AlmaとAlephセントラル目録の統合

Note

Alephセントラル目録プロファイルは、特定のAlephセントラル目録の統合プロファイルです。このプロファイルは、Alma実装の一部として設定する必要があり、Ex Librisサポートの承認なしに設定することはできません。

Almaは、共有Alephセントラル目録環境への参加を可能にする統合ツールを提供します。この機能を使用すると、Almaで作成された書誌および所蔵レコードとアイテムレベルの目録をAlephセントラル目録に反映できます。さらに、Almaデータベースに対応するレコードがあるAlephセントラル目録のレコードが変更された場合、Z39.50プロトコルを使用して、変更されたAlephセントラル目録レコードを検索できます。

Almaは、Almaの次の機関レベルの機能を通じて、Alephセントラル目録との互換性をサポートしています：

- セントラル目録からAlmaにカタログをコピーします
- Almaからセントラル目録への変更に貢献する
- Almaからセントラル目録に目録を公開します
- セントラル目録からAlmaに変更をインポートする

Alephセントラル目録を使用するには、Alephセントラル目録システムと、各Almaメンバー機関の両方を設定する必要があります。以下のセクションでは、AlephとAlmaの両方を設定するために必要な手順を定義します：

- Alephの設定
- Almaの設定

Alephの設定

このセクションでは、Alma機関をAlephセントラル目録と統合するために、Alephで実行する必要がある設定について説明します。

AlmaメンバーをAlephに統合するには：

VSTサーバー定義を設定します。

1. 以下を入力します：

```
vim $alephe_root/aleph_start.private
```

リクエストは、vst_libraryとして定義された図書館に保存されます。デフォルトでは、これは（aleph_startで定義されている）USR00図書館です。別の図書館を使用したい場合は、aleph_start.privateで次のように定義します：

```
setenv vst_library XXXNN
```

通常、USR00を使用する必要があります。そのため、必要なaleph_start.privateには、追加のエントリがありません。

データ複製インターフェイスのサーバー（VSTサーバー）は、デフォルトでポート522xで実行されます。別のポートを使用する場合は、aleph_start.privateで入力することができます：

```
setenv VST_SERVER_PORT nnnn
```

例：

```
setenv VST_SERVER_PORT 4711
```

2. 以下を入力します：

```
vim $alephe_tab/tab_low_group
```

tab_low_groupでは、ローカル目録IDが定義されています。Z115レコードは、ローカル目録IDが存在する場合にのみ作成されます。ローカル目録IDは、1〜n個のサブライブラリコードを表す場合があります。Z115メッセージを生成するローカルシステムごとに、サブライブラリコードを定義する必要があります。

図書館が書誌のLOWフィールドに関係なくVSTサーバーへのZ115メッセージをトリガーしたい場合は、tab_low_group を次のように設定します。

```
! 1 2 3 4 5 6 7 !!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-! ALMA ALL
```

上記の設定では、LOWタグに依存することなく、書誌レコードが更新されるたびにメッセージが表示されます。

例：

```
! 1 2 3 4 5 6 7 !!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-! VST01 LOC28 VST02 LOC69 LOC70 LOC75 VST05 LOC75
VST06 VST06
```

VST02ローカル目録IDには、いくつかのサブライブラリコードが含まれています。VST02リストから少なくとも1つの所有権確認を含む書誌レコードを修正すると、修正用のZ115レコードが作成されます。BIBレコードに対して最初の所有権検証が生成されるか、最後の所有権検証が削除されると、作成または削除用のZ115レコードが生成されます。

シリーズまたは集合タイトルを修正すると、すべてのローカル目録IDに対してZ115レコードが作成されます。これ

は、典拠レコードにも適用されます。所蔵レコードについては、`tab_low_group`のテーブルが索引として機能します。これは、所蔵レコードごとに常に1つのサブライブラリコード（`OWN`）があるためです。

Almaの場合：

```
! 1 2 3 4 5 6 7 !!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-! ALMA #####
```

レコード間で親子リンクを使用する図書館の場合、子のBIBレコードがアップデートされると、子のZ115レコードに加えて、親のZ115が作成されるように設定できます。

上記の機能をアクティブにするために、`tab_low_group`の列7は「`Y`」に設定する必要があります。

例：

```
! 1 2 3 4 5 6 7 !!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-! ALMA ##### Y
```

親レコードリンクは、`./[bib_library]/tab/tab_relink_link`テーブルの定義に基づいています。したがって、親を送信機能を有効にするには、すべての親リンクで `tab_relink_lin` を定義します。

例：

```
! 1 2 3 4 5 6 7 8 9 !!!!!-!-!!!-!-!!!!!-!-!!!-!-! 010## a 001 DN Y
```

3. 以下を入力します：

```
vim $alephe_unicode/tab_character_conversion_line
```

VSTサーバー経由でメッセージを送信する場合、文字変換をアクティブにすることができます。変換が定義されていない場合、サーバーは、UNICODE文字（UTF8）としてデータを送信します。エントリ「`UTF_TO_VST`」に変換を割り当てます。

例：

```
!!!!!!!!!!!!!!!!!!!!-!!!!!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!-! UTF_TO_VST ##### L line_utf2line_sb unicode_
to_8859_1 Y
```

必要に応じて、さらに変換プログラムとテーブルを実装できます。

4. VST ライブラリをアクティブにします。

この関数がVST ライブラリでアクティブになっている場合にのみ、リクエストが生成されます。これを行うには、VSTライブラリの`file_list`に次のエントリを追加し、UTIL A 17/1を介してZ115のテーブルを初期化する必要があります。

> TAB z115 100K,128K,128K,128K,128K,128K 0K ts0,TS1D,TS1D,TS1D,TS1D,TS1D IND z115_id 100K,128K,128K,128K,128K,128K 0K ts1,TS1X,TS1X,TS1X,TS1X,TS1X

Note

テーブルのサイズは、予想されるリクエストの数によって異なります。リクエストは自動的に削除されません。定期的に手動で削除するか、アーカイブする必要があります。

さらに、カウンター「last-vst-number」は、UTIL G / 2で定義する必要があります。プレフィックスは割り当てられていません。先頭のゼロを削除しないでください。

Z115が確立され、関連するカウンターが定義されるとすぐに、tab_low_groupからのローカル目録IDのリクエストが書き込まれます。VSTサーバーを再起動します。

Z39.50定義をAlephz39_serverに設定する

1. `$alephe_tab/z39_server/z39_server.conf`でポートが正しいことを確認します。

2. **MAB環境：**
   「Z39」スタッフをADMIN（ユーザーライブラリ）に変更し、「MAB01」ライブラリのアクセス許可を追加します（すべての共通タブを許可します）。

   USMARC 環境の場合：
   「Z39」スタッフをADMIN（ユーザーライブラリ）に変更し、「USM01」ライブラリのアクセス許可を追加します（すべての共通タブを許可します）。

3. Z39経由でUTFを返すには、`./alephe/tab/z39_server/z39_server_MAB01.conf`（または`z39_server_`**`USM01`**`.conf`）を次のように変更します。を次のように変更します。

   MAB環境：
   MAB2MARCセクションにはtab_expandに一致するセクション、MB2MCセクションにはtab_fixに一致するセクションがなくてはなりません。

   USMARC環境：
   Z39_SERVERセクションにはtab_expandに一致するセクション、「Z39_server」セクションにはtab_fixまたはtab_expandに一致するセクションがなくてはなりません。を選択します。

   - out-record-syntax USMARC
   - out-record-format USMARC
   - out-record-char-conv
   - out-record-expand MAB2MARC
   - out-record-fix MB2MC
   - out-record-syntax USMARC
   - out-record-format USMARC
   - out-record-char-conv
   - out-record-expand Z39_SERVER

4. オプションで（**MAB環境のみ**）- 元の設定を保持するために新しい`MB2MC`変換を使用するAlma Z39.50検索の新しいベースを定義します（旧MAB→MARC変換）。以下のセクションで詳細を参照してください：MABからMARCへの変換

5. オプションで（**USMARC環境**）- 事前定義されたコードを含む003フィールドと、001からのシステム番号と003からのプレフィックスを含む035フィールドを追加する新しいエキスパンドである「`Z39_SERVER`」を使用するAlmaZ39.50検索の新しいベースを定義します。拡張 - 003と035を追加するを参照してください。

MABからMARCへの変換：（**MAB環境のみ**）

1. 次の例のような新しいセクションを`./[bib_library]/tab_expand`に追加します：

> !!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!> MAB2MARC expand_doc_bib_bib_mab2marc DOWN-MAX=10 MAB2MARC fix_doc_clear DUP-FLD=Y

2. 次の例のような新しいセクションを`./[bib_library]/tab_fix`（Almaの場合）エントリに追加します：

> !!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!> MB2MC fixexp_docx_parallel INIT MB2MC fixexp_docx_parallel COPY,PTR-FROM=0,PTR-TO=I MB2MC expdoc_43obv_bib_bib_mab2marc DN-MAX=10,SRD-MAX=10 MB2MC fix_doc_mab FUNC=OUTPUT,FILE=alma_obvsg.tab_fix_mab.bib_mab2marc MB2MC fix_doc_mab FILE=alma_obvsg.tab_fix_mabaseq.bib_mab2marc MB2MC fix_doc_clear DUP-FLD=Y MB2MC fix_doc_43obvos_recode_ind2tag MB2MC fixexp_docx_parallel COPY,PTR-FROM=0,PTR-TO=B MB2MC fixexp_docx_parallel FIXROUT,MBPR1,PTR-FROM=0,PTR-TO=H MB2MC fixexp_docx_parallel CAT,PTR-FROM=0,PTR2FROM=H MB2MC fixexp_docx_parallel CAT,PTR-FROM=0,PTR2FROM=H MB2MC fixexp_docx_parallel FIXROUT,MBPR2,PTR-FROM=0,PTR-TO=W MB2MC fixexp_docx_parallel CAT,PTR-FROM=0,PTR2FROM=W MB2MC fix_doc_convtb FILE=alma_obvsg.convtb.bib_mab2marc.main MB2MC fixexp_docx_parallel FIXROUT,MBOS1,PTR-FROM=B,PTR-TO=1 MB2MC fixexp_docx_parallel PARFUNC,copyfld,"LDR,LDRMC",PTR-FROM=0,PTR-TO=1 MB2MC fixexp_docx_parallel FIXROUT,MBOS2,PTR-FROM=1,PTR-TO=2 MB2MC fixexp_docx_parallel PARFUNC,43obvos_catlink,PTR-FROM=0,PTR2FROM=2,PTR-TO=0 MB2MC fix_doc_mab2marc_synind_880lnk MB2MC fix_doc_convtb FILE=alma_obvsg.convtb.bib_mab2marc.post_01 MB2MC fix_doc_convtb FILE=alma_obvsg.convtb.bib_mab2marc.upd_lnkidn MB2MC fixexp_docx_parallel FIXROUT,MBNOM,PTR-FROM=I,PTR-TO=M MB2MC fixexp_docx_parallel CAT,PTR-FROM=0,PTR2FROM=M MB2MC fix_doc_sort_marc21 USECODE=Y MB2MC fix_doc_clear DUP-FLD=Y MBPR1 fix_doc_convtb FILE=alma_obvsg.convtb.bib_mab2marc.pre_h_fields MBPR2 fix_doc_convtb FILE=alma_obvsg.convtb.bib_mab2marc.pre_wk_fields MBOS1 fix_doc_convtb FILE=alma_obvsg.convtb.bib_mab2marc.os1 MBOS2 fix_doc_convtb FILE=alma_obvsg.convtb.bib_mab2marc.os2 MBNOM fix_doc_convtb FILE=alma_obvsg.convtb.bib_mab2marc.nomapp

   MAB環境で - この修正手順は、ラインのために`$alephe_tab/z39_server/z39_server_MAB01.conf`で使用される修正手順を一致させる必要があります：

   > out-record-fix MB2MC

3. **MAB環境で**グローバル変数テーブルを介してユーザー固有の値を定義します。

$data_tab内のテーブルの名前は、tab_globalvar_convtb.m2mです。

テーブル./mab01/tab/ttt_49bvb.tab_globalvar_convtb.m2mを./[bib_library]/tab/tab_globalvar_convtb.m2mにコピーし、ISIL-Codeユーザーを変更します。

MARCからMABへの変換：（MAB環境のみ）

1. ./[bib_library]/tab_expandおよび/または./[bib_library]/tab_fix（Almaについては、MAB01）に、新しいセクションを追加します。

