
# Parallel Indexing

Versions 16 and later

## CONFIDENTIAL INFORMATION



## DISCLAIMER



## TRADEMARKS



Document released: November 6, 2012

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

# Table of Contents

# Introduction

Parallel indexing is used to re-build an OPAC index, parallel to the online ALEPH system, with no downtime while the index is being created. This process also enables changing indexing parameters, and checking the results, without losing current indexes. This document outlines how to accomplish this.

The indexing is done in a separate library. This library is set up with a pointer to the document records in the actual library, and indexes are located in the indexing library. After indexing has been completed, a pointer is created in the actual library to the index in the indexing library.

In this document, USM01 is being used as the actual library, and USM21 as the indexing library.

To start the indexing process in the USM21 library, we define all index tables (e.g., for Word tables this would be Z97, Z98, etc.) as local in the files list, and the documents file (z00, z103, z0102) as a logical synonym to USM01. We can now run the indexing job (e.g., p_manage_01) in USM21. It reads records from USM01, via the logical synonym, but creates index tables locally. We can also change index setup in the indexing library to create different index codes or to use different filing procedures, etc.

After the index is built successfully, we can check it in USM21 using the Web OPAC.

Finally, when we want to switch to the new index we create a logical synonym from USM01 to USM21 for all relevant tables. Thus there is no downtime whatsoever.

Re-indexing word (W-nnn) and direct (IND) indexes is a process which is complete within itself and does not require any other indexing. Re-indexing the headings (ACC) index, on the other hand, requires a series of indexing jobs.

If you have used option 1 under Step 9, the next time that you want to re-index, you have to create an additional indexing library (in order that both the bib documents library and the indexing library remain unlocked).

# Step 1: Open a Library

Open a new BIB library, parallel to the library that is going to be indexed.

# Step 2: Add the Indexing Library to library_relation

The /alephe/tab/library_relation table defines relationships among various libraries. For parallel indexing a PID relationship must be defined, as in the following example:

```
PID USM21 USM01
```

In addition the relationship between the indexing library and the ADM and HOL libraries should be defined in exactly as the same way as the relationship between the actual library and its related ADM and HOL libraries. For example:

```
ADM USM01 USM50 USM51
HOL USM01 USM60

ADM USM21 USM50 USM51
HOL USM21 USM60

PID USM21 USM01
```

## Step 3: Adjust the Library's file_list

In the root directory of each ALEPH library there is a configuration table called file_list. This table lists all of the library's Oracle tables, their size, extents and location. In this configuration table, it is possible to define that the library uses an Oracle table of a different library, instead of its own Oracle table. This is accomplished by setting a pointer to the other library, using a "logical symbol" (LS) definition.

Initially, in the root directory of the new library, there should be a copy of the actual library's file_list. At this stage, the file_list should contain all the Oracle tables listed in the actual library's file_list, using the same definitions. Later, the definitions are changed, in both the actual and the indexing library, as required. That is, if you are running p_manage_01 (for Words) in the parallel library, you need to have the following entries in the parallel library's file_list:

```
TAB z95              100K            0K              ts0
TAB z97              100K            0K              ts1
TAB z98              100K            0K              ts2
IND z95_id           100K            0K              ts0
IND z97_id           100K            0K              ts1
IND z97_id1          100K            0K              ts1
IND z97_id2          100K            0K              ts1
IND z97_id3          100K            0K              ts1
IND z98_id           100K            0K              ts2
TAB z980             100K            0K              ts3
IND z980_id          100K            0K              ts3
```

If you are running p_manage_102 / p_manage_02 (for Headings), you need to have the following entries:

```
TAB z01              100K            0K              ts0
TAB z02              100K            0K              ts0
IND z01_id           100K            0K              ts0
IND z01_id2          100K            0K              ts0
IND z01_id3          100K            0K              ts0
IND z01_id4          100K            0K              ts0
IND z01_id5          100K            0K              ts0
IND z02_id           100K            0K              ts0
IND z02_id1          100K            0K              ts0
```

If you are running p_manage_05, you need to have the following entries:

```
TAB z11                 100K            0K              ts0
IND z11_id              100K            0K              ts0
IND z11_id1             100K            0K              ts0
```

If you are running p_manage_07, you need to have the following entries:

```
TAB z13                 100K            0K              ts0
IND z13_id              100K            0K              ts0
IND z13_id1             100K            0K              ts0
```

If you are running p_manage_27, you need to have the following entries:

```
TAB z101                100K            0K              ts0
IND z101_id             100K            0K              ts0
```

If you are running p_manage_35, you need to have the following entries:

```
TAB z0101               100K            0K              ts0
IND z0101_id            100K            0K              ts0
IND z0101_id1           100K            0K              ts0
IND z0101_id2           100K            0K              ts0
```

These entries are appropriate for versions 18, 19, and 20. If you are working with a later version, check the source library's file_list in case there have been any additions.

