



ALEPH VERSION 19.01

How to Use the MARCIVE Document Loader (file-99)

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.

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 Ex Libris has no liability for materials on those Web sites.

Copyright Ex Libris Limited, 2008. All rights reserved.

Documentation produced January 2008

Document version 1.0

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

Table of Contents

1	LOADING MARCIVE RECORDS INTO ALEPH - OVERVIEW	4
1.1	Running the MARCIVE Loader	5
1.2	Help on Fields	5
1.3	Loader Logger – The Z73 Oracle Table	7
2	CATALOG RECORD LOADING TABLES.....	8
2.1	tab_merge.....	8
2.2	tab_merge_overlay.....	8
2.3	tab_match.....	9
2.4	tab_fix	9
2.5	tab_loader table.....	9
2.6	tab_loader_def table.....	11
2.6.1	Default Values Handled	12
2.7	tab_mapping table	14
3	LOADING WORKFLOW.....	15

1 Loading MARCIVE Records into ALEPH - Overview

[MARCIVE](#) is a commercial service which provides catalog records for U.S. government publications (retrospective and current) that can be loaded into a library's local online catalog and a Shipping List Service that provides SuDocs labels, brief MARC records, smart barcode labels, and shelflist cards.

The MARCIVE document loader is a batch process designed to be run by libraries on a periodic basis as they receive MARCIVE data files. These files are normally received on a weekly basis and then updated on a monthly basis. There are two types of MARCIVE government document records files:

- **Weekly shipping lists** – A brief catalog of the documents the library receives from the GPO.
- **Monthly full cataloging** - Full MARC cataloging of government documents, which may or may not correspond to the shipping list records.

The MARCIVE loader is run from the Load Catalog Records option from the Services menu in the ALEPH Cataloging module. It can be added to the libraries' job_list (UTIL E/5) for regular automatic initiation.

There are several configuration tables associated with this process, some of which supply default values for automatically created records and others which determine how the loaded records will be merged and what fix procedures should be carried out.

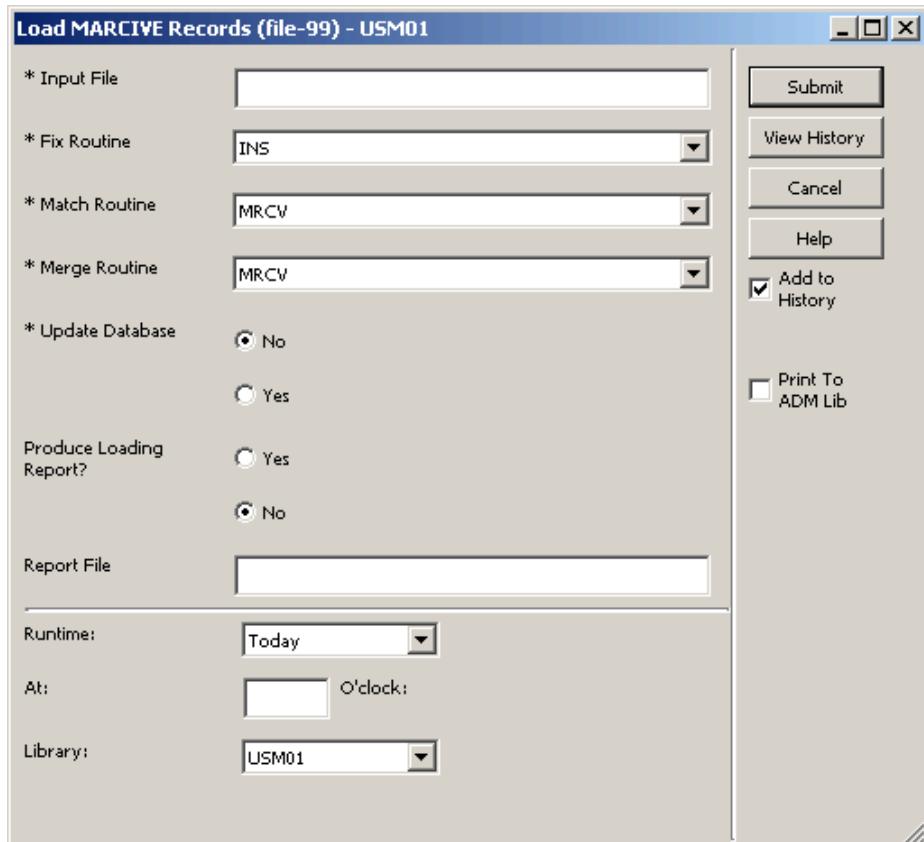


Figure 1 - The MARCIVE loader (file_99)

