



Aleph 21 Release Notes

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, 2012. All rights reserved.

Document released: June 24, 2012

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

Table of Contents

Chapter 1	Introduction	7
	Purpose of This Document.....	7
	Aleph 21 Documentation.....	7
Chapter 2	Acquisitions	9
	Verde Order Counter	9
	Order Closure Upon Arrival Completion.....	10
	Renew Serial Encumbrance Based on Paid Amount.....	10
Chapter 3	Cataloging	13
	Expand from Item Barcode	14
	Additional LKR Field Upon Item Enumeration and Chronology Modification	15
	Fix Routine preferred_doc_suppl_prio – Merge Bibliographic Records According to Supplier Priority	16
	Fix Routine fix_doc_assign_issn – Automatic Creation and Assignment of an ISSN	17
	Support for Additional Filters in the LKR Field	18
	Rosetta Loader (p-rosetta-01).....	21
	Show Hidden Characters.....	22
	Automatic Translation of Bibliographic Note Fields.....	22
	Create Additional Subject Headings from Authority (manage-46) ..	23
	Hot Key Activation for the Catalog Record Edit Menus	24
	New Generic Fix Doc Operation	25
	Support Added for External Expand Routines	26
	Configuring TYP Fields for Multiple Subfield Occurrences	27
	CAT Field Creation.....	28
Chapter 4	Circulation.....	29
	Hold/Photocopy Request Service Hours.....	30
	Photocopy Workflow Enhancements	31
	NCIP Temporary Item	44
	Circulation Request Refusal.....	44

	Cash Transaction Report (cash-02) Enhancements.....	47
	Hold Checks	48
	Booking Checks	51
	Requester Name Field for Hold/Photocopy Requests	52
	Cash Transaction Types (tab18.lng) Enhancements.....	58
	Renew Items on Loan (cir-08)	58
	E-Mail Validation Check	60
	Check for Similar Item On Loan.....	62
	Loan Session Enhancements	62
	Log for Patron Record Changes	63
	Cash Transactions Enhancements.....	65
	Enhancement to Print Only Hold Letters Without Printing Hold Slips in the Hold Shelf Report Service (cir-06).....	66
	Restricting Cash Limit Related Check Routines to Specific Cash Types.....	67
	Return During Loan in NCIP and SIP2	68
Chapter 5	General.....	71
	Performance Improvements	71
	Identifying PC Workstation with the Logon Name	74
	HTML in e-Mail.....	75
	Aleph Extended Topologies.....	75
	Retrieving Availability Information by Collection.....	76
	New Barcode Fonts to Support IE 9.....	77
	Aleph 21 and JBoss 5	77
Chapter 6	Indexing.....	79
	APAC Indexing.....	79
Chapter 7	Integrated ILL	83
	Patron Information in ILL Borrowing Request	83
	Multi-Item Borrowing Request.....	85
Chapter 8	Items	93
	Configuring the Statistic Field to be Mandatory.....	93
	Printing Labels for Multiple Items.....	94
	Update of Monograph Item Arrival Date	94

	Sort Items by Copy ID.....	95
	Running Multiple Simultaneous Instances of the Update Item Records Service (p-manage-62)	96
	Advanced General Retrieval Form (ret-adm-03)	96
	Notification When Item Changed to be Part of a Requested Group.	97
	Item Changes Alert.....	98
	Enhancement to the Item Call Number Duplication Check to Check for Duplicate Titles	98
	Item List Filter	99
Chapter 9	Serials	101
	Batch Service: Print Claim Letters/Report (serial-44)	101
	Order Information Tab in Serials Module.....	103
	Ordering Item Arrival and Label Printout by Copy ID	104
	SISAC Barcode	105
Chapter 10	Web OPAC	107
	Adding an OWN Field to a BIB Record Created as a Result of an ACQ Request	107
	Hold Request Confirmation Page Enhanced.....	108
	New Photocopy Request Confirmation Page.....	112
	Configuring Apache To Send Minimal Details.....	114
	Displaying Requested in the Due Date Column of the Holding Screen.....	115
	User Registration Form – Default Values.....	115
	Converting From Wide-Char to Standard ASCII	116
	My E-Shelf Enhancement	116
	Hold Request History Displayed in Library Card.....	118
	Server Side Scripts in the Aleph OPAC.....	121
Chapter 11	DLF API	123
	RENEW-PRE Added to tab_check_circ.....	123
	New Parameters for the Items DLF Service.....	124
	Hold Request History DLF API Service	124
Chapter 12	ARC.....	127
	Bibliographic Field Extraction (tab22_arc).....	127
	ETL – New Complete Incremental Mode	127

1

Introduction

This section includes:

- [Purpose of This Document](#) on page 7
- [Aleph 21 Documentation](#) on page 7

Purpose of This Document

This document lists the various enhancements and changes that are new for Aleph 21. You can find them listed under the following topics:

- [Acquisitions](#) on page 9
- [Cataloging](#) on page 13
- [Circulation](#) on page 29
- [General](#) on page 71
- [Indexing](#) on page 79
- [Integrated ILL](#) on page 83
- [Items](#) on page 93
- [Serials](#) on page 101
- [Web OPAC](#) on page 107
- [DLF API](#) on page 123
- [ARC](#) on page 127

Aleph 21 Documentation

The following is a list of the primary documentation for Aleph 21:

- *Aleph 21 Staff User's Guide*

- *Aleph 21 System Librarian's Guide*
- *Aleph 21 System Configuration Guide*
- *Aleph 21 Release Notes*

2

Acquisitions

This section includes:

- **Verde Order Counter** on page 9
- **Order Closure Upon Arrival Completion** on page 10
- **Renew Serial Encumbrance Based on Paid Amount** on page 10

Verde Order Counter

Purpose of Enhancement

A new order counter is available to be used when a new order record is created from Verde.

Description

It is possible to set up an order counter (from `Util/g/2`) to be used when a new order record is created from Verde. The `[adm_lib]/tab/verde_open_order` table, Col 10 contains up to 5 characters and matches the definition of the order counters in `util/g/2`. Only the first line is filled in. For example, if the column is set to `s` in `verde_open_order` of `XXX50`, the system uses the counter `last-order-no-s` of the `XXX50` library. If the column is empty or it has a value but the counter does not exist, the default counter `last-order-number` is used.

Reference to Technical Documentation

For more information, refer to the **Order Number Counters** chapter of the *Aleph 21 System Librarian's Guide-Acquisitions*.

Order Closure Upon Arrival Completion

Purpose of Enhancement

A new option has been added to control the automatic closure of monograph and standing orders after the order arrival is completed.

Description

To support this new option, a new tab100 flag, `CLS-ORDER-UPON-ARRIVE`, has been added to `./<adm_library>/tab/ tab100`.

Possible values: 0/1

0 – Default. After the completion of monograph or standing order arrival, the order status is not automatically changed to `Closed`. The automatic closure of the order occurs when the order is marked with the following three statuses:

- Completely Arrived
- Completely Invoiced
- Paid

1 – (New functionality). After the complete arrival of a monograph or standing order, the order status is automatically changed to `Closed`. This occurs regardless of the order's invoice information.

Reference to Technical Documentation

For more information, refer to the **Acquisitions-related Entries in tab100** section of the *Aleph 21 System Librarian's Guide-Acquisitions*.

Implementation Notes

Set the following line to `./xxx50/tab/tab100`:

```
CLS-ORDER-UPON-ARRIVE=1
```

Renew Serial Encumbrance Based on Paid Amount

Purpose of Enhancement

The Renew Order Encumbrances for Serials and Standing Orders (acq-06-b) service has been enhanced with a new field that enables the renewing of the encumbrance based on the invoice paid amount of the previous year.

Description

The `Renew Based on` field has been added to the Renew Order Encumbrances for Serials and Standing Orders (acq-06-b) batch service.

The new field allows you to choose how Aleph calculates the new year order's encumbrance amount.

- **Encumbrance Amount** – Same functionality as before the fix. The newly created encumbrance amount is based on the previous encumbrance amount plus the percent increase.
- **Paid Amount** – New functionality. The newly created encumbrance amount is based on the last year paid invoice amount and budget. If there are no paid amounts, Aleph renews the encumbrance based on the encumbered amount plus the percent increase.

A new column has been added to the batch output report that indicates whether the new encumbrance is based on the encumbrance amount or the invoice paid amount.

Reference to Technical Documentation

For more information, refer to the batch Help.

Implementation Notes

If localization is required, do the following:

- 1 Add the followings to the following files (after ORDER-UNIT control):

- `./alephe/pc_b_<lng>/p-acq-06-b.xml`
- `./alephe/pc_b_<lng>/p-acq-06.xml:`

```
<control>
  <radio>
    <argname>F11</argname>
    <label>Renew Based on</label>
    <source>explicit</source>
    <default>E</default>
    <ritem>
      <value>E</value>
      <display>Encumbrance Amount</display>
    </ritem>
    <ritem>
      <value>P</value>
      <display>Paid Amount</display>
    </ritem>
  </radio>
</control>
```

- 2 Add the following to `./form_<lng>/renew-order-encumbrance.xml` (after the `/z601-currency` field):

```
<xsl:call-template name="display-grid-gen">
  <xsl:with-param name="label" select="'Base Amount:'"/>
  <xsl:with-param name="value" select="./base-amount"/>
  <xsl:with-param name="header" select="$header"/>
</xsl:call-template>
```

- 3 Replace the following in `./form_<lng>/renew-order-encumbrance.xml`:

```
<xsl:with-param name="label" select="'Original sum enc:'"/>
```

With:

```
<xsl:with-param name="label" select="'Original sum enc/
inv:'"/>
```

- 4 Remove the following from `./form_<lng>/renew-order-encumbrance.xml` (if the same column exists twice):

```
<xsl:call-template name="display-grid-gen">
  <xsl:with-param name="label" select="'Credit/Debit:'"/>
  <xsl:with-param name="value" select="./z601-credit-
debit"/>
  <xsl:with-param name="header" select="$header"/>
</xsl:call-template>
```

- 5 Add the following to `./alephe/error_<lng>/p_acq_06`:

```
5005 0000 L Order type M(Monograph) can't work with "Paid
Amount".
```

3

Cataloging

This section includes:

- **Expand from Item Barcode** on page 14
- **Additional LKR Field Upon Item Enumeration and Chronology Modification** on page 15
- **Fix Routine preferred_doc_suppl_prio – Merge Bibliographic Records According to Supplier Priority** on page 16
- **Fix Routine fix_doc_assign_issn – Automatic Creation and Assignment of an ISSN** on page 17
- **Support for Additional Filters in the LKR Field** on page 18
- **Rosetta Loader (p-rosetta-01)** on page 21
- **Show Hidden Characters** on page 22
- **Automatic Translation of Bibliographic Note Fields** on page 22
- **Create Additional Subject Headings from Authority (manage-46)** on page 23
- **Hot Key Activation for the Catalog Record Edit Menus** on page 24
- **New Generic Fix Doc Operation** on page 25
- **Support Added for External Expand Routines** on page 26
- **Configuring TYP Fields for Multiple Subfield Occurrences** on page 27
- **CAT Field Creation** on page 28

Expand from Item Barcode

Purpose of Enhancement

The Expand Item Barcode option has been added to the Cataloging module. This option allows you to enter an item's barcode and transfer the item's information from the item record to the LKR subfields of the bibliographic record.

Description

To use the Expand Item Barcode option, select it from the Edit Actions menu or right-click the LKR field. Enter an item's barcode in the dialog box.

After you use the Expand Item Barcode option, the LKR subfields are filled in from the item's chronology and enumeration information, as follows:

Table 1. Expand from Item Barcode

LKR Subfield	Item Record
LKR\$\$v	item enumeration level a
LKR\$\$i	item enumeration level b
LKR\$\$p	item enumeration level c
LKR\$\$d	item enumeration level d
LKR\$\$e	item enumeration level e
LKR\$\$f	item enumeration level f
LKR\$\$g	item enumeration level g
LKR\$\$h	item enumeration level h
LKR\$\$y	item chronology level i
LKR\$\$j	item chronology level j
LKR\$\$w	item chronology level k
LKR\$\$o	item chronology level l
LKR\$\$q	item chronology level m

In addition, subfield 245\$\$a is filled in with the item's description.

NOTE:

Expand from Item Barcode does not affect other LKR subfields.

Note that the LKR filters are applied for all LKR subfields only if Z103X (the extended linkage mechanism between documents) is applied. If Z103X is not

applied, the following LKR subfields are not taken into account when linking the BIB record to specific items: d, e, f, g, h, w, o, and q.

Reference to Technical Documentation

For more information, refer to the **Table of Available Editing Functions** section of the *Aleph 21 Staff User's Guide-Cataloging*.

Implementation Notes

If localization is needed, add the following lines:

- `.\catalog\tab\lng\menu.dat:`
EXPAND_FROM_BARCODE Expand From Item Barcode
- `.\catalog\tab\lng\hint.dat:`
EXPAND_FROM_BARCODE Expand the LKR field From Item Barcode
- `.\catalog\tab\lng>window.dat:`
OpenArticleRecord.Title Expand From Item Barcode
OpenArticleRecord.Static.Title Enter Item Barcode:

Additional LKR Field Upon Item Enumeration and Chronology Modification

Purpose of Enhancement

This new development supports the adding of an additional LKR field to the BIB record when modifying the related item's enumeration and chronology information.

Description

The LKR field of type ANA is used to link bibliographic records in order to create a hierarchy between BIB records. This structure supports, for example, maintaining the following three levels of linking between bibliographic records:

- journal title
- issue title
- article title

The items are created for the journal title. The issue title and article title are linked to the journal title's items via the LKR ANA mechanism.

This mechanism has been enhanced to ensure that updating the item enumeration or chronology information updates the issue title and article title with an additional LKR field with the new item filters.

The addition of the new LKR field occurs when the item enumeration or chronology information is modified using one of the following sections of the Aleph interface:

- Item Form – Serial Level tab
- Cataloging module – Binding
- Batch service: Update Item Records (manage-62)

Reference to Technical Documentation

For configuration instructions and more information, see the **LKR Update Upon Item Enumeration and Chronology Modification** section of the *Aleph 21 System Librarian's Guide-Cataloging*.

Implementation Notes

- 1 Set PERIOIC-INEX message type to `i` in `tab_z105` of the ADM library.
- 2 In Column 3 (target library), set the BIB library in which to update the LKR field, for example, `./usm50/tab/tab_z105`

```
!      1      2  3      4      5      6      7      8      9      10
!!!!!!!!!!!!!!!!!!!!-!-!!!!-!!!!-!!!!-!!!!-!!!!-!!!!-!!!!-!!!!-
!!!!->
PERIODIC-INEX  i  USM01
```

- 3 In the `z105_library` (for example, `USR00`), re-activate `util E/11` (the message between libraries daemon).

Fix Routine `preferred_doc_suppl_prio` – Merge Bibliographic Records According to Supplier Priority

Purpose of Enhancement

A new preferred routine, `preferred_doc_suppl_prio`, determines which of two matching records is preferred.

Description

The `preferred_doc_suppl_prio` routine defined in `tab_preferred` consults the `data_supplier_priorities` table for the supplier code located in field 970\$v. The supplier with the lower number has the higher priority, so that 001 is highest possible priority and 999 is the lowest.

The `preferred_doc_suppl_prio` routine in `tab_preferred` also consults a weighing table (tag STA) when determining the supplier priority.

Reference to Technical Documentation

For more information on the `data_supplier_priorities` table, refer to the `data_supplier_priorities` section of the *Aleph 21 Configuration Guide*.

Implementation Notes

To use the new fix routine:

- 1 Add the following in `<BIB library>/tab/tab_preferred`:

```
!1                2                3
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!>
xxx              preferred_doc_suppl_prio  data_supplier_priorities
```

- 2 Add the new table `<BIB library>/tab/data_supplier_priorities`.

Fix Routine `fix_doc_assign_issn` – Automatic Creation and Assignment of an ISSN

Purpose of Enhancement

The `fix_doc_assign_issn` fix routine has been introduced to enable the automatic creation and assignment of an ISSN to a record.

Description

The routine creates and assigns ISSNs (022 \$a) and Centre codes (022\$2) to bibliographic records.

The `fix_doc_assign_issn` routine uses two parameters in column 3. The first parameter (\$\$2) contains the Centre code, which indicates the ISSN National Centre responsible for assigning the ISSN. The second parameter (limit) contains the upper limit of the ISSN counter.

Reference to Technical Documentation

For more information, see the `fix_doc_assign_issn` entry in the `tab_fix` section of the *Aleph 21 System Librarian's Guide-Cataloging*.

Implementation Notes

To use the new fix routine:

- 1 Add the following line in `./<library>/pc_tab/catalog/fix_doc.<lng>`:

```
ISSN N L Assign a new ISSN to the record
```

- 2 Add the following line in `$data_tab/tab_fix`:

```
ISSN fix_doc_assign_issn          $$2=9 limit=nnnnnnn  
where nnnnnn is the upper limit of issn range
```

- 3 Run `util G/2` and create the `create-issn` counter with the start point of the ISSN range.

Support for Additional Filters in the LKR Field

Purpose of Enhancement

Additional filters are now available in the `LKR` field.

Description

These filters are used to filter items using all the available levels of numeration and chronology. The following are the new `LKR` subfields:

- `$$d` – Fourth level of enumeration
- `$$e` – Fifth level of enumeration
- `$$f` – Sixth level of enumeration
- `$$g` – Alternative first level of enumeration
- `$$h` – Alternative second level of enumeration
- `$$j` – Second level of chronology
- `$$w` – Third level of chronology
- `$$o` – Forth level of chronology
- `$$q` – Alternative chronology

A new Oracle table, Z103X, has been introduced to support the new filters. This table is an expansion of the Z103 linking table and contains information on the new filters.