**Note**:
The above values of COL 3 (Initial Extent Allocation), COL 4 (Next Extent Allocation), and COL 5 (Tablespace Name) are only examples.

In the indexing library the sequence numbers table, Z52, should always be local:

```
TAB z52                 10K             10K             ts0
IND z52_id              10K             10K             ts1
```

**Important**:
In the indexing library the bibliographic documents table, Z00, as well the Z103 and Z0102* tables should *always* be a logical synonym:

```
LS  z00                 USM01
LS  z103                usm01
LS  z0102               usm01 *
```

Before initiating parallel indexing, the above tables should be dropped in the indexing library (the parallel library).

Before doing the following SQL drops, do the following select:

>>s+ usm21
SQL-USM21>select count(*) from Z00;

The result should be 0. If it is not, it indicates that you either:

(1) are not in the parallel library or

(2) are LS-ed from the parallel library to the production.

**Quit immediately.**

In the case of #1, do "s+" to the correct, parallel library.

In the case of #2, do util a/17/5/1 to confirm that the synonyms already exist. If so, you do not need to do anything more: the fact that the synonyms exist indicates that the tables have been dropped. (Oracle does not let you create the synonym when the table exists in the parallel library.)

If the result of the "select count" is 0, then proceed with the SQL commands for doing the drop:

SQL-USM21>drop table Z00;
SQL-USM21>drop table Z103;
SQL-USM21>drop table Z0102;*

After changing the file_list and dropping the tables, create the logical synonyms (UTIL A/17/5/2) in the parallel library.

* In v17-up, and v16 with new_ue_01, p_manage_32 (which builds the z0102) can also be done in the parallel library.

Do not forget to check if there is enough free space to build the index in the parallel library tablespace (util o/14/1). You can see the size of the index tables in the actual (USM01) library with util a/17/11/2. (Note that the latter result is in KB whereas the first is in MB). The new tables require an equal amount of space in the parallel library tablespaces ( -- unless you change the number of indexes or fields indexed).

If the p_manage_01 Word indexing job is running and if the actual library has z113, z114, or z970 tables, add the following to the indexing library's file list where usm01 is your bib library and then do util a/17/5/2 in the indexing library to create the synonyms:

```
LS  z113              usm01
LS  z114              usm01
LS  z970              usm01
```

**Note**: If you prefer, you can instead copy these tables from the actual library to the indexing library.

## Step 4: Adjust the Library's Z52 Table

Before initiating parallel indexing, make sure that the following counters are defined as listed. Use UTIL/G/2 to add missing counters, and update the counter values.

Required values are:

last-doc-number (set to the same value as last-doc-number of the actual library)
last-acc-number ( set to 0)

last-long-acc-number (set to 0)
last-similar-acc-number (set to 0)
last-word-number (set to 0)

## Step 5: Set up Indexing Configuration Tables

Define the setup of the indexing tables in the tab directory of the indexing library.
You can choose to copy the tables that are used for indexing in the actual library into the tab directory of the indexing library, or you can use the path_convert configuration table to direct the system to the actual library's configuration tables. This option is feasible only if you do not want to change the indexing setup.

You may change $data_tab values in the indexing library in order to improve the indexing. If you do this, be sure to copy these changed tables to the actual library in Step 9.

## Step 6: Save Interim Indexing Updates

While the indexing jobs are running, new and updated records in the actual library are indexed, through the ue_01 and Z07 mechanism, on the old indexes. These Z07 records must be saved, for re-indexing after the new indexes have been built.
In order to save these records, before running the indexing process, perform the utility E/5/1 in the actual library.
Util-e-05 allows the creation of a history table for z07 entries handled by ue_01.
These stored entries may be later used to re-execute ue_01 on the same records. (See Step 10, below.)

## Step 7: Run the Indexing Jobs

### Word

Run p_manage_01 to rebuild the Word index of the database.