1.1 Running the MARCIVE Loader

To run the MARCIVE loader:

- 1 This process can either use the MARC file as an input file directly or as a file converted to ALEPH Sequential format.

An institution can run the Modify MARC Records (p_file_08) batch utility before running p_file_99 if global changes to the MARC records themselves are desired. This can only be done only if the file is in ALEPH Sequential format. If you want to use the Modify MARC Records (p_file_08) service before running the loader, then convert the MARC file to ALEPH Sequential format using the Convert MARC Records - Step 1 (file_01) and Convert MARC Records - Step 2 (file_02) batch jobs consecutively (these batch jobs are also run from the Load Catalog Records option of the Services menu in the Cataloging module).

- 2 Click **Load Catalog Records** and then click **Load MARCIVE Records (file-99)** from the **Services** menu to open the MARCIVE loader from the Cataloging GUI. The Load MARCIVE Records (file-99) batch service is displayed.
- 3 Fill in the fields according to the instructions in the Help for the batch service.
- 4 Click **Submit** to send the batch job to the queue.

1.2 Help on Fields

Input File

This field is mandatory. Enter the name of the input file. If the Input file is located in the library's scratch directory, simply enter the name of the file as listed. If the Input file is located in another directory, enter the path relative to the scratch directory or enter the full path name.

An example of a relative path is `../source/filename`.

An example of a full path is `/aleph/a50_5/exu50/source/filename`.

Fix Routine

This field is optional. You can leave the Fix routine parameter blank or set it to one of the codes in the `tab_fix` table of the bibliographic library. Fix routines are standard library-defined procedures that automatically "fix" or make changes to cataloging records.

For example, if you select INS, the system will "fix" the records that are being loaded according to the fix programs linked to the INS routine which is the fix routine performed when records are sent to the server from the Cataloging module. The specific bibliographic record fixes that should be applied to incoming MARCIVE records are a matter of library policy. Fixes that are specific to the incoming

MARCIVE data should be added as a fix_doc group to the tab_fix table. For example, the code MRCV in column 1 can be used as the fix routine section to be used for the MARCIVE loader.

Match Routine

Select the match routine that specifies the match specifications for checking the input file against the database. Match routines are defined in the tab_match table of the library's tab directory.

Note:

If no merge routine is selected, the F99 routine will be used as the default.

Merge Routine

Select the merge routine to be performed. Merge specifications (which include the merging records or the replacing of records old records by new incoming records) are defined in the tab_merge table.

Note:

If no merge routine is selected, the F99 routine will be used as the default.

Update Database

Yes or No. If you choose not to update the database, the system performs all checks and prepares a log file with messages, but no records are written on the database files.

The Load MARCIVE Records (file-99) batch process does not produce output files, with the exception of the log file. In order to load the records to the library's catalog, it is necessary to set the update database flag to "Y". Records updated or created by this service have the cataloger stamp set to "MARCIVE" unless otherwise specified in the default values table.

When running the batch service with "Update Database - No", the job is executed in a simulation mode and no actual records are created. The job adds a notice about creating an item without actually performing the necessary checks and validations.

For the item creation step, the full checks and validations procedures are performed when the batch service is executed with "Update Database – Yes".

Produce Loading Report

This field determines whether to produce a report regarding the current load.

Report File

Enter the report's file name.

The output file contains the following information:

Administrative library, OWN Filed, Open Date, Log No., Action (Success, Failed or Information), Message information.

This information is saved as well in the Z73 Oracle table (Loader Logger).

See Section [1.3 Loader Logger – The Z73 ORACLE Table](#) on page 7 for more information.