In order for the new filters to be applicable:

- Define the Z103X table in all relevant libraries (BIB and ADM libraries).
- Configure the `update_z103_lkr_extended` program in `tab_z103`.
- Run `manage-12` to re-create the links.

Reference to Technical Documentation

For more information, refer to the **LKR Field** section of the *Aleph 21 Staff User's Guide-Cataloging* and the **tab_z103** section of the *Aleph 21 Configuration Guide*.

Implementation Notes

1 Create the Z103X Oracle table:

- a Add the Z103X table to `file_list` in all libraries that have the Z103 table, for example: `./<BIB library>/file_list`:

```

!1      2      3      4      5
6
!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!
TAB z103x      100K      0K      ts0
IND z103x_id   100K      0K      ts1

```

- b Create the Z103X Oracle table in all libraries that have the Z103 table using `util/a/1/17`.

2 Activate the extended linking routine:

- a In `./<bib library>/tab/tab_z103` add the following line:

```

!      1      2
!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!->
update_z103_lkr_extended

```

- b In `./<adm library>/tab/tab_z103` add the following line:

```

!      1      2
!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!->
update_z103_lkr_extended

```

3 Run `manage_12` to recreate the Z103 table.

4 Modify the LKR form. Change `./<bib library>/pc_tab/catalog/lkr_xx.lng`:

a Change the following lines:

From:

```
^1^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
^2^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^&2
^3^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_3_
^4^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^5^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^6^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^7^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^8^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^9^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^10^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^11^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^12^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^13^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^14^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
```

To:

```
^1^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
^2^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^&2
^3^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_3_
^4^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^5^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^6^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^7^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^8^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^9^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^15^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^21^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^22^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^23^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^10^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^11^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^12^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^16^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^17^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^18^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^19^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^20^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^13^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
^14^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^_30_
```

b Replace the following line:

From:

```
F ablnmryvipks
```

To:

```
F ablnmryjwoqvipdefghks
```

- c** Add the following lines to the end of the file:

```
^15=Second level of chronology (j)
^16=Fourth level of enumeration (d)
^17=Fifth level of enumeration (e)
^18=Sixth level of enumeration (f)
^19=Alternative Volume link (g)
^20=Alternative Issue link (h)
^21=Third level of chronology (w)
^22=Fourth level of chronology (o)
^23=Alternative chronology (q)
```

- 5** Add the following to `./<bib library>/tab/check_doc_line`:

```
!AL:
! 2 - 4 5 6 7
!!-!!-!!!!-!!!!---!-!-!
AL XX LKR d 0 1
AL XX LKR e 0 1
AL XX LKR f 0 1
AL XX LKR g 0 1
AL XX LKR h 0 1
AL XX LKR w 0 1
AL XX LKR o 0 1
AL XX LKR q 0 1
AL XX LKR j 0 1
```

- 6** Update the GUI HTML Help for LKR tag from `demo setup $aleph_dev/usm01/pc_tab/catalog/html/lkr_xx_<lng>.html`
- 7** Restart the Update Doc Index job, using `util/e/2` and `util/e/1`

Rosetta Loader (p-rosetta-01)

Purpose of Enhancement

The `p-rosetta-01` service has been created to convert and load bibliographic records harvested from Rosetta to Aleph.

Description

The Aleph OAI harvester (`ue_13`) transfers data that has been provided from the Rosetta OAI provider to the `ue_13_(date).(seq number)_(set name)` file in the `$alephe_scratch` Aleph directory.

NOTE:

`ue_13` can be configured to activate the Aleph service `rosetta-01` automatically.

To obtain this new service, add `p-rosetta-01.xml` to the list of library batch services menu.

Reference to Technical Documentation

For more information, refer to the *Rosetta – Aleph Synchronization Guide* and the Batch service help file (`p-rosetta-01.pcbh`).

Show Hidden Characters

Purpose of Enhancement

The Show Hidden Characters option is now available from the Edit Text menu.

Description

This new option displays a bubble above the inserted text in the BIB record, showing the text with its hidden characters (including directionality characters) but excluding combined characters. Note that this functionality is available only when combined characters functionality is supported.

Reference to Technical Documentation

For more information, refer to the **Available Editing Functions** section of the *Aleph 21 Staff User's Guide-Cataloging* and the **Combining Diacritics** section of the *Aleph 21 System Librarian's Guide-Cataloging*.

Implementation Notes

The intended hotkey for this functionality is Alt+F2. You can change it to be the Alt key only in the `accel.dat` file.

Automatic Translation of Bibliographic Note Fields

Purpose of Enhancement

A new mechanism has been added to automatically translate bibliographic note fields (for example, from English to French). This development has been implanted for Aleph 20 as well.

Description

A new Aleph fix record routine, `fix_doc_notes`, has been developed to support the automatic translation of Bibliographic note fields.

This can be used, for example, to translate bibliographic English note fields to French note fields. This procedure invokes a new Aleph table, `tab_fix_notes`, which contains the translation of phrases.

Reference to Documentation

For more information, refer to the Automatic Translate of Bibliographic Note Fields section of the *Aleph 21 System Librarian Guide-Cataloging*.

Implementation Notes

To activate the automatic translation mechanism:

- 1 Define `fix_doc_notes` in the relevant instances of `tab_fix`.

The following is an example of `./usm01/tab/tab_fix`:

```
! 1                2                3
!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!>
INS2  fix_doc_notes
```

- 2 Set a list of phrases and their translation in `./<bib_+library>/tab/tab_fix_notes`.

Create Additional Subject Headings from Authority (manage-46)

Purpose of Enhancement

A new cataloging batch service has been introduced – Create Additional Subject Heading(s) from Authority (manage-46). (This development has been implemented for Aleph 20 as well.)

The service creates a translated subject heading within the bibliographic record based on authority matching indexes. This can be used, for example, to update bibliographic records with French RVM subject headings based on matching terms in the authority heading.

Description

The manage-46 service automatically creates 6XXX fields (RVM Subject Heading), using the authority record field 1xx or 7xx.

The service uses a new bibliographic library table: `tab_bib_aut_match`.

When the bibliographic record 6xx field finds a matching record in RVM (Authority database), the batch procedure copies the RVM record's French heading (1xx field) to the bibliographic record with tag 6XXx6, or the RVM record's English heading (7xx field) to the bibliographic record with tag 6XXx0 in order to have a translated subject heading within the bibliographic record.

Reference to Documentation

For more information, refer to the **Automatic Creation of 6XX Fields** section of the *Aleph 21 System Librarian's Guide-Cataloging*.

Implementation Notes

- 1 Set the relevant 6XX tags and AUT Index codes in `./bib_library/tab/tab_bib_aut_match`.

The following is an example of `./usm01/tab/tab_bib_aut_match`:

```
! 1      2      3      4      5
!!!!-!!!!!!!!!!!!-!!!!-!!!!-!!
650#0      USM10 LCS
650#2      USM10 MLC
650#7 aat  USM10 AAT
6###6      USM10
```

- 2 If localization is required, add the following lines to `./alephe/pc_b_lng/menu-catalog.xml`:

```
<item>
  <display>Create Additional Subject Heading(s) from
  Authority (manage-46)</display>
  <file>p-manage-46</file>
</item>
```

Hot Key Activation for the Catalog Record Edit Menus

Purpose of Enhancement

A new configuration option has been added to allow you to define the hot keys for Catalog Record Edit menus.

Description

Previously, the only way to open the Edit Text and Edit Action cataloging editor menus in the Cataloging module was to right click for the Edit actions menu or press shift+right click for the Edit text menu.

It is now possible to reverse the hot keys for these menus.

To support this, a new `catalog.ini` entry has been added to the Editor section:

```
[Editor]
RightClickMenu=
```

Possible values: `EditActions` or `EditText`

- **EditActions** – (default) Right click to activate the Edit Actions menu. Press shift+right click to activate the Edit Text menu.
- **EditText** – Right click to activate the Edit Text menu. Press shift+right click to activate the Edit Actions menu.

Reference to Technical Documentation

For more information, refer to the **Catalog.ini Settings** section of the *Aleph 21 System Librarian's Guide-Cataloging*.

Implementation Notes

Set the relevant value in the Editor section of the `catalog.ini` file:
`\Catalog\Tab\catalog.ini`

For example:

```
[Editor]
RightClickMenu=EditActions
```

New Generic Fix Doc Operation

Purpose of Enhancement

A new generic fix doc operation has been added. It replaces a string with another string using the user's defined delimiter between the source and the target string.

Description

The external expand routines can be written in any programming language and can be executed without being linked to Aleph. These expand programs are defined in `tab_expand` and can be used for extended bibliographic information.

Reference to Technical Documentation

For more information, refer to the `expand_routines` section of the *Aleph 21 System Librarian's Guide-Indexing*.

Implementation Notes

Add the expand routine to `/BIB library/tab/tab_expand`. For example:

```
! 1                2                3
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!>
WEB-FULL  expand_doc_245_add_subfield
```

NOTE:

The external program must reside in `$aleph_exe` and should not have an extension.

Configuring TYP Fields for Multiple Subfield Occurrences

Purpose of Enhancement

A new parameter has been added to take into account all subfield occurrences.

Description

It is now possible to create `TYP` fields for multiple subfield occurrences. Parameter `A (all)` has been added to col 5. of `tab_type_config`. This means that all occurrences of the subfield are checked.

Implementation Notes

To implement this functionality, set col 5 in `tab_type_config.lng` with the following characters: The first character is an `s` (for subfield), the second character is the subfield itself (`b`), and the third character is the subfield

occurrence (1-9,A). To enable the expand to check the match for multiple subfield occurrences, set the third character to A (all). For example:

```
! 1 2 3 4 5 6
7
!!!!-!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!-!!!!-!!!!-!!!!-
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!
TYP SHB 042## SbA MATCH
SHB
```

CAT Field Creation

Purpose of Enhancement

It is now possible to prevent the creation of the CAT field.

Description

Libraries whose records frequently change, might want to prevent the creation of CAT fields in order to reduce record size.

Reference to Documentation

For more information, refer to the **tab100** section of the *Aleph 21 Configuration Guide*.

Implementation Notes

In the library where you want to prevent CAT field creation, add the following to tab100:

```
CREATE-CAT-FIELD=Y
```

4

Circulation

This section includes:

- **Hold/Photocopy Request Service Hours** on page 30
- **Photocopy Workflow Enhancements** on page 31
- **NCIP Temporary Item** on page 44
- **Circulation Request Refusal** on page 44
- **Cash Transaction Report (cash-02) Enhancements** on page 47
- **Hold Checks** on page 48
- **Booking Checks** on page 51
- **Requester Name Field for Hold/Photocopy Requests** on page 52
- **Cash Transaction Types (tab18.lng) Enhancements** on page 58
- **Renew Items on Loan (cir-08)** on page 58
- **E-Mail Validation Check** on page 60
- **Check for Similar Item On Loan** on page 62
- **Loan Session Enhancements** on page 62
- **Log for Patron Record Changes** on page 63
- **Cash Transactions Enhancements** on page 65
- **Enhancement to Print Only Hold Letters Without Printing Hold Slips in the Hold Shelf Report Service (cir-06)** on page 66
- **Restricting Cash Limit Related Check Routines to Specific Cash Types** on page 67
- **Return During Loan in NCIP and SIP2** on page 68

Hold/Photocopy Request Service Hours

Purpose of Enhancement

A new configuration table, `tab_service_hour`, defines the desk availability for hold and photocopy requests.

Description

This table is set in the ADM library. Depending on the request check routines setup, the system consults the `tab_service_hour` table to determine if the selected pickup location can fill the request within the department's service hours.

In addition to the new `tab_service_hour` table, new check hold routines `check_hold_request_sh` and `check_photo_request_sh` have been developed to be set in `tab_hold_request` and `tab_photo_request` respectively.

These check routines are invoked after hold or photocopy requests are submitted via GUI, OPAC, and DLF. The request submission time (day+hour) is checked against the pickup location service hours that are set in `tab_service_hour`.

Upon setting those checks, `tab_service_hour` is consulted as part of the request validation flow.

Reference to Technical Documentation

For setup and additional information, see the **Request Service Hours Checks (`tab_service_hour`)** section of the *Aleph 21 System Librarian's Guide-Circulation*.

Implementation Notes

- 1 Add the following lines to `<ADM library>/tab/tab_hold_request`:

```
! 1                2
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
POST      check_hold_request_sh
CIRC      check_hold_request_sh
XML-CIRC  check_hold_request_sh
```

- 2 Add the following lines to `<ADM library>/tab/tab_photo_request`:

```
! 1                2
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
POST      check_photo_request_sh
CIRC      check_photo_request_sh
```

- 3 Add the new `<ADM library>/tab/tab_service_hour` table.

Photocopy Workflow Enhancements

The photocopy workflow in Aleph has been enhanced to allow better support of photocopy services. This section describes the following enhancements:

- **Photocopy Availability Checks** on page 31
- **Photocopy Request Forms in the Web OPAC** on page 33
- **OPAC Photocopy Submitting Form** on page 34
- **New Sort Option for Photocopy List** on page 35
- **Slip Printing Using Request daemon UE_06** on page 36
- **Photocopy Filled Form Enhancements** on page 36
- **Photocopy Shipping and Supply** on page 37
- **New Events for Photocopy Status Changes** on page 38
- **Photocopy Request History** on page 39
- **Additional Info Field** on page 40
- **Like Items in Photocopy Requests** on page 43

For more information, refer to the **Photocopy Requests** section of the *Aleph 21 Staff User's Guide-Circulation*.

Photocopy Availability Checks

Purpose of Enhancement

Two new check routines are now available in order to determine whether placing photocopy request is allowed.

Description

- `check_photo_request_d` – placing a photocopy request on unavailable items is not allowed.

An item is considered unavailable if at least one of the following is true:

- A photocopy request of this item exists for another patron
- The item is loaned to another patron

- A hold or booking request of this item exists

The `check_photo_request_d` routine retrieves a list of item statuses as a parameter. The photocopy request is not blocked, even if the item is unavailable, for items with the specified item status.

For example:

```
! 1                2                3
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!
PRE      check_photo_request_d      01,02
```

- `check_photo_request_e` – specific patron statuses are blocked from placing photocopy requests on specific item statuses.

This routine retrieves a list of item statuses and patron statuses as parameters. If the selected item has one of the given item statuses and the requesting patron has one of the given patron statuses, placing a photocopy request is not allowed.

For example:

```
! 1                2                3
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!
PRE      check_photo_request_e      ITEM=##;BOR=02,01
```

NOTE:

To block all item statuses from placing a photocopy request, set `##` as a parameter in the `tab_photo_request` table for `check_photo_request_e`.

Implementation Notes

- 1 Add the new checks in `./<ADM library>/tab/tab_photo_request`, for example:

```
! 1                2                3
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
PRE      check_photo_request_d      01,03
PRE      check_photo_request_e      ITEM=01,44;BOR=01,04
```

- 2 If localization is required:

- a Add the following messages to `$alephe_error_lng/check_photo_request`:

```
1250 L Item is not available for photocopy request.
1251 L Request cannot be placed on an already reserved item.
1260 L Patron cannot place a photocopy request for this item
status.
```


b Add the following messages to `$alephe_error_lng/pc_cir_c0426`:

```
1250 0002 L Item is not available for photocopy request.  
1251 0002 L Request cannot be placed on an already reserved  
item.  
1260 0002 L Patron cannot place a photocopy request for this  
item status.
```

c Add the following messages to `$alephe_error_lng/pc_cir_c0420`:

```
1250 0004 L Item is not available for photocopy request.  
1251 0004 L Request cannot be placed on an already reserved  
item.  
1260 0004 L Patron cannot place a photocopy request for this  
item status.
```

Photocopy Request Forms in the Web OPAC

Purpose of Enhancement

It is now possible to configure the system to use different photocopy request forms when the patron places a request through the Web OPAC.

Description

A new table – `tab_photo_request_form` – has been introduced in the Administrative library. This table sets the option to configure the system to use different photocopy request forms when a patron places a request through the Web OPAC.

When a request is placed, the system looks for a matching line in the table. The correct line is selected according to item and patron information. A photocopy form with the configured suffix is displayed.

Reference to Technical Documentation

For more information, refer to the **Different Photocopy Request Forms** section of the *Aleph 21 System Librarian's Guide-Circulation*.

Implementation Notes

- 1 Configure the `ADM library/tab/tab_photo_request_form` according to your library's policy.
- 2 Add the following OPAC photocopy forms with the suffix you configured in `tab_photo_request_form`:
 - `item-photo-request-body-<suffix>`
 - `item-phot-request-head-<suffix>`

- `item-photo-request-tail-<suffix>`
- `item-photo-request-tail-an-<suffix>`

OPAC Photocopy Submitting Form

Purpose of Enhancement

The photocopy request form in OPAC has been enhanced to fill the `Author of Article`, `Title of Article`, and `Pages` fields with bibliographic information from the Article document. (The `TYP` field of the document has subfield `$$aAN`). These fields are read only.

Description

When using the special photocopy form `alephe/www_f_lng/item-photo-request-tail-an`, if the requested document is an article (the document has a `TYP` field with subfield `$$aAN`), the photocopy request record is displayed with information from the document.

In addition, the `Z38-NOTE-2` field of the photocopy request is populated with the value in field `001`.

Reference to Technical Documentation

For more information, refer to the **Submitting a Photocopy Form** section of the *Aleph 21 System Librarian's Guide-Web OPAC*.

