ALEPH
INTEGRATED LIBRARY SYSTEM

ALEPH VERSIONS 16 AND 17

How to Set Up Label Printing

Ex Libris

CONFIDENTIAL INFORMATION

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

This document is provided for limited and restricted purposes in accordance with a binding contract with Ex Libris. The information herein includes trade secrets and should all be considered 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, errors and omissions excepted.

Any references in this document to non-Ex Libris Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Ex Libris product and use of those Web sites is at your own risk.

Table of Contents

This document focuses on printing spine labels (that is, labels which fit on the spine of the book), but the principles described can also be used to set up printing of other labels, such as book pocket labels.

This document explains how you can prepare data for labels, how you can print labels using an external program, and how you can use Ex Libris' **LABEL_PRINT** tool. This **LABEL_PRINT** software is delivered to libraries that request it, and is neither supported nor maintained by Ex Libris. Section 4b in this document details how to configure **LABEL_PRINT.**

1 How to Print Labels

Labels are printed from the **Items** tab in the **Cataloging** module, or from the **Items List or from the Arrival Form** in the **Serials** tab of the **Acquisitions/Serials** module.

In the Acquisition/Serials module there is one method for printing labels:

Highlight the item in the **Items List** and click the **Label** button, or highlight the item in the Check In list and click the **Label** button in the Arrival Form.

In the Cataloging module there are two methods to print labels

Method 1:

Highlight the item in the **Items List** and click the **Label** button.

Method 2:

Select the items menu from the menu bar and select the **Print Item Labels** option. The system will display the **Print Item Labels** window. For each item you wish to print a label for, enter the item's bar code and click **Add.** Click **Print** to print the label(s).

2 How to Set Up Label Printing

Setting up label printing involves the following seven steps:

- **Step 1**: Specify prefixes for each sublibrary/collection in the `tab_label.lng` table.

- **Step 2**: Specify how to split the call number and item description in the `tab_label_parse` table.

- **Step 3**: Determine the printing method.

- **Step 4-a**: Set up the forms `item-copy-label-nn.xsl` and `item-issue-label-nn.xsl`.

- **Step 4-b**: Tailor the label definitions in `prt_srv.ini`.

- **Step 5**: Specify the form(s) to use for your client in the `CAT\TAB\print.ini` file.

- **Step 6**: Supplemental tab_label_parse information
- **Step 7**: Set up `XXX50/tab/edit_field.lng`

Each step is described below in greater detail.

2.1   Step 1: Specify prefixes for each sublibrary/collection in the tab_label.lng table

You can determine the prefix that is printed in front of the call number for each sublibrary/collection. This is specified in the `XXX50/tab/tab_label.lng` (**UTIL I/12**). The structure of the table is:

Col. 1   Sublibrary code

Col. 2   Collection code

Col. 3 Prefix (or suffix) delimited by semicolon '**;**' which denotes a line feed

Example of the table:

```
! 1      2                3
!!!!!-!!!!!-!!!!!!!!!!!!!!!!!!!!>

UGDOC GEN    UGDOC;General
UHLTH REF    UHLTH;Reference
UMUSI        UMUSI;
UEDUC GEN    UEDUC;General
```

In order for these prefixes to appear on the label, lines for `tab-label-01` and `tab-label-02` must be set up in the forms `item-copy-label-nn.xsl` and `item-copy-issue-nn.xsl`.

For example:

```
<xsl:call-template name="generic-line">
   <xsl:with-param name="line" select="./tab-label-01"/>
   <xsl:with-param name="width" select="'30%'"/>
</xsl:call-template>
```

See *Step 4-a: Set up the forms item-copy-label-nn.xsl and item-issue-label-nn.xsl* on page *7*.

2.2   Step 2: Specify how to split the call number and item description in the tab_label_parse table

Note:

The parsed call number and item description fields produced by `tab_label_parse` are only relevant to the printing of spine labels.

You can control, for each call number type, how the call number and the Z30-description are split into separate lines. This is done in the `XXX50/tab/tab_label_parse` table (**UTIL I/15**). For more details see *Step 6: Supplemental tab_label_parse information* on page *15*.

In order for the parsed call number to appear on the label, the lines `call-no-piece-<number>` must be set up for numbers 01-05 in the forms `item-copy-label.xsl` and `item-issue-label-nn.xsl`.

