# Z00 – DOCUMENT RECORD (DOC) and Z00H – DELETED RECORD

## 1. Z00 and Z00H

A document record in the ALEPH system is the basic information record. Each record is uniquely identified by a 9-digit system number assigned sequentially by the system.

In the bibliographic, authorities and holdings libraries, the document record is the "cataloging" record. In most instances these records will be in standard MARC 21, UKMARC, UNIMARC, MAB, etc. format for fields and data. The setup for fields and data validity is set in library tables, and can, therefore, depart from standards. The ADM (administrative) library also uses a document record, but the fields and data are ALEPH standard.

In the database the document record is stored in the Z00 table. The structure of the Z00 record in ALEPH is as follows[1]:

```
01  Z00-DOC.
  02 Z00-DOC-NUMBER            PICTURE 9(9).
  02 Z00-NO-LINES              PICTURE 9(4).
  02 Z00-DATA-LEN              PICTURE 9(6).
  02 Z00-DATA                  PICTURE X(45000).
```

| Z00-DOC-NUMBER | M | 9(9) Leading zeroes. | **DESC:** Document system number. |
|---|---|---|---|
| Z00-NO-LINES | M | 9(4) | **DESC:** Number of fields (lines) in the record |
| Z00-DATA-LEN | M | 9(6) | **DESC:** Length of the record's data content. |
| Z00-DATA | M | X(45000) | **DESC:** Record's content. The information in this field is compressed according to an internal ALEPH algorithm. For the loading and manipulation of the data, you can create a file in ALEPH Sequential format (refer to the Data Loading and Manipulation section below). |

---

[1] In the table, (X) denotes an alphanumeric field and (9) denotes a numeric field. All alphanumeric fields are left justified with trailing spaces; all numeric fields are right justified with leading zeroes. Decimal positions are indicated by the symbol V. The total number of decimal digits is designated by the number of nines following the V.

Deleted documents are stored in the Z00H table. When a document is deleted it is stored in the Z00H table if the CREATE-Z00H flag in the library's tab100 table is set to "Y". The Z00H table is for statistical purposes only, and does not imply that there is capability to restore.

Following is the structure of the table:

```
01  Z00H-DOC.
   02 Z00H-DOC-NUMBER              PICTURE 9(9).
   02 Z00H-NO-LINES                PICTURE 9(4).
   02 Z00H-DATA-LEN                PICTURE 9(6).
   02 Z00H-DATA                    PICTURE X(45000).
```

## 2. Data Loading and Manipulation

Cataloging records can be loaded into the ALEPH system using an ALEPH Sequential format flat file. This file is an intermediate file for the loading and the handling of the data that cannot be directly manipulated in the format needed for the Z00-DATA field.

Users can create the ALEPH Sequential format file according to the specifications below. In addition, there are procedures that convert external files that are in standard format (MARC format) to ALEPH Sequential. This is done by two procedures that can be run from the Services menu of the Cataloging module, Convert MARC Records - Step 1 (p_file_01) and Convert MARC Records - Step 2 (file-02)[2].

Following format is the structure of the ALEPH Sequential record:

| SECTION | TYPE | DESCRIPTION |
|---|---|---|
| Document number | 9 digits – 9(9) | Unique identifier of the record |
| Filler | 1 character – X(1) | Blank |
| Field code | 5 characters – X(5) | Tag + indicators. Note that The LDR and the FMT are mandatory fields[3]. |
| Filler | 1 character – X(1) | Blank |
| Alpha | 1 character – X(1) | Character set of the field. **L** = For all scripts[4] |

---

[2] Note that currently the maximum limit size for records is 62000 bytes. When this limit is reached, an error message is displayed in the log file of the p_file_02 process and the record is converted up to the limit.

[3] The LDR is a standard MARC field. The FMT contains a two-character code that specifies the format of the record (for example, BK for book or MP map). When the ALEPH Sequential file is created by the Convert Records processes – p_file_01 and p_file_02 – these fields are automatically added.

[4] For non-UTF data (sequential files produced from pre-Unicode ALEPH systems or external data that contains fields in different encoding systems), this section can be used to identify the character set to

| Filler | 1 character – X(1) | Blank |
|---|---|---|
| Text | Up to 2000 characters | Field text in UTF-8 encoding. Note that subfield codes are denoted by two consecutive dollar signs ($$). In addition, use the caret (^) sign to denote a blank in MARC 21 fixed fields. For UNIMARC fixed fields, use the hyphen (-) sign. |

Following is an example of a bibliographic record in ALEPH sequential format:

```
000001059 FMT   L BK
000001059 LDR   L 00641nam^^22002291^^4500
000001059 001   L AAJ0078
000001059 008   L 000814s1959^^^^ilua^^^^^^^^^^00000^eng^^
000001059 010   L $$a58011949
000001059 035   L $$a(OCoLC)99058213
000001059 040   L $$cCarP
000001059 0500  L $$aHB871$$b.H37
000001059 051   L $$cCopy 2.
000001059 051   L $$cCopy 3.
000001059 090   L $$aHB871$$b.H376
000001059 10010 L $$aHauser, Philip Morris,$$d1909-$$eed.
000001059 24504 L $$aThe study of population:$$ban inventory
and appraisal.
000001059 2600  L $$a[Chicago]$$bUniversity of Chicago
Press$$c[1959]
000001059 300   L $$a864 p.$$billus.$$c25 cm.
000001059 650-0 L $$aDemography.
000001059 650-0 L $$aPopulation.
000001059 70010 L $$aDuncan, Otis Dudley,$$ejoint ed.
```

To summarize, ALEPH records are limited to:

- 5000 subfields
- 45000 bytes

Each field is limited to 2000 bytes.

Note that if the file to be converted contain fields longer than 2000 bytes in MARC records, these fields are converted into ALEPH Sequential format in the following manner:

---

perform the appropriate conversion. Use L for Latin, H for Hebrew, S for Cyrillic, R for Greek, A for Arabic, C for CJK.

1. The text of a long field is not cut arbitrarily in the middle of a word (by finding the cutting point according to the number of characters only). Instead, the text is cut immediately after the delimiter "-- " (dash,dash, space), which is used as a separator between entries in fields representing "Table of Contents" (for example, tag 505).

2. You can determine whether or not a field is actually a part of a longer field that has been cut, in the following manner:

   o If a field begins with "$$9^^$$a", that is, subfield 9, caret, caret, and another subfield (subfield a is just an example) - this means that it is only a part of a cut field, and not a complete field. The "^^" means that the text of this part continues the subfield of the former part (the subfield that follows the "^^").

   o If a field begins with "$$9^$$t", that is, subfield 9, caret, and another subfield (subfield t is just an example) - this also means that it is only a part of a cut field. However, in this case, the former part has been cut at the **end** of a subfield, and the current part begins with a new subfield (the one that follows the "^").

   To sum up, any field longer than 2000 bytes is divided into multiple parts (at least two), so that subsequent parts begin with "$$9".

The Load ALEPH Sequential MARC Records (p_manage_18) procedure loads the records in ALEPH Sequential format into the system. In other words, it populates the Z00 table.