   このセクションは、次の例のようなものです（fix-linesのみ）：

   > !!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!> MC2MB fix_doc_remove_punctuation_usm MC2MB fix_doc_convtb FILE=ttt.tab_fix_convtb.bib_marc2mab.49kobv_p0_tt_pp MC2MB fix_doc_convtb FILE=ttt.tab_fix_convtb.bib_marc2mab.49kobv_p1 MC2MB fix_doc_convtb FILE=ttt.tab_fix_convtb.bib_marc2mab.49kobv_p2 MC2MB fix_doc_clear DUP-FLD=Y MC2MB fix_doc_overlay TMARC2MABI MC2MB fix_doc_sort_mab

2. tab_fixでfix_doc_overlay手順について定義したとおり、統合セクションを追加することによって、入力レコードとDBレコードのマージを設定します。
   ./mab01/tab/addon_ttt_m2m.bib.tab_mergeテーブルでテンプレートオプションを参照してください。

   例えば、./[bib_library]/tab/tab_mergeで、上記のTMARC2MABIに以下を追加します。

   > ! 1 2 3 !!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!> TMARC2MABI merge_doc_overlay 08

3. ./[bib_library]/tab/tab_merge_overlayで、統合定義を追加します。統合セットはtab_mergeで定義された統合セットと一致する必要があります。
   ./mab01/tab/addon_ttt_m2m.bib.tab_merge_overlayテーブルでテンプレートオプションを参照してください。

   例えば、上記の統合セット「08」に、次のラインを追加します：

   > !1 2 3 4 !!-!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! ... ! 08 1 N ##### 08 1 Y CAT## 08 1 Y LOW## 08 1 Y OWN## 08 2 Y ######

Note

LOWフィールドが./[bib_library]/tab/tab_cat_hidden_fieldsで隠しフィールドとして定義されている場合、上記tab_merge_overlayの定義から、次のラインを削除します：

> 08 1 Y LOW##

拡張 - 003（プレフィックス）と035（システム番号+プレフィックス）を追加します（USMARC環境）。

タブ_expandに(xxx01 $data_tab
）を追加します：

> ! Z39_SERVER expand_doc_sysno Z39_SERVER fix_doc_do_file_08 alma.fix Z39_SERVER fix_doc_create_035_1 !Z39_SERVER fix_doc_create_035

「alma.fix」は次のように定義されます（xxx01 $data_tab/import/）：

> ! 2 3 4 5 6 7 8 9 !-!!!!!-!!-!-!!!-!!!-!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!! ! 1 FMT DELETE-FIELD ! Handles system number etc. 1 003## DELETE-FIELD 2 035## DELETE-FIELD-COND Y,$$a(ALM) 2 LDR ADD-FIELD 003 ,L,ALM

上記の例の
プレフィクス「ALM」は、Aleph統合プロファイルのAlmaでも定義されています。
これは、X-serviceを介したAlmaからAlephへの寄与に使用されます。

update-doc-almaのX-serviceを設定する：

1. **MAB**環境で：「update-doc-alma」のX-serviceでMARC-MAB変換を処理するには、./[bib_library]/tab/tab_fixにある「ALMA1」のセクションに修正手順を追加します。
   MAB-ALEPH-Central目録の「MARCからMAB」への変換の例：

   > !!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!> ALMA1 fixexp_docx_parallel INIT ALMA1 fix_doc_clear DUP-FLD=Y ALMA1 fix_doc_mab2marc_rm_punct_usm ALMA1 fixexp_docx_parallel COPY,PTR-FROM=0,PTR-TO=M ALMA1 fix_doc_convtb FILE=alma_obvsg.convtb.bib_marc2mab.pre_01 ALMA1 fixexp_docx_parallel COPY,PTR-FROM=0,PTR-TO=O ALMA1 fix_doc_convtb FILE=alma_obvsg.convtb.bib_marc2mab.49dnb_p0_tt_pp ALMA1 fix_doc_convtb FILE=alma_obvsg.convtb.bib_marc2mab.49dnb_p1 ALMA1 fix_doc_convtb FILE=alma_obvsg.convtb.bib_marc2mab.49dnb_p2 ALMA1 fix_doc_convtb FILE=alma_obvsg.convtb.bib_marc2mab.post_01 ALMA1 fixexp_docx_parallel FIXFUNC,fix_doc_convtb,"FILE=alma_obvsg.convtb.bib_marc2mab.osc",PTR-FROM=O,PTR-TO=O ALMA1 fixexp_docx_parallel CAT,PTR-FROM=0,PTR2FROM=O ALMA1 fixexp_docx_parallel FIXFUNC,fix_doc_convtb,"FILE=alma_obvsg.convtb.bib_marc2mab.nomapp",PTR-FROM=M,PTR-TO=M ALMA1 fixexp_docx_parallel CAT,PTR-FROM=0,PTR2FROM=M ALMA1 fix_doc_char_conv_z GND-UTF-TO-UTF ALMA1 fix_doc_clear DUP-FLD=Y ALMA1 fix_doc_sort ALMA1 fix_doc_overlay ALMA1MC2MB ALMA1 fix_doc_sort

Note

修正セクションは、パラメーターとしてXサービスに送信できます。この場合、各メンバーには、上記のラインを含むtab_fixで定義されたセクションが必要です。

ロングフィールド（2000文字以上）を持つレコードの読み込みを有効にするには、. /alephe/aleph_start.private： に次のラインを追加します。

```
setenv ENABLE_LONG_FIELDS Y
```

2. 入力レコードとDBレコードの統合を設定するために、tab_fixでfix_doc_overlay手順について定義したとおりマージセクションを追加します。各メンバーは異なる統合セクションを持つことができます。
例えば、./[bib_library]/tab/tab_mergeで、上記のALMA1MC2MBに以下を追加します：

```
! 1 2 3 !!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!> ALMA1MC2MB merge_doc_overlay AA
```

3. ./[bib_library]/tab/tab_merge_overlayで、統合定義を追加します。統合セットはtab_mergeで定義された統合セットと一致する必要があります。

例えば、、上記の統合セットAAの 場合 、次のラインを追加します。

```
!1 2 3 4 !!-!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!! ... ! For Alma X-service AA 1 N ##### AA 1 Y CAT## AA 1 Y OWN##
AA 1 Y OSC## AA 2 Y #####
```

所蔵情報の読込みを設定します（書誌レコードのLOWフィールド）。

このセクションでは、LOWフィールドを使用して、書誌レコードの所蔵情報の読み込みを説明しています。

1. LOWフィールドの削除を扱うには、

   ./[bib_library]/tab/tab_fix inに次のラインを必要なセクションとともに追加します：

```
!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!> LOWRM fix_doc_do_file_08 delete_low_fields
```

2. 修正手順を介してLOWフィールドを削除するには、./[bib_library]/tab/import/delete_low_fieldsテーブルが次のラインを含むことを確認します 。

```
! 2 3 4 5 6 7 8 9 !-!!!!!-!!-!-!!!-!!!-!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! 1 LOW## DELETE-FIELD-COND Y,$$uDEL
```

- ./[bib_library]/tab/tab00.LNG（フィールドコードと名前）にLOW索引を追加します。

> ! 2 3 4 5 6 7 8 9 10 11 !-!!!!!-!!!!!-!-!!-!!-!-!---!!-!!!!!-!!!!!!!!!!!!!!!!!!!! H LOW## IND 07 00 00 LOW - Local Owner

- ./[bib_library]/tab/tab_filing（ファイリングおよび正規化手順の定義）にLOWファイリングを追加します。ファイリングセクションはtab00.LNG内の定義と一致する必要があります。
例えば（tab_filingの）：

> !1 2 3 4 !!-!-!!!!!!!!!!!!!!!!!!!!-!! ! * for LOW 07 suppress 07 to_blank - 07 to_lower 07 compress_blank 07 get_subfields a

- ./[bib_library]/tab/tab_matchの一致設定に次のラインが含まれていることを確認します。

> ! 1 2 3 !!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!! ALM match_doc_gen TYPE=IND,TAG=001,CODE=IDN ...

- ./[bib_library]/tab/tab_mergeに次のラインを追加します。

> ! 1 2 3 !!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!> LOADHOL merge_doc_alma_overlay 09

- ./[bib_library]/tab/tab_merge_adv_overlayに次のラインを追加します。

> !1 2 3 4 5 6 7 !!-!-!-!-!!!!!-!-!!!!!!!!!!!> ! Alma load holdings 09 1 # Y ##### 09 1 # U LOW## 09 2 # N ##### 09 2 # Y LOW##

- 所蔵情報をAlmaからAlephに読み込むには、following ./alephe/tab/job_listに次の行を追加します。

  これにより、Almaからの変更が毎日読み込まれます：

> ! 2 3 4 5 !!-!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!> ! 2 3 4 5 6 7 !!-!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!!-!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!! !Load holdings W2 12:35:00 N untar-alma csh -f /exlibris/aleph/a23_3/aleph/proc/untar_files /exlibris/aleph/alma/bvb/ /exlibris/aleph/a23_3/mab01/scratch alma_inventory W2 08:56:00 Y MAB01 p_file_02 MAB01,all_alma_inventory,all_alma_inventory.seq,06, W2 08:56:00 Y MAB01 p_manage_36 MAB01,all_alma_inventory.seq,all_alma_inventory.seq.new,all_alma_inventory.seq.upd,all_alma_inventory.seq.dup,IDN, W2 08:56:00 Y MAB01 p_manage_18 MAB01,all_alma_inventory.seq.upd,all_alma_inventory.seq.upd.rejected,all_alma_inventory.seq.upd.doc_log,OLD,,,FULL,MERGE,M,,LOADHOL, W2 08:56:00 Y MAB01 p_manage_18 MAB01,all_alma_inventory.seq.new,all_alma_inventory.seq.new.rejected,all_alma_inventory.seq.new.doc_log,NEW,,,FULL,MERGE,M,,LOADHOL, W2 08:56:00 Y MAB01 p_manage_37 MAB01,DOC_LIST,b_manage_36,000000000,000000000,manage_37.out,LOWRM,Y,MASTER,

これにより、次のことが実行されます：

1. 「untar - alma」スクリプトが実行され、Almaで公開されたファイルを解凍し、それを$data_scratch（[bib_library]の下）に移動します。

2. p_file_02が、Alma MARC XMLファイルをAlephシーケンシャルファイルに変換します。
3. p_manage_36が実行され、セントラルDB内で一致するAlephレコードを見つけます。
4. p_manage_18が2回実行されます：
   - 「LOADHOL」統合手順を使用して、一致したレコードをAlephのレコードと統合します
   - Alephに新しいレコードを挿入します
5. p_manage_37は、LOWRM修正手順を使用してLOWフィールドを削除します

- Almaが所蔵情報の公開を毎日実行するように、job_list.confに最初のパラメータ（「W2」）を定義します。
- mab01を[bib_library]に置き換えます。
- untar_filesのパラメータをご使用の環境の正しいパスに置き換えます。
- 読み込み所蔵レコード情報を設定します。

このセクションでは、HOL図書館への所蔵レコードの読み込みについて説明します。

1. 「001」索引が./[hol_library]/tab/tab11_indに存在することを確認します。

   例：

   ```
   ! 1 2 3 4 5 6 7 8 !!!!!-!!!!!-!-!!!!!!!!!!-!!!!!-!!!!!!!!!!!!!!!!!!!!-!-! 001 IDN
   ```

2. ./[hol_library]/tab/tab_matchの一致設定に次のラインが含まれていることを確認します。

   ```
   ! 1 2 3 !!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! HOL
   match_doc_uid I-IDN ...
   ```

3. ./[hol_library]/tab/tab_mergeに次のラインを追加します：

   ```
   ! 1 2 3 !!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!> LOADMABHOL merge_doc_overlay 07
   ```

4. ./[hol_library]/tab/tab_merge_overlayに次のラインを追加します。

   ```
   !1 2 3 4 !!-!-!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! 07 2 C ##### 07 1 Y #####
   ```

5. 新しい設定ファイル./[hol_library]/tab/import/fix_alma_001に次の行を追加します：

   ```
   ! 2 3 4 5 6 7 8 9 !-!!!!!-!!-!-!!!-!!!-!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!!!! 1 001 CHANGE-SUBFIELD ,a
   ```

6. 新規またはアップデートされた所蔵レコード：
   以下の行を./alephe/tab/job_listに追加して、新規またはアップデートされた所蔵レコードをAlmaからAlephに読み込みます。
   これにより、Almaからの変更が毎日読み込まれます：

Note

公開されたファイルはAlephサーバーにあると想定されています。
そうでない場合は、ftpユーティリティまたはコマンドを使用してFTPサーバーから転送します。たとえば、次を実行します：

```
wget --ftp-user=USER --ftp-password='PASSWORD' ftp://ftp.exlibris.co.il/Aleph/Pub_Aleph_mab_hol_..._new.xml.tar.gz
```