For example:

```
<xsl:call-template name="generic-line">
    <xsl:with-param name="line" select="./call-no-piece-01"/>
    <xsl:with-param name="width" select="'30%'"/>
</xsl:call-template>
```

See *Step 4-a: Set up the forms item-copy-label-nn.xsl and item-issue-label-nn.xsl* on page *7*.

2.3   Step 3: Determine the printing method

By default, labels will be printed using ALEPH's **XML-XSL** printing method. Since this method produces HTML output it is somewhat limited. Instead of the standard XML-XSL print method you can use external software. Ex Libris provides an unsupported **LABEL_PRINT** software package as an external tool, for printing of the forms: *item-copy-label* and *item-issue-label* (only), which provides control of the label size and format.

The `form_print_method` table in the `XXX50/tab` directory determines the printing method, when an external program, such as Ex Libris' **LABEL_PRINT** product, is used.

XML_XSL

If you are going to use the ALEPH **XML_XSL** printing method, do not define `item-copy-label` and `item-issue-label` in this table. Continue to *Step 4-a: Set up the forms item-copy-label-nn.xsl and item-issue-label-nn.xsl* on page *7*.

ALEPH's LABEL_PRINT Tool

If you are going to use ALEPH's **LABEL_PRINT** tool define the printing method in the `form_print_method` table as follows.

Please note that the ALEPH LABEL_PRINT tool supports only the printing of the forms *item-copy-label* and the *item-issue-label*

```
!1                          2  3                        4 5
!!!!!!!!!!!!!!!!!!11!!!!!!!!-!!-!!!!!!!!!!!!!!!!!1111!!!-!-!
item-copy-label             20 EXECUTE LABEL_PRINT
item-copy-label             00 EXECUTE LABEL_PRINT
item-issue-label            20 EXECUTE LABEL_PRINT
item-issue-label            00 EXECUTE LABEL_PRINT
```

Column 1 defines the form name. Column 2 defines the print template format. In this document – **20** is used for a spine label and **–00** is used for an internal label. Column 3 defines that the **LABEL_PRINT** tool will be used.

Note that when **LABEL_PRINT** was installed, a definition was automatically inserted in `alephcom.ini` in the `[PrintExecute]` section pointing to the location of the print program. For example:

```
LABEL_PRINT=D:\Tmp\AL500_16\PrintLabel\prt_srv.exe $1 $2
```
For more details on the `form_print_method` table refer to the table header. Continue to *Step 4-b: Tailor the label definitions in prt_srv.ini* on page 8.

Other External Label Printing Tools

If you are going to use an external tool not supplied by Ex Libris, define the printing method in the `form_print_method` table as follows.

```
!1                               2  3                        4 5
!!!!!!!!!!!!!!!!!!11!!!!!!!!!-!!-!!!!!!!!!!!!!!!!1111!!!-!-!
item-copy-label          20 EXECUTE <COMMAND>
item-copy-label          00 EXECUTE <COMMAND>
item-issue-label         20 EXECUTE <COMMAND>
item-issue-label         00 EXECUTE <COMMAND>
```

In alephcom.ini, define the path of the external label printing tool, in the [Print Execute] section.

Please note that printing of forms other than *item-copy-label* and *item-issue-label* as labels and not as HTML files, can be done only with other external tools and with the ALEPH LABEL_PRINT tool

2.4 Step 4-a: Set up the forms item-copy-label-nn.xsl and item-issue-label-nn.xsl

The form used for printing labels for ISSUE items is `/form_lng/item-issue-label-nn.xsl` (where `nn` is the suffix specified in your `print.ini` file. See *Step 5: Specify the form(s) to use for your client* on page *15*. The form used for printing labels for non-ISSUE items is `/form_eng/item-copy-label-nn.xsl`.

Notes:

The **call-number** and **call-number-2** fields can be parsed (split) for use in spine-labels with the `XXX50/tab/tab_label_parse` table (see *Step 2: Specify how to split the call number and item description in the tab_label_parse table* on page *5*).

Similarly, the item description can be split into logical pieces. This is also controlled by `tab_label_parse`.

You have the option of taking the unbroken call number or the parsed call number. Similarly, you can use an unbroken item description or the parsed item description.

For spine labels, you will want to control the prefix associated with each sublibrary/collection through the use of `XXX50/tab/tab_label.lng`. (See *Step 1: Specify prefixes for each sublibrary/collection in the tab_label.lng table* on page *5)*

2.5   Step 4-b: Tailor the label definitions in prt_srv.ini

This step is only relevant if you are using ALEPH's **LABEL_PRINT** tool.

