ALEPH

INTEGRATED LIBRARY SYSTEM

ALEPH VERSION 17.01

Upgrade Express 16.02 to 17.01 User Guide

Ex Libris

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.

Table of Contents

Introduction

The ALEPH integrated library system is constantly under development in order to accommodate ongoing changes in the automated library market's requirements; to respond to ALEPH customers' development requests; and to adopt and integrate new hardware and software technologies. Ex Libris releases a new version of ALEPH annually that includes new features, enhancements to existing features and one that is compliant with new hardware and software technologies.

The ALEPH upgrade process is the process of replacing a version of ALEPH by a newer version.

The main objective of the Upgrade Express is to perform the upgrade of ALEPH versions quickly and efficiently so that the cutover phase will take less than 36 hours (a weekend). It is important to note that during this period, the library's OPAC can function in read-only mode (in other words, activities such as hold requests, photocopy requests, and so on, are disabled).

The methodology of Upgrade Express is simple. Upon the new version - ALEPH 17.01 - a copy of the 16.02 **utree** and database is added. This creates an ALEPH 17.01 software environment with 16.02 configuration and data. Minor changes need to be made to allow login. After these changes, the Upgrade process is executed and transforms the configuration and data into 17.01 format. The whole process is performed without download and/or upload of data.

This process can be performed several times for testing before the final cutover. We highly recommend performing at least one test.

Although much of the Upgrade is done automatically, some configuration has to be done manually, for example, the implementation of changes to Web files. Some of these changes might take longer than 36 hours. For this reason, there is a mechanism to store these changes during the test phase and restore them instantly during the cutover phase.

This document explains step-by-step how to perform the upgrade from ALEPH version 16.02 to version 17.01.

Upgrade Conventions and Requirements

Conventions

In Upgrade Express documentation, we refer to the "source" and "target" versions or environments.

The **source version** is the current production platform and ALEPH version environment that the customer would like to upgrade. In this document the source version is ALEPH 16.02.

The **target version** is the new production platform and ALEPH version environment that is the target of the upgrade process. In this document, the target version is ALEPH 17.01.

Upgrade Options

1. Upgrade Express can be run "inside the database". In this case version 17 software is installed in parallel with version 16 software and is connected to the same database as version 16. Only one version can be active at a time (version 16 before the upgrade and version 17 after the upgrade). The advantage of this method is less downtime. The disadvantage is that in case of failure, restore from backup will be required. The risk of such a "failure" is not any higher than during usual work of the system, as the upgrade kit runs SQL scripts inside the database in order to upgrade the data. A full cold backup before the upgrade is absolutely necessary.

2. Upgrade Express can also be run in a separately installed database. In this case ver 17 software is installed together with a new database. Advantage of this method: in case of failure you can continue working on the old database while the problem is being fixed. Disadvantage: more downtime, more disc space required.

If you've chosen option 2, before you can run Upgrade Express, you will need to bring your data to the newly installed instance. There are 2 methods proposed for doing it:

  a. Import and Export method. This method requires having additional space on the server, and is only suitable for "smaller" sites, as the export/import routines can take a long time for larger systems. The Installation team will be able to create the same number of user tablespaces on target as you have on the source installation. The Import and Export can be performed using Oracle export and import utility:

      Download upx_exp_imp_utility.tar from the ftp server:

       ftp ftp.exlibirs.co.il
       ver17ux/password to be obtained from a local representative

      The upx_exp_imp_utility.tar is composed of 4 files

      oracle_exp_aleph_libs
      oracle_exp_aleph_libs_a

      oracle_imp_aleph_libs
      oracle_imp_aleph_libs_a

      1. copy the oracle_exp_aleph_libs and oracle_exp_aleph_libs_a to the source aleph_proc
      2. copy the oracle_imp_aleph_libs and oracle_imp_aleph_libs_a to the target aleph_proc
      3. on the source
         csh -f $aleph_proc/oracle_exp_aleph_libs
         The job will echo the libraries which will be exported, the log file and ask for confirmation. The export will run in the background