`job_list`に追加：

```
! 2 3 4 5 !!-!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!> ! 2 3 4 5 6 7 !!-!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!-!!!!!-!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!! ... !Alma load holdings !Load holdings W2 10:02:00 N move-prev-alma-hol csh -f /exlibris/aleph/a23_3/aleph/proc/move_files /exlibris/aleph/a23_3/mab60/scratch /exlibris/aleph/a23_3/mab60/scratch/backup pub_aleph_mab_hol W2 10:03:00 N untar-alma-hol csh -f /exlibris/aleph/a23_3/aleph/proc/untar_files /exlibris/aleph/load_holdings /exlibris/aleph/a23_3/mab60/scratch pub_aleph_mab_hol W2 10:04:00 Y MAB60 p_file_02 MAB60,all_new_pub_aleph_mab_hol,all_pub_aleph_mab_hol_pre.seq,06, W2 10:05:00 Y MAB60 p_file_08 MAB60,all_pub_aleph_mab_hol_pre.seq,all_pub_aleph_mab_hol.seq,all_pub_aleph_mab_hol_pre.rej,fix_alma_001 W2 10:06:00 Y MAB60 p_manage_36 MAB60,all_pub_aleph_mab_hol.seq,all_pub_aleph_mab_hol.seq.new,all_pub_aleph_mab_hol.seq.upd,all_pub_aleph_mab_hol.seq.dup,HOL, W2 10:08:00 Y MAB60 p_manage_18 MAB60,all_pub_aleph_mab_hol.seq.new,all_pub_aleph_mab_hol.seq.new.rejected,all_pub_aleph_mab_hol.seq.new.doc_log,NEW,,,FULL,MERGE,M,,LOADMABHOL, W2 10:10:00 Y MAB60 p_manage_18 MAB60,all_pub_aleph_mab_hol.seq.upd,all_pub_aleph_mab_hol.seq.upd.rejected,all_pub_aleph_mab_hol.seq.upd.doc_log,OLD,,,FULL,MERGE,M,,LOADMABHOL,
```

これにより、次のことが実行されます：

1. `move_files`スクリプトが実行され、前の実行で生成された入力ファイルを除去/移動します。
2. `untar-alma`スクリプトが実行され、Almaで公開されたファイルを解凍し、`$data_scratch`（[hol_library]の下）に移動します。
3. `p_file_02`が、Alma MARC XMLファイルをAlephシーケンシャルファイルに変換します。
4. `p_file_08`がサブフィールド「a」をタグ「001」に追加します
5. `p_manage_36`が実行され、セントラルDB内で一致するAlephレコードを見つけます。
6. `p_manage_18`が2回実行されます：
   - 「`LOADHOL`」統合手順を使用して、一致したレコードをAlephのレコードと統合します
   - Alephに新しいレコードを挿入します
7. Almaが所蔵情報の公開を毎日実行するように、`job_list.conf`に最初のパラメータ（「`W2`」）を定義します。
8. `mab60`を[hol_library]に置き換えます。
9. `untar_files`のパラメータをご使用の環境の正しいパスに置き換えます。

10. 所蔵レコードの削除
：AlmaからAlephへの所蔵レコードを削除するために、`./alephe/tab/job_list`に次のラインを追加します。
これにより、Almaからの変更が毎日読み込まれます：

Note

公開されたファイルはAlephサーバーにあると想定されています。そうでない場合は、ftpユーティリティまたはコマンドを使用してFTPサーバーから転送します。たとえば、次を実行します：

> wget --ftp-user=USER --ftp-password='PASSWORD' ftp://ftp.exlibris.co.il/Aleph/Pub_Aleph_mab_hol_..._del.xml.tar.gz

---

`job_list`に追加：

> !Alma delete holdings !Delete holdings !W2 13:00:00 N untar-alma csh -f /exlibris/aleph/a23_2/aleph/proc/untar_files /exlibris/aleph/load_holdings /exlibris/aleph/a23_2/mab60/scratch pub_aleph_mab_hol W2 11:23:00 Y MAB60 p_file_02 MAB60,all_delete_pub_aleph_mab_hol,all_del_pub_aleph_mab_hol_pre.seq,06, W2 11:23:00 Y MAB60 p_file_08 MAB60,all_del_pub_aleph_mab_hol_pre.seq,all_del_pub_aleph_mab_hol.seq,all_del_pub_aleph_mab_hol_pre.rej,fix_alma_001 W2 11:23:00 Y MAB60 p_manage_36 MAB60,all_del_pub_aleph_mab_hol.seq,all_del_pub_aleph_mab_hol.seq.new,all_del_pub_aleph_mab_hol.seq.upd,all_del_pub_aleph_mab_hol.seq.dup,HOL, W2 11:23:00 Y MAB60 p_manage_18 MAB60,all_del_pub_aleph_mab_hol.seq.upd,all_del_pub_aleph_mab_hol.seq.upd.rejected,all_del_pub_aleph_mab_hol.seq.upd.doc_log,OLD,,,FULL,DELDOC,M,,LOADMABHOL,

これにより、次のことが実行されます：

1. `untar-alma`スクリプトが実行され、Almaで公開された削除済みファイルを解凍し、$data_scratch（[hol_library]の下）に移動します。
2. `p_file_02`が、Alma MARC XMLファイルをAlephシーケンシャルファイルに変換します。
3. `p_file_08`がサブフィールド「a」をタグ「001」に追加します
4. `p_manage_36`が実行され、セントラルDB内で一致するAlephレコードを見つけます。
5. `p_manage_18`が、一致したレコードをAlephから削除します（一致なし= エラー）。

11. Almaが所蔵情報の公開を毎日実行するように、`job_list.conf`に最初のパラメータ（`W2`）を定義します。
12. mab60を[hol_library]に置き換えます。
13. `untar_files`のパラメータをご使用の環境の正しいパスに置き換えます。

- 読み込みアイテム情報を設定します。

このセクションでは、Alephセンター（manage-60）でアイテムを作成するプロセスについて説明します。

1. BIB図書館で、次のディレクトリを作成します。読み込むアイテムのtarファイルをこのディレクトリに配置します。

```
mkdir ./[BIB library]/items_to_center
```

Note

このディレクトリ内のファイルは`tar.gz`形式で終わる、ZIP形式のtarファイルでなければなりません。ファイル名に`new`という単語が含まれている場合、ファイル内のアイテムは新規の/アップデートされたアイテムとして読み込まれます。ファイル名に`delete`という単語が含まれている場合、ファイル内のアイテムは削除済みアイテムとして扱われます。

2. 以下のラインを追加して、AlmaからAlephに新規、アップデート、または削除されたアイテムを読み込みます。

Note

公開されたファイルはAlephサーバーにあると想定されています。そうでない場合は、ftpユーティリティまたはコマンドを使用してFTPサーバーから転送します。たとえば、次を実行します：

```
wget --ftp-user=USER --ftp-password='PASSWORD' ftp://ftp.exlibris.co.il/Aleph/Pub_Aleph_
mab_hol_..._new.xml.tar.gz
```

`job_list`に追加：

```
! 2 3 4 5 !!-!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!-!!!!!!!!!!!!!!> ! 2 3 4 5 6 7 !!-!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!-!!!!!-
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!-!!!!!! !Alma load items W2 10:10:00 Y MAB01 p_manage_60 MAB01
```

このサービスは、異なる実行時間で`p_manage_60`ジョブの追加行を定義することによって、一日に数回実行するように設定することができます。

Almaの設定

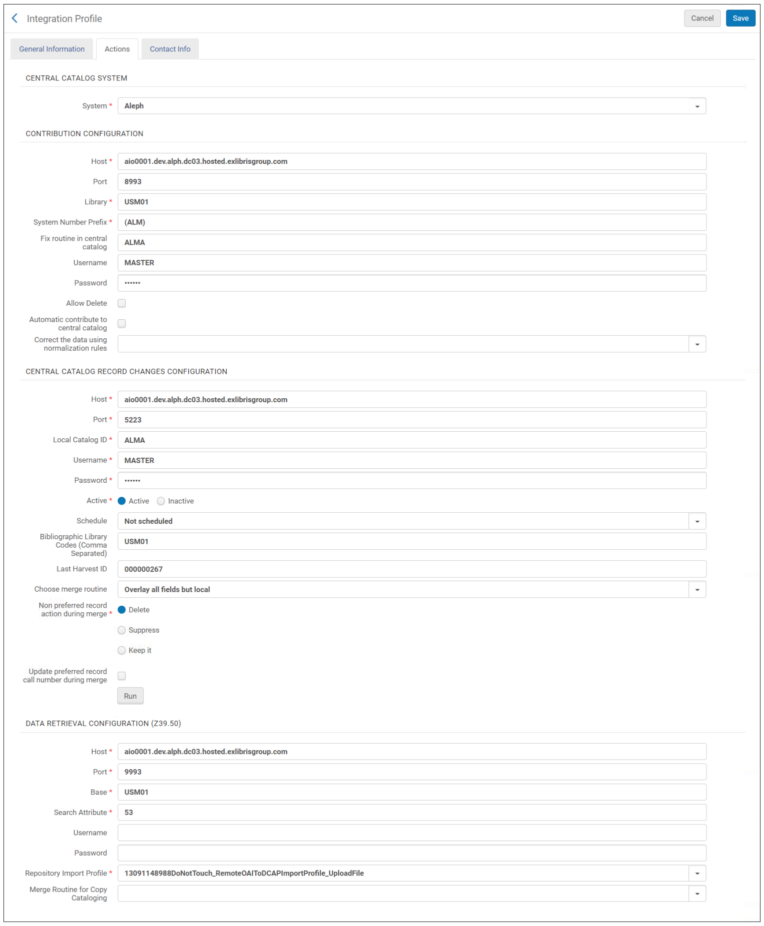
このセクションでは、Alma機関をAlephセントラル目録と統合するためにAlmaで必要な設定について説明します。

Alephのセントラル目録統合プロファイルの設定

Alephセントラル目録を使用するには、セントラル目録統合プロファイルを設定する必要があります。

総合セントラル目?統合プロファイルを設定するには：

1. 統合プロファイルリストページ（設定メニュー > 一般 > 外部システム > 統合プロファイル）で、**統合プロファイルを追加**を選択します。
2. 統合プロファイルの**コード**と**名前**を入力します。
3. **統合タイプ**ドロップダウンリストから**セントラル目録統合**を選択します。
4. **システム**で、システムタイプとして**Aleph**を選択します。
5. **次へ**を選択します。統合プロファイル設定の次のページが表示されます。



セントラル目録統合

6. 次の表に従ってフィールドに入力します：

セントラル目録統合

フィールド	説明
中央カタログシステム	
システム	設定するセントラル目録システムタイプとして**Aleph**を選択します。 **Note** ◦ このページで**System > Aleph**を選択する必要があります（最初のページで既に選択している場合でも）。 ◦ 機関ごとに設定できるセントラル目録統合プロファイルは1つだけです。
コントリビューション設定（コントリビューションサーバーのセットアップ）	
ホスト	Alephセントラル目録のIPアドレス。
ポート	Alephセントラル目録のポート。
図書館	ドキュメントがアップデートされるAleph図書館。

フィールド	説明
システム番号プレフィックス	図書館固有のシステム番号プレフィックス（たとえば、DE-604）。
セントラル目録の手順修正	Alephが使用する修正手順を入力します。
ユーザー名	Alephのユーザー名。
パスワード	Alephのパスワード。
削除を許可	中央目録内のレコードの削除を有効にする場合に選択します。 **「中央目録への自動投稿」** オプションが選択されている場合（以下を参照）、メタデータ エディタで一元管理されたレコードを削除すると、ネットワークゾーンとAlephの 両方で削除されます（いずれのシステムにおいても、両方のシステムで 削除が阻止されない場合にのみ削除が行われます）。
セントラル目録への自動投稿	この チェックボックス は、 機関がネットワークゾーンのメンバーである場合にのみ使用できます。 メタデータエディターで一元管理されたレコードを保存する場合は、 Alephのセントラル目録 とネットワークゾーンのレコードをアップデートすることを選択します。 このチェックボックスが選択されていない場合、レコードを保存すると、ネットワークゾーンには保存されますが、Alephセントラル目録には保存されません。レコードをAlephセントラル目録に保存するために、目録作成者は 「レコードをセントラル目録に投稿する」アクションをアクティブにします。Alephセントラル目録では、一元管理されたレコードのみが更新されます。 一元管理されていないレコード（新しく作成されたレコードおよび035 識別子のない既存のレコード） の場合、目録作成者は 「レコードを中央カタログに提供する」アクションをアクティブにします。
セントラル目録レコード変更設定（**VST**サーバーセットアップ）	
ホスト	Alephセントラル目録内のVSTサーバーのIPアドレス。
ポート	TCP/IP呼び出しを送信するためのAlephセントラル目録ポート。
ローカル目録ID	Alephの図書館を識別するために使用されます。
ユーザー名	Alephセントラル目録に送信するユーザー名。
パスワード	Alephセントラル目録に送信するパスワード。