When **LABEL_PRINT** was installed, the `prt_srv.ini` label definition file was automatically copied into the **LABEL_PRINT** directory. This file contains all the definitions for printing labels. This includes both layout definitions, such as label size and fonts, and content definitions that determine what information will be included in the label.

The `prt_srv.ini` file includes general definitions used for all labels, and specific content and layout definitions for each type of label. The definitions that are not used by default are commented out using the **;** character at the beginning of the row. The order of the definitions is not important. Note that all possible layout definitions are in this file.

2.5.1   General Definitions

```
[label_general]
suppress_empty_lines=y
```

Use this global definition to determine whether empty lines should be suppressed in all label printouts.

`suppress_empty_lines=y` empty lines will not printed.

`suppress_empty_lines=n`   empty lines will be printed. In other words, there will be blank rows wherever there is no data. This

ensures that each label will always contain the same

number of lines. This is important for correct alignment

on label stock.

```
[label_font]
bold=n
name=Arial
```

bold

Determines whether all the characters will be bold (`bold=y`) or not (`bold=n`). The default is not bold.

Name

Determines a default font for all languages. The default font will be used in all languages that do not have a font definition and in all languages that have a font definition that does not include the `face=<font name>` line.

```
        name=<font name>
```

;Font Definitions for Various Character Sets

[lat]
```
face=arial
;face=tahoma
charset=0
codepage=1252
```

[eng]
```
face=arial
;face=tahoma
charset=0
codepage=1252
```

[rus]
```
face=tahoma (cyr)
charset=204
codepage=1251
```

[gre]
```
face=Arial Greek
charset=161
codepage=1253
```

[heb]
```
face=arial (hebrew)
charset=177
codepage=1255
```

[ara]
```
face = Arial (Arabic)
charset=178
codepage=1256
```

[chi]
```
face=ms song
charset=134
codepage=936
```

[jpn]
```
face=ms gothic
charset=128
codepage=932
```

[kor]
```
charset=129
codepage=1361
```

face

Determines the font type.

```
        face=<font name>
```

charset

Determines the language.

```
charset=<number>
```

codepage

Used to convert the Unicode characters to Codepage in order to determine the encoding for the specific language.

```
codepage=<number>
```

```
[barcode]
;fontname=UPC-A
fontname=CarolinaBar-B39-2.5-22x158x720
height=-6
with_chars=Y
short=N
```

The [barcode] definition is used when printing an actual bar code. It is called using the `use_barcode=z30-barcode` command, from within the labels definition.

fontname

Determines whether the barcode will be printed as a number

```
fontname=UPC-A
```

or a barcode

```
fontname=CarolinaBar-B39-2.5-22x158x720
```

Note that the CarolinaBar font can be tailored.

height

Determines the height of the barcode. Note that if this parameter is defined then the short parameter must be set to N.

```
height=<number>
```

with_chars

Determines whether the barcode characters will be displayed under the barcode
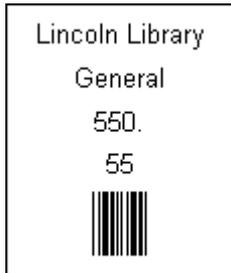
```
with_chars=Y
```
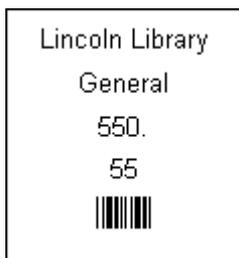
or not

```
      with_chars=N
```

The default is N.

short

Determines whether the barcode will be regular height (short=N)

or short (short=Y)

Definitions for each label

Each label has two sections:

`[<label-template>]` contains the content definitions.

`[<label-template>-layout]` contains the layout definitions.

For example `[item-copy-label-20]` and `[item-copy-label-20-layout]`.

2.5.2   Content Definitions
Example: item-copy-label-20

```
[item-copy-label-20]
copies=4
z30-sub-library=1
z30-collection=2
call-no-piece-01=3
call-no-piece-02=4
```

```
call-no-piece-03=5
call-no-piece-04=6
call-no-piece-05=7
use_barcode=z30-barcode
```

copies

Determines how many copies of the label will be printed.

```
copies=<number>
```

<field>

Determines the fields that will be displayed and their order. The syntax is:

 <field>=<line number>,<maximum number of characters>

The `<maximum number of characters>` parameter is optional. The number of characters that fits in a line depends on the size of the label, the width definition in the layout section and the font that is being used. Note that if the content of a field is longer that the width of the label, and the `<maximum number of characters>` parameter is not defined, the text will continue onto the next line.