Implementation Notes

- 1 Add in `$alephe_root/www_f_lng/item-photo-request-head`:

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

- 2 Add the `$alephe_root/www_f_lng/item-photo-request-tail-an` file.
- 3 To identify the article documents, add `$$aAN` to the `TYP` field of the document. Configure `./<bib library>/tab/tab_expand` and `./<bib library>/tab/tab_type_config.eng`:

```
!      1              2              3
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!>
WEB-FULL  expand_doc_type              tab_type_config.eng,Y
```

- 4 Add the following in `$alephe_root/www_server.conf`:

```
www_photo_article_prefill          Y
```

New Sort Option for Photocopy List

Purpose of Enhancement

It is now possible to sort the list of photocopy requests based on various sort types.

Description

The sort is based on the following sort types:

- Pickup Location
- Sub library
- Pickup Location / Sub library
- Pickup Location / Status
- Sub library / Status
- Pickup Location / Sub library / Status
- Pickup Location / Status / Sub library

In order to use these sort options, you must configure the new `tab_z38_sort` table.

In addition, two new columns have been added to the photocopy requests list:

- item's sub library (Z30-SUB-LIBRARY)
- request's pickup location (Z38-PICKUP-LOCATION)

Implementation Notes

- 1 Add the following lines to `./<ADM library>/tab/pc_tab_exp_field.lng`:

PHOTO-SORT-TYPE	L Photocopy No	PHOTO-1
PHOTO-SORT-TYPE	L Sublib/Req name/Number	PHOTO-2
PHOTO-SORT-TYPE	L Pickup/Sublib/Req name/number	PHOTO-3
PHOTO-SORT-TYPE	L Pickup/Status/Req name/number	PHOTO-4
PHOTO-SORT-TYPE	L Sublib/Status/Req name/number	PHOTO-5
PHOTO-SORT-TYPE	L Pickup/Sublib/Status/Req name/number	PHOTO-6
PHOTO-SORT-TYPE	L Pickup/Status/Sublib/req name/number	PHOTO-7
PHOTO-SORT-TYPE	L Pickup/Req name/number	PHOTO-8

- 2 Add the new `./<ADM library>/tab/tab_z38_sort` table.

Slip Printing Using Request daemon UE_06

Purpose of Enhancement

A new table – `tab_ue_pickups` – has been created in the Administrative library.

Description

The `tab_ue_pickups` table lists the photocopy pickup locations that are handled by the UE_06 request daemon. Photocopy pickup locations that are not listed in the table are skipped by the request daemon.

The table structure is as follows:

```
! COL 1. 5; ALPHA_NUM, UPPER; #;
! Pickup location;
```

NOTE:

If the table does not exist, all pickup locations are handled.

Implementation Notes

Add the required information to `./<ADM library>/tab/tab_ue_pickups`

Photocopy Filled Form Enhancements

Purpose of Enhancement

The Photocopy Filled form has been enhanced.

Description

The following enhancements have been done for the Photocopy Filled form:

- The paper size field is now a drop down menu with a list of possible values. It is also possible to enter a value that is not defined in the list.
- It is now possible to enter up to 6 paper size and number of pages combinations in the photocopy filled form. Cash transaction are created for each combination.
- A new flag has been introduced in the `./adm-library/tab/tab100` table that indicates if the request should be removed after the Filled action or should be kept with Filled status.

If `KEEP-FILLED-PHOTO` is set to `N`, the request is removed; otherwise, it is kept with Filled status.

Implementation Notes

- 1 Add paper sizes to `./<adm library>/tab/pc_tab_exp_field.lng`. For example:

```
!      1      2      3
4
!!!!!!!!!!!!!!!!!!!!!!-----!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!>
PHOTO-PAPER-SIZE      L A4 form
A4
```

- 2 Add paper sizes to `./<adm library>/tab/tab18.lng`.
- 3 Add the following line to `./<adm-library>/tab/TAB100:`

```
KEEP-FILLED-PHOTO=Y
```

Photocopy Shipping and Supply

Purpose of Enhancement

Two new buttons have been introduced in the photocopy requests list under the Patron tab.

Description

The buttons are as follows:

- **Shipping** – (enabled also for multi-photocopy requests selection). When you click the **Shipping** button the following occurs:
 - Cash transactions for photocopy shipment are created.
 - All related cash transactions are marked as transferred to vendor.
 - An invoice of all related cash transactions is printed.

After the invoice is printed, the status of the requests is changed to `Supplied`, and the requests are moved to the `z38h Photocopy Requests History` table. (See also [Photocopy Request History](#) on page 39).

- **Supply** – When you click the **Supply** button, the request is moved to the `z38h Photocopy Requests History` table with `Supplied` status. (See also [Photocopy Request History](#) on page 39).

Implementation Notes

- 1 Add a shipment charge to <ADM library>/tab/tab18.lng, for example:

```
0056 ##### ## ## Y 10.00          Photocopy package
shipment
```

- 2 Create a new sequence in the photo-shipment-id by util g/2 ADM library.
- 3 Add the following line to alephe/tab/user_function.lng

```
REQUESTS          L Circulation Requests          PHOTO-
SHIPPING          L Photocopy request - Ship
```

- 4 Add the following lines to <ADM library>/tab/tab_z31_sort:

```
CASH-9           A 08
CASH-9-D         D 08
```

- 5 Add the following lines to <ADM library>/tab/pc_tab_exp_field.lng

```
CASH-SORT-TYPE   L Transfer Number/Transaction Type
CASH-9
CASH-SORT-TYPE   L Transfer Number/Transaction
Type (Descending) CASH-9-D
```

- 6 Add the following fields to <BIB library>/tab/pc_tab_col.lng - PC_CASH_SUMMARY:

```
PC_CASH_SUMMARY L Transfer Number    20 000 04 C04
Transfer Number
PC_CASH_SUMMARY L Photocopy          21 000 04
C04              Photocopy
```

- 7 Add a new XSL file – <BIB library>/form_lng/cash-ship-invoice.xsl
- 8 Add the following lines to circ\tab\lng>window.dat

```
UserPhoto.Btn.Shipping          &Shipping
UserPhoto.Btn.Supply            S&upply
```

New Events for Photocopy Status Changes

Purpose of Enhancement

New event types have been introduced to support the recording of photocopy status changes in the z35 events table.

Description

The new event types are as follows:

- Event type 83 - Photocopy status changed to `Waiting`
- Event type 84 - Photocopy status changed to `Filled`
- Event type 85 - Photocopy status changed to `Supplied`
- Event type 86 - Photocopy status changed to `Refused`
- Event type 87 - Photocopy status changed to `Deleted`

In order to support the creation of the above events, configure the `./adm-library/tab/tab_events.lng` table to activate the event creation.

Implementation Notes

Add the following lines to `./<ADM library>/tab/tab_events.lng`:

```
83 Y Photocopy status changed to "Waiting"
84 Y Photocopy status changed to "Filled"
85 Y Photocopy status changed to "Supplied"
86 Y Photocopy status changed to "Refused"
87 Y Photocopy status changed to "Deleted"
```

Photocopy Request History

Purpose of Enhancement

A new table – `Z38H Photocopy Request History` – has been introduced.

Description

This table behaves in a similar manner to `Z36H Loan History` and `Z37H Hold Request History`.

When a photocopy request is supplied or deleted, it is moved to the `Photocopy Requests History` table.

To view requests in the `Photocopy Requests History` table, click the **Historical Photocopy Requests** node from the Patron tab.

Note that in order to use the `Z38H` table, the `CREATE-Z38H` variable must be set to `Y` in the `tab100` table.

Implementation Notes

- 1 Add the following section to ./<bib_library>/tab/pc_tab_col.<lng>:

```
!*
!*  Circulation - Patron Photo Request History List
!*
PC_CIR_B_PHOTOH_LIST L Doc No.          01 000 04 C04
Document No
PC_CIR_B_PHOTOH_LIST L Bibliographic Info 02 050 04 C04
BIB Doc Heading
PC_CIR_B_PHOTOH_LIST L Date            03 025 04 C04
Open Date
PC_CIR_B_PHOTOH_LIST L No.            04 010 04 C04
Photo #
PC_CIR_B_PHOTOH_LIST L Status          05 015 04 C01 04
C06 Status
```

- 2 If localization is required add the pc_cir_c0483 to /alephe/error_lng file:

```
!!!!-!!!!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!>
0011 0000 L Unable to find photocopy request
0021 0000 L No photocopy requests
0022 0000 L Starting point beyond end of list

0050 0000 L In process
0051 0000 L Waiting in queue
0052 0000 L Unrecognized status
0053 0000 L History
0054 0000 L Filled
0055 0000 L Supplied
0056 0000 L Refused
0057 0000 L Deleted
```

- 3 Add the following line to ./<adm-library>/tab/TAB100:

```
CREATE-Z38H=Y
```

Additional Info Field

Purpose of Enhancement

The Additional Info field has been added to the Photocopy request form.

Description

The new field can be used by patrons in the photocopy request form of the Web OPAC and by staff to add a free text note of up to 600 characters.

Reference to Technical Documentation

For more information, refer to the **Photocopy Requests** section of the *Aleph 21 Staff User's Guide-Circulation*.

Implementation Notes

The following items are relevant for the configuration of this new feature:

- The Additional Info field is set in PC file: `./circ/tab/lng/window.dat`:

```
ItemCreatePhoto.Static.AddlInfo      Additional info:
UtilCreatePhoto.Static.AddlInfo      Additional info:
ItemPhotoExpand.Static.AddlInfo      Additional info:
UserPhotoExpand.Static.AddlInfo      Additional info:
UtilUpdatePhoto.Static.AddlInfo      Additional info:
```

- Make the following changes to the following Web OPAC configuration files:

- `item-photo-request-head`

Change:

```
document.form1.elements[i].name == "note2")  {
```

To:

```
document.form1.elements[i].name == "note2" ||
document.form1.elements[i].name == "addlinfo")  {
```

- `item-photo-request-tail`

After:

```
<tr>
  <td class=td2>Note 2 </td>
  <td class=td2> </td>
  <td class=td2>
    <input size=30 maxlength=100 name="note2"
value="$3600"></td>
</tr>
```

Add:

```
<tr>
  <td class=td2>Additional Info </td>
  <td class=td2> </td>
  <td class=td2>
    <textarea name="addlinfo" cols=60 rows=10>$3700</
textarea></td>
</tr>
```

■ bor-photo-exp

After:

```
<tr>
  <td class=td1 id=bold align=left>Location 2:</td>
  <td class=td1 align=left>&nbsp;$1300</td>
</tr>
```

Add:

```
<tr>
  <td class=td1 id=bold align=left>Additional Information:</td>
  <td class=td1 align=left>&nbsp;$1400</td>
</tr>
```

■ item-photo-request-tail-an

After:

```
<tr>
  <td class=td2>Note </td>
  <td class=td2> </td>
  <td class=td2>
    <input size=30 maxlength=50 name="note"
value="$3300"></td>
</tr>
```

Add:

```
<tr>
  <td class=td2>Additional Info </td>
  <td class=td2> </td>
  <td class=td2>
    <textarea name="addlinfo" cols=60 rows=10>$3700</
textarea></td>
</tr>
```

■ item-photo-request-details

After:

```
<input type=hidden name="note2" value="$4600">
```

Add:

```
<input type=hidden name="addlinfo" value="$4700">
```

After:

```
<tr>
  <td class="td2">Second Note </td>
  <td class="td2">$4600</td>
</tr>
```

Add:

```
<tr>
  <td class="td2">Additional Info </td>
  <td class="td2" style="word-wrap: break-word"
width="1000">$4700</td>
</tr>
```

Change:

```
<a href="&server_f?func=item-photo-
request&doc_library=$0100&adm_doc_number=$0200&item_sequence=
$0300&year=$2500&volume=$2600&sub_library=$2700&type=$2800&no
_loaned=$2900&start_rec_key=$3000&end_rec_key=$3100&bib_doc_n
um=$0600&bib_library=$0500&source_form=PHOTO_DETAILS&author=$
4200&title=$4300&pages=$4400&note1=$4500&note2=$4600&pickup=$
4100" >
```

To:

```
<a href="&server_f?func=item-photo-
request&doc_library=$0100&adm_doc_number=$0200&item_sequence=
$0300&year=$2500&volume=$2600&sub_library=$2700&type=$2800&no
_loaned=$2900&start_rec_key=$3000&end_rec_key=$3100&bib_doc_n
um=$0600&bib_library=$0500&source_form=PHOTO_DETAILS&author=$
4200&title=$4300&pages=$4400&note1=$4500&note2=$4600&pickup=$
4100&addlinfo=$4700" >
```

Like Items in Photocopy Requests

Purpose of Enhancement

A third level of chronology is now taken into account for item comparison for photocopy requests.

Description

A third level of chronology (`CHRONOLOGICAL K`) is now used as a filter for photocopy requests. Only items that have the same value in the `CHRONOLOGICAL K` field as the requested item are considered as like items when fulfilling photocopy requests.

NCIP Temporary Item

Purpose of Enhancement

It is now possible to create a temporary Item with the external barcode of an item without attaching the prefix that is taken from the agency code.

Description

When a temporary item is created, Aleph creates a unique barcode. The barcode should be attached to the item with which circulation activities can be performed. If, for some reason, the temporary item barcode is not attached to the item, the library can still perform all circulation activities, such as retrieval, by using the item's call number.

Reference to Technical Documentation

For more information, see the **Creation and Retrieval of a Temporary Item** section of the *How to Set Up and Use the NCIP Server* document.

Circulation Request Refusal

Purpose of Enhancement

It is now possible to refuse hold, photocopy, and booking requests.

Description

A new option has been added to the Requests menu in the Circulation module – Print letter - refusal.

This option enables the printing of a refusal letter, including the refusal reason. The request is moved to the History log with `Refused` status. A Circulation Logger record (Z309) is created, including the refusal reason.

Reference to Technical Documentation

For more information, see the **Hold Request Refused, Photocopy Request Refused, and Refusing Booking Request** sections of the *Aleph 21 Staff User's Guide-Circulation* and the **Request Refusal** section of the *Aleph 21 System Librarian's Guide-Circulation*.

Implementation Notes

- 1 Add the following forms to the `./<Bib Lib>/form_lng` directory:

```
hold-refused-letter-s.xml
photocopy-refused-letter-s.xml
booking-refused-letter-s.xml
```

- 2 Add the following expand screens to the `./<Adm Lib>/pc_display_lng` directory:

```
booking-request-info
booking-request-info.trn
```

- 3 Add the following lines to the `./<Bib Lib>tab/pc_tab_col.lng` file:

```
PC_CIR_LOGGER          L Request Number      11 000 01 C04          Z309-REQUEST-NUMBER
PC_COM_HOLD_H_LIST    L Status              12 010 04 C04          Request Status
PC_CIR_REF_HOLD_LIST  L Patron Name         01 030 04 C04          username
PC_CIR_REF_HOLD_LIST  L Patron ID           11 020 04 C04          Patron ID
PC_CIR_REF_HOLD_LIST  L Req Date            02 020 04 C04          request date
PC_CIR_REF_HOLD_LIST  L Pickup              03 020 04 C04          pickup location
PC_CIR_REF_HOLD_LIST  L Status              04 025 04 C01 04 C06 status
PC_CIR_REF_HOLD_LIST  L Note 1              05 000 04 C04          note 1
PC_CIR_REF_HOLD_LIST  L Note 2              06 000 04 C04          note 2
PC_CIR_REF_HOLD_LIST  L Priority             07 010 04 C04          priority
PC_CIR_REF_HOLD_LIST  L Proxy               08 005 it map          Proxy
PC_CIR_REF_HOLD_LIST  L Proxy ID            09 000 04 C04          Proxy ID
PC_CIR_REF_HOLD_LIST  L Proxy Name          10 000 04 C04          Proxy Name
PC_CIR_REF_HOLD_LIST  L End Req             12 015 04 C04          End Request Date
PC_CIR_REF_HOLD_LIST  L Hold Date           13 020 06 C06          Hold Date
PC_CIR_REF_BOOK_LIST  L Patron Name         01 030 04 C04          username
PC_CIR_REF_BOOK_LIST  L Patron ID           11 020 04 C04          Patron ID
PC_CIR_REF_BOOK_LIST  L Req Date            02 020 04 C04          request date
PC_CIR_REF_BOOK_LIST  L Pickup              03 020 04 C04          Pickup Location
PC_CIR_REF_BOOK_LIST  L Open Date           14 000 04 C04          Open Date
PC_CIR_REF_BOOK_LIST  L Status              04 025 04 C01 04 C06 status
PC_CIR_REF_BOOK_LIST  L Note 1              05 000 04 C04          note 1
PC_CIR_REF_BOOK_LIST  L Note 2              06 000 04 C04          note 2
PC_CIR_REF_BOOK_LIST  L Booking Start Date  15 000 04 C04          Booking Start Date
PC_CIR_REF_BOOK_LIST  L Booking End Date    16 000 04 C04          Booking End Date
PC_CIR_REF_BOOK_LIST  L Booking Start Hour  17 000 04 C04          Booking Start Hour
PC_CIR_REF_BOOK_LIST  L Booking End Hour    18 000 04 C04          Booking End Hour
PC_CIR_REF_BOOK_LIST  L Priority             07 010 04 C04          priority
PC_CIR_REF_BOOK_LIST  L Proxy               08 005 it map          Proxy
PC_CIR_REF_BOOK_LIST  L Proxy ID            09 000 04 C04          Proxy ID
PC_CIR_REF_BOOK_LIST  L Proxy Name          10 000 04 C04          Proxy Name
```

- 4 Add the following lines to the `./<Adm Lib>/tab/tab_circ_log.lng` file:

```
64 Y Y LRefusal Hold Request note
65 Y Y LRefusal Photocopy Request note
66 Y Y LRefusal Booking Request note
```