<table>
<tr><th>フィールド</th><th>説明</th></tr>
<tr><td>アクティブ</td><td>インポートプロファイルをアクティブにするためにアクティブを選択するか、またはプロファイルを非アクティブにするために非アクティブを選択します。</td></tr>
<tr><td>スケジュール</td><td>インポートプロセスをスケジュールするには、ドロップダウンリストからスケジュールを選択します。利用可能なオプションは次のとおりです：
◦ 未スケジュール
◦ 11:00に開始して、12時間おきに
◦ 05:00に開始して、6時間おきに
◦ 毎時

Note
このジョブを実行すると、編集中のレコードのロックは無視され、レコードがインポートされます。</td></tr>
<tr><td>書誌事項図書館コード（コンマ区切り）</td><td>書誌レコード（コンマ区切り）を保持するAleph図書館コードを入力します。</td></tr>
<tr><td>最終ハーベストID</td><td>このフィールドは、プロセスの最初の正常な実行後に入力され、次の実行の開始点を示します。特定のレコードIDからデータを取得する場合は、この値を編集できます。</td></tr>
<tr><td>統合手順の選択</td><td>レコードがリダイレクトされるとき、古いレコードを新しいレコードと統合する方法を選択できます。機関によって定義された統合ルールに基づいて統合された手順を選択します。例えば：
◦ ローカル以外のすべてのフィールドのオーバーレイ
◦ 存在する場合、245および035 OCoLCの置き換え
◦ 条件付き主題見出し
◦ 古い値のみを保持

Note
レコードをマージするとき、セカンダリレコードから識別子を取得し、プライマリレコードにそれを配置するためのAlmaの設定については、BIBリダイレクトフィールドの設定を参照してください。</td></tr>
<tr><td>統合中の優先レコードアクション</td><td>削除、抑制、または保持のオプションから1つを選択し、（Alephセントラル目録内の）レコードの統合によるセカンダリ（非優先）レコードが、Almaデータベースのレコードとの同期時にどのように処理されるかを示します。

詳細については、Alephセントラル目録からの変更されたレコードの検索を参照してください。</td></tr>
</table>

フィールド	説明
統合中に優先レコード請求番号をアップデートする	このオプションを選択して、レコードがAlephセントラル目録に統合され、その後、Almaデータベースと同期されるとき（Alephセントラル目録ジョブが実行されるとき）、所蔵請求番号が、Almaデータベースに残っている優先レコードでアップデートされるべきであることを示す。
実行	クリックしてプロセスを開始します。プロセスには以下が含まれます： 1. VST サーバーを使用して hbz ID を取得します。 2. z39 サーバーを使用して完全なレコードを取得します。 3. メタデータのインポート経由でこれらのレコードをAlmaで読み込みます。
データ検索設定（**Z39.50**）	
ホスト	Alephセントラル目録のIPアドレス。
ポート	Z39.50コールを送信するためのAlephセントラル目録ポート。
土台	Z39.50クエリを送信するAlephのデータベース。
検索属性	セントラル目録内のシステム番号の場所の一意の識別子（Marc 001フィールドに相当）。
ユーザー名	z39.50のAlephセントラル目録に送信するユーザー名。
パスワード	z39.50のAlephセントラル目録に送信するパスワード。
リポジトリインポートプロファイル	書誌インポートのベースとなるリポジトリインポートプロファイルを選択します。具体的には、インポートプロファイルは、Alephセントラル目録プレフィックスの一致を指定する必要があります。詳細については、Alephセントラル目録統合プロファイルのためのインポートプロファイルの設定を参照してください。
目録コピーの統合手順	機関で定義された統合ルールから、目録コピーの統合手順を選択します。これらのルールは、目録コピーを実行する際に、以下の統合の考慮事項のいくつかに対処します： ◦ ローカル以外のすべてのフィールドのオーバーレイ ◦ 存在する場合、245および035 OCoLCの置き換え ◦ 条件付き主題見出し ◦ 古い値のみを保持 この統合手順は、目録のコピー時に外部レコードを既存のレコードと統合する方法を決定します。

フィールド	説明
	詳細については、統合ルールへの対応を参照してください。

7. **[保存]**を選択します。

外部システムとの統合プロファイルの設定に関する詳細情報については、外部システムを参照してください。

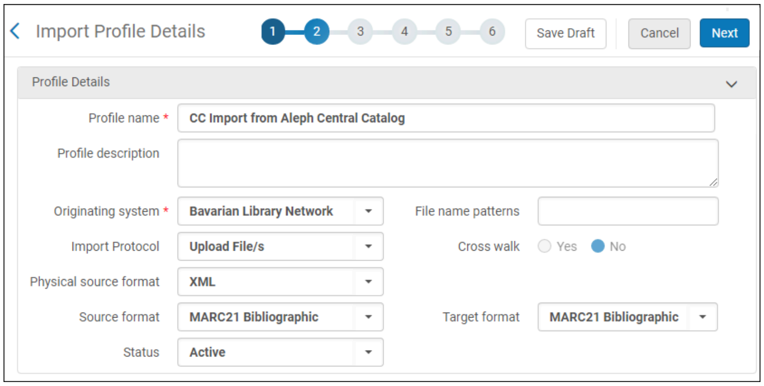
Alephセントラル目録統合プロファイルのためのインポートプロファイルの設定

Alephセントラル目録で変更され、Alma機関に関連する書誌レコードは、この目的のために設定されたAlmaインポートプロファイルを使用してAlmaにインポートされます。作成したインポートプロファイルは、Alephセントラル目録の統合プロファイル内のリポジトリインポートプロファイルパラメータで指定されます（リポジトリインポートプロファイルを参照してください）。

タイプリポジトリのインポートプロファイルを作成する手順（リポジトリを参照）と、特定のAlephセントラル目録設定用の以下の設定指示を使用して、新しいAlmaインポートプロファイルを作成します。

プロファイルの詳細セクションで、次のオプションを選択します：

- 発信元システム-ドロップダウンリストから、Alephコンソーシャル目録発信元システムを選択します。
- インポートプロトコルファイルのアップロードを選択します。
- 冊子ソースフォーマット**-XML**を選択。
- ソースフォーマット–**MARC 21**書誌を選択します。
- ステータス-アクティブを選択します。
- ファイル名のパターン–このオプションは空白のままにすることができます。
- クロスウォーク–いいえを選択します。
- ターゲットフォーマット–**MARC 21**書誌を選択します。



Alephセントラル目録からインポートするためのインポートプロファイル詳細セクション

プロファイルに一致させる、アクションに一致させる、自動複数一致処理、および一致なしセクションで、以下で説明するオプションを選択/入力します：

プロファイルに一致させるセクション

- 逐次刊行物一致方法- **035（他のシステム識別子）一致方法**を選択します。
- システム識別子プレフィックス（逐次刊行物一致方法用）– Alephセントラル目録の図書館ISILコード（図書館および関連機関の国際標準識別子）を入力します。これは、照合するMARC 035フィールドのシステム番号に対応します。
- 非逐次刊行物一致方法- **035（他のシステム識別子）一致方法**を選択します。
- システム識別子プレフィックス（非逐次刊行物一致方法用）– Alephセントラル目録の図書館ISILコード（図書館および関連機関の国際標準識別子）を入力します。これは、照合するMARC 035フィールドのシステム番号に対応します。

一致アクション

- 処理方法–**自動**を選択します。
- 照合の際に- **統合**を選択します。

一致アクション-統合/オーバーレイ

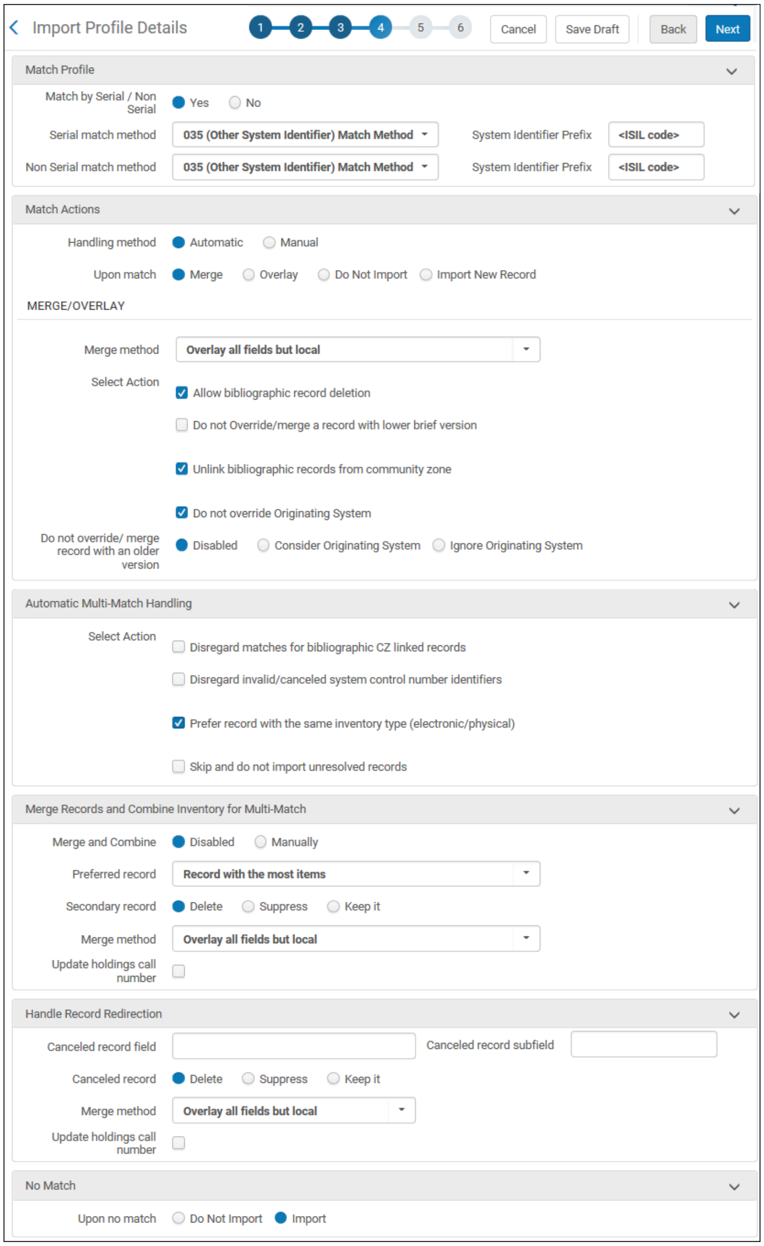
- 統合方法-関連する統合方法を選択します。**ローカル以外のすべてのフィールドをオーバーレイ**を選択すると、すべてのローカルフィールドが保存され、他のすべてのフィールドがAlephセントラル目録のレコードに置き換えられます。
- 次のアクションを選択します：
  - **書誌レコードの削除を許可**（Alephセントラル目録で削除されたレコードをAlma機関から削除する必要がある場合）
  - **コミュニティゾーンからの書誌レコードのリンク解除**
  - **古いバージョンではレコードを上書き/統合しない－無効化**

自動複数一致処理

- **同じ目録タイプ（電子/冊子）のレコードを優先**を選択する

一致なし

- **一致なしの場合**パラメータで、インポートオプションを選択します。



一致するレコードのインポートプロファイル詳細

Alephセントラル目録と統合するための公開プロファイルの設定

Alma機関からAlephセントラル目録に目録情報を公開するには、公開プロファイルを設定する必要があります。統合を機能させるには、利用可否を公開する必要があります。任意で、アイテムと所蔵情報を公開できます。利用可否、所蔵、およびアイテム情報を共有するには、利用可否（LOWフィールド）、所蔵、およびアイテムの3つの公開プロファイルを作成する必要があります。公開プロファイルを作成する手順および詳細については、Alephセントラル目録と統合するための設定の仕様を説明する次のセクションで、公開と目録の強化（一般公開）参照してください：

- 利用可否公開の設定
- 所蔵公開の設定
- アイテム公開の設定

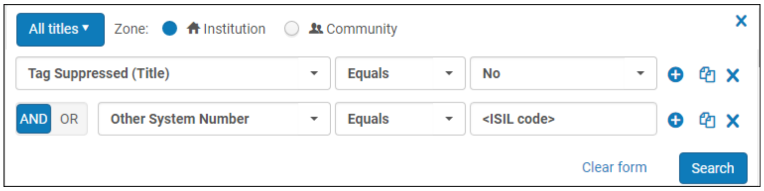
公開プロファイルの前提条件には、各正規化ルールのセット、正規化ルール、およびプロセスが含まれます（以下を参