Run Time/at

This field is mandatory. You must enter the day and hour the batch process is to run. The system uses a 24-hour clock.

Library

Select the library on which you wish to run the service. The libraries that are displayed are filtered according to the user's permitted libraries.

The log file for the MARCIVE loader can be found in \$alephe_scratch. The log file contains a record of the number of documents that have been merged.

1.3 Loader Logger – The Z73 Oracle Table

The Z73 Oracle table stores information on the running of the MARCIVE (p-file-99) and OCLC loaders. The log has a running number. This number also serves as the key of the Z73 table. The sequence has to be defined under Z52 of the Bibliographic library – “last-loader-log-no”.

Each message created by the MARCIVE or OCLC loaders is saved in a different Z73 record.

Each message contains a type (I=Information/ S=success /F=Fail) and a text. The text is defined under \$aleph_error_eng/generic_loader.

Note that a new Z73 log number will be generated for each file-99 batch run. For each loaded record in a specific run a new log sequence will be generated.

2 Catalog Record Loading Tables

The following are examples of the tables involved in the loading mechanism. The tables reproduced below are: tab_merge, tab_merge_overlay, tab_match, tab_fix, tab_loader, tab_loader_def and tab_mapping.

2.1 tab_merge

1	2	3
OVERLAY-03	merge_doc_overlay	03
OVERLAY-05	merge_doc_overlay	05
OCLC	merge_doc_adv_overlay	06
HVD	merge_doc_adv_overlay	01
MRCV	merge_doc_overlay	02
RLIN	merge_doc_adv_overlay	03
F99	merge_doc_overlay	07

The tab_merge table is used to define the merging routine to be performed. Column 3 contains the merging procedure to be performed from tab_merge_overlay (column 1).

Note:

If no merge routine is selected, the F99 routine will be used as the default.

2.2 tab_merge_overlay

1	2	3	4
02	1	Y	#####
02	1	N	051##
02	1	N	245##
02	2	Y	245##
02	2	Y	650##
02	2	Y	008
02	2	N	001
07	1	N	#####
07	1	Y	035##
07	1	Y	930##
07	1	Y	940##
07	1	Y	946##
07	1	Y	948##
07	1	Y	960##
07	1	Y	961##
07	2	Y	#####

The tab_merge table is used to define the merging procedures, which fields to retain and which fields to take from the incoming record.

2.3 tab_match

! 1	2	3
!!!!!!-!!!!!!!!!!!!!!-!!!!!!-!!!!!!-!!!!!!->		
F99	match_doc_gen	TYPE=IND, TAG=909, SUBFIELD=a, CODE=909
YBP	match_doc_uid	I-ISBN
RLIN	match_doc_uid	T-020
STIDN	match_doc_uid	I-STIDN
OCLC	match_doc_uid	T-010
OCLC2	match_doc_uid_2	035
MRCV	match_doc_uid	T-909
ISSN	match_doc_gen	TYPE=IND, TAG=022, SUBFIELD=a, CODE=ISSN

This table is used to define the match specifications for checking the input file against the documents in the database.

Note:

If no merge routine is selected, the F99 routine will be used as the default.

2.4 tab_fix

! 1	2	3
!!!!!!-!!!!!!!!!!!!!!-!!!!!!-!!!!!!-!!!!!!->		
MRCV	fix_doc_notis_1_c	
MRCV	fix_doc_notis_3	
MRCV	fix_doc_notis_4	

This table is used to define the fix procedures that the loaded records will undergo.

2.5 tab_loader table

The ADM library table – tab_loader defines processing regarding the creation of the holding records, items, orders, budget transactions, and load information.

COL 1. – 049 Code – Holding code in tag 049\$\$a of incoming BIB record. If 049\$\$a is missing from the BIB input, the value entered in the batch service is taken into account.

COL 2. – OWN code for the bibliographic record – Optional – The data entered in this column will be used for the OWN field of records added/updated by the OCLC server processes.