- 5 Add the following lines to the GUI configuration file under `Circ/Tab/Lng/hint.dat`:

```
REFUSE_HOLD_BARCODE      Print Letter - Hold Request Refused
REFUSE_PHOTO              Print Letter - Photocopy Request Refused
REFUSE_BOOKING            Print Letter - Booking Request Refused
```

- 6 Add the following lines to the GUI configuration file under `Circ/Tab/Lng/menu.dat`:

```
REFUSE_HOLD_BARCODE      Print Letter - Hold Request &Refused
REFUSE_PHOTO              Print Letter - Photocopy Request &Refused
REFUSE_BOOKING            Print Letter - Booking Request &Refused
```

- 7 Add the following lines to the GUI configuration file under `Circ/Tab/Lng/message.dat`:

```
ConfirmHoldRefusal      Confirm Refusal      YN  2  Q  Are you
sure you want to refuse the hold request?
ConfirmPhotoRefusal     Confirm Refusal      YN  2  Q  Are you
sure you want to refuse the Photocopy request?
ConfirmBookingRefusal   Confirm Refusal      YN  2  Q  Are you
sure you want to refuse the booking request?
TitleReqRefusal         Title Deletion        O    A  This hold
Request is attached to a Title Request.\nThe title Request
will be deleted.
TitleReqRefNonActive    Title Deletion        O    A  Cannot
delete a non active stop of Title Request.
UtilBookingListEmpty    Utils Booking List    O    E  No booking
requests found for item: %s
PhotoRequestFilled      Photocopy Refusal     O    E  Unable to
Refuse Filled request.
```

- 8 Add the following lines to the GUI configuration file under `Circ/Tab/Lng/window.dat`:

```

RefuseHold.Title          Print Letter - Hold Request Refused
RefuseHold.Static.Title  Enter Item Barcode:
RefusePhoto.Title        Print Letter - Photocopy Request Refused
RefusePhoto.Static.Title Enter Photocopy Request No:
RefuseBooking.Title      Print Letter - Booking Request Refused
RefuseBooking.Static.Title Enter Item Barcode:
RefuseHoldList.Title     Refuse Hold Request by Barcode - %s
RefuseHoldList.Static.Doc Description:
RefuseHoldList.Btn.Close &Close
RefuseHoldList.Btn.Expand &Expand
RefuseHoldList.Btn.Refusal &Refusal
RefuseHoldList.Btn.Print &Print
RefuseBookingList.Title  Refuse Booking Request by Barcode - %s
RefuseBookingList.Static.Doc Description:
RefuseBookingList.Btn.Close &Close
RefuseBookingList.Btn.Expand &Expand
RefuseBookingList.Btn.Refusal &Refusal
RefuseBookingList.Btn.Print &Print
RefusePhotoNote.Title    Refuse Photocopy Request - %s - Note for Patron
RefusePhotoNote.Static.Title Patron Note:
RefuseHoldNote.Title     Refuse Hold Request - %s - Note for Patron
RefuseHoldNote.Static.Title Patron Note:
RefuseBookingNote.Title  Refuse Booking Request - %s - Note for Patron
RefuseBookingNote.Static.Title Patron Note:
ExpandBookingRequest.Title Booking Request Details

```

Cash Transaction Report (cash-02) Enhancements

Purpose of Enhancement

New filters have been added to the Cash Transaction Report (cash-02) service.

Description

The new filters are `transfer date from` and `transfer date to`.

These filters can be used to print reports of all cash transactions that have been transferred to the vendor between a specific range of dates. Only cash transactions that have been transferred to an external vendor between the `transfer date from` and the `transfer date to` are included in the report.

In addition, `Special report format - 15` is now introduced for the Cash Transaction Report (cash-02) service. This report includes patron information on each line (and not only in first line of each new patron).

Reference to Technical Documentation

For more information, refer to the **Cash Transaction Report** section of the *Aleph 21 Staff User's Guide-Circulation* and the **Cash Reports** section of the *Aleph 21 System Librarian's Guide-Circulation*.

Implementation Notes

To add the new filters:

- 1 Add the following lines to `./aleph/pc_b_lng/p-cash-02.xml`:

```
<date_time>
    <argname>F08</argname>
    <label>Transfer Date (From)</label>
</date_time>
</control>
<control>
    <date_time>
        <argname>F09</argname>
        <label>Transfer Date (To)</label>
    </date_time>
</control>
```

- 2 Add the following lines to `./aleph/pc_b_lng/p-cash-02.xml`:

```
<!-- Param:   P-TRANS-DATE-FROM           9(8) .
F08 -->
<!-- Param:   P-TRANS-DATE-TO           9(8) .
F09 -->
```

To add format 15:

Add the following lines to `./aleph/pc_b_lng/p-cash-02.xml`, as part of the Report Format options:

```
<cbitem>
    <value>15</value>
    <display> Include all information in
every line </display>
</cbitem>
```

Hold Checks

Purpose of Enhancement

New check routines have been introduced in order to determine whether placing hold requests are allowed.

Description

- `check_hold_request_d1` – Hold requests are allowed only for items that are available.

An item is considered unavailable if at least one of the following is true:

- A photocopy request of this item exists
- The item is loaned
- A hold or booking request of this item exists

Item statuses that are excluded from this check are configured as parameters. (For an example, see the implementation notes).

- `check_hold_request_h_1` – checks the maximum number of hold requests defined in tab 16 column 13. This routine is similar to `check_hold_request_h`, but also includes in the check existing loans that were originally hold requests.
- `check_hold_request_q_1` – checks the maximum number of hold requests defined in tab 16 column 13 for the sub-library (line 99). This routine is similar to `check_hold_request_q`, but also includes in the check existing loans that were originally hold requests.
- `check_hold_request_t_1` – checks the maximum number of hold requests defined in tab 16 column 13 for the ADM library. This routine is similar to `check_hold_request_t`, but also includes in the check existing loans that were originally hold requests.
- `check_hold_request_x_1` – checks the maximum number of hold requests defined in tab 16 column 13 for the sub-libraries group. This routine is similar to `check_hold_request_x`, but also includes in the check existing loans that were originally hold requests.
- `check_hold_request_1_a` – checks patron global blocks. The error message includes block codes and descriptions from the `tab_delinq.lng` table.
- `check_hold_request_1_b` – checks patron local blocks. The error message includes block codes and descriptions from the `tab/tab_delinq.lng` table.
- `check_hold_request_1_d` – checks if patron has overdue loans. A loan is considered overdue if the due date has passed and `n` or more notices have been sent, as defined in the library's `tab/tab100 OVERDUE-LETTER-NO` directory.
- `check_hold_request_1_e` – checks the patron block date. (`z305_end_block_date`)
- `check_hold_request_4_a` – checks if the patron has exceeded the loan limit for the item status (tab16 column 12).
- `check_hold_request_4_b` – checks if the patron has exceeded the loan limit for the sublibrary (tab16 column 12).

- `check_hold_request_4_c` – checks if the patron has exceeded the loan limit for the ADM library (tab16 column 12).
- `check_hold_request_4_d` – checks if the patron has exceeded the loan limit for the sublibrary group (tab16 column 12).
- `check_hold_request_5` – checks if the patron has exceeded the cash limit allowed in `z305_cash_limit` - for both O(pen) and T(ransferred) transactions.
- `check_hold_request_5_b` – checks if the patron has exceeded the cash limit allowed only for O(pen) transactions.
- `check_hold_request_9_a` – checks the limits defined in `tab/tab_block_circ` (maximum number of overdue loans, maximum number of recalled overdue loans, and maximum cash owing).
- `check_hold_request_9_b` – checks the limits defined in `tab/tab_block_circ` as `9_a` at the item sublibrary level.
- `check_hold_request_10_a` – checks as `9_a` but counts all types of loans – active, lost, and claimed return.
- `check_hold_request_10_b` – checks as `9_b` but counts all types of loans – active, lost, and claimed return.
- `check_hold_request_h_2` – checks if the maximum number of hold requests reached. The maximum number is received as a parameter.

Reference to Technical Documentation

For more information, refer to the `tab_hold_request` section of the *Aleph 21 Configuration Guide*.

Implementation Notes

Add the new checks to `/adm library/tab/tab_hold_request`. For example:

```

! 1                2                3
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!
POST      check_hold_request_d1                01,02
POST      check_hold_request_h_1
POST      check_hold_request_q_1
POST      check_hold_request_t_1
POST      check_hold_request_x_1
POST      check_hold_request_1_a
POST      check_hold_request_1_b
POST      check_hold_request_1_d
POST      check_hold_request_1_e
POST      check_hold_request_4_a
POST      check_hold_request_4_b
POST      check_hold_request_4_c
POST      check_hold_request_4_d
POST      check_hold_request_5
POST      check_hold_request_5_b
POST      check_hold_request_9_a
POST      check_hold_request_9_b
POST      check_hold_request_10_a
POST      check_hold_request_10_b
POST      check_hold_request_h_2                3

```

Booking Checks

Purpose of Enhancement

New check routines have been introduced in order to determine whether placing booking request is allowed.

Description

- `check_booking_request_d` – Booking requests are allowed only for items that are available.

An item is considered unavailable if at least one of the following is true:

- A photocopy request of this item exists
- The item is loaned
- A hold or booking request of this item exists

Item statuses that are excluded from this check are configured as parameters. (For an example, see the implementation notes).

- `check_booking_request_h_1` – checks the maximum number of booking requests defined in tab 16 column 26. This routine is similar to `check_booking_request_h`, but also includes in the check existing loans that were originally booking requests.
- `check_booking_request_q_1` – checks the maximum number of booking requests defined in tab 16 column 26 for the sub-library (line 99). This routine is similar to `check_booking_request_q`, but also includes in the check existing loans that were originally booking requests.
- `check_booking_request_t_1` – checks the maximum number of booking requests defined in tab 16 column 26 for the ADM library. This routine is similar to `check_booking_request_t`, but also includes in the check existing loans that were originally booking requests.
- `check_booking_request_x_1` – checks the maximum number of booking requests defined in tab 16 column 26 for the sub-libraries group. This routine is similar to `check_booking_request_x`, but also includes in the check existing loans that were originally booking requests.

Reference to Technical Documentation

For more information, refer to the `tab_hold_request` section of the *Aleph 21 Configuration Guide*.

Implementation Notes

Add the new checks to `/adm library/tab/tab_hold_request`. For example:

```
! 1                2                3
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!
!!!!!!!!!!!!
BK-POST    check_booking_request_h_1
BK-POST    check_booking_request_q_1
BK-POST    check_booking_request_t_1
BK-POST    check_booking_request_x_1
```

Requester Name Field for Hold/Photocopy Requests

Purpose of Enhancement

It is now possible to store requester name information for hold and photocopy requests.

Description

Subscribed libraries may request material from a central library on behalf of their patrons. These libraries are identified in the central library, but their patrons are not. When a patron of the subscribed library wants to request material from the central library, the library staff creates a request using the library's account in the OPAC. The request is written in the borrowing library's account.

It is now possible to keep track of the requesting library's patron information. In addition, this information is written for related cash transactions and loan records. This information is written in the requester name field of the hold or photocopy request.

A new requester name field has been added to the hold request (Z37) and photocopy request (Z38). The requester name information can be viewed in the GUI / OPAC lists and can be used for sorting purposes.

Reference to Documentation

For more information, refer to the **Staff Request** section of the *Aleph 21 Staff User's Guide-Circulation* and the **Staff Requests** section of the *Aleph 21 System Librarian's Guide-Circulation*.

Implementation Notes

- 1 Add the requester name field to hold and photocopy request creation forms in the OPAC:

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 to the 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 following lines:

```
<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 to the 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 lines:

```
<a href="%server_f?func=item-photo-
request&doc_library=$0100&adm_doc_number=$0200&item_sequence=
$0300&year=$2500&volume=$2600&sub_library=$2700&type=$2800&no
_loaned=$2900&start_rec_key=$3000&end_rec_key=$3100&bib_doc_n
um=$0600&bib_library=$0500&source_form=PHOTO_DETAILS&author=$
4200&title=$4300&pages=$4400&note1=$4500&note2=$4600&pickup=$
4100&addlinfo=$4700&requestername=$3800" >

```

4 Display the requester name field in the 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">
```

5 View requester name information in the GUI lists:

- a** Display requester name information in the Hold Requests List in the Patron Tab, by adding the following line to `./<BIB library>/tab/pc_tab_col.lng`:

```
PC_CIR_B_HOLD_LIST L Requester Name 21 000 04 C04
Requester name
```

- b** Display requester name information in the History Hold Requests List of the Item Tab, by adding the following line to `./<BIB library>/tab/pc_tab_col.lng`:

```
PC_COM_HOLD_H_LIST L Requester Name 11 000 04 C04
Requester Name
```

- c** Display requester name information in the Photocopy Requests List of the Patron Tab, by adding the following line to `./<BIB library>/tab/pc_tab_col.lng`:

```
PC_CIR_B_PHOTO_LIST L Requester Name 08 000 04 C04
Requester name
```

- d** Display requester name information in the Historical Photocopy Requests List of the Patron Tab, by adding the following line to `./<BIB library>/tab/pc_tab_col.lng`:

```
PC_CIR_B_PHOTOH_LIST L Requester Name 06 000 04 C04
Requester Name
```

- e** Display requester name information in the Cash List of the Patron Tab, by adding the following line to `./<BIB library>/tab/pc_tab_col.lng`:

```
C_CASH_SUMMARY L Requester Name 22 000 04 C04
Requester Name
```

6 View requester name information in OPAC lists:

- a** Display requester name information in cash list by inserting the following line into `alephe/www_f_lng/bor-cash-head`:

```
<th class="text3">Requester Name</th>
```

- b** Insert the following line into `alephe/www_f_lng/bor-cash-body`:

```
<td class=td1 valign=top>$1100</td>
```

- c** Display requester name information in the requests list by inserting the following line into `alephe/www_f_lng/bor-hold-head`:

```
<th class="text3">Requester Name</th>
```

- d** Insert the following line into `alephe/www_f_lng/bor-hold-body`:

```
<td class=td1 valign=top>$1900</td>
```

- e** Display requester name information in requests list by inserting the following line into `alephe/www_f_lng/bor-photo-head`:

```
<th class="text3">
Requester Name
</th>
```

- f** Insert the following line into `alephe/www_f_lng/bor-photo-body`:

```
<td class=td1 valign=top>$1100 </td>
```

- g** Display requester name information in request expanded information by inserting the following line into `alephe/www_f_lng/bor-hold-exp`:

```
<tr>
<td class=td1 id=bold align=left>Requester Name:</td>
<td class=td1>&nbsp;$2400</td>
</tr>
```

- h** Display requester name information in request expanded information by inserting the following line into `alephe/www_f_lng/bor-photo-exp`:

```
<tr>
<td class=td1 id=bold align=left>Requester Name:</td>
<td class=td1 align=left>&nbsp;$1500</td>
</tr>
```

- 7** Sort the Hold Request list of the Patron Tab according to sublib, pickup location, open date, and requester name, by setting `06` in the `circ_user_z37_sort_routine` environment variable (defined in `pc_server_defaults`).
- 8** Sort the photocopy requests list of the Patron Tab according to the following sort options:
- Pickup Location / requester name / photocopy number
 - Sub library / requester name / photocopy number
 - Pickup Location / Sub library / requester name / photocopy number

- Pickup Location / Status / requester name / photocopy number
- Sub library / Status / requester name / photocopy number
- Pickup Location / Sub library / Status / requester name / photocopy number
- Pickup Location / Status / Sub library / requester name / photocopy number

These options are defined in `./<ADM library>/tab/atb_z38_sort` and in matching entries in `./<ADM library>/tab/pc_tab_exp_field.lng` under PHOTO-SORT-TYPE.

NOTE:

Refer to the **tab_z38_sort** section of the *Aleph 21 Configuration Guide* for more details.

- a Sort the Cash List of the Patron Tab according to transfer number, date, and requester name with the 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.
- b Sort the Cash List of the Patron Tab according to requester name, with the 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.
- c Sort Hold Requests List of My Library Card according to open date and requester name, by setting the 05 in `www_user_z37_sort_routine` and the environment variable (defined in `www_server.conf`).
- d Sort the Photocopy Requests List of My Library Card according to the open date and requester name, by setting 08 in `www_user_z38_sort_routine` and the environment variable (defined in `www_server.conf`).
- e Sort the Cash List of My Library Card according to transfer number, date, and requester name, with the 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.
- f Sort the Cash List of My Library Card according to requester name, with the 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.

Cash Transaction Types (tab18.lng) Enhancements

Purpose of Enhancement

A new parameter has been added in column 5 of the Cash Transaction Types table.

Description

The new `z` parameter activates the cash transaction (similar to `Y`) but only when the cash rate does not equal zero. Note that the `z` parameter influences only new transactions.

Note that ZERO-FINE-HANDLING in `tab100` is not relevant anymore now that this definition can be set by defining `z` in col. 5 of `tab18.lng`.

Reference to Technical Documentation

For more information, refer to the **Cash Transactions** section of the *Aleph 21 System librarian Guide-Circulation*.

Implementation Notes

- 1 To implement this feature, set col 5. with the `z` parameter, for example:

```
0040 ##### ## ## z           Lost material -  
Handling
```

- 2 Make sure that ZERO-FINE-HANDLING is not defined in `tab100` and set up types 0003 and 0014, if necessary, with the relevant parameter in `tab18.lng`.

Renew Items on Loan (cir-08)

Purpose of Enhancement

The Renew Items on Loan service have been enhanced by the addition of two new fields.

