		bor-renew-all-gbl-error	meta-tags replace-page-js head-replace copyrights
		bor-renew-all-body	
		bor-renew-all-tail	Copyrights
	History-Loan Information	bor-history-loan-head	meta-tags replace-page-js head-replace bor-include-3
		bor-history-loan-body	
		bor-history-loan-tail	Copyrights
		bor-history-loan-exp-head	meta-tags replace-page-js head-replace
		bor-history-loan-exp	
		bor-history-loan-no-loan	meta-tags replace-page-js head-replace bor-include-1 copyrights
	History-Loan Information Expanded details for a loan	bor-history-loan-exp-head	meta-tags
		bor-history-loan-exp	
		bor-history-loan-exp-1	
		bor-history-loan-exp-2	copyrights
	Hold Information	bor-hold-head	meta-tags replace-page-js head-replace bor-include-3
		bor-hold-body	
		bor-hold-tail	copyrights
		bor-hold-exp-head	meta-tags replace-page-js head-replace
		bor-hold-exp	copyrights
		bor-hold-no-hold	meta-tags replace-page-js head-replace bor-include-1 copyrights

Function	Subfunction	Filename	Include Files
	History-Hold Request Information	bor-history-hold-head	bor-include-3
		bor-history-hold-no-hold	bor-include-1
		bor-history-hold-tail	
		bor-history-hold-exp	
		bor-history-hold-exp-head	
	Photo Information	bor-photo-head	meta-tags replace-page-js head-replace bor-include-3
		bor-photo-body	
		bor-photo-tail	copyrights
		bor-photo-exp-head	meta-tags replace-page-js head-replace
		bor-photo-exp	copyrights
		bor-photo-no-photo	meta-tags replace-page-js head-replace bor-include-1 copyrights
	Advance Booking Information	bor-item-schedule-head	meta-tags replace-page-js head-replace bor-include-1
		bor-item-schedule-body	
		bor-item-schedule-tail	copyrights
		bor-item-schedule-exp	meta-tags replace-page-js head-replace copyrights
		bor-item-schedule-no-hold	meta-tags replace-page-js head-replace bor-include-1 copyrights
	ILL Information - List of ILL Requests	bor-ill-head	meta-tags replace-page-js head-replace bor-include-1
		bor-ill-body	
		bor-ill-tail	copyrights
		bor-ill-exp	meta-tags replace-page-js head-replace copyrights

Function	Subfunction	Filename	Include Files
		bor-ill-no-ill	meta-tags replace-page-js head-replace bor-include-1 copyrights
	ILL request details	bor-ill-exp-ext	meta-tags replace-page-js head-replace
		bor-ill-exp-ext-tail	copyrights
	Cash Information	bor-cash-sub-head	meta-tags replace-page-js head-replace bor-include-3
		bor-cash-sub-body	
		bor-cash-head	
		bor-cash-body	
		bor-cash-tail	copyrights
		bor-cash-exp-head	meta-tags replace-page-js head-replace
		bor-cash-exp	
		bor-cash-exp-1	
		bor-cash-exp-tail	Copyrights
		bor-cash-no-cash	meta-tags replace-page-js head-replace bor-include-1 copyrights
	ACQ Requests for patron	bor-acq-no-acq	
		bor-acq-head	meta-tags replace-page-js head-replace bor-include-1
		bor-acq-body	
		bor-acq-tail	copyrights
	Title Request Information	bor-title-head	meta-tags replace-page-js head-replace bor-include-1
		bor-title-body	
		bor-title-tail	copyrights
		bor-title-exp-head	meta-tags replace-page-js head-replace
		bor-title-exp	copyrights

Function	Subfunction	Filename	Include Files
		bor-title-no-hold	meta-tags replace-page-js head-replace bor-include-1 copyrights
	Proxy Information	bor-proxy-head	meta-tags replace-page-js head-replace bor-include-1
		bor-proxy-body	
		bor-proxy-tail	Copyrights
		bor-proxy-no-proxy	meta-tags replace-page-js head-replace bor-include-1 copyrights
	Update Address	bor-update	meta-tags replace-page-js head-replace copyrights
		bor-update-without-date	meta-tags replace-page-js head-replace copyrights
	Profile	bor-profile-display	meta-tags replace-page-js head-replace sub-library-select-usm50 copyrights
	SDI	bor-sdi-head	meta-tags submit-form-include replace-page-js head-replace
		bor-sdi-body	
		bor-sdi-tail	Copyrights
		bor-sdi-update	meta-tags replace-page-js js-calendar head-replace adm-library-select sub-library-select-usm50 sub-library-select-usm51 copyrights
		bor-sdi-delete	meta-tags replace-page-js head-replace copyrights
		bor-sdi-no-sdi	meta-tags replace-page-js head-replace copyrights

Function	Subfunction	Filename	Include Files
		sdi-refine	meta-tags head-1 find-code-include find-include-filter copyrights
	Password	bor-update-password	meta-tags replace-page-js head-replace copyrights
	Note for borrower	bor-note-new	meta-tags replace-page-js head-replace sub-library-select-usm50 copyrights
		bor-note-display-head	meta-tags replace-page-js head-replace
		bor-note-display-body	
		bor-note-display-tail	Copyrights
		bor-info-no-z305	meta-tags replace-page-js head-replace copyrights
	Mail from Activities	user-info-mail	replace-page-js include-replace-hash replace-textarea-hash-js meta-tags head-1 copyrights
Mail			
		short-mail	meta-tags replace-textarea-js head-1-bor/nobor include-select-format copyrights
		full-mail	meta-tags replace-textarea-js head-1-bor/nobor include-select-format copyrights
SFX			
		sfx-title	
		sfx-subject	
		sfx-au	

Function	Subfunction	Filename	Include Files
		sfx-record	
		sfx-empty	
		sfx-admin	
		sfx-record-ver-0.1	
		sfx-record-ver-1.0	
		sfx-button-data	
Databases		base-list	meta-tags head-1-bor/nobor copyrights
Help		help-1	meta-tags copyrights
Error		forbidden-access	
Calendar		calendar	
Feedback		feedback	meta-tags replace-textarea-js head-1-bor/nobor copyrights
User Limit Error		license-limit	