### Direct indexes

Run p_manage_05 to rebuild the Direct index of the database.

### Headings Indexes

1. Optional: Run p_manage_102; to pre-enrich the bibliographic headings index, based on the Authority Database.
2. Run p_manage_02.
3. Run p_manage_105 in the AUT libraries in order to add untraced references.
4. Run p_manage_17 in order to alphabetize long headings.
5. Run p_manage_35 in order to create brief records.

6. Run p_manage_32 to build counters for logical bases.

**Note:** In version 16 or earlier, this last job should be run in the actual library, not in the indexing library. This procedure locks the ALEPH system and should only be run when the library is closed. In version 17 and up and version 16 with new_ue_01,p_manage_32 can be run in the parallel library.

**Note**: When re-building headings (browse) indexes, you MUST run the additional indexing processes listed above.
If your AUT database does not include untraced headings, there is no need to run p_manage_105.
If you do not have logical bases, or you have not set "Y" in column 8 of tab_base.lng for any of the bases, then there is no need to run p_manage_32.
If you are not using the "brief records" there is no need to run p_manage_35.

### Other Indexing Jobs to Run

1. Run p_ manage-07 to **update short bibliographic records**.
2. Run p_manage_27 to **update the sort index**.

## Step 8: Check the New Index

Add the indexing library to /alephe/tab/tab_base.eng and add the library to the base list for the Web OPAC (/alephe/www_f_eng/base_list). Access the WEB OPAC, choose the indexing library, and check the new index. If all appears satisfactory, continue with Step 9.

**Note:** The location does not display in the OPAC Brief or Full displays when performing this test. (This is because there is no ADM or HOL for the parallel library.)

If this run of the parallel indexing is just for the Keywords, then the Browse and the Browse links in the Full display do not work in performing this test. If you feel that it is important that the Browse work in doing the testing in this case, see Appendix A.

If this run of the parallel indexing is for just the Browse, then Keyword does not work in performing this test. If you feel that it is important that Keyword work in doing the testing in this case, see Appendix A.

If your site is using Union Catalog or Union View, note that they do not work in performing this test unless you add LS's for the z120 and z127. See Appendix A.

# Step 9: Applying the New Indexes

## Option 1: Logical Synonyms

This option uses logical synonyms to point from the actual library to the indexing library Oracle tables in order to apply the new indexes.

When the indexing has been completed stop all running daemons in the actual library (UTIL E…).
The next step required is switching from the current (old) index to the new index.
Create a pointer from the actual library Oracle table to the Oracle table in the indexing library, by changing the definition in the actual library's file_list to a logical synonym.
The following example shows the new setup after re-indexing headings:

**Replace:**

```
TAB z01              2M              1M              ts0
IND z01_id           1M              1M              ts1
IND z01_id2          300K            100K            ts1
IND z01_id3          200K            100K            ts1
IND z01_id4          200K            100K            ts1
IND z01_id5          200K            100K            ts1
IND z01_id6          200K            100K            ts1
```

**By:**

```
LS  z01              USM21
```

**Replace:**

```
TAB z02              400K            100K            ts0
IND z02_id           300K            100K            ts1
IND z02_id1          400K            100K            ts1
```

**By:**

```
LS  z02              USM21
```

The following example shows the new setup after re-indexing words:

**Replace:**

```
TAB z95              1M              1M              ts0
IND z95_id           1M              1M              ts1
TAB z97              2M              1M              ts0
IND z97_id           1M              1M              ts1
IND z97_id1          1M              1M              ts1
IND z97_id2          1M              1M              ts1
IND z97_id3          1M              1M              ts1
TAB z970           100K            100K              ts0
IND z970_id        100K            100K              ts1
IND z970_id1       100K            100K              ts1
TAB z98              3M              1M              ts0
IND z98_id           2M              1M              ts1
TAB z980             1M              1M              ts0
IND z980_id          1M              1M              ts1
```

**By:**

```
LS   z95              USM21
LS   z97              USM21
LS   z970             USM21
LS   z98              USM21
LS   z980             USM21
```

Drop the relevant Z tables (above mentioned) in the actual library by using the SQL command, as in the following example:

>>s+ usm01
SQL-USM01>drop table Z01;
SQL-USM01>drop table Z02;
etc.
Then create logical synonyms to the actual library, using UTIL A/17/5.