Description

The new fields limit the loan renewal when running the batch. The `Maximum renewals` field contains the maximum number of previous renewals that a loan

can have in order for the batch to be performed on it. For example, if the value 3 is entered, the batch performs a renewal on loans that have had up to three previous renewals. The `Maximum Loan days` field contains the maximum number of loan days that a loan can have in order for the batch to be performed on it. For example, if the value 15 is entered, the batch performs a renewal on a loan that has been loaned up to 15 days.

Implementation Notes

If localization is required:

- 1 Add to `./alephe/pc_b_lng/p-cir-08.xml` the following:

After the following entry:

```
<control>
  <radio>
    <argname>F10</argname>
    <label>Ignore Due Date</label>
    <source>explicit</source>
    <ritem>
      <value>Y</value>
      <display>Yes</display>
    </ritem>
    <ritem>
      <value>N</value>
      <display>No</display>
    </ritem>
  </radio>
</control>
```

Add the following:

```
<control>
  <edit_num>
    <argname>F13</argname>
    <label>Maximum renewals to be include</label>
    <size>01</size>
  </edit_num>
</control>
<control>
  <edit_num>
    <argname>F14</argname>
    <label>Maximum Loan days to be include</label>
    <size>03</size>
  </edit_num>
</control>
```

At the end of the file, add the following lines:

```
<!-- Param: P-MAX-NO-OF-RENEWS-X. F13 -->
<!-- Param: P-MAX-NO-OF-RENEWS 9(1). -->
<!-- Param: P-MAX-RENEW-DAYS-X. F14 -->
<!-- Param: P-MAX-RENEW-DAYS 9(3). -->
```

2 Add the following to ./aleph/error_lng/p-cir-08:

```
5004 0000 L Maximum renewals not numeric.
5005 0000 L Maximum loan Days not numeric.
5006 0000 L Value Zero in maximum renewals field is not
allowed.
5007 0000 L Value Zero in maximum loan Days field is not
allowed.
```

E-Mail Validation Check

Purpose of Enhancement

New functionality has been added to check the validity of a patron's e-mail address.

Description

The following new tab100 flag has been added to the ADM library setup: EMAIL-VALIDATION.

Possible values:

Y – the system performs a validation check on the e-mail address. A notification is displayed if the e-mail address is not valid.

N – default. No validity check is performed.

Reference to Technical Documentation

For more information, refer to the **tab100-Related Entries in Circulation** section of the *Aleph 21 System Librarian's Guide-Circulation*.

Implementation Notes

To activate the e-mail validation check, set ./xxx50/tab/tab100 with:

```
EMAIL-VALIDATION=Y
```

If localization is required:

1 Add to ./aleph/error_lng/pc_cir_c0468 the following:

```
0096 0001 L Email address is not valid.
0097 0001 L Email domain could not be identified. Entered
address might be wrong.
```

2 Add to ./aleph/error_lng/pc_com_c0125 the following:

```
0096 0001 L Email address is not valid.
0097 0001 L Email domain could not be identified. Entered
address might be wrong.
```

3 Add to ./aleph/error_lng/www_f_heading the following:

```
0096 0001 L Email address is not valid.
0097 0001 L Email domain could not be identified. Entered
address might be wrong.
```

4 Add to ./aleph/error_lng/xml_update_bor the following:

```
0096 Succeeded to $2 table $1. cur-id $3. Email address is not
valid.
0097 Succeeded to $2 table $1. cur-id $3. Email domain could
not be identified. Entered address might be wrong.
0098 Succeeded to $2 table $1 with modification restrictions.
cur-id $3. Email address is not valid.
0099 Succeeded to $2 table $1 with modification restrictions.
cur-id $3. Email domain could not be identified. Entered
address might be wrong.
```

5 Add to ./aleph/error_lng/p_file_20 the following:

```
0096 Succeeded to $2 table $1. cur-id $3. Email address is not
valid.
0097 Succeeded to $2 table $1. cur-id $3. Email domain could
not be identified. Entered address might be wrong.
0098 Succeeded to $2 table $1 with modification restrictions.
cur-id $3. Email address is not valid.
0099 Succeeded to $2 table $1 with modification restrictions.
cur-id $3. Email domain could not be identified. Entered
address might be wrong.
```

6 Add to ./aleph/error_lng/dlf_global the following:

```
0096 0001 L Email address is not valid.
0097 0001 L Email domain could not be identified. Entered
address might be wrong.
```

Check for Similar Item On Loan

Purpose of Enhancement

A new check routine has been introduced in order to determine if loaning an item is allowed based on the existence of a loan on a similar item.

Description

The `check_circ_14_b` routine checks if the patron already has a similar item on loan in order to block the loaning of two similar items at the same time.

A similar item is an item that has the same enumeration/chronology and is of the same material type. The sublibrary, collection, and status of an item are not taken into account for a determination of similarity.

The check disregards the definitions for Like Copy in tab100 (SUB-LIBRARY-DIVISION, HOLD-REQUEST-ITM-STATUS, HOLDREQ-PROCESS-STATUS, and HOLD-REQUEST-COLLECTION).

Reference to Technical Documentation

For more information, refer to the `tab_check_circ` section of the *Aleph 21 Configuration Guide*.

Implementation Notes

Add the new check to `/adm library/tab/tab_check_circ`. For example:

```
!      1                      2                      3
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!>
LOAN      check_circ_14_b
```

Loan Session Enhancements

Purpose of Enhancement

Two enhancements have been made to the loan session window.

Description

The following two enhancements have been made to the loan session window:

- The number of loans performed in the session is now displayed in the Loan Session tab.

- A new column has been added to the table in which the item call number for the item is displayed.

Implementation Notes

To display the loan counter:

Add a new line to the PC file: `.\circ\tab\lng>window.dat`. After the line:

```
LoanSession.Tab.Screen0          Loan Session
```

Add:

```
LoanSession.Tab.Screen01        Loan Session: %d
```

To display the item call number:

Update `./tab/pc_tab_col.lng` by adding the following line to the `PC_CIRC_LOAN_LINE` menu. Adjust column 5 of all of the other lines to have a proper width display.

```
PC_CIR_LOAN_LINE      L Call Number          05 015 01 C03
Call Number
```

Log for Patron Record Changes

Purpose of Enhancement

A new log records changes made to the patron record, the name of the user who implemented the change, and the time and date the change was made.

Description

The log is created when the global patron record, local patron information, address information, and patron's IDs of the patron record is added, updated, or deleted.

The following new components have been added:

- A new ORACLE table, `Z307` of `usr_library`, for recording changes made on patron's related entities.
- A new configuration table, `tab_patron_log.lng`, which defines the actions that trigger patron log creation (`Z307`).
- A new batch service, `cir-27`, which produces a report of changes made for patron records.

Reference to Technical Documentation

For more information, refer to the **Log for Patron Record Changes** section of the *Aleph 21 System Librarian's Guide-Circulation*.

Implementation Notes

- 1 Make sure that the z307 Oracle table is set in `usr_library`:
 - If the `usr_library` has type USR then create table z307 with `util/a/17/1`.
 - If an ADM library is used as `usr_library`:
 - a Define z307 in `file_list` of adm library:

```
TAB z307          100K          0K          ts0
IND z307_id       100K          0K          ts1
IND z307_id2      100K          0K          ts1
```

- b Create table z307 with `util/a/17/1`.
- 2 Define the actions that trigger log creation by configuring `./<usr_library>/tab/ tab_patron_log.lng`. For example, in order to activate all logs, set the following:

```
1 2          3
!!!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!->
001 Y Update of global patron record-Details
002 Y Update of global patron record-Blocks
003 Y Update of local patron record-Details
004 Y Update of local patron record-Blocks
005 Y Update of local patron record-Privileges
006 Y Update of patron addresses
007 Y Update of patron IDs

010 Y Delete of global patron record
011 Y Delete of local patron record
012 Y Delete of patron address record
013 Y Delete of patron ID
!
020 Y New global patron record created
021 Y New local patron record created
022 Y New patron address record created
023 Y New patron ID created
```


- 3 If localization is required, add the following lines to `./alephe/pc_b_lng/menu-circ.xml` (under the Patron Records sub menu):

```
<item>
    <display>Patron Record Log Changes
    Report (cir-27)</display>
    <file>p-cir-27</file>
</item>
```

Cash Transactions Enhancements

Purpose of Enhancement

Several enhancements have been introduced for cash transactions handling.

Description

- It is now possible to waive cash transactions without entering a note.
- It is now possible to add a note to cash transactions from the circulation cash list and view it as part of the cash details.
- It is now possible to display the cash list in the Web OPAC, including bibliographic information and the author in each line (and in lines that are related to the same item).

Reference to Technical Documentation

For more information, refer to the **Cash Transactions from Patron Tab** section of the *Aleph 21 Staff User's Guide-Circulation*.

Implementation Notes

- Add the new cash note in the following files:
 - `<ADM library>/form_lng/cash-pay-receipt-pay.xml`
 - `<ADM library>/form_lng/cash-pay-receipt-refund.xml`
 - `<ADM library>/form_lng/cash-line-details.xml`

```
<xsl:call-template name="display-grid-gen">
  <xsl:with-param name="label" select="'Cash Note'"/>
  <xsl:with-param name="value" select="./z31-note"/>
  <xsl:with-param name="header" select="$header"/>
</xsl:call-template>
```

- Add the cash note in `<ADM library>/pc_display_lng/user-cash:`

```
<tr>
  <th class="table-1" id="bold" align="right">
    Cash Note
  </th>
  <td class=table-1 STYLE="-ms-word-wrap:break-word; word-
wrap:break-word; width:500px; ">
    ./z31-note
  </td>
</tr>
```

- If localization is needed, add the following line to `<ADM library>/tab/pc_tab_col.lng:`

```
PC_CASH_SUMMARY      L Note                23 000 it map
Cash Note
```

- To display bibliographic information and author in each line, in `/exlibris/aleph/a21_1/alephe/www_f_lng/bor-cash-body`

Replace:

```
<td class=td1 valign=top>$0900</td>
<td class=td1 valign=top>$0200</td>
```

With:

```
<td class=td1 valign=top>$1200</td>
<td class=td1 valign=top>$1300</td>
```

Enhancement to Print Only Hold Letters Without Printing Hold Slips in the Hold Shelf Report Service (cir-06)

Purpose of Enhancement

This enhancement allows for printing only letters for the hold requester without printing hold slips.

Description

When an item with a hold placed on it becomes available, there is an option to print a letter that informs the requester that the item is available and print a book hold slip to be placed in the book that indicates that the book is reserved. Until now it was not possible to print the letter without printing the slip. There

is now an option to print the letter without printing the slip. To enable this feature, the **format 25: Print only letters to next requester (no slips)** option has been added to Next Requester Letter Format field of the Cir-06 service (Hold Shelf Report). Select this option to print only letters and not slips.

Reference to Technical Documentation

For more information, refer to the batch service Help.

Implementation Notes

If localization is required, add the following new format (25) to the Next Requester Letter Format field (F13 argument) option in `./alephe/pc_b_<LNG>/p-cir-06.xml`:

```
<cbitem>
  <value>25</value>
  <display>Print only letters to next requester (no
  slips)</display>
</cbitem>
```

Restricting Cash Limit Related Check Routines to Specific Cash Types

Purpose of Enhancement

It is now possible to configure which cash types are taken into account by the cash limit related check routines.

Description

A new parameter has been added that indicates to the cash limit related check routines which cash types to take into account when calculating the patron's cash limit.

The following is the structure of the new parameter:

```
CHARGES=<include/exclude sign><Cash charges to include/
exclude>
```

The parameter consists of the following sections:

- Include/exclude sign:
 - Plus sign (+) – Include the listed cash charges
 - Minus sign (-) – Exclude the listed cash charges

- Cash charges to include/exclude – Cash charge numbers (4 digits each), separated by a comma.

When using the include sign, only the listed cash charges are taken into account in order to check if the patron exceeds the cash limit.

When using the exclude sign, all cash charges that are not listed are taken into account in order to check if the patron exceeds the cash limit.

If the parameter is not configured, all cash charges are taken into account.

The new parameter is supported by the following check routines:

- Check_circ_5
- check_circ_5_b
- check_circ_9_a
- check_circ_10_a,
- Check_hold_request_5
- check_hold_request_5_b
- check_hold_request_9_a
- check_hold_request_10_a

Reference to Technical Documentation

For more information, refer to the **tab_hold_request** and **tab_check_circ** sections of the *Aleph 21 Configuration Guide*.

Implementation Notes

Define the list of charges to include or exclude in `/adm library/tab/tab_hold_request` and `/adm library/tab/tab_check_circ`. For example:

```
! 1                2                3
!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!
CIRC      check_hold_request_5          CHARGES=+0001,0004
```

Return During Loan in NCIP and SIP2

Purpose of Enhancement

If an item is mistakenly registered as loaned and a patron attempts to borrow it, the NCIP and SIP2 servers now can automatically return the item and then loan it to the patron.

Description

If an item is mistakenly registered as loaned, a patron can borrow the item without the item having to be manually returned. If the `return_during_loan` option is set to `Y`, when the patron attempts to borrow the item, it is automatically registered as returned and then registered as loaned to the patron.

Reference to Technical Documentation

For more information, refer to the *How to Set Up and Use the NCIP Server* and *How to Set Up a SIP2 Server for Self-Check Systems* documents.

Implementation Notes

Set `return_during_loan` to `Y` in `/adm library/tab/ tab_ncip.conf`:

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[Default]
    return_during_loan = Y
```

Set `return_during_loan` to `Y` in `/adm library/tab/ tab_sip2.conf`:

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[SIP]
    return_during_loan = Y
```


5

General

This section includes:

- **Performance Improvements** on page 71
- **Identifying PC Workstation with the Logon Name** on page 74
- **HTML in e-Mail** on page 75
- **Aleph Extended Topologies** on page 75
- **Retrieving Availability Information by Collection** on page 76
- **New Barcode Fonts to Support IE 9** on page 77
- **Aleph 21 and JBoss 5** on page 77

Performance Improvements

Retrieve Catalog Records (ret-01)

Purpose of Enhancement

A new parameter, `Number of Processes`, has been added.

Description

The `Number of Processes` parameter builds a cycle file and runs in parallel on different ranges of documents.

Reference to Technical Documentation

For more information, refer to the batch service help.

Implementation Notes

If localization is required, add the following to `./alephe/pc_b_lng/p-ret-01.xml`:

- Before the line `</dialog>`:

```
<control>
  <edit_num>
    <argname>F38</argname>
    <label>Processes to Create</label>
    <size>2</size>
    <default>1</default>
  </edit_num>
</control>
```

- At the end of the file:

```
<!-- Param:   P-NO-PROCESS-X.           F38 -->
<!-- Param:   P-NO-PROCESS             9 (2) . -->
```

Rebuild Word Index (manage-01)

Purpose of Enhancement

A new parameter, Rebuild the Words dictionary Z97 (Y/N), has been added.

Description

The default is `Y`. If set to `N`, the service runs without deleting the previous `z97` table. If the service has already been run in the past, it saves the building and loading of the words dictionary.

Reference to Technical Documentation

For more information, refer to the batch service help.

Implementation Notes

If localization is required, add the following to `./alephe/pc_b_lng/p-manage-01.xml`:

- Before the paragraph about the number of processes:

```
<control>
  <drop_down>
    <argname>F06</argname>
    <label>Rebuild the Words dictionary (Z97)</
label>
    <default>Y</default>
    <cbitem>
      <value>Y</value>
      <display>Yes</display>
    </cbitem>
    <cbitem>
      <value>N</value>
      <display>No</display>
    </cbitem>
  </drop_down>
</control>
```

- At the end of the file:

```
<!-- Param: P-REBUILD-DICT X(1) . F06 -->
```

Indexing

- **ue_01** – Word indexing has been split into 10 separate processes.
- **p-manage-01** – A new service parameter has been added: Rebuild the Words dictionary Z97 (Y/N).

List Display

The display of lists with scrolling has been improved by optimizing the data transfer between the PC and the server.

Item List Display

Item lists are now displayed faster.

Cataloging

The `p-ret-01` batch service can now be split into parallel processes.

Circulation Logger Clean Up Service (cir-78) Enhancement

Purpose of Enhancement

The performance of the Circulation Logger Clean Up service (cir-78) has been improved.

Description

The Circulation Logger records activities performed in the Circulation module. In order to increase the speed of the Circulation Logger Clean Up service (cir-78), a new index has been added to the Z309-ACTION field.

Identifying PC Workstation with the Logon Name

Purpose of Enhancement

A new option identifies a PC workstation instead of the IP address.

Description

It is now possible to use the Windows logon name to identify a PC workstation instead of the IP address. This solution is relevant only for customers who use a single IP address for all PC workstations.

Reference to Technical Documentation

For more information, refer to the **Setting Up User Logons** section of the *Aleph 21 System Librarian's Guide – General*.

Implementation Notes

1 Run `$alephe_root/aleph_shutdown` and make sure that no Aleph processes are running.

2 Run the following command:

```
csh -f $aleph_proc/clear_vir01 VIR01
```

3 Add the following line to the `$alephe_root/aleph_start.private` file:

```
setenv use_login_for_lock Y
```

4 Run the command `source $alephe_root/aleph_start` or log on again to the server.

5 Run `$alephe_root/aleph_startup`.

HTML in e-Mail

Purpose of Enhancement

It is now possible to send e-mails to patrons, vendors, and partners with formatted HTML in the body of the e-mail.

Description

The **HTML Body Only** option has been added to the Mail Attachment drop-down list. Selecting it causes e-mail to be sent with formatted HTML inside the body of the e-mail. The related HTML is built according to the defined XSL files, which are located under the `form_LNG` directory.