Note that the width of the label is defined either in centimeters or inches, whereas the <maximum number of characters> is defined in actual characters.

Example:

The sublibrary name will be printed on the first line of the label:

```
z30-sub-library=1
```

The first 20 characters of the sublibrary name will be printed on the first line of the label:
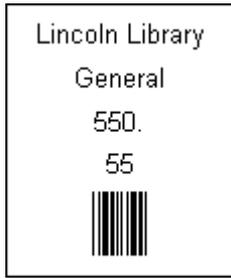
```
z30-sub-library=1,20
```

use_barcode=z30-barcode

Used to print an actual barcode and not the barcode number. The barcode fonts defined in the `[barcode]` section will be used.

Example:

Barcode:

```
Lincoln Library
General
550.
55
▌▌║▌
```

2.5.3 Layout Definitions

Example: item-copy-label-20-layout

```
[item-copy-label-20-layout]
columns=3
rows=4
; all numbers depend on 'units'
; cm means 1/1000 cm
; inch means 1/1000 inch
units=cm
width=3400
height=4600
rows.spacing=500
columns.spacing=500
;topdown=no
;sheet=yes
;print.margin.left = 250
;print.margin.right = 250
;print.margin.top = 250
;print.margin.bottom = 250
; orientation
;        0 - Printer default (default)
;        1 - Landscape
;        2 - Portrait
;print.orientation = 0
```

Columns

Determines the number of labels across the page.

```
        columns=<number>
```

Rows

Determines the number of labels down the page.

```
        rows=<number>
```

Units

Determines whether the various measurements are in units of 1/1000 centimeters or 1/1000 inches.

```
        units=<cm/inch>
```

Width

Determines the width of the label.

```
width=<number>
```

In the example above the width is 3.4 centimeters, because the units are defined as 1/1000 centimeter and the width is set to 3400.

Height

Determines the height of the label.

```
height=<number>
```

In the example above the height is 4.6 centimeters, because the units are defined as 1/1000 centimeter and the height is set to 4600.

rows.spacing

Determines the space between each row of labels.

```
rows.spacing=<number>
```

In the example above the space between each row is 0.5 centimeters, because the units are defined as 1/1000 centimeter and the parameter is set to 500.

columns.spacing

Determines the space between each column of labels.

```
columns.spacing=<number>
```

In the example above the space between each column is 0.5 centimeters, because the units are defined as 1/1000 centimeter and the parameter is set to 500.

topdown

If `topdown=yes` the labels will be printed by columns, that is the left column of labels will be printed first, then the second column, and so on. If `topdown=no`, the labels will be printed by rows.

The default is **no**.

sheet

Use `sheet=yes` for printing separate pages of labels. Use `sheet=no` for printing with continuous pages.

The default is yes.

print.margin

The print.margin definitions determine the page's printing margins.

```
print.margin.<direction>=<number>
```

In our example, the page has a margin of 0.25 centimeters because the units are defined as 1/1000 centimeter and the parameters are all set to 250.

print.orientation

Determines whether the page will be printed in Landscape (`1`), Portrait (`2`), or according to the printer's default definition (`0`).

```
print.orientation=<number>
```

```
The default is 0.
```

2.6   Step 5: Specify the form(s) to use for your client

```
Catalog\Tab\print.ini
```

```
Alephcom\Tab\print.ini
```

```
Acq\Tab\print.ini
```

Column 2, **Version ID,** in the <module>`\Tab\print.ini` file determines which form is used for printing the label. The forms for **Method 1** (highlight the item in the **Items List** and click the **Label** button) are determined in the **ItemLabel** line and the forms for **Method 2** (Select the **Utilities** menu from the menu bar and select the **Print Item Labels** option) are determined in the **ItemLabelPrint** line.

If you specify "00" as the version ID, the system will look for a version with "-00" as a suffix.

You need to specify the suffixes for the form(s) that were set up in *Step 3: Determine the printing method* on page 6.

```
!      1                              2   3 4      5
```

ItemLabel   00 N P label.prn

2.7   Step 6: Supplemental tab_label_parse information

A. Call Number Parsing Routines
There are currently five different routines that can be used to parse call numbers.

```
1. parse_call_no_default
```

```
2. parse_call_no_lc_1
```

```
3. parse_call_no_lc_2
```

```
4. parse_call_no_lc_3
```

```
5. parse_call_no_dw_1
```

Routines 2-4 are intended to be used with LC (Library of Congress) and NLM (National Library of Medicine) call numbers, although some sites have chosen to use the parse_call_no_default even for LC.