COL 3. OWN Group code – Optional – This column is used in order to group different OWN codes to a common code. This will be used when a library uses different 049 codes and it creates a different OWN code for each 049 code. If no OWN group is defined, a separate BIB record will be created for each separate OWN code. If several OWN codes are assigned to the same OWN group, each OWN code is added to a single shared BIB record. The OWN group code is not written in the BIB record, it is used only in this table for grouping.

COL 4. – Item creation method. Possible values:

- 0 – Do not create items. This is the default value.
- 1 – Create item for new records only. The system will generate a barcode.
- 2 – Create first and additional items using the definitions in 949/852 fields.
 - If field is missing (or barcode is missing) the system will generate a barcode.
 - Create items even though BIB already has items.
- 3 – Create first items using the definitions in 949/852 fields. If field is missing or barcode is missing) the system will generate a barcode.
 - Does not create item if BIB already has items.
- 4 – Create first and additional items using the definitions in 949/852 fields.
 - If field is missing (or barcode is missing) don't create item. In this case ADM record will be Created.
 - Create items even though BIB already has items.
- 5 – Create first items using the definitions in 949/852 fields. If field is missing (or barcode is missing) do not create items. In this case ADM record will be created.
 - Do not create items if BIB already has items.

Item creation in the load is based on two fields in the BIB record:

- Field 949: \$\$a (for barcode), \$\$c (for Shelf list)
- Field 852: \$\$p (for barcode), \$\$a (for Shelf list)

An Item is created for every 949/852 field+Shelf list occurrence.

The uniqueness check is performed.

For the system generated barcode, the program looks for
CSCR-OCLC-Z30-BARCODE in tab_checksum

COL 5. – HOL record creation method:

- 0 – Do not create HOL record. This is the default value.
- 1 – Create the first HOL for new records only.
- 2 – Create HOL records (check uniqueness with 852 \$\$b + \$\$c -if matching HOL record exists, a new one will not be created). Add information that is in brackets in 049 to 852 subfields k,m
- 3 – Create HOL records (check uniqueness with 852
\$\$b + \$\$c -if matching HOL record exists, a new one will not be created).
Do not add information that is in brackets in 049 to 852;
- 5 – Always create new HOL record.

The information for the HOL record is located in the ADM library table:
tab_mapping.

COL 6. – Order record creation method;

- 0 – Do not create Order record. This is the default value.
- 1 – Create the first Order for new ADM record, or additional order for existing ADM record.

COL 7. – Budget transaction creation method

- 0 – Do not create a Budget transaction record. This is the default value.
- 1 – Create new Budget transaction records.

COL 8. – Multi match bibliographical record load method

0 – Do not create bibliographical record. This is the default value.

1 - Create new bibliographical record.

1	2	3	4	5	6	7	8
!!!!!!	!!!!!!	!!!!!!	!!!!!!	!!!!!!	!!!!!!	!!!!!!	!!!!!!
NO-OWN			3	3	1	1	0
WID	MASTER50		0	0	0	0	0
CUV7	MASTER50		1	1	1	0	1

The use of 949 is for vendors that supply extra item information in the input file.

The information in the vendor input file should be the same as for p_file_96.

The information for the holding records is located in tab_mapping table the minimum configuration should be for the holding's sublibrary.

The information in the table cannot contain hashes (#).

There should be a single tab_loader table for all libraries that are using p_file_90 and all the 049 should be unique to the library.

Note that tab_loader is used also by the following load serviced (in addition to file-99):

- Advanced Generic Vendor Records Loader (file-90)
- Load OCLC Records (file-93)
- OCLC server

2.6 tab_loader_def table

tab_loader_def table is located in the administrative library's /tab/ directory and includes default values for fields in the items(Z30), orders (Z68), and budget transaction records (Z601) that are created using this service.

The default information for the holding record should be populated in tab_mapping table.

COL 1.– Z30/Z68/Z601 field name.

COL 2. – 10; Match 049 value in the incoming bibliographic record or hashes to indicate that the value in column 3 will be used regardless of the bibliographic record.