Reference to Technical Documentation

For more information, refer to the **Sending Attachments in E-mail Messages** section of the *Aleph 21 System Librarian's Guide-Printing*.

Implementation Notes

Add the following lines to `<ADM Lib>/tab/ pc_tab_exp_field.LNG` and in `<ILL Lib>/tab/pc_tab_exp_field.LNG`:

```
!      1          2      3
4
!!!!!!!!!!!!!!!!!!!!!!-----!-
!!!!!!!!!!!!!!!!!!!!!!-----!-
PLAIN-HTML          L HTML Body Only
L
```

If you want to disable the option to send e-mail attachments and send only HTML format in the body of the e-mail, set `$alephe_tab/tab100` with the following:

```
PLAIN-ONLY=H
```

Aleph Extended Topologies

Purpose of Enhancement

Several extended topologies are now supported.

Description

Multi-server topologies can now be used to improve performance, scalability, and recovery in order to solve security issues.

The Aleph system now supports the usage of different servers for the database, the Apache server, and the Aleph software and also supports the usage of different machines for batches and daemons.

Reference to Technical Documentation

For more information, refer to the *Aleph 21 Topologies* document.

Retrieving Availability Information by Collection

Purpose of Enhancement

It is now possible to retrieve the availability information by collection (in addition to sublibrary).

Description

The `collection` option has been added to the `expand_doc_bib_avail` `expand` routine. This option retrieves the availability information by collection (in addition to sublibrary).

Reference to Technical Documentation

For more information, refer to the `expand_doc_bib_avail` section of the *Aleph Publishing Mechanism* document.

Implementation Notes

Add the collection parameter to `/bib library/tab/tab_expand`. For example:

```
! 1                               2                               3
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
PRM1          expand_doc_bib_avail          COLLECTION=Y
```

New Barcode Fonts to Support IE 9

Purpose of Enhancement

It is now possible to view and print barcodes when using IE 9.

Description

New GUI fronts have been added in order to support the viewing and printing of barcodes when using IE 9.

Implementation Notes

Replace the existing barcode fonts in the relevant XSL with the new fonts in the windows system directory.

Aleph 21 and JBoss 5

Purpose of Enhancement

Aleph version 21 now works with JBoss version 5.

Jboss 5 is part of the Aleph 21 installation.

Implementation Notes

During version 21 installation, the old `ng` tree will be replaced in order to match the new Jboss.

Therefore, manually copy the external adapters (JAR files) from the old `ng` tree (if used) to the Aleph 21 `ng` tree.

6

Indexing

This section includes:

- [APAC Indexing](#) on page 79

APAC Indexing

Adjacency Words Creation for Latin Only

Purpose of Enhancement

It is now possible to configure the system to create adjacency words for Latin words only.

Description

Some of the CJK indexing routines (2gram and morpheme word breaking routines) have special rules for handling adjacent CJK words. Therefore, when these routines are used in the indexing process, only Latin adjacency words are created.

Reference to Technical Documentation

For more information, refer to the `setenv ADJACENCY` section of the *Aleph 21 Configuration Guide*.

Implementation Notes

In `./alephe/ aleph_start` set:

```
ADJACENCY_TYPE=3
```

New Indexing/Searching Routines

Purpose of Enhancement

New indexing and searching word breaking routines have been introduced for handling CJK text.

Description

The following word breaking routines have been introduced:

- `mopheme2_index` – To be used only for word breaking during the indexing process.

This routine creates all possible substrings that are a concatenation of successive CJK words that are less than the character limit in the given field.

This routine is used in parallel with the `2gram` routine.

- `mopheme2_search` – To be used when the `morphem2_index` and `2gram` routines are both used together in the indexing process.

If the searched word is longer than the character limit, this routine uses the `2gram` routine. Otherwise, the `morpheme` routine is used.

Reference to Technical Documentation

For more information, refer to the **APAC Indexing** section of the *Aleph 21 System Librarian's Guide-Indexing*.

Implementation Notes

- 1 The default value for character limit is 8. This is the recommended value. If a different value is required, define it in `aleph_start`:

```
setenv MORPHEME_LIMIT_CHAR 9
```

- 2 Set the new word breaking routines in `<BIB library>/tab/tab_word_breaking`. For example:

```
!1 2          3          4
!!-!-!!!!!!-
!!!!!!!!!!!!!!
01 # del_subfield
01 # cjk_2gram_lng      CONC
!*
03 # del_subfield
03 # morpheme2_index

93 # cjk_2gram_lng      NO-CONC
94 # morpheme2_search
```


7

Integrated ILL

This section includes:

- **Patron Information in ILL Borrowing Request** on page 83
- **Multi-Item Borrowing Request** on page 85

Patron Information in ILL Borrowing Request

Purpose of Enhancement

The following fields have been added to ILL borrowing request information

- Patron's status
- Type
- Expiry date

Description

Patron status, type, and expiry date have been added to the following:

- ILL Module>Borrowing Request>Request Info tab 1 (read-only info)
- ILL Module>Borrowing printouts: Arrival Letter, Info Letter, Cancel Letter, etc.

The patron information is taken from the patron local information as registered in the ADM library (for example, local patron record at USM50). If there is no local patron information at the ADM level, the system selects an alternative local patron record (as defined in `tab_sub_library.lng` - col. 9 through 13).

Implementation Notes

- 1 Add the following lines to `./ill_libray/pc_display_lng/bor-request-expand` (see example in `./usm40/pc_display_eng/bor-request-expand`)

```
<tr>
  <th class="table-1" id="bold" align="left" width="50%">
    Patron Status
  </th>
  <td class=table-1>
    ./z305-bor-status
  </td>
</tr>

<tr>
  <th class="table-1" id="bold" align="left" width="50%">
    Patron Type
  </th>
  <td class=table-1>
    ./z305-bor-type
  </td>
</tr>

<tr>
  <th class="table-1" id="bold" align="left" width="50%">
    Patron Expiry Date
  </th>
  <td class=table-1>
    ./z305-expiry-date
  </td>
</tr>
```

- 2 Change the following line in `./ill_libray/pc_display_lng/bor-request-expand`

Replace this:

```
<tr>
  <th class="table-1" id="bold" align="left" width="50%">
    Patron Status
  </th>
  <td class=table-1>
    ./z36-bor-status
  </td>
</tr>
```

With this:

```
<tr>
  <th class="table-1" id="bold" align="left" width="50%">
    Patron Status (during loan)
  </th>
  <td class=table-1>
    ./z36-bor-status
  </td>
</tr>
```

- 3** Make sure the following lines are set in `./ill_library/form_lng/global.trn`

```
z305-bor-status          PC-TAB-EXP S BOR-STATUS z305-sub-
library
z305-bor-type            PC-TAB-EXP S BOR-TYPE
z305-expiry-date        DATE          S
```

- 4** Add the following lines to `./ill_library/form_lng/funcs-address.xml` (see an example in `./usm40/form_lng/funcs-address.xml`)

```
<xsl:call-template name="display-gen">
  <xsl:with-param name="label" select="'Patron Status'"/>
</xsl:call-template>
<xsl:call-template name="display-gen">
  <xsl:with-param name="label" select="'Patron Type'"/>
  <xsl:with-param name="value" select="//z305-bor-
status"/>
</xsl:call-template>
<xsl:call-template name="display-gen">
  <xsl:with-param name="label" select="'Patron Expiry
Date'"/>
  <xsl:with-param name="value" select="//z305-expiry-
date"/>
</xsl:call-template>
```

Multi-Item Borrowing Request

Purpose of Enhancement

It is now possible to handle more than one item in each borrowing request.

Description

The borrowing workflow now supports the receiving, managing, and returning of multiple items as part of a single request and recording the charge for the items together.

Reference to Technical Documentation

For more information, refer to the **Borrowing Requests** section of the *Aleph 21 Staff User's Guide-Integrated ILL* and the **ILL Temporary Items** section of the *Aleph 21 System Librarian's Guide-Integrated ILL*.

Implementation Notes

- In `./<ILL_lib>/form_lng/ill-patron-letter-4.xsl`:

Add:

```
<!--section-02 (SPLIT)-->
  <xsl:for-each select="//section-02">
    <xsl:call-template name="section-02"/>
  </xsl:for-each>
```

Replace the line:

```
<xsl:with-param name="line" select="'salutation_string##We
would like to inform you that your ILL item is overdue.'"/>
```

With:

```
<xsl:with-param name="line" select="'salutation_string##We
would like to inform you that your ILL item(s) is overdue.'"/>
```

Remove:

```
<xsl:call-template name="display-gen">
  <xsl:with-param name="label" select="'Item
Barcode:'"/>
  <xsl:with-param name="value" select="./z411-
item-barcode"/>
</xsl:call-template>
```

Add:

```

<!--SECTION-02 (SPLIT)-->
<xsl:template name="section-02">
  <xsl:call-template name="table-split-open"/>
  <xsl:call-template name="free-title">
<xsl:with-param name="free-title" select="'Item:'"/>
  </xsl:call-template>
  <xsl:call-template name="display-gen">
    <xsl:with-param name="label" select="'Item
Barcode:'"/>
    <xsl:with-param name="value" select="./z30-
barcode"/>
  </xsl:call-template>
  <xsl:call-template name="table-split-close"/>
</xsl:template>

```

- In `./<ILL_lib>/form_lng/ill-patron-letter-5.xsl`:

Add:

```

<!--section-02 (SPLIT)-->
  <xsl:for-each select="//section-02">
    <xsl:call-template name="section-02"/>
  </xsl:for-each>

```

Replace the line:

```

<xsl:with-param name="line"
select='concat("salutation_string##We regret to inform you
that the due date for your ILL loaned item (detailed
below)##was advanced to: ",//z36-due-date, "##Please return
the item on the requested date.") '/>

```

With:

```

<xsl:with-param name="line"
select='concat("salutation_string##We regret to inform you
that the due date for your ILL loaned item(s) (detailed
below)##was advanced to: ",//z36-due-date, "##Please return
the item(s) on the requested date.") '/>

```

Remove:

```

<xsl:call-template name="display-gen">
  <xsl:with-param name="label" select="'Item
Barcode:'"/>
  <xsl:with-param name="value" select="./z411-
item-barcode"/>
</xsl:call-template>

```

Add:

```
<!--SECTION-02 (SPLIT)-->

    <xsl:template name="section-02">
        <xsl:call-template name="table-split-open"/>
        <xsl:call-template name="free-title">
            <xsl:with-param name="free-title"
select="'Item:'"/>
        </xsl:call-template>
        <xsl:call-template name="display-gen">
            <xsl:with-param name="label" select="'Item
Barcode:'"/>
            <xsl:with-param name="value" select="./z30-
barcode"/>
        </xsl:call-template>

        <xsl:call-template name="table-split-close"/>
    </xsl:template>
```

- In ./<ILL Lib>\form_lng\ill-arrival-slip-new.xsl:

Replace:

```
<xsl:with-param name="value" select="./z411-item-barcode"/>
```

With:

```
<xsl:with-param name="value" select="./z30-barcode"/>
```

Replace:

```
<xsl:with-param name="value" select="./z411-circulation-
note"/>
```

With:

```
<xsl:with-param name="value" select="./z30-note-circulation"/
>
```

- In ./<ILL Lib>\form_lng\ill-patron-letter-a.xsl:

After:

```
<!--section-01 (FREE)-->
    <xsl:for-each select="//section-01">
        <xsl:call-template name="section-01"/>
    </xsl:for-each>
```


Add:

```
<!--section-02 (SPLIT)-->
  <xsl:for-each select="//section-02">
    <xsl:call-template name="section-02"/>
  </xsl:for-each>
```

Replace:

```
<xsl:with-param name="line" select='"salutation_string##We
would like to inform you that your ILL request item has
arrived to the library. "'/>
```

With:

```
<xsl:with-param name="line" select='"salutation_string##We
would like to inform you that your ILL request item(s) have
arrived to the library. "'/>
```

Replace:

```
<xsl:with-param name="line" select='"Please come to the
library at your earliest convenience to pick up the item"/>
```

With:

```
<xsl:with-param name="line" select='"Please come to the
library at your earliest convenience to pick up the item(s)"/>
```

After:

```
<xsl:call-template name="display-gen">
  <xsl:with-param name="label"
select="'Supplied media:'"/>
  <xsl:with-param name="value" select="./z411-
supp-media"/>
</xsl:call-template>
<xsl:call-template name="table-close"/>
<xsl:call-template name="table-open"/>
</xsl:template>
```

Add:

```
<!--SECTION-02 (SPLIT)-->
  <xsl:template name="section-02">
    <xsl:call-template name="table-split-open"/>
    <xsl:call-template name="free-title">
      <xsl:with-param name="free-title"
select="'Item:'"/>
    </xsl:call-template>
    <xsl:call-template name="display-gen">
      <xsl:with-param name="label" select="'Item
barcode:'"/>
      <xsl:with-param name="value" select="./z30-
barcode"/>
    </xsl:call-template>
    <xsl:call-template name="display-gen">
      <xsl:with-param name="label"
select="'Circulation note:'"/>
      <xsl:with-param name="value" select="./z30-
note-circulation"/>
    </xsl:call-template>
    <xsl:call-template name="table-split-close"/>
  </xsl:template>
```

Remove:

```
<xsl:call-template name="display-gen">
  <xsl:with-param name="label" select="'Item
barcode:'"/>
  <xsl:with-param name="value" select="./z411-
item-barcode"/>
</xsl:call-template>
```

Remove:

```
<xsl:call-template name="display-gen">
  <xsl:with-param name="label"
select="'Circulation note:'"/>
  <xsl:with-param name="value" select="./z411-
circulation-note"/>
</xsl:call-template>
```

- In `./<ill library>/pc_display_lng/bor-request-expand`:

Add:

```
SECTION-01
.....
END-SECTION-01
In section 01 live all exsiting rows
SECTION-02
CONDITION z30-barcode
<tr>
  <th class="table-1" id="bold" align="left"
width="30%">
    Barcode
  </th>
  <td class=table-1>
    ./z30-barcode
  </td>
</tr>
END-CONDITION
END-SECTION-02
SECTION-01 SINGLE
```

- If localization is needed, in GUI, `\Circ\Tab\lng\Message.dat`:

Add:

```
MultiIll          Multi Item ILL          O    I
There are %d additional items that should also be loaned.
```

- If localization is needed, in GUI, `.\Ill\Tab\Lng\Message.dat`:

Add:

```
ReturnMultiItems          ALEPH ILL          OC    Q
This request has %d related items. Continue?
```

- If localization is needed, in GUI, `.\ill\tab<lng>\window.dat`:

Add:

```
MaterialType.Radio.ReturnableMulti          Returnable &Multi
```

Add:

```
ReceiveReturn.Tab.Screen0          Request
  ReceiveReturn.Tab.Screen1          Items
  ReceiveReturn.Btn.Add              &Add
  ReceiveReturn.Btn.Delete           &Delete
  ReceiveReturn.Btn.Modify           &Modify
  ReceiveReturn.MultiItems.EmptyBC   System
Generated Barcode
```

Add:

BulkReceiveRet.Tab.Screen1	Items
----------------------------	-------

Add:

BulkReceiveRet.Btn.Add	&Add
BulkReceiveRet.Btn.Delete	&Delete
BulkReceiveRet.Btn.Modify	&Modify
BulkReceiveRet.MultiItems.EmptyBC	System
Generated Barcode	

Add:

ReceiveMultiInput.Title	Multiple Items
ReceiveMultiInput.Btn.Add	&Add
ReceiveMultiInput.Static.Barcode	Item Barcode
ReceiveMultiInput.Static.Note	Circulation
Note	

- In `.\alephcom\tab\<lng>\tab_col.dat`

Add:

ILL_MULTI_ITEMS	L Barcode	01 030 01 C01
barcode		
ILL_MULTI_ITEMS	L Circulation Note	02 070 01
C01	note	

8

Items

This section includes:

- **Configuring the Statistic Field to be Mandatory** on page 93
- **Printing Labels for Multiple Items** on page 94
- **Update of Monograph Item Arrival Date** on page 94
- **Sort Items by Copy ID** on page 95
- **Running Multiple Simultaneous Instances of the Update Item Records Service (p-manage-62)** on page 96
- **Advanced General Retrieval Form (ret-adm-03)** on page 96
- **Notification When Item Changed to be Part of a Requested Group** on page 97
- **Item Changes Alert** on page 98
- **Enhancement to the Item Call Number Duplication Check to Check for Duplicate Titles** on page 98
- **Item List Filter** on page 99

Configuring the Statistic Field to be Mandatory

Purpose of Enhancement

The Statistic field of Item records can be set to be a mandatory field. This refers to the main Item form that is used in the Cataloging and ACQ/Serials modules.

Description

To support this, a new tab100 flag, ITEM-FORM-STATISTIC, has been added to `./<adm_library>/tab/ tab100`.

Possible values: Y/N

N - Default. The item's statistic field (Z30-ITEM-STATISTIC) is not a mandatory field. The item record can be saved with null in the Statistic field (as is now, before implementing the new functionality).

Y - The item's Statistic field is a mandatory field (new functionality). If the Statistic field is left blank when an item record is saved, a message is displayed forcing the user to populate the Statistic field.

Reference to Technical Documentation

For more information, refer to the **Item Statistic Options** section of the *Aleph 21 System Librarian's Guide-Items*.

Implementation Notes

Set the following line to `./xxx50/tab/tab100`:

```
ITEM-FORM-STATISTIC=Y
```

Printing Labels for Multiple Items

Purpose of Enhancement

You can now print labels for several items at one time.

Description

You can now select up to 100 items from the Item list using the <CTRL> and <SHIFT> keys and print labels for them by clicking **Label**. Before this enhancement, you could only print a label for one item at a time.