In v.15 only**:**

```
TAB z950 1M 1M ts0
IND z950_id 1M 1M ts1
```

In v.15 only**:**

```
LS z950 USM21
```

### Option2: Oracle Import

Installations that have Oracle DBA expertise can choose to copy the new indexes (i.e. the Oracle tables) from the indexing library to the actual library.

When the new index is a result of the p_manage_02 process, update the last-acc-number counter in the actual library (using UTIL G/2) to the same value as the counter in the indexing library.

When the new index is a result of the p_manage_01 process, update the last-word-number counter in the actual library (using UTIL G/2) to the same value as the counter in the indexing library.

## Step 10: Index records that were updated in the interim

When the new index is a result of the p_manage_02 process, update the last-acc-number and the last-similar-acc-number counters in the actual library (using UTIL G/2) to the same value as the counter in the indexing library.

If the p_manage_02 service has been run in "Update headings index" procedure and Duplicate Mode: Yes; update the last-long-acc-number counter in the actual library (using UTIL G/2) to the same value as the counter in the indexing library.

When the new index is a result of the p_manage_01 process, update the last-word-number counter in the actual library (using UTIL G/2) with the same value as the counter in the indexing library.

When the new index is a result of the p_manage_35 process, update the last-z0101-sequence counter in the actual library (using UTIL G/2) with the same value as the counter in the indexing library.

When the new index is a result of the p_manage_17 process, update the last-long-acc-number counter in the actual library (using UTIL G/2) with the same value as the counter in the indexing library.

If you have changed $data_tab tables in the indexing library in order to improve the indexing (see Step 5, above), copy these changed tables to the actual library. In case some of these files have an earlier timestamp than the ones they are replacing, do util x/7 to clean out the utf_files.

**The last action in this section of the re-indexing process is restarting the daemons (UTIL E…) in the actual library.**

## Step 11: Index records that were updated in the interim

In order to include records that were updated while indexing was running in the indexing library, perform UTIL/E/5/2 in the actual library. This copies the saved Z07H records to Z07, deleting duplicate entries. The ongoing UTIL E/1 (ue_01) process in the actual library re-indexes the records stored in Z07.

## Appendix A: Optional Steps for the "Check New Index" Testing (Step 8)

As noted in Step 8, if this run of the parallel indexing is for just the Keywords, then the Browse and the Browse links in the Full display does not work. To make the Browse work in this case, you need to:

1. Comment out the z01, z02, z0101, and z0102 TAB and IND entries in the parallel library's file list.

2. Add the following LS's in the parallel library's file list:

```
LS   z01       usm01
LS   z02       usm01
LS   z0101     usm01
LS   z0102     usm01
```

3. s+ usm21

SQL> select count(*) from z01;
**If not zero, quit immediately and consult Step 3 above.**

SQL-USM21> drop table z01
SQL-USM21> drop table z02
SQL-USM21> drop table z0101
SQL-USM21> drop table z0102

4. dlib usm21 and do util a/17/5/2 to recreate the Oracle synonyms for usm21.

As noted in Step 8, if this run of the parallel indexing is for just the Browse, then Keyword does not work. To make Keyword work in this case, you need to:

1. Comment out the z95, z97, z970, z98, and z980 TAB and IND entries in the parallel library's file list.

2. Add the following LS's in the parallel library's file list:

SQL> select count(*) from z95;

**If not zero, quit immediately and consult Step 3 above.**

```
LS   z95       usm01
LS   z97       usm01
LS   z970      usm01
LS   z98       usm01
LS   z980      usm01
```

3. s+ usm21
   SQL-USM21> drop table z95
   SQL-USM21> drop table z97
   SQL-USM21> drop table z970
   SQL-USM21> drop table z98
   SQL-USM21> drop table z980

4. dlib usm21 and do util a/17/5/2 to recreate the Oracle synonyms for usm21.

As noted in Step 8, if your site is using Union Catalog or Union View, it does not work unless you add LS's for the z120 and z127. To be able to test using the union catalog / union view:

1. Comment out the z120 and z127 TAB and IND entries in the parallel library's file list.

2. Add the following LS's in the parallel library's file list:

```
LS   z120      usm01
LS   z127      usm01
```

3. s+ usm21

SQL> select count(*) from z120;
**If not zero, quit immediately and consult Step 3 above.**

   SQL-USM21> drop table z120
   SQL-USM21> drop table z127

4. dlib usm21 and do util a/17/5/2 to recreate the Oracle synonyms for usm21.