COL 3. Value to place in the field defined in column 1

1	2	3
!!!!!!	!!!!!!	!!!!!!>
Z30-SUB-LIBRARY	BXMM	WID
Z30-SUB-LIBRARY	BXMA	MED
Z30-SUB-LIBRARY	YBPDD	WID
Z30-SUB-LIBRARY	#####	WID

Z30-COLLECTION	BXMM	GEN
Z30-COLLECTION	BXMA	GEN
Z30-COLLECTION	#####	GEN
Z30-MATERIAL	#####	BOOK
Z30-ITEM-STATUS	#####	01
Z30-CATALOGER	#####	MASTER
Z30-CALL-NO-TYPE	#####	0
Z30-ITEM-PROCESS-STATUS	#####	OR
Z68-SUB-LIBRARY	BXMM	WID
Z68-SUB-LIBRARY	BXMA	MED
Z68-SUB-LIBRARY	YBPDD	WID
Z68-SUB-LIBRARY	#####	WID
Z68-ORDER-TYPE	#####	S
Z68-ORDER-STATUS	#####	SV
Z68-ARRIVAL-STATUS	#####	P
Z601-VENDOR-CODE	#####	YBP
Z601-USER-NAME	#####	MASTER
Z601-CURRENCY	#####	USD
Z601-TYPE	#####	ENC
Z601-ORIGINAL-SUM	#####	0010

2.6.1 Default Values Handled

Be sure that configuration is correctly set up for creation of items, orders, and encumbrances; incorrect configuration can lead to meaningless data.

Column 1 in tab_loader_def can contain the following values (Bold fields are mandatory):

For Z30 ITEM creation:

Z30-SUB-LIBRARY
Z30-COLLECTION
Z30-MATERIAL
Z30-ITEM-STATUS
Z30-CATALOGER
Z30-NO-LOANS
Z30-ALPHA
Z30-CALL-NO-TYPE
Z30-CALL-NO
Z30-CALL-NO-KEY
Z30-CALL-NO-2-TYPE
Z30-CALL-NO-2
Z30-CALL-NO-2-KEY
Z30-DESCRIPTION
Z30-NOTE-OPAC
Z30-NOTE-CIRCULATION
Z30-NOTE-INTERNAL
Z30-INVENTORY-NUMBER
Z30-INVENTORY-NUMBER-DATE
Z30-LAST-SHELF-REPORT-DATE
Z30-PRICE

Z30-DOC-NUMBER-2
Z30-SCHEDULE-SEQUENCE-2
Z30-COPY-SEQUENCE-2
Z30-VENDOR-CODE
Z30-INVOICE-NUMBER
Z30-LINE-NUMBER
Z30-PAGES
Z30-ISSUE-DATE
Z30-EXPECTED-ARRIVAL-DATE
Z30-ARRIVAL-DATE
Z30-ITEM-STATISTIC
Z30-ITEM-PROCESS-STATUS
Z30-COPY-ID
Z30-HOL-DOC-NUMBER
Z30-TEMP-LOCATION
Z30-ENUMERATION-A
Z30-ENUMERATION-B
Z30-ENUMERATION-C
Z30-ENUMERATION-D
Z30-ENUMERATION-E
Z30-ENUMERATION-F
Z30-ENUMERATION-G
Z30-ENUMERATION-H
Z30-CHRONOLOGICAL-I
Z30-CHRONOLOGICAL-J
Z30-CHRONOLOGICAL-K
Z30-CHRONOLOGICAL-L
Z30-85X-TYPE

For Z68 ORDER creation:

Z68-ORDER-DATE
Z68-E-TERM-PERCENT
Z68-ORDER-TYPE
Z68-ORDER-STATUS
Z68-ARRIVAL-STATUS
Z68-VENDOR-CODE
Z68-RUSH
Z68-DELIVERY-TYPE
Z68-ORDER-DELIVERY-TYPE
Z68-TARGET-FLAG
Z68-E-CURRENCY
Z68-METHOD-OF-ACQUISITION
Z68-SEND-METHOD
Z68-MATERIAL-TYPE
Z68-AUTO-CLAIM
Z68-ORDERING-UNIT
Z68-VENDOR-NOTE
Z68-INVOICE-STATUS
Z68-ORDER-NUMBER-1