The order of printed labels is the same as the order that the items are displayed in the Item List (as set in the Sort Option).

Update of Monograph Item Arrival Date

Purpose of Enhancement

A new ADM library tab100 flag, ITEM-DATE-FROM-ORDER, has been created to control the automatic update of the arrival date of monograph orders.

Description

The new flag has the following possible values: 0/1:

- 0 - Default. The arrival date is not recorded (old functionality).
- 1 - The arrival date is recorded in the item record for all related items (new functionality).

Reference to Technical Documentation

For more information, refer to the **Related Entries in tab100** section of the *Aleph 21 System Librarian's Guide-Items*.

Implementation Notes

Set the following line in `/xxx50/tab/tab100`:

```
ITEM-DATE-FROM-ORDER=1
```

Sort Items by Copy ID

Purpose of Enhancement

A new sort routine, 16, has been added for issue items.

Description

Routine 16 sorts items by the following hierarchy of an item's chronology, enumeration, and copy ID (ascending and/or descending order):

- Chronology level 1 .Year. (Z30-CHRONOLOGICAL-I)
- Chronology level 2 (Z30-CHRONOLOGICAL-J)
- Chronology level 3 (Z30-CHRONOLOGICAL-K)
- Enumeration level 1 .Volume. (Z30-ENUMERATION-A)
- Enumeration level 2 (Z30-ENUMERATION-B)
- Enumeration level 3 (Z30-ENUMERATION-C)
- Copy number (Z30-COPY-ID)

Items that are not assigned with a copy ID are sorted by their sequence number (Z30-ITEM-SEQUENCE) at the top of the relevant chronology/enumeration list for an ascending sort and at the bottom of the list for a descending sort.

Reference to Technical Documentation

For more information refer to the `tab_z30_sort` section of the *Aleph 21 Configuration Guide* and to Table header `./xxx50/tab/tab_z30_sort`.

Implementation Notes

- 1 Set the new item list sort routine, 16, in `./xxx50/tab/tab_z30_sort`. It can be set on ascending and/or descending order as in the following example:

```
ITEM-12          A 16 A 16
ITEM-13          D 16 D 16
```

- 2 Update `./xxx50/tab/pc_tab_exp_field.lng` with the relevant entries (in this example setup: ITEM-12 and ITEM-13):

```
ITEM-SORT-TYPE   L Chron./Enumer./Copy ID-Ascend
ITEM-12
ITEM-SORT-TYPE   L Chron./Enumer./Copy ID-Descend
ITEM-13
```

Running Multiple Simultaneous Instances of the Update Item Records Service (p-manage-62)

Purpose of Enhancement

Multiple instances of the Update Item Records manage-62 batch service can now be run simultaneously.

Description

The new Process to Create parameter enables multiple instances of the Update Item Records manage-62 batch service to be run simultaneously in order to shorten the run time.

The user can indicate the number of simultaneous instances of this service to be run.

Reference to Technical Documentation

For more information, refer to the **Update Item Records (p-manage-62)** batch online help.

Advanced General Retrieval Form (ret-adm-03)

Purpose of Enhancement

This new service is an advanced version of the General Retrieval Form (ret-adm-01). It is available from the Items menu of Cataloging and Circulation modules.

Description

Similar to the General Retrieval Form (ret-adm-01), this service retrieves items, places them in an output file, and produces a batch report.

It has the following advanced features beyond the General Retrieval Form:

- The Advanced Retrieval Form can be run in parallel process (divided into cycles) in order to improve performance. This option is enabled only if the service is run without an input file.
- Instead of five value fields for each filter (as in ret-adm-01), there is only one field where it is possible to enter up to 20 values, separated by semicolons.
- For each filter, the user can include or exclude the listed values.

Reference to Technical Documentation

For more details, see the batch online help service.

Implementation Notes

To have the new service available in the Items menu of the Cataloging and Circulation modules, define `p-ret-adm-03` in the `menu-catalog.xml` and `menu-circ.xml` files.

For an example, see the following demo files: `./aleph/a21_1/aleph/pc_b_eng/menu-catalog.xml` and `menu-circ.xml`.

Notification When Item Changed to be Part of a Requested Group

Purpose of Enhancement

A notification is displayed when an item's information is changed so that it is now part of a requested group.

Description

When an item's information is changed from the Change Item Process Status or Change Item Info Item menus, a new check determines if the item is now part of a requested group. If it is, a notification is displayed indicating that the new item is now a member of a like item group of an existing hold request.

Item Changes Alert

Purpose of Enhancement

This new development prevents the user from losing changes made to item data. This new functionality is flag dependent (`NotifyItemChanged` in `Alephcom.ini`)

Description

If a user updates an item and then moves to another item, a message is displayed requesting confirmation that the user wants to move to the new item without saving the changes. This functionality is flag dependent.

Reference to Technical Documentation

For more information, refer to the **Item changes Alert** section of the *Aleph 21 System Librarian Guide-Items*.

Implementation Notes

- 1 Configure the following in `alephcom.ini`:

```
[ItemList]
NotifyItemChanged=Y
```

- 2 If localization is required, add the following line to `alepcom/tab/lng/message.dat`:

```
ItemChange                ALEPH GUI                YN  2 Q
Item %d-%d has been changed locally, but the changes have not
been saved on the server. Continue anyway?
```

Enhancement to the Item Call Number Duplication Check to Check for Duplicate Titles

Purpose of Enhancement

When entering or updating item information in Aleph, you can now configure Aleph to check if an item with the same call number and title already exists in the system. If one does, a message is displayed informing the user.

Description

Tab100 flag: CHECK-DUP-CALL-NO has been enhanced.

Until now the check only looked for items with the same call number but a different title (a different ADM record). If one was found, the user was blocked from adding the item. Now, the check also looks for items with the same call number and the same title (the same ADM record). If one is found, a message is displayed informing the user, but the user is not blocked from adding or updating the item.

Reference to Technical Documentation

For more information, refer to the **CHECK-DUP-CALL-NO** section of the *Aleph 21 System Librarian's Guide-Items*.

Implementation Notes

To activate the item call number duplication check, set the following in `./adm_lib/tab/tab100`:

```
CHECK-DUP-CALL-NO=Y
```

Item List Filter

Purpose of Enhancement

The Item List has been enhanced to allow filtering by sublibraries and year.

Description

The following new buttons have been added to the bottom of the Item List in the Aleph Cataloging and Acquisitions/Serials modules to allow for filtering the Item List by sublibraries and year:

- **Filter by** – Click to display the Item List Filter window that allows filtering the Item List by year and sublibraries.
- **Clear Filter** – Click to clear the filters and refresh the item list to display all items according to the user privileges and sort routines. When the full list is displayed without filtering this button is disabled.

Reference to Technical Documentation

For more information, refer to the **Item List Filter** topic in the Aleph Online Help.

9

Serials

This section includes:

- **Batch Service: Print Claim Letters/Report (serial-44)** on page 101
- **Order Information Tab in Serials Module** on page 103
- **Ordering Item Arrival and Label Printout by Copy ID** on page 104
- **SISAC Barcode** on page 105

Batch Service: Print Claim Letters/Report (serial-44)

Purpose of Enhancement

Two new fields have been added to the batch service Print Claim Letters/Report (serial-44) – Order Unit and Acquisition Method.

In addition, a new parameter has been added to allow the running of multiple instances of the batch service simultaneously.

Description

The new fields allow filtering the input of the batch so that claim reports and letters are produced for items that are related to the specified order units and acquisition methods.

Note that the Order Unit filter is functional only for libraries that are set to work with Ordering Unit as part of their acquisition workflow (TAB100- USE-ORDER-UNIT=Y).

In addition, the batch Output Letters and Reports has been enhanced to include full order (Z68) and Vendor (Z70) information and not just selective data as was the case before this enhancement. Libraries may formulate print templates to

include Order and Vendor data. The following are the relevant template file names:

- batch-serial-claim-letter-vend-m.xml
- batch-serial-claim-letter-vend.xml
- batch-serial-claim-letter.xml
- serials-claim-report.xml

For examples of print templates, see the `./usm01/form_eng` demo setup.

The new Process to Create parameter enables multiple instances of the batch service to be run simultaneously in order to shorten the run time. The user indicates the number of simultaneous instances of this service to be run.

Reference to Technical Documentation

For more information, refer to the batch online help.

Implementation Notes

The batch interface `./aleph/pc_b_lng/p-serials-44.xml` has been updated with the following new controls.

```
<control>
  <expand_checkbox>
    <label>Order Unit</label>
    <empty_line>NONE</empty_line>
    <source>ORDER-UNIT</source>
    <argname>F19</argname>
  </expand_checkbox>
</control>
<control>
  <expand_checkbox>
    <label>Method of Acquisition</label>
    <empty_line>NONE</empty_line>
    <source>ACQ-ORDER-METHOD</source>
    <argname>F20</argname>
  </expand_checkbox>
</control>
<control>
  <edit_num>
    <argname>F21</argname>
    <label>Processes to Create</label>
    <size>2</size>
    <default>1</default>
  </edit_num>
</control>
```

For batch output XSL examples, see the relevant template file names in demo setup: `./usm01/form_eng`.

Order Information Tab in Serials Module

Purpose of Enhancement

The Order Information tab has been added in the Control>Group node of the Serials module.

Description

Tab 6: Order Information has been added to the Control>Group node of the Series module. This tab displays, in a single line, all items that share the same enumeration, chronology, and order number.

The column headings are:

- Order Number
- Order Unit/Sublibrary
- Vendor name
- Acquisition method

A new sort routine, 15, has been added for the display of lines within tab 6: Order Information. Routine 15 sorts by chronology, enumeration, and order number by ascending and/or descending order.

Reference to Technical Documentation

For more information, refer to the **Group Node** section of the *Aleph 21 Staff User's Guide-Serials*, the online help, and table header `./xxx50/tab/tab_z30_sort`.

Implementation Notes

The following items are relevant for the configuration of this new feature:

- Column heading of the new tab: `./xxx01/tab/pc_tab_col.lng`:

```
PC_SERIAL_GROUPS_ORD L No. Items.          01 015 01 C01
PC_SERIAL_GROUPS_ORD L Arrived             02 015 01 C01
PC_SERIAL_GROUPS_ORD L Expected            03 015 01 C01
PC_SERIAL_GROUPS_ORD L Desc.               04 035 01 C01
PC_SERIAL_GROUPS_ORD L Order No.          05 015 01 C01
PC_SERIAL_GROUPS_ORD L Order Unit         06 035 01 C01
PC_SERIAL_GROUPS_ORD L Vendor              07 030 01 C01
PC_SERIAL_GROUPS_ORD L ACQ Method          08 015 01 C01
```

- The new tab description is set in PC file: `./acq/tab/lng/window.dat`:

```
GroupList.Tab.Screen5          Order Information
```

- A new error has been added to `./aleph/error_lng/pc_serial_c0306:`

```
0014 0000 L Unable to retrieve order.
```

- The new sort routine for the Order Information tab.

- 1 Set sort routine 15 in `./xxx50/tab/tab_z30_sort`. It can be set to ascending and/or descending order as in the following example:

```
ITEM-10          A 15 A 15
ITEM-11          D 15 D 15
```

- 2 Update `./xxx50/tab/pc_tab_exp_field.lng` with the relevant entries (in this setup example: ITEM-10 and ITEM-11):

```
SERIAL-GRP-SORT-TYPE L Chron./Enumer./Order Number-Ascend
ITEM-10
SERIAL-GRP-SORT-TYPE L Chron./Enumer./Order Number-Descend
ITEM-11
```

Ordering Item Arrival and Label Printout by Copy ID

Purpose of Enhancement

Aleph now uses the Copy ID (Copy Number) to perform the following actions:

- Arrived items are checked in in the order of the Copy ID (Copy Number)
- Labels are printed for checked-in items in the order of the Copy ID (Copy Number).

Description

Aleph can be configured to check in items and print labels according to the order of the Copy ID (Z30-COPY-NUMBER). This new feature can be applied only when the system is configured to register the arrival of all grouped selected items.

If some of the items are set with a null in the Copy Number field, the ones that are registered with a value are prioritized. The items without a copy number are ordered by their sequence number (Z30-ITEM-SEQUENCE).

Before this change, the check-in was conducted by the item sequence number – with no option to change it.

Reference to Technical Documentation

For more information, refer to the **Item Arrival Form** section of the *Aleph 21 System Librarian's Guide-Serials*.

Implementation Notes

- To configure arrived items to be checked in in the order of the Copy ID, set the following variables in `$alephe_root/pc_server_defaults`:

```
setenv modify_item_arrival      N
setenv arrival_group_order     1
```

- To configure labels to be printed in the order of the Copy ID, set the following in `./acq/tab/acq.ini`

```
[GroupList]
ArrivePrintLabel=Y
```

SISAC Barcode

Purpose of Enhancement

It is now possible to search for a serial publication and its items in the GUI Acquisitions module by SISAC Barcode.

Description

The SISAC Barcode is an automated representation of the SICI code. To search by SISAC Barcode, select **SISAC Barcode** from the Serial Bar drop-down list and enter the SISAC Barcode (manually or via a barcode scanner) in the adjacent edit box.

10

Web OPAC

This section includes:

- **Adding an OWN Field to a BIB Record Created as a Result of an ACQ Request** on page 107
- **Hold Request Confirmation Page Enhanced** on page 108
- **New Photocopy Request Confirmation Page** on page 112
- **Configuring Apache To Send Minimal Details** on page 114
- **Displaying Requested in the Due Date Column of the Holding Screen** on page 115
- **User Registration Form – Default Values** on page 115
- **Converting From Wide-Char to Standard ASCII** on page 116
- **My E-Shelf Enhancement** on page 116
- **Hold Request History Displayed in Library Card** on page 118
- **Server Side Scripts in the Aleph OPAC** on page 121

Adding an OWN Field to a BIB Record Created as a Result of an ACQ Request

Purpose of Enhancement

It is now possible to set up a value for the OWN field to be added to a new BIB record created as a result of the ACQ request.

Description

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
```

Reference to Technical Documentation

For more information, refer to the **Local Own Check Procedure** section of the *Aleph 21 System Librarian's Guide-Web OPAC*.

Hold Request Confirmation Page Enhanced

Purpose of Enhancement

The Hold Request Confirmation page has been enhanced.

Description

The following enhancements have been implemented to the Hold Request Confirmation page:

- The actual confirmation of the hold request is now done after reviewing all of the details of the request and clicking the **GO** button.
- The Hold Request Confirmation form contains more information regarding the requested item.
- The Hold Request Confirmation form contains a new **Back** button that enables you to go back to the Hold Request Confirmation form and change the details.

Reference to Technical Documentation

For more information, refer to the **Hold Request** section of the *Aleph 21 Staff User's Guide-Web OPAC*.

Implementation Notes

The changes in the following `www_f_lng` files are mandatory:

- `item-hold-request-details`

1 Change:

```
<br>Item has been requested for pickup at $0300.$0500<br></td>
```

To:

```
<br>Note that the request will not be active unless the 'Go'
button is clicked<br></td>
```

2 After:

```
<input type=hidden name="adm_doc_number" value=$0200>
```

Add:

```
<input type=hidden name="item_sequence" value=$4200>
  <input type=hidden name="bib_request" value=$0800>
  <input type=hidden name="year" value=$4300>
  <input type=hidden name="volume" value=$4400>
  <input type=hidden name="sub_library" value=$4500>
  <input type=hidden name="type" value=$4600>
  <input type=hidden name="no_loaned" value=$4700>
  <input type=hidden name="start_rec_key" value=$4800>
  <input type=hidden name="end_rec_key" value=$4900>
  <input type=hidden name="pickup" value=$4100>
  <input type=hidden name="from" value=$3900>
  <input type=hidden name="to" value=$5700>
  <input type=hidden name="note1" value="$3300">
  <input type=hidden name="note2" value="$3400">
  <input type=hidden name="author" value="$3000">
  <input type=hidden name="title" value="$3100">
  <input type=hidden name="pages" value="$3200">
  <input type=hidden name="rush_request" value=$3500>
```

3 Remove:

```
<tr>
  <td class="td2" colspan=2><br>
    <input type="image"
      alt="Order"
      src="&icon_path/f-go.gif" border=0>
    <a href="javascript:window.print();"
      </a>
  </td>
</tr>
```

4 After:

```
<tr>
  <td class="td2">Rush Handling </td>
  <td class="td2">$3500</td>
</tr>
</table>
```

Add:

```
<div class=title>
  <br>Requested Item Details <br>
</div>
<table border=0 cellspacing=2 width="90%">
  <tr>
    <td class="td2" nowrap>Bibliographic Information:</td>
  </tr>
  <tr>
    <td colspan=2 class="td2"> $5600 </td>
  </tr>
</table>
<table border=0 cellspacing=2 width="90%">
  <tr>
    <td class="td2" nowrap>Item status:</td>
    <td class="td2" nowrap>&nbsp;$5000</td>
  </tr>
  <tr>
    <td class="td2" nowrap>Sublibrary:</td>
    <td class="td2" nowrap>&nbsp;$5100</td>
  </tr>
  <tr>
    <td class="td2" nowrap>Collection:</td>
    <td class="td2" nowrap>&nbsp;$5200</td>
  </tr>
  <tr>
    <td class="td2" nowrap>Location:</td>
    <td class="td2" nowrap>&nbsp;$5300</td>
  </tr>
  <tr>
    <td class="td2" nowrap>Location-2:</td>
    <td class="td2" nowrap>&nbsp;$5400</td>
  </tr>
  <tr>
    <td class="td2" nowrap>Description:</td>
    <td class="td2" nowrap>&nbsp;$5500</td>
  </tr>
</table>
  Note that the request will not be active unless the 'Go'
  button is clicked &nbsp; &nbsp; &nbsp; &nbsp;
  <input type="image"
    alt="Order"
    src="&icon_path/f-go.gif" border=0>
  <a href="javascript:window.print();">
    </a>
```