照）。

セット

公開プロファイルを作成する場合、プロファイルパラメータでは、公開用のセットを指定する必要があります。公開プロファイルを構成する前に、次の高度なリポジトリ検索条件を使用してセットを作成します：

- 検索-すべてのタイトル
- タグ抑制条件=いいえ
- その他のシステム番号条件 = <ISIL code>



Aleph中央目録への公開の設定

正規化ルール

利用可否、所蔵、およびアイテム公開プロファイルの両方に対して正規化ルールを作成する必要があります。例については以下を、 指示については正規化プロセスへの対応セクションを参照してください。

次の正規化ルールの例は、利用可否公開プロファイル用です：

LOWフィールドの正規化

LOWフィールドの正規化の例：

```
rule "Move Aleph BIB ID to 001"
priority 6
when
(exists "035.a.(<ISIL code>)*")
then
removeControlField "001"
removeField "900" if (exists "900")
copyField "035.a" to "900.a" if (exists "035.a.*<ISIL code>*")
copyControlField "900.a" to "001"
replacecontrolcontents "001.{0,8}" with ""
end

rule "Combine 852"
priority 5
when
(exists "852")
then
combineFields "852" excluding "a"
end

rule "Add Library code to LOW field when 852 exist"
priority 4
when
(exists "852")
```

LOWフィールドの正規化の例：

```
then
addField "LOW.a.<library code> " if (not exists "LOW.a.<library code>") end<library code> " if (not exists "LOW.a.<library code>")
end

rule "Add Delete tag to 852.u for deleted BIB"
priority 3
when
(existsControl "LDR.{5,1}.d")
then
addField "LOW.a.<library code>" if (not exists "LOW.a.<library code>")
replacecontents "LOW.u" with "DEL" if (exists "LOW.u")
addSubField "LOW.u.DEL" if (not exists "LOW.u")
end

rule "Add Delete tag & Library code when there is no 852"
priority 2
when
(not exists "852")
then
addField "LOW.a.<library code>" if (not exists "LOW.a.<library code>")
addSubField "LOW.u.DEL" if (not exists "LOW.u")
end

rule "Remove all BIB fields but 001 & LOW"
priority 1
when
(TRUE)
then
removeControlField "002"
removeControlField "003"
removeControlField "004"
removeControlField "005"
removeControlField "006"
removeControlField "007"
removeControlField "008"
removeControlField "009"
removeField "01*"
removeField "02*"
removeField "03*"
removeField "04*"
removeField "05*"
removeField "06*"
removeField "07*"
removeField "08*"
removeField "09*"
removeField "1*"
removeField "2*"
removeField "3*"
removeField "4*"
removeField "5*"
removeField "6*"
removeField "7*"
removeField "8*"
removeField "9*"
end
```

次の正規化ルールの例は、所蔵公開プロファイル用です。

所蔵の正規化の例：

```
rule "Delete 852 when 852.8 doesn't exist"
priority 7
when
(exists "852")
then
removeField "852" if (not exists "852.8")
end

rule "Move Aleph BIB ID to 012.a"
priority 6
when
(exists "035.a.(<ISIL code>)*")
then
copyField "035.a" to "012.a" if (exists "035.a.*<ISIL code>*")
replaceContents "012.a.(<ISIL code>)" with ""
end

rule "Add Library code to 012 $l"
priority 5
when
(TRUE)
then
addField "012.l.<library code>"
end

rule "Convert holdings Marc to Mab when 852 exist"
priority 4
when
(exists "852")
then
removeControlField "001"
copyControlField "852.8" to "001"
copyField "852.a" to "200.b" if (exists "852.a")
copyField "852.b" to "OWN.a" if (exists "852.b")
copyField "852.c" to "200.g" if (exists "852.c")
copyField "852.h" to "200.f" if (exists "852.h")
end

rule "Remove BIB LDR only if holdings LDR lies in 009"
priority 3
when
existsControl "009"
then
removeControlField "LDR"
changecontrolField "009" to "LDR"
end

rule "Remove all BIB fields but 001 & 012"
priority 2
when
(TRUE)
then
removeControlField "002"
removeControlField "003"
removeControlField "004"
removeControlField "005"
```

所蔵の正規化の例：

```
removeControlField "006"
removeControlField "007"
removeControlField "008"
removeControlField "009"
removeField "01*" if (not exists "012")
removeField "02*"
removeField "03*"
removeField "04*"
removeField "05*"
removeField "06*"
removeField "07*"
removeField "08*"
removeField "09*"
removeField "1*"
removeField "2*" if (not exists "200")
removeField "3*"
removeField "4*"
removeField "5*"
removeField "6*"
removeField "7*"
removeField "8*"
removeField "9*"
end

rule "Combine 012 & 200"
priority 1
when
(TRUE)
then
combineFields "012" excluding ""
combineFields "200" excluding ""
end
```

次の正規化ルールの例は、アイテム公開プロファイル用です：

アイテムの正規化

アイテムの正規化の例：

```
rule "Move Aleph BIB ID to 001"
priority 2
when
TRUE
then
removeControlField "001"
copyField "035.a" to "999.a" if (exists "035.a.*<ISIL code>*")
replacecontents "999.a.(<ISIL code>)" with ""
copycontrolField "999.a" to "001"
end

rule "Remove all BIB fields but 001"
priority 1
when
(TRUE)
then
```

アイテムの正規化の例：

```
removeControlField "002"
removeControlField "003"
removeControlField "004"
removeControlField "005"
removeControlField "006"
removeControlField "007"
removeControlField "008"
removeControlField "009"
addControlField "009.<library code>"
removeField "01*"
removeField "02*"
removeField "03*"
removeField "04*"
removeField "05*"
removeField "06*"
removeField "07*"
removeField "08*"
removeField "09*"
removeField "1*"
removeField "2*"
removeField "3*"
removeField "4*"
removeField "5*"
removeField "6*"
removeField "7*"
removeField "8*"
removeField "9*"
end
```

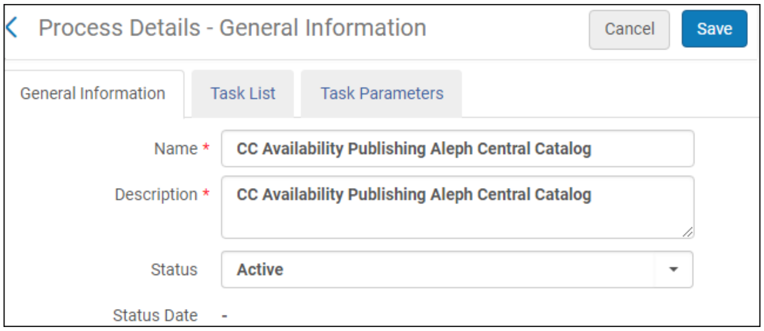
プロセス

作成したそれぞれの正規化ルールについて、作成した公開プロファイルの**正規化ルールを使用してデータを修正**ドロップダウンリストから選択することができるプロセスを作成する必要があります。詳細については、正規化プロセスの操作を参照してください。

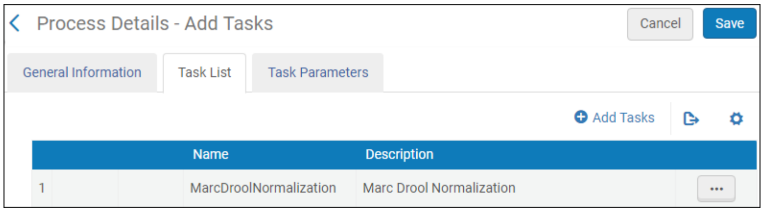
Alephセントラル目録を使用するために作成するプロセスについては、以下を指定する必要があります：

- ビジネスエンティティ-**書誌タイトル**を選択します。
- タイプ-**Marc 21 書誌正規化**を選択します。
- プロセス-**MarcDroolNormalization**を選択します。
- Droolsファイルキー-作成した正規化ルールの名前を選択します。

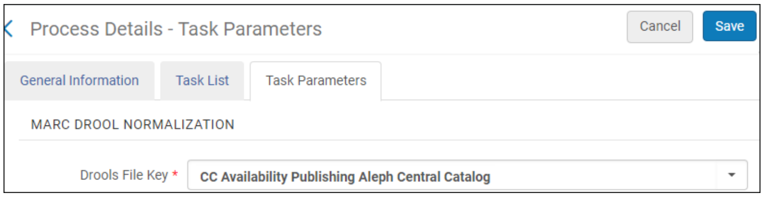
利用可否の正規化ルール用に作成されたプロセスの例については、以下のスクリーンショットを参照してください。



プロセス設定の一般情報



プロセス設定選択プロセス



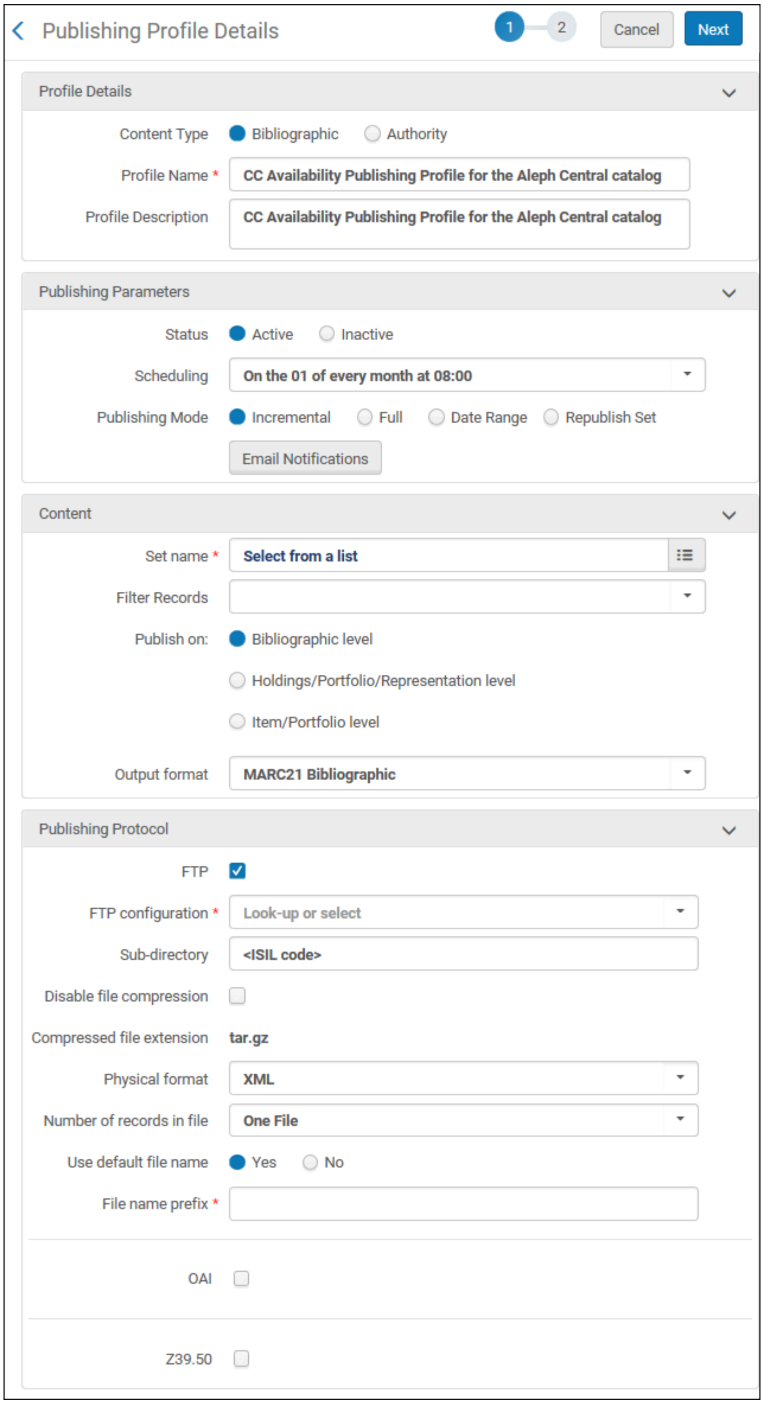
プロセス設定Droolsファイルキー

利用可否公開の設定

Alephセントラル目録で使用するために作成する利用可否公開プロファイルについては、以下の仕様で一般プロファイルを追加します：

ウィザードステップ1

- プロファイル名を入力します。
- ドロップダウンリストから日次スケジュールオプションのいずれかを選択します。
- 公開プロファイル用に過去に作成したセットの名前を指定します。
- 公開レベルパラメータで、書誌レベルを選択します。
- MARC 21書誌出力フォーマットを選択します。
- インクリメンタル公開モードを選択します。
- 公開プロトコルでFTPを選択します。FTPアカウントを設定する手順については、S/FTP接続の設定を参照してください。