4. after the export finishes you must move or copy or ftp or ln -s the files to the same place in the target $TMPDIR 5 on the target

  csh -f $aleph_proc/oracle_imp_aleph_libs |& tee $TMPDIR/oracle_imp_aleph_libs.log

You can then view the logfile in $TMPDIR. Note that z98 in the bibliographic and the authority libraries make take particularly long to import. You can follow the progress of the import of a particular table by regularly logging into SQL and running couns against the table concerned. If the count is growing, it means that the import is proceeding as expected.

  b. Cloning database method. Please contact your local representative for complete instructions. This requires at least as much free disk space as the source database.

Prerequisites

1. Install the Target Version (17.01)
2. Download the Upgrade Express Package

Installing the Target Version (17.01)

The first step in the actual upgrade process is the preparation of the target environment. The target version (17.01) can be installed on the same server as 16.02 or on a separate server. If installed on the same server, the target version (17.01) can share a database with ver 16, or have a separate database. In any case, the hardware and operating system requirements for 17.01 and the size of the customer's database must be taken into account. Please check the hardware requirements documentation before arranging installation.

The installation is scheduled with the local Ex Libris representative, and the completion of the "Request for Installation" form may be required.

Downloading the Upgrade Express Package

Download Upgrade Express package in the target environment. The Upgrade Express Package is available for download from the ftp server and is a tar file which can be extracted anywhere but we recommend to extract it under $ALEPH_MOUNT.

ftp ftp.exlibris.co.il

ver17ux/password to be obtained from a local representative

bin

get <filename>

e.g. get upgrade_express_1602_1701.tar.1.10.gz

To open the kit:

>>gzip –d <filename>

e.g. gzip –d upgrade_express_1602_1701.tar.1.10.gz

>>tar –xf <filename>

Upgrade Process

This section is divided into two subsections. The first subsection presents a generalized list of the actions to be performed during the upgrade process. The second subsection returns to this list and details the conditions and procedures for each step in the upgrade process.

Basic Outline of the Upgrade Process

The following is a basic list of the steps to be performed during the upgrade process NB make sure you are logged into the target version when running any Upgrade Express step:

1. Copy the source **utree** (16.02) into the target installation (17.01) (see *Copying the Utree* on page 8).

2. If the upgrade is performed in a separate database, copy the complete source ALEPH database (16.02) into the target installation (17.01) (see *Copying the Database* on page 8).

3. Make sure that archiving is switched off in both source and target versions (util O 7 in both v. 16 and v. 17)

4. Perform the minimal required manual changes to selected configuration files such as updates to `$alephe_root/aleph_start` in order to allow basic functions such as login (see *Making the Minimal Required Manual Changes* on page 8).

5. Copy the apache directory from the **utree** provided with the target version (17.01) into the new **utree** brought from the source version (see *Copying Apache* on page 9).

6. Perform import of the demo libraries in the target version (see *Importing Demo Libraries* on page 10).

7. Configure the Upgrade Express setup (see *Configuring Upgrade Express* on page 10).

8. Backup Oracle tables to be upgraded (optional) (see *Backing Up Oracle Tables* on page 10).

9. Run the Upgrade Express (see *Running the Upgrade Express* on page 11).

10. Check upgrade results (logs, functionality, and so on) (see *Analyzing the Upgrade Express Results* on page 12).

11. Implement changes described in the implementation notes (see *Implementation Notes* on page 12).

12. Perform standard final actions (such as clear_vir, repackaging of the cataloging tables, and so on) (see *Packaging and Final Steps* on page 13).

Upgrade Process in Detail

The following is the detailed description of the steps to be performed during the Upgrade Express process. Please note that in case of running the upgrade more than once, it is recommended to repeat the whole series of steps for each run.