```
<a href="&server_f?func=item-hold-
request&doc_library=$0100&adm_doc_number=$0200&item_sequence=
$4200&year=$4300&volume=$4400&sub_library=$4500&type=$4600&no
_loaned=$4700&start_rec_key=$4800&end_rec_key=$4900&source_fo
rm=HOLD_DETAILS&from_date=$3900&to_date=$5700&author=$3000&ti
tle=$3100&pages=$3200&note1=$3300&note2=$3400&rush_ind=$3500&
pickup_loc=$4100">
</a>
```

- item-hold-request-tail

1 Change:

```
<a href="javascript:document.form1.reset();">
```

To:

```
<a href="javascript:my_reset();">
```

- item-hold-request-head

After:

```
<head>
```

Add:

```
<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 == "note1" ||
        document.form1.elements[i].name == "note2") {
          document.form1.elements[i].value = "";
        }
        if (document.form1.elements[i].name ==
"rush_request")
          document.form1.elements[i].checked = false ;
        }
      }
    // -->
</script>
```

New Photocopy Request Confirmation Page

Purpose of Enhancement

A confirmation page for the photocopy request has been introduced.

Description

This new page contains the details of the request. The user must confirm the request by clicking **GO**; otherwise, the request is not preformed.

Reference to Documentation

For more information, refer to the **Photocopy Request** section of the *Aleph 21 Staff User's Guide-Web OPAC*.

Implementation Notes

The changes in the following `www_f_lng` files are mandatory:

- `item-photo-request-head`

After:

```
<head>
```

Add:

```
<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].value = "";
            }
        }
    }
}
// -->
</script>
```

- `item-photo-request-tail`

1 Change:

```
<input size=30 maxlength=50 name="author" value=$3000></td>
```


To:

```
<input size=30 maxlength=50 name="author" value="$3000"></td>
```

2 Change:

```
<input size=30 maxlength=100 name="title" value=$3100></td>
```

To:

```
<input size=30 maxlength=100 name="title" value="$3100"></td>
```

3 Change:

```
<input size=30 maxlength=30 name="pages" value=$3200></td>
```

To:

```
<input size=30 maxlength=30 name="pages" value="$3200"></td>
```

4 Change:

```
<input size=30 maxlength=100 name="note" value=$3300></td>
```

To:

```
---
<input size=30 maxlength=100 name="note" value="$3300"></td>
```

5 After:

```
<tr>
  <td class=td2>Note </td>
  <td class=td2> </td>
  <td class=td2>
    <input size=30 maxlength=100 name="note"
value="$3300"></td>
</tr>
```

Add:

```
<tr>
  <td class=td2>Note 2 </td>
  <td class=td2> </td>
  <td class=td2>
    <input size=30 maxlength=100 name="note2"
value="$3600"></td>
</tr>
```

6 Change:

```
<a href="javascript:document.form1.reset();">
```

To:

```
<a href="javascript:my_reset();">
```

Configuring Apache To Send Minimal Details

Purpose

The Apache server can be configured to send minimal details in order to improve Apache security.

Description

In order to improve security the Apache server can be configured to send to the browser minimal details in the HTTP headers and Apache signature, leaving out the server version and the installed modules.

Implementation Notes

Change the following lines in `$alephe_root/apache/conf/httpd.conf` and restart the Apache server:

1 From:

```
ServerSignature On
```

To:

```
ServerSignature Off
```

2 From:

```
ServerTokens Full
```

To:

```
ServerTokens Prod
```

Displaying Requested in the Due Date Column of the Holding Screen

Purpose of Enhancement

It is now possible to display `Requested` in the Due Date column of the holding screen for items that have only photocopy requests.

Description

Until now, `Requested` was displayed in the Due Date column of the holding screen, only if the item had hold or booking requests.

Implementation Instructions

In order to apply this functionality:

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

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

With:

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

Reference to Technical Documentation

For more information, refer to the **Holding Screens** section of the *Aleph 21 System Librarian's Guide-Web OPAC*.

User Registration Form – Default Values

Purpose of Enhancement

The hidden fields of the registration page are now filled with values from the `tab31` table.

Description

When a new user is registered via the Web OPAC, if hidden fields are not filled on the registration page, they are filled with the default values from the `tab31` table and not with the values from the code.

Converting From Wide-Char to Standard ASCII

Purpose of Enhancement

It is now possible to convert the fields in the New Borrower Registration form from Wide-Char to standard ASCII.

Description

When registering a new borrower in the Web OPAC by using a Japanese keyboard (which sends Wide-Char characters of numbers, English letters, and other standard signs), newly created borrowers cannot be located by name in the Aleph Circulation module, because names are registered in the database as Wide-Char, and Aleph automatically converts Wide-Char to standard ASCII characters if the text contains English letters, numbers, or other standard signs. To address this problem, the `WEB_BOR_REGISTRATION` program and `widechar_to_ascii` conversion table have been created to support the conversion from Wide-Char to standard ASCII.

Implementation Notes:

This feature is configured in the `WEB_BOR_REGISTRATION` section in `$alephe_unicode/tab_character_conversion_line` table:

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!
WEB_BOR_REGISTRATION ##### # line_utf2line_utf
widechar_to_ascii
```

Always use a hash (#) in column 3 in the `WEB_BOR_REGISTRATION` section.

The `widechar_to_ascii` table is a new conversion table. Its name and contents are configurable.

My E-Shelf Enhancement

Purpose of Enhancement

My E-Shelf have been enhanced to supply better information regarding request submissions.

Description

My E-Shelf now includes the following information regarding hold requests:

- A new column in My E-shelf that indicates if a patron has a hold request on a specific record.
- A new column in the report for Records Which Were Not Requested that details the reason for rejecting a hold request.

Reference to Technical Documentation

For more information, refer to the **My E-Shelf** section of the *Aleph 21 Staff User's Guide-Web OPAC* and to the **My E-Shelf** section of the *Aleph 21 System Librarian's Guide-Web OPAC*.

Implementation Notes

- 1 Change the following files to support a new column that indicates if a patron has hold request for a specific record:

- `./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>
```

- `./alephe/www_f_lng/myshelf-short-head-2-no-course`

After:

```
Documents in My e-Shelf
```

Add:

```

submitted a hold request
```

After:

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

Add:

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

- 2 Change the following files to support a new column which will specify the reason for rejecting a hold request

- /alephe/www_f_lng/group-request-exec-head:

After:

```
<th class=text3>Title</th>
```

Add:

```
<th class=text3>Note</th>
```

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

After:

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

Add:

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

- 3 If localization is required, add the following error messages to ./aleph/error_lng/www_f_heading:

```
1107 L Fail to create hold request. Apply to the system
administrator.
1151 L Not all items are like item - Please apply to hold
request on a specific item.
```

Hold Request History Displayed in Library Card

Purpose of Enhancement

A history of the patron's hold requests can now be displayed in the Library Card section of the Web OPAC.

Description

Aleph libraries that save Hold Request History data (Z37H) can configure the My Library Card page in the Web OPAC to display the patron's Hold Request History. This includes a list of the patron's hold requests and information about each request.

The API-DLF services have been enriched to provide the patron's Hold Request History data.

Reference to Technical Documentation

For more information, refer to the *Aleph 21 Staff User's Guide-Web OPAC* and the *Aleph 21 System Librarian's Guide-Web OPAC*.

Implementation Notes

- The following parameter has been added to `./alephe/www_server.conf`:
`www_z37h_max_record_limit`.

This parameter sets the maximum number of history hold request that are listed. The default value is 80.

For example:

```
setenv www_z37h_max_record_limit      35
```

- A new form file has been created for the borrower history hold detail save/mail function. This file is located under `./xxx01/form_lng/bor-history-hold-list-00`.
 - New HTML files have been created for the borrower history hold window. The files are located under `./alephe/www_f_lng`:
 - `bor-history-hold-head`
 - `bor-history-hold-body`
 - `bor-history-hold-tail`
 - `bor-history-hold-no-hold`
 - New HTML files have been created for the borrower history hold detail window. The files are located under `./alephe/www_f_lng`:
 - `bor-history-hold-exp`
 - `bor-history-hold-exp-head`
- 1 Update `./alephe/www_f_lng/bor-info`:

After:

```
</td>
  $1500
</tr>
```

Add:

```
<tr>
  <td class=td1>
    Hold Requests History
  </td>
  $9000
</tr>
```

2 Update ./alephe/www_f_lng/bor-info-single-adm:**After:**

```
</td>
  $1500
</tr>
```

Add:

```
<tr>
  <td class=td1>
    Hold Requests History
  </td>
  $9000
</tr>
```

3 Update ./alephe/www_f_lng/bor-include-1:**After:**

```
History Loans</a>
</td>
```

Add:

```
<td class="bar" align="center" nowrap>
  &nbsp;|&nbsp;
</td>
<td class="bar" align="center" nowrap>
  <a href="javascript:replacePage('&server_f?func=bor-
history-loan&adm_library=$5000');"
  title="History Loans" class="blue">
    History Loans</a>
</td>
```

4 Update ./alephe/www_f_lng/bor-include-3:**After:**

```
History Loans</a>
</td>
```


Add:

```
<td class="bar" align="center" nowrap>
  &nbsp;|&nbsp;
</td>
<td class="bar" align="center" nowrap>
  <a href="javascript:replacePage('&server_f?func=bor-
history-loan&adm_library=$5000');"
    title="History Loans" class="blue">
    History Loans&nbsp;</a>
</td>
```

5 Update ./alephe/www_f_lng/bor-include-4:

After:

```
History Loans&nbsp;</a>
</td>
```

Add:

```
<td class="bar" align="center" nowrap>
  &nbsp;|&nbsp;
</td>
<td class="bar" align="center" nowrap>
  <a href="javascript:replacePage('&server_f?func=bor-
history-loan&adm_library=$5000');"
    title="History Loans" class="blue">
    History Loans&nbsp;</a>
</td>
```

Server Side Scripts in the Aleph OPAC

Purpose of Enhancement

This new development enables you to add server-side scripts (other than Javascript) to Aleph OPAC HTML pages, which allows libraries to enrich these pages with content otherwise unavailable.

Description

A new tag is now supported by Aleph OPAC and can be added in HTML pages. This tag enables you to include in the HTML pages a call to server-side scripts that are written and maintained by the library.

Reference to Technical Documentation

For more information, refer to the **Server Side Scripts in the Aleph OPAC** section of the *Aleph 21 System Librarian's Guide-Web OPAC*.

Implementation Notes

For details, refer to the **Server Side Scripts in the Aleph OPAC** section of the *Aleph 21 System Librarian's Guide-Web OPAC*.

11

DLF API

This section includes:

- **RENEW-PRE Added to `tab_check_circ`** on page 123
- **New Parameters for the Items DLF Service** on page 124
- **Hold Request History DLF API Service** on page 124

RENEW-PRE Added to `tab_check_circ`

Purpose of Enhancement

The `RENEW-PRE` section has been added to `tab_check_circ`.

Description

The `RENEW-PRE` section checks renew permissions in the Loans List DLF service for all of the patron's loans.

For performance considerations, it is recommended to avoid defining the following checks:

- `check_circ_1_d`
- `check_circ_9_a`
- `check_circ_10_a`

Reference to Technical Documentation

For more information, refer to the `tab_check_circ` section of the *Aleph 21 System Configuration Guide*.

Implementation Notes

Add the desired checks in the RENEW-PRE section under `/adm/tab/tab_check_circ.`

New Parameters for the Items DLF Service

Purpose of Enhancement

Two new parameters have been added to the Items DLF service (Base URL>: `/record/<record ID>/items`)

Description

The new parameters are:

- `startPos` – Used to indicate the starting position of the returned items list. Possible values are a number between 0 and 990. Optional.
- `sublibs` – Used to indicate which sublibraries are included. When this parameter is populated, the returned items list is per sublibrary. Possible values are a list of sublibraries delimited by a comma. Optional.

Hold Request History DLF API Service

Purpose of Enhancement

This new DLF API service supports the display of a patron's Hold Request History list.

Description

It is now possible to receive brief and full information for the Hold Request History list and the full information of a specific request.

The parameter `type=history` is used to indicate history information of hold request.

Note that this service retrieves history information only for hold requests that are included by `z37h-type=h` (title requests and others are not included).

The following are usage examples:

- Full list of history hold requests:

<http://il-aleph09:1891/rest-dlf/patron/00000012/circulationActions/requests/holds?type=history&view=full>

- Brief list of history hold requests:
<http://il-aleph09:1891/rest-dlf/patron/00000012/circulationActions/requests/holds?type=history&view=brief>
- Specific history hold request information:
<http://il-aleph09:1891/rest-dlf/patron/00000012/circulationActions/requests/holds/USM500000513780000100001201110021150212?type=history>

Implementation Notes

If localization is required, add the following lines to `./aleph/dlf_lng/dlf-global.trn`:

```
!Tag                                Type          S Parameters
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!-!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!->
z37h-pickup-location                 SUB-LIB       S
z37h-status                          TEXT          S
W=Waiting_in_queue A=In_process S=Hold_shelf
```


12

ARC

This section includes:

- **Bibliographic Field Extraction (tab22_arc)** on page 127
- **ETL – New Complete Incremental Mode** on page 127

Bibliographic Field Extraction (tab22_arc)

Purpose of Enhancement

It is now possible to define up to 70 bibliographic fields that are extracted to ARC.

Description

A new configuration table – `tab22_arc` – is used for ARC extraction (ETL, `p_arc-01`) process. It defines up to 70 bibliographic fields that are extracted to ARC.

Reference to Technical Documentation

For more information, refer to the *ARC 3.0 Release Notes* and the **Addition of User Defined Fields** section of the *Aleph Configuration for ARC 3.0*.

ETL – New Complete Incremental Mode

Purpose of Enhancement

The ETL process (`p_arc_01`, extraction of data from Aleph to ARC) can now run in a new Complete Incremental mode. This new mode is designated `C`.

Description

In this mode, the ETL process performs an Extract and Load only to records that have been changed, added, or deleted in Aleph since the last run.

To support this new mode of the ETL process (`p_arc_01`), the Update Time Stamp field has been added to several Aleph Oracle tables.

A new table has been added to the Aleph database – Z900. This table contains data about records deleted in Aleph. When a deletion is made to a record in one of the tables, a trigger records the table and the deletion in the Z900 Oracle table.

Reference to Technical Documentation

For more information, refer to the *ARC 3.0 Release Notes* and the **Run ETL Process** section and **Appendix C Aleph Z900 Oracle table, Index, and Related Triggers** of the *Aleph Configuration for ARC 3.0*.

Implementation Notes

Verify that the Z900 table, Z900 index, and the related triggers are configured in the local `file_list` file of the ADM, BIB, USR, ILL, and Course Reading libraries as in the example below.

- 1 Run `util A - 17 - 1` (Drop & Create Table and Index) and `util A - 17 - 6 - 2` (Enable Trigger).
- 2 Implement the definitions in your local libraries in the `file_list` file, located under `$data_root`.

The following are example DEMO definitions, located in the `file_list` file under the `$aleph_root/tab/directory`.

■ file_list - BIB Library

```

!1      2          3          4          5
6
!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!
TAB z900          1M          0K          ts0
ram36.aleph1
IND z900_id      100K        0K          ts1
ram36.aleph1
IND z900_id1    100K        0K          ts1
ram36.aleph1

"file_list - ADM Library

!1      2          3          4          5
6
!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!
TAB z900          1M          0K          ts0
ram36.aleph1
IND z900_id      100K        0K          ts1
ram36.aleph1
IND z900_id1    100K        0K          ts1
ram36.aleph1
SEQ last_result_set
TRI z76_trigger_1
TRI z601_trigger_1
TRI z31_trigger_1
TRI z77_trigger_1
TRI z30_trigger_1
TRI z30h_trigger_1
TRI z36_trigger_1
TRI z36h_trigger_1
TRI z37_trigger_1
TRI z20_trigger_1
TRI z68_trigger_1
TRI z16_trigger_1
TRI z501_trigger_1
TRI z70_trigger_1
TRI z08_trigger_1
TRI z38_trigger_1
TRI z309_trigger_1
TRI z78_trigger_1

```

■ file_list – COURSE Reading library

```

!1      2          3          4          5
6
!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!
TAB z900          1M          0K          ts0
ram36.aleph1
IND z900_id      100K        0K          ts1
ram36.aleph1
IND z900_id1    100K        0K          ts1
ram36.aleph1
SEQ last_temp_file
SEQ last_media_number
TRI z127_trigger
TRI z127_trigger_1
TRI z127_trigger_2
TRI z13_trigger_1
    
```

■ file_list – ILL Library

```

!1      2          3          4          5
6
!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!
TAB z900          1M          0K          ts0
ram36.aleph1
IND z900_id      100K        0K          ts1
ram36.aleph1
IND z900_id1    100K        0K          ts1
ram36.aleph1
TRI z13_trigger_1
    
```

■ file_list – USR Library:

```

!1      2          3          4          5
6
!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!
TAB z900          1M          0K          ts0
ram36.aleph1
IND z900_id      100K        0K          ts1
ram36.aleph1
IND z900_id1    100K        0K          ts1
ram36.aleph1
SEQ last_publish_seq
TRI z303_trigger_1
TRI z304_trigger_1
TRI z305_trigger_1
TRI z308_trigger_1
    
```