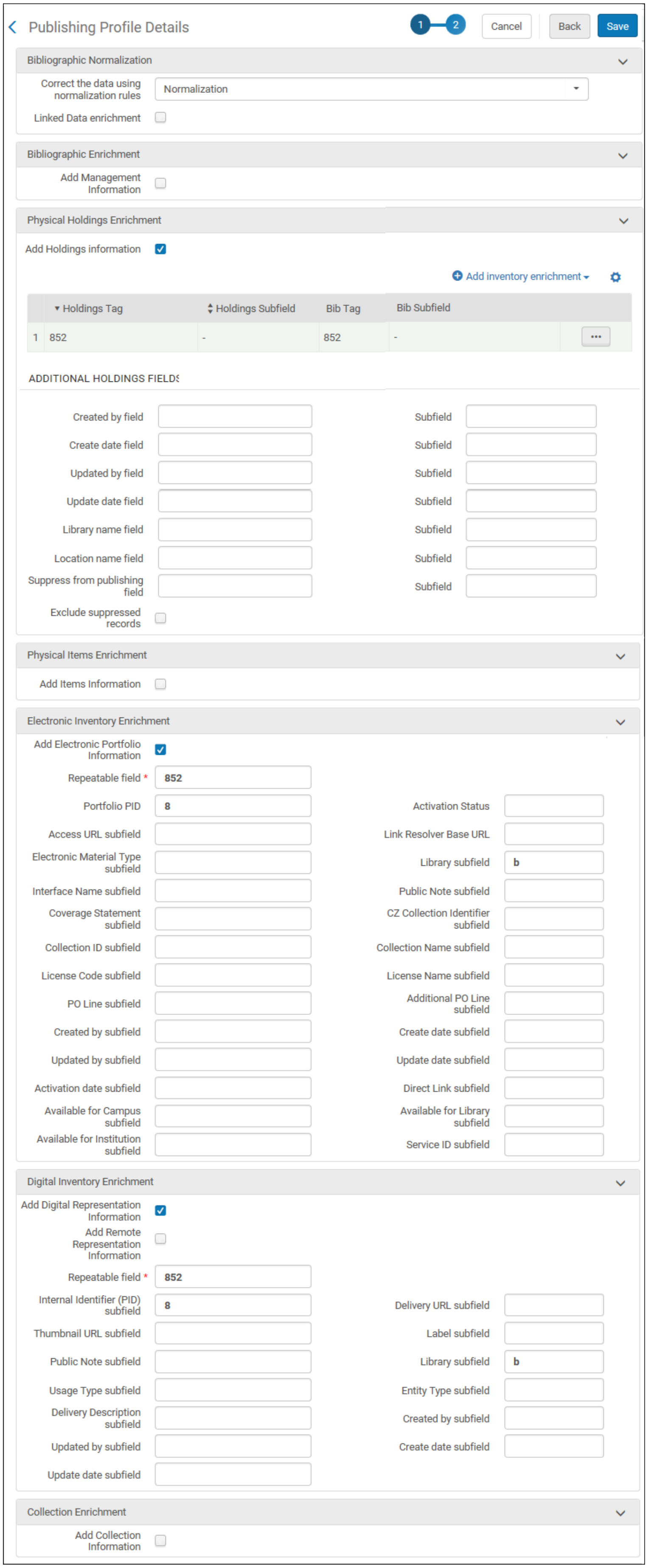
利用可否公開プロファイルウィザードステップ1

ウィザードステップ2

- 正規化ルールを使用してデータを修正ドロップダウンリストから、利用可否公開プロファイル用に作成したLOWフィールド正規化ルールを選択します。
- 所蔵情報を追加を選択し、852書誌タグに852の所蔵タグをマッピングします。
- 電子ポートフォリオ情報を追加を選択し、以下を指定します：
  - 繰り返し可能なフィールドとしての852へのマッピング
  - ポートフォリオPIDを$8にマッピング

- 図書館を$bにマッピング

- デジタル表記情報を追加を選択し、以下を指定します：
  - 繰り返し可能なフィールドとしての852へのマッピング
  - 内部識別子（PID）を$8ドルにマッピング
  - 図書館を$bにマッピング



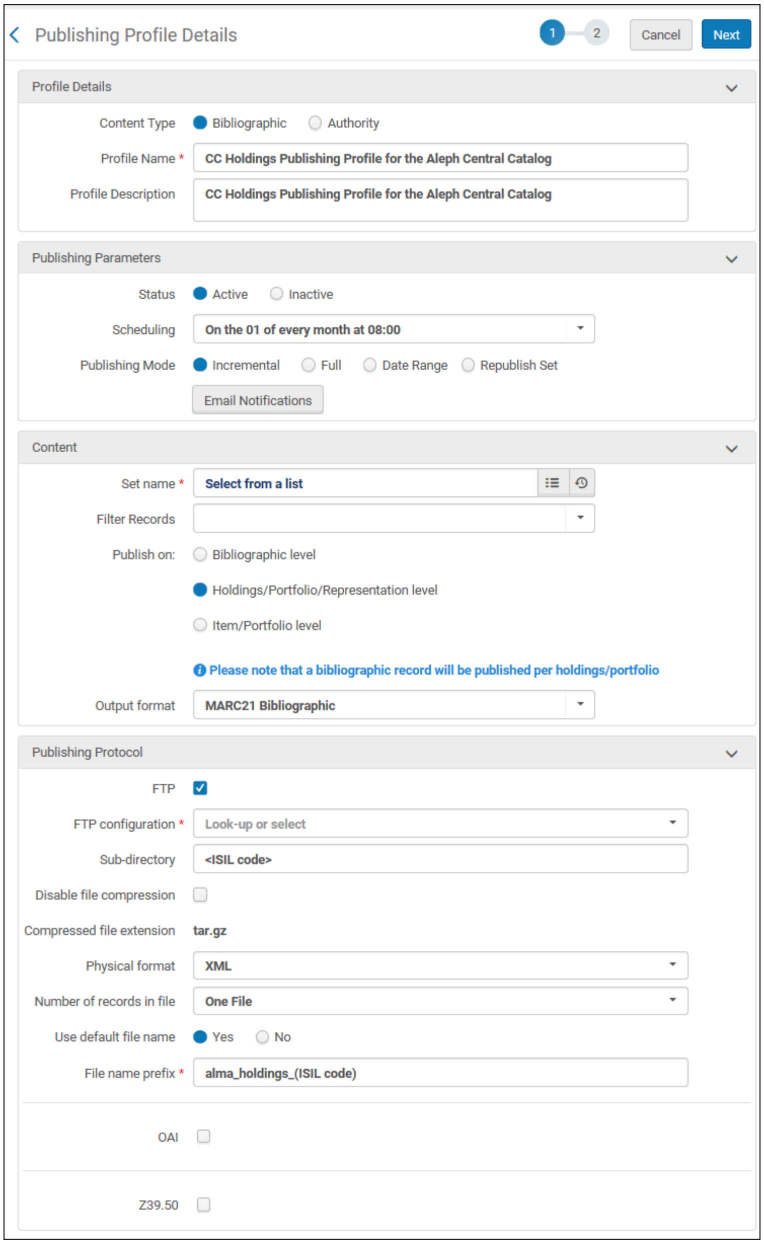
Publishing Profile Details
1
2
Cancel
Back
Save
Bibliographic Normalization
Correct the data using normalization rules
Normalization
Linked Data enrichment
Bibliographic Enrichment
Add Management Information
Physical Holdings Enrichment
Add Holdings information
Add inventory enrichment
Holdings Tag
Holdings Subfield
Bib Tag
Bib Subfield
1
852
-
852
-
ADDITIONAL HOLDINGS FIELDS
Created by field
Subfield
Create date field
Subfield
Updated by field
Subfield
Update date field
Subfield
Library name field
Subfield
Location name field
Subfield
Suppress from publishing field
Subfield
Exclude suppressed records
Physical Items Enrichment
Add Items Information
Electronic Inventory Enrichment
Add Electronic Portfolio Information
Repeatable field *
852
Portfolio PID
8
Activation Status
Access URL subfield
Link Resolver Base URL
Electronic Material Type subfield
Library subfield
b
Interface Name subfield
Public Note subfield
Coverage Statement subfield
CZ Collection Identifier subfield
Collection ID subfield
Collection Name subfield
License Code subfield
License Name subfield
PO Line subfield
Additional PO Line subfield
Created by subfield
Create date subfield
Updated by subfield
Update date subfield
Activation date subfield
Direct Link subfield
Available for Campus subfield
Available for Library subfield
Available for Institution subfield
Service ID subfield
Digital Inventory Enrichment
Add Digital Representation Information
Add Remote Representation Information
Repeatable field *
852
Internal Identifier (PID) subfield
8
Delivery URL subfield
Thumbnail URL subfield
Label subfield
Public Note subfield
Library subfield
b
Usage Type subfield
Entity Type subfield
Delivery Description subfield
Created by subfield
Updated by subfield
Create date subfield
Update date subfield
Collection Enrichment
Add Collection Information

所蔵公開の設定

Alephセントラル目録で使用するために作成する所蔵公開プロファイルについては、以下の仕様で一般プロファイルを追加します：

ウィザードステップ1

- プロファイル名を入力します。
- ドロップダウンリストからスケジュールオプションを選択します。
- 公開プロファイル用に過去に作成したセットの名前を指定します。
- 公開レベルパラメータで、**所蔵/ポートフォリオ/代表レベル**を選択します。
- **MARC 21書誌**出力フォーマットを選択します。
- インクリメンタル公開モードを選択します。
- 公開プロトコルで**FTP**を選択します。FTPアカウントを設定する手順については、S/FTP接続の設定を参照してください。

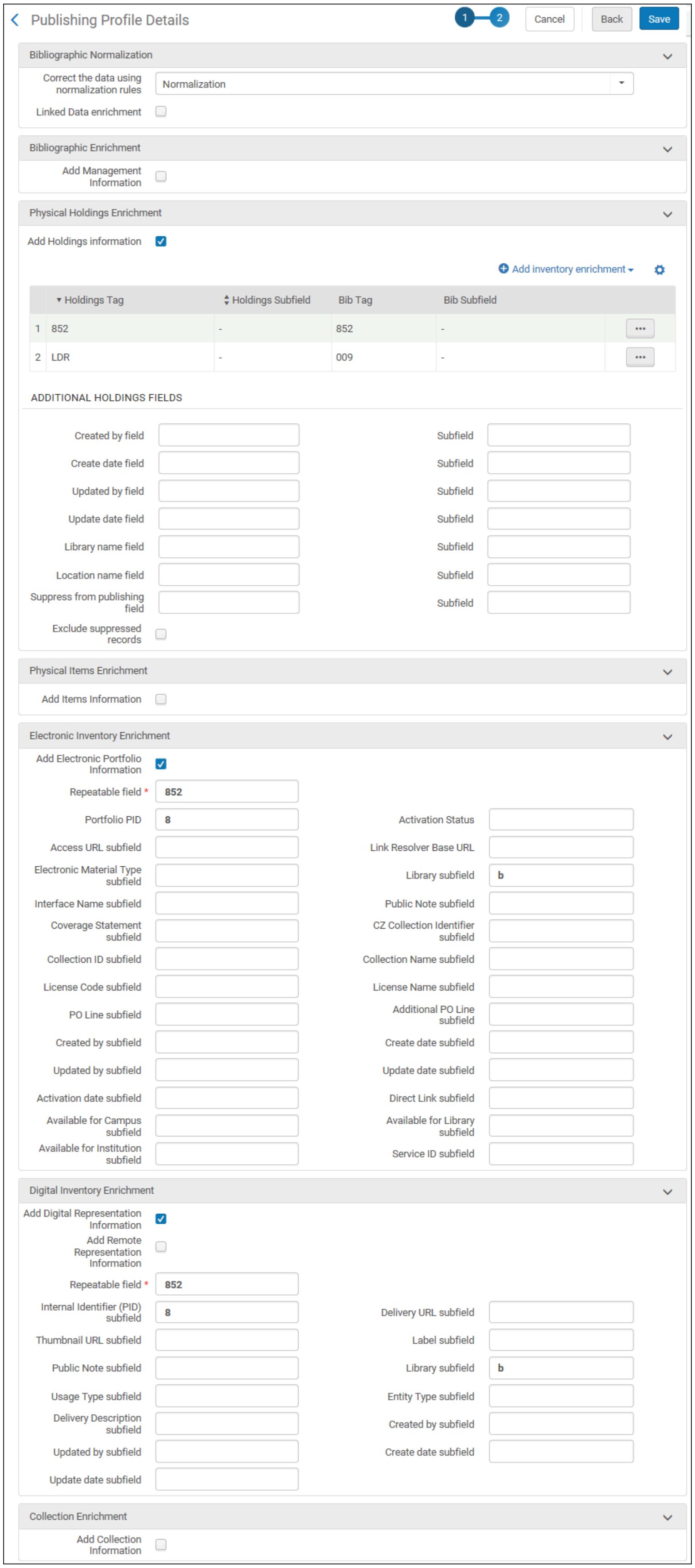


所蔵公開プロファイルウィザードステップ1

ウィザードステップ2

- 正規化ルールを使用してデータを修正ドロップダウンリストから、所蔵公開プロファイル用に作成した所蔵正規化ルールを選択します。
- 所蔵情報を追加を選択し、以下の通りマッピングします：
  - 852所蔵タグを852書誌タグへ
  - LDRを009書誌タグへ
- 電子ポートフォリオ情報を追加を選択し、以下を指定します：
  - 繰り返し可能なフィールドとしての852へのマッピング
  - ポートフォリオPIDを$8にマッピング
  - 図書館を$bにマッピング
- デジタル表記情報を追加を選択し、以下を指定します：