Routines 1-4 break on a space or a $$i. {Note: The caret ("^") prints as a space but does **not** cause a line break.}

1. parse_call_no_default
This routine breaks on a space or a $$i.

2. parse_call_no_lc_1
In addition to parse_call_no_default, this routine:

- breaks between the letter and number components of the classification part of the call number (even if there is no space) and

- breaks on a decimal preceding a letter (even if it is not preceded by a space).

3. parse_call_no_lc_2

This routine is like parse_call_no_lc_1, except that in addition, it breaks pieces longer than 8 characters on the nearest preceding decimal.

4. parse_call_no_lc_3 I
This routine is like parse_call_no_lc_2, except that no space is added to force a line break after the class code.

Here is how the following call numbers would be parsed under routines 1-4:

$$hKJV444.21804 A7$$iL63 1805

and

$$hG635.H4$$iA3 1989^a

1. parse_call_no_default:

KJV444.21804

A7

L63

1805

and

G635.H4

A3

1989 a

2. parse_call_no_lc_1:
KJV

444.21804

A7

L63

1805

and

G

635

H4

A3

1989 a

3. parse_call_no_lc_2:

KJV

444

21804

A7

L63

1805

and

G

635

H4

A3

1989 a

4. parse_call_no_lc_3:

KJV444

21804

A7

L63

1805

and

G635

H4

A3

1989 a

5. parse_call_no_dw_1 I
This is a site-specific parsing routine. Break at the end of the first string of numerals terminated with a space or a full stop. The full stop remains on the first line. Then the following rules apply:

Non-periodicals: Start a new line after every three numeric characters or after every space (whichever occurs first).

Periodicals: All call numbers that begin with P (other than those beginning with P/) and end with either a suffix consisting of a five-digit string or a five-digit string followed by a decimal point and a two-digit string, break at the suffixes decimal point, or after the five-digit string. All characters/digits preceding the suffix for these "P" call numbers are handled as non-periodicals.

B. Description Parsing Routines
There are currently 4 different routines that can be used to parse descriptions.

1. parse_desc_default

2. parse_desc_semicolon

3. parse_desc_semi_noparens

4. parse_desc_niso_8

1. parse_desc_default:

This routine splits the item description at spaces.

2. parse_desc_semicolon:

If a semicolon is present in the description, it splits on semicolons. Otherwise, the item description splits on commas.

3. parse_desc_semi_no_parens:

This routine is the same as parse_desc_semicolon, except that only the part of the item description up to the first "(", "[" or "<" is used.

4. parse_desc_niso_8

The parse_desc_niso_8 routine can be used in tab_label_parse to parse item descriptions formatted according to the ANSI/NISO Z39.71 standard (**Holdings Statements for Bibliographic Records**). This routine also attempts to limit each line of the item description to 8 characters, using the following algorithm:

1. Remove any trailing information in parentheses, square brackets, or angle brackets.

2. Break at "+" or "&", which indicate bibliographic units, retaining the "+" or "&".

3. Within a bibliographic unit, break on "**:**" but do not retain the colon in the label.

4. Within a level, break on "**,**" and "**;**", retaining the punctuation.

5. If a segment produced by step 4 is longer than 8 characters, break up to two times on the "-" and "/" characters, retaining them on the label.

6. If more than ten lines of item description would be produced based on steps 2 - 5, append the rest of the item description to the tenth line, even if it ends up longer than 8 characters.

Example: The item description

v.120:no.1:pt.A-B,D + Index + Supplement (1998)

would be parsed into

v.120

no.1

pt.A-B,

D

+ Index

+ Supplement

Note: if no matching program is specified, or if the program does not exist, parse_desc_default will be used.

2.8   Step 7: Set up XXX50/tab/edit_field.lng

Edit   all   parsing   routines   using   the   following   definition   in XXX50/tab/edit_field.lng:

```
1 L LOC## D B -2
2                    k A ^
2                    h A ^
2                    l A ^
2                    j A ^
2                    i A ^
2                    m A ^
```

Note that carets (^) are mandatory for the LOC D definition.