
How Does the MARC 880 Field Display and Index?

- **Article Type:** General
 - **Product:** Aleph
 - **Product Version:** 20, 21, 22, 23
-

Description:

880 Field - In MARC, when the 245 field contains non-Latin character set there should be an 880 field containing the parallel title. Our current ILS does not support this, so we create a Romanized form in 246. If the 880 field was supported, the 246 would not be necessary.

Can Aleph index the 880s and original field. Will 880 display?

Resolution:

The 880 field behavior will depend on local indexing and display definitions. The following information is related to this question and may be helpful in deciding how to deal with 880 fields.

There are several ways the non-Roman form can be represented. It can be in a separate authority record altogether. This appears to be the most common practice among ALEPH libraries that have a single authority database with both Roman and non-Roman forms.

It can be coded in a 5xx field, which is the most common practice among US ALEPH libraries. It allows the non-Roman form to be present without risking unanticipated heading correction in Bib records.

The MARC21 standard itself calls for use of the 880 field for representing these alternate forms in Authority records.

Finally, there is the possibility of using the 4xx field. This has generally been avoided among US ALEPH libraries. The 4xx represents the non-preferred form of the heading, and this doesn't actually represent the nature of the non-Roman form. However, beginning in June 2008, OCLC began distributing Authority records from the NACO project with the non-Roman form in a 4xx. This was deemed inappropriate by several ALEPH libraries, who approached OCLC and asked that this non-standard use of the 4xx be somehow flagged so that ALEPH could treat the records as a separate case. OCLC declined to do so. Consequently, Ex Libris, with input from the participating ALEPH libraries, modified the ALEPH software to insure that unexpected heading changes wouldn't occur from these NACO records. The rep_change is included in the following versions of ALEPH (and higher):

V.16 - rpc #2166

V.18 - rpc #1729

V.19 - rpc #332

Here are the details of the rep_change:

Description: A new parameter has been implemented for the fix_doc_ref_1 program. This program is responsible for the update of the bibliographic record from the authority linked record. The new parameter enables Aleph customers that desire to "turn-off" the automatic authorities update mechanism in cases where the bibliographic heading is a non-Latin heading and the authority heading derives from a Latin authorized heading (or the opposite). In other words, the link to the authority record is created but if there is a difference in the scripts (Latin vs. non-Latin), then the update of the bibliographic record does not take place. This enhancement has been performed due to an announcement made by The Library of Congress on the addition of non-Latin script (such as Arabic, Cyrillic and Hebrew) to the LC/NACO name authority records distributed.

These references will be under the 4XX reference fields. The Romanized form will continue to be the authorized heading (authority record 1XX field). Distribution of the new records is planned for mid-July.

Related Fix Routines

fix_doc_880

This routine replaces the tag number of the alternate graphic representation field (880) by the associated tag registered in subfield \$6 of the field. In addition, the tag number of the associated field is removed from subfield \$6 but the occurrence number is retained. For non-880 fields, the tag number of the associated field (for example, 880) is removed from subfield \$6 and the occurrence number is retained.

This program creates parallel fields, both containing the same tag number. For example:

1001 L \$\$6880-01\$\$a[Name in Chinese script].

8801 L \$\$6100-01/(B\$\$aShen, Wei-pin.

Is changed to:

1001 L \$\$601\$\$a[Name in Chinese script].

1001 L \$\$601\$\$aShen, Wei-pin.

Note that for this fix to work, subfield \$6 must be the first subfield in both linked fields.

fix_doc_sort_sub6

This program should be used after performing the fix_doc_880 program that creates parallel fields that are different script representations of each other. The fix_doc_sort_sub6 program is used to sort the linked-parallel fields. The fields are sorted by the occurrence number stored in subfield \$6.

fix_doc_redo_880

This routine reverses the effects of the fix_doc_880 program. This program restores the tag number of the alternate graphic representation field (880). For example:

1001 L \$\$601\$\$a[Name in Chinese script].

1001 L \$\$601\$\$aShen, Wei-pin.

Is changed to:

1001 L \$\$6880-01\$\$a[Name in Chinese script].

8801 L \$\$6100-01/(B\$\$aShen, Wei-pin.

Note that the order of the paired fields is important, because the tag of the first of a pair is left as is, and the second of a pair is transferred to 880.

In addition, note that the input for this program must be in MARC8 (not in UTF) encoding. The reason for this is that this fix routine sets the escape sequence and orientation for the language code, and in order to do so, the record must be in MARC8 encoding. The fix_doc_redo_880 program will work correctly on UTF records, but will not set the escape sequence and orientation for the language code.

expand_doc_bib_880_n

The expand_doc_bib_880_n program creates two new concatenated fields out of 2 fields linked by subfield \$6. Subfield \$6 contains data that links fields that are d

-
- **Article last edited:** 10/8/2013