- 繰り返し可能なフィールドとしての852へのマッピング
- 内部識別子（PID）を$8ドルにマッピング
- 図書館を$bにマッピング

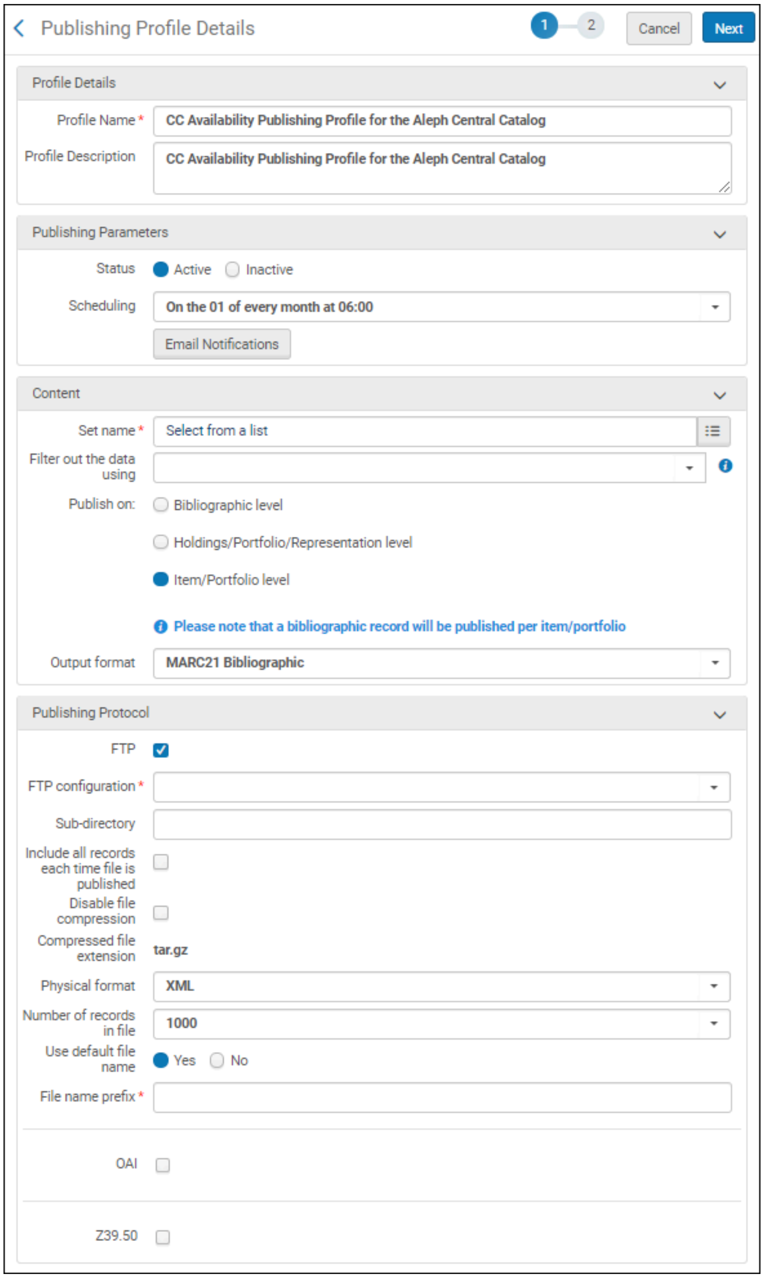


アイテム公開の設定

Alephセントラル目録で使用するために作成するアイテム公開プロファイルについては、以下の仕様で一般プロファイルを追加します：

ウィザードステップ1

- プロファイル名を入力します。
- ドロップダウンリストからスケジュールオプションを選択します。
- 公開プロファイル用に過去に作成したセットの名前を指定します。
- 公開レベルパラメータで、アイテム/ポートフォリオレベルを選択します。
- **MARC 21**書誌出力フォーマットを選択します。
- インクリメンタル公開モードを選択します。
- 公開プロトコルで**FTP**を選択します。FTPアカウントを設定する手順については、S/FTP接続の設定を参照してください。



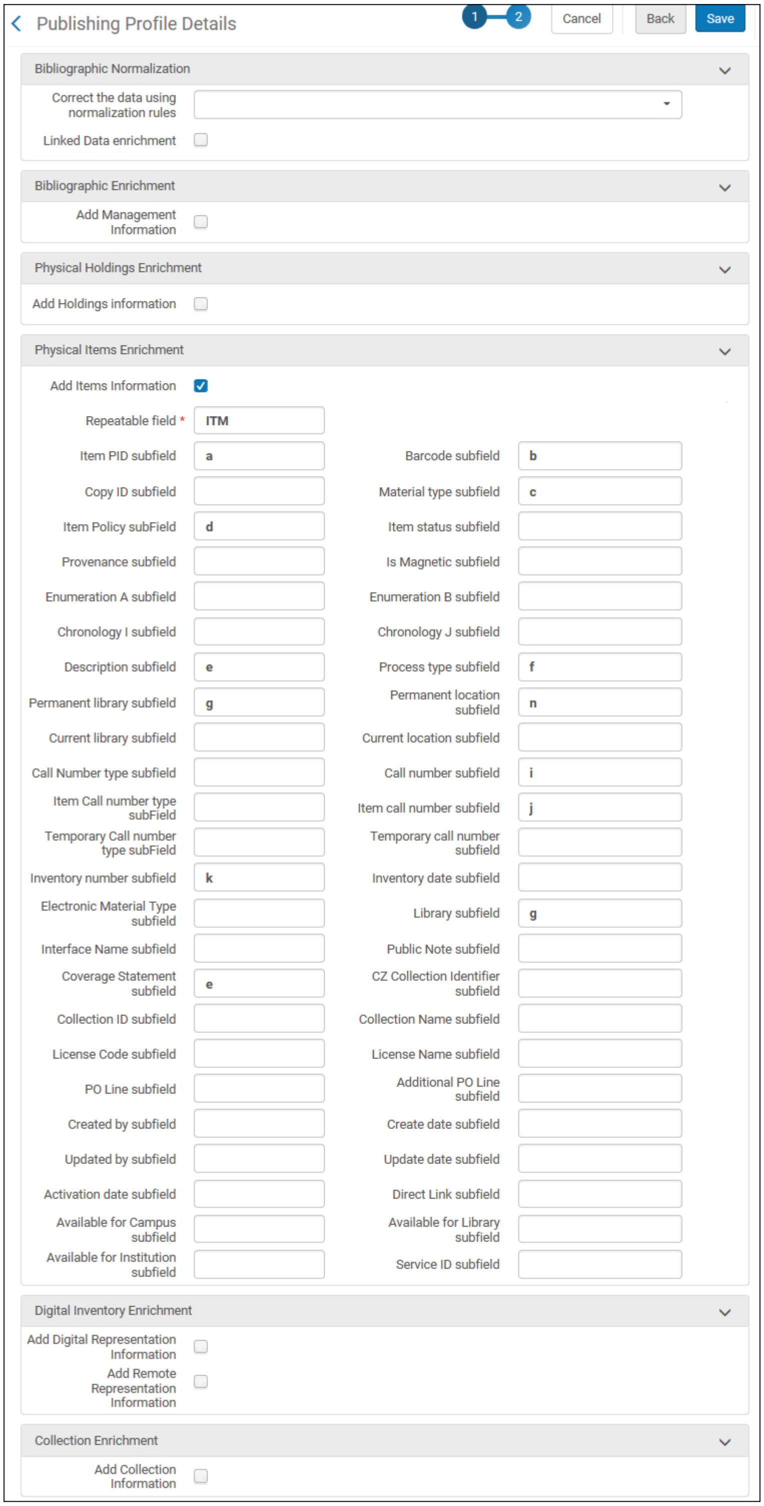
アイテム公開プロファイルウィザードステップ1

ウィザードステップ2

- 正規化ルールを使用してデータを修正ドロップダウンリストから、所蔵公開プロファイル用に作成したアイテム正規化ルールを選択します。
- アイテム情報を追加を選択し、以下を指定します。
  - 繰り返し可能なフィールドとしてのITMフィールドへのマッピング
  - アイテムPIDサブフィールドの$aへのマッピング
  - アイテムポリシー サブフィールド$dへのマッピング
  - 説明サブフィールドの$eへのマッピング
  - 恒久的図書館サブフィールドの$gへのマッピング
  - 目録番号サブフィールドの$kへのマッピング

- バーコードサブフィールドの$bへのマッピング
- 資料タイプサブフィールドの$cへのマッピング
- プロセスタイプサブフィールドの$fへのマッピング
- 恒久的場所サブフィールドの$nへのマッピング
- 請求番号サブフィールドの$iへのマッピング
- アイテム請求番号サブフィールドの$jへのマッピング

- **電子ポートフォリオ情報を追加**を選択し、以下を指定します：
  - 繰り返し可能なフィールドとしてのPRTフィールドへのマッピング
  - ポートフォリオPIDの$aへのマッピング
  - カバレッジ明細サブフィールドの$eへのマッピングする
  - 図書館の$gへのマッピング



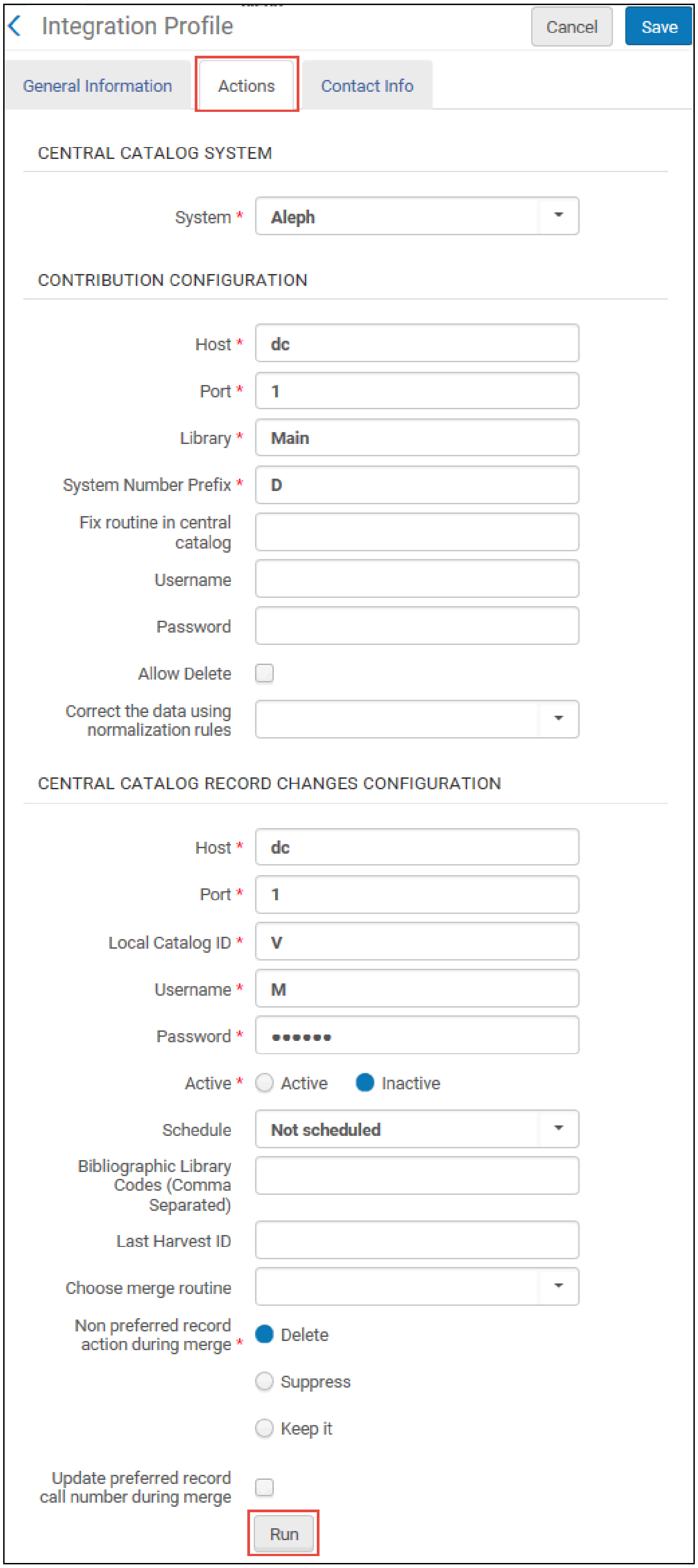
アイテム公開プロファイルウィザード ステップ2

Alephセントラル目録からの変更されたレコードの検索

Aleph中央目録の統合プロファイルを作成して保存した後、統合プロファイルのアクションタブには、Aleph中央目録から変更されたレコードを検索するジョブを手動で実行するオプションが表示されます。