Z68-ORDER-NUMBER-2
Z68-ORDER-GROUP
Z68-LIBRARY-NOTE
Z68-VENDOR-REFERENCE-NO
Z68-QUANTITY-TEXT
Z68-E-NOTE
Z68-E-TERM-PERCENT
Z68-AUTO-CLAIM

For Z601 BUDGET TRANSACTION creation:

Z601-BUDGET-NUMBER

Z601-ORIGINAL-SUM
Z601-CURRENCY
Z601-SEQUENCE
Z601-VENDOR-CODE
Z601-INVOICE-NUMBER
Z601-LINE-NUMBER
Z601-USER-NAME
Z601-OPEN-DATE
Z601-TYPE
Z601-NOTE
Z601-CREDIT-DEBIT
Z601-CURRENCY-RATIO
Z601-PAID

Note that tab_loader is used also by the following load serviced (in addition to file-99):

- Advanced Generic Vendor Records Loader (file-90)
- Load OCLC Records (file-93)
- OCLC server

2.7 tab_mapping table

The ADM library table: tab_mapping, defines mapping values for the holding records.

COL 1. – Holding Code

COL 2. – Source field Code

COL 3. – Source Subfields

COL 4. – Target Code

COL 5. – Target Subfields

COL 6. – Text Command

COL 7. – Overlay Flag

COL 8. – New Line Flag

Y – for all occurrences

N – for first occurrence of source tags

	1	2	3	4	5	6	7	8
MAIN	!!!!!!	-!!!!-	!!!!-	!!!!-	!!!!-	!!!!-	!!!!!!	-!-!
MAIN		050##	a	8528	h		Y	N
MAIN		050##	b	8528	i		Y	N
LCN-2-HOL	541##	abcde	541	abcde			Y	Y
LCN-2-HOL	541##	fho39	541	fho39			Y	Y
LCN-2-HOL	561##	ab39	561	ab39			Y	Y
LCN-2-HOL	590##	ab9	590	ab9			Y	Y
LCN-2-HOL	690##	ab9	690	ab9			Y	Y

3 Loading Workflow

After p-file-99 has been submitted, the MARCIVE loader will handle records as follows:

1. The loader looks for the tab_loader table which contains the holding code from field 049. The ADM library where tab_loader is the ADM environment for the loading.
2. The loader looks for matching records using the match section given as a parameter in the batch.
3. The loader filters the matched records according to the owner group from tab_loader col.3. It leaves only those records which have an OWN field that belongs to the OWN group as defined in tab_loader.

If there is more than one match, an error is returned.

If there is one matched record, then the loader merges the new loaded record with the existing one according to the merge program given as a parameter in the batch, adds a new OWN field, and saves it.

If there is no matched record, then the loader creates a new bibliographic record, adds OWN field, and then executes the fix routine given as parameter in the batch and saves the record.

4. The loader creates an ADM record in the ADM library.
5. The loader creates an HOL record, according to the definition in tab_loader . The HOL library is determined using tab_library_relation.

The HOL fields are created using the tab_mapping table.

The HOL record will not be written if a record already exists with the same 852 field with the following subfields: \$b, \$c, \$h, \$i, \$j, \$k, \$m, \$l.

6. The loader creates item record(s) according to the definitions in tab_loader_def.

The new item(s) will be linked to a HOL record if the later record's 852 \$b and \$c subfields match the Z30-SUB-LIBRARY and Z30-COLLECTION fields of the first record.

7. If an item with the same barcode already exists, no item will be created, but if the item does not have a HOL link then it will be linked to the HOL record if it exists. Again, the match is done by comparing subfields \$b and \$c of the 852 field with the Z30-SUB-LIBRARY and Z30-COLLECTION fields of the item.

Note

If position 05 of the LDR field of the input record contains "d" or position 23 holds X'03', then the matching bibliographic record and all the associated administrative records will be deleted. A STA \$\$a DELETED will be added to the record.

However, if the record has any of the following records, it will not be deleted and an appropriate warning message will be displayed:

Loans (Z36)

Hold Requests (Z37)

Photocopy Request (Z38)

Acquisition Order (Z68)