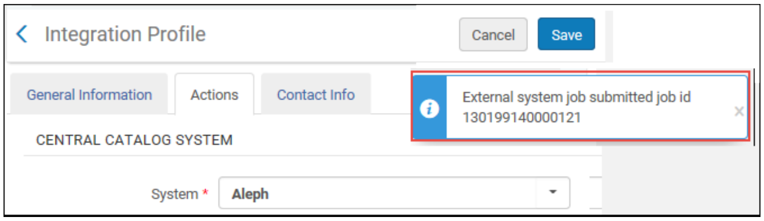
レコードの作業が終わったら、レコードが リリース されていることを確認してください。Aleph 中央目録統合ジョブでは

下書きが削除されないためです。レコードがリリースされていない場合は、レコードにアクセスする際のエラーを回避するため、レコードを再ロードする必要があります。



Alephセントラル目録統合プロファイルアクションタブ

実行を選択して、ジョブを送信します。



実行を使用して、**Aleph**セントラル目録から変更されたレコードを検索する

Note

このジョブを実行すると、編集中のレコードのロックは無視され、レコードがインポートされます。

Note

階層関係についてですが、現在のAleph中央目録 統合の設計では、削除を目的として子レコードがAlephからAlmaに送信された場合、関連する親レコードの削除はサポートされません。

Alephセントラル目録統合ジョブリポート

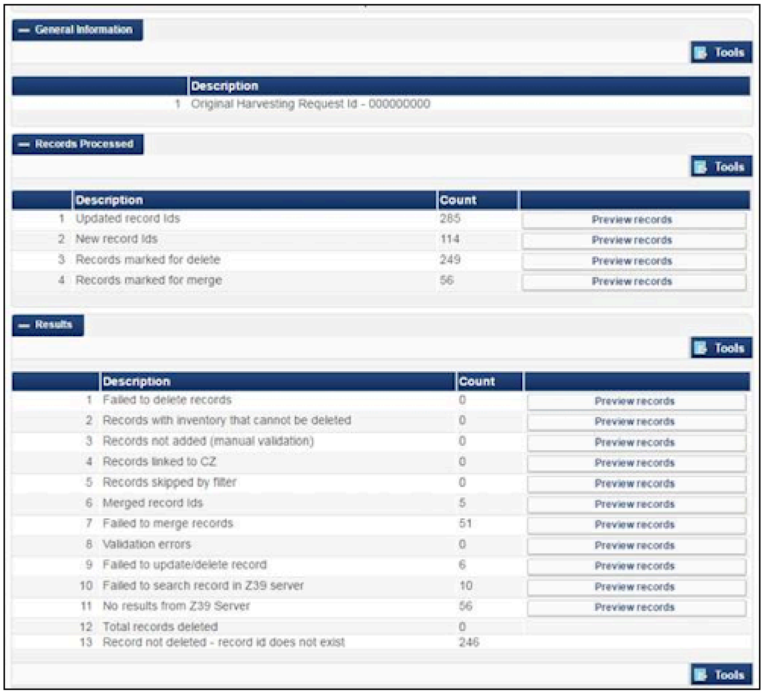
Aleph中央目録統合ジョブを実行した後、作業の監視ページの履歴タブ（**[管理] > [作業の管理] > [作業の監視]**）を確認して、結果を表示します。

また、その代わりに、統合プロファイルリストページ（設定 メニュー **>** 一般 **>** 外部システム **>** 統合プロファイル）から、Alephセントラル目録の統合プロファイルを含む行でアクション **>** ジョブ履歴を選択します。ジョブ履歴ページが表示されます。ジョブリポートのページを表示するために、ジョブ履歴のページから、完了したジョブのアクション**>** ビューを選択します。処理済みレコードセクションと結果セクションを展開して、以下の件数（レコード数）を表示します：

Alephセントラル目録統合ジョブリポート

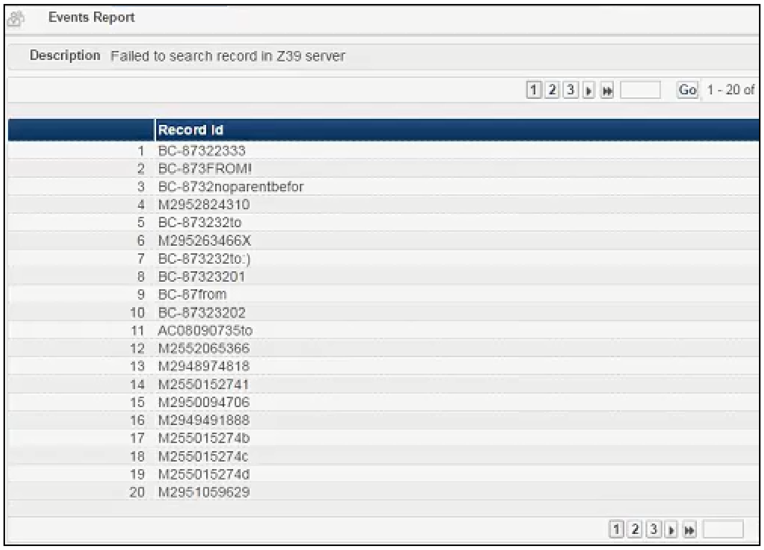
セクション	説明
処理済みレコード	• アップデート済みレコードID • 新しいレコードID • レコードが削除対象としてマークされています • レコードが統合対象としてマークされています
検索結果	• レコードの削除に失敗しました • 削除できない目録を含むレコード • 追加されていないレコード（手動検証） • CZにリンクされたレコード • レコードがフィルタによりスキップされました

セクション	説明
	• 統合済のレコードID • レコードの統合に失敗しました • 検証エラー • レコードのアップデート/削除に失敗しました • Z39サーバーでのレコードの検索に失敗しました • Z39サーバーからの検索結果がありません • 削除されたレコードの総数 • レコードが削除されませんでした - データIDが存在しません



Alephセントラル目録統合ジョブリポート

これらのセクションでは、オプションとして、レコードをプレビューを選択してレコードの詳細を表示することができます。

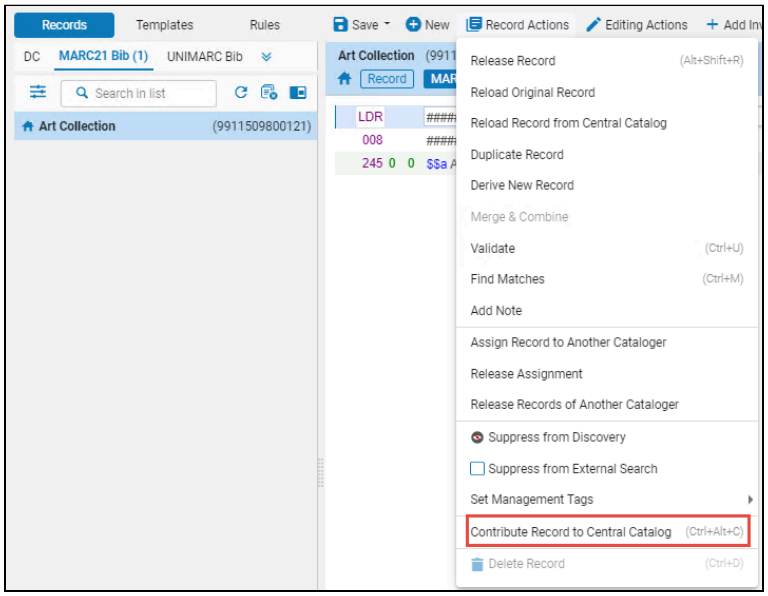


Z39サーバーからの結果がない障害

Z39サーバ障害がジョブ結果で発生した場合、ジョブ結果には「**Z39**サーバーでレコードを**検**索できませんでした」と「**Z39**サーバーからの結果はありません」が表示されます。Almaはこれらの失敗試行の記**録**を保存し、 次回ジョブが実行されるときに再度処理します。

レコードをセントラル目**録**に提供の使用

Alephセントラル目**録**統合プロファイルを設定した後、MDエディタのレコードをセントラル目**録**に提供オプション（レコードアクション **>** レコードをセントラル目**録**に提供）が利用可能になります。



詳細については、レコードをセントラル目**録**に提供を参照してください。

Note

統合プロファイルで「セントラル目**録**に自動的に提供する」オプションが選**択**されている場合（上記を参照）、「セントラル目**録**にレコードを提供する」 アクションは、Alephにすでに提供されているレコード（システムプレフ

ィックスが付いた035 $aのレコード（例：DE-605））に対してアクティブになりません。このようなレコードは、保存すると自動的にAlephに送信されます。
「セントラル目録にレコードを提供する」 アクションは、レコードが「保存時のMarc21検証」手順 のいずれかに失敗した場合、レコードをAlephに送信しません（プロセスの設定を参照）。

Note

英語以外の言語を使用するには、次のパラメーターをPOST URLに挿入します：

> $form {'con_lng'} = 'lng' ;

例：

> $form{'con_lng'} = 'ger' ;

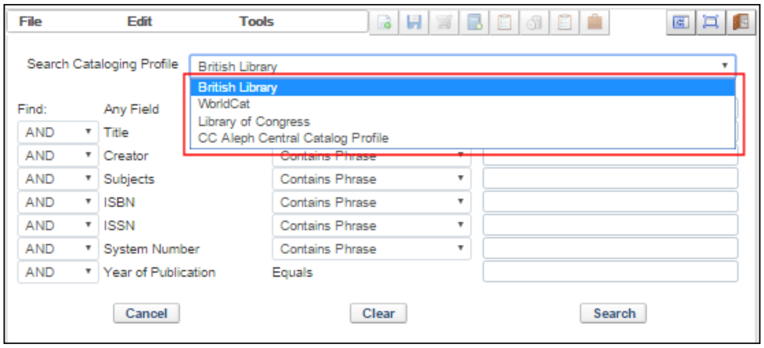
Alephセントラル目録の元のシステムバージョン

Almaに保存されている書誌レコードの場合、005フィールドにはAlmaの日付と時刻のバージョン情報が含まれます。Alephセントラル目録データベースに対応するレコードがある、AlmaがAlmaデータベースに保存している各レコードについて、AlmaはAlephレコードの元のシステムバージョンを別の場所に保存します。

その後、AlmaがAlephセントラル目録にレコードを提供すると、005フィールドのAlmaバージョン情報が、過去に個別に保存されたAleph発信システムバージョン情報に置き換えられます。これは、Alephセントラル目録がそのレコードのどのバージョンがAlmaの提供によってアップデートされているかを識別するために行われます。

Alephセントラル目録統合プロファイルで外部検索の使い方

Alephセントラル目録の統合プロファイルを設定した後、プロファイルはAlma MDエディタで利用できる外部リソースを検索するオプション（**ツール > 検索外部リソース**）の**検索カタログプロファイル**のドロップダウンリストから選択するためのオプションとして表示されます。



Alephセントラル目録の外部検索

詳細については外部リソースを検索するを参照してください。

MDエディタの、外部リソースを検索する、からAlephセントラル目録プロファイルを使用することを選択すると、システム番号パラメータは、Alephセントラル目録内のレコードの001フィールドを検索するために使用することができます。