Copying the Utree

Make sure that ALEPH is down on both source and target versions (run $alephe_root/aleph_shutdown). Copy the complete **utree** from the source (16.02) to the target (17.01). Use the tar command in order to copy the tree. This is necessary in order to copy the symbolic links correctly. It is recommended to clean the copied **utree** as much as possible. Remove all unnecessary files from all directories (such as scratch, files, print, and so on). The **utree** delivered with the target installation (17.01) by Ex Libris must be backed up, as some steps in the upgrade process make use of files in this tree.

Example of the commands that can be used:

cd /exlibris/aleph/
Backup u17_1 tree: mv u17_1/ u17_1.orig/
Tar u16_1 tree: tar -cf u16_1.tar u16_1
Move u16_1 to u17_1: mv u16_1/ u17_1/
Restore u16_1 tree: tar –xvf u16_1.tar

Copying the Database

Copy the full ALEPH database from source to target. This step should be ignored if the upgrade is performed "inside a database". The amount of time it takes to copy a database depends on the size of the database, system resources and the method chosen. A database with 10,000,000 bibliographic records takes about 5 hours to copy using the cloning database method.[contact local rep for complete instructions on cloning a database]. The export may take 2 hours and the import 20 hours on a site with 1 million bibliographic records. [For instructions on "How to export/import Oracle data see the "Upgrade Options" section].

During the upgrade of Oracle tables, the processed table is renamed for backup in case the upgrade fails. This means that in the target installation, you need additional space in each table space. The additional space should be at least twice the size of the largest table/index in the tablespace.

Making the Minimal Required Manual Changes

To enable login into the new version, make the following minimal changes:

1. Add the following line to the ./alephe/tab/tab_version file:

```
17.01  ALEPH Y 0
```

2. Add a valid ./alephe/tab/license file. Either copy a license if it has been supplied by Ex Libris or copy the license file from the $aleph_dev/alephe/tab directory. (If you have already copied the u tree, this will be located in your u17_1.orig directory).

3. Add a valid alephe/tab/license.www_x file if the x-server is required. Either copy a license if it has been supplied by Ex Libris or copy the license file from the $aleph_dev/alephe/tab directory. (If you have already copied the u tree, this will be located in your u17_1.orig directory).

4. Synchronize the new /alephe/aleph_start, aleph_startup and aleph_shutdown files with the relevant file supplied with the target installation under the **utree**. It is recommended to merge the aleph_start by taking the file supplied with the installation and adding local specifications into the new file. Login into version 17 and test aleph_startup and aleph_shutdown. Make sure that the correct version has been shutdown and startup. (If you are upgrading from 16 to 18 don't do this step until the 17 to 18 upgrade.)

5.

After you have made these changes, you will be able to log in to the version and to work with SQL *Plus.

Please refer to the *Manual Changes: Store and Restore* section on page 13. This section describes how to save modified files so that potential subsequent upgrade runs do not override them.

Copying Apache

In order to have a functional Apache configuration, you must copy the apache directory supplied by the version (under the backed up **utree, u17_1.orig**) into the new **utree** to be upgraded. Copy the directory by using the tar command in order to preserve the links and to avoid copying them as directories. Note that the links might point to locations that do not exist and it is still necessary to recreate the links.

tar Apache directory in the original u17_1/ tree:
cd /exlibris/aleph/u17_7.orig/alephe
tar -cf apache.tar apache
Move apache.tar into the "real" ver 17 tree:
mv apache.tar /exlibris/aleph/u17_1/alephe
cd /exlibris/aleph/u17_1/alephe
Untar: tar -xvf apache.tar

After copying the apache directory, perform the following:

1. If necessary, add any local configuration (such as local files, and so on) that was part of your previous apache.

2. Clean up the logs directory (./alephe/apache/logs) by removing all files (rm *).

3. Recreate the apache links by running the following command from the **apm** directory (get to this directory by typing apm at the command line):

```
create_htdocs_links
```

You will be required to restart the Apache server (util w/3/6).

(If you are upgrading from 16 to 18 wait until the 17 to 18 upgrade before running this step?)

Importing Demo Libraries

At this point the data of the demo libraries is also in the format of the source database (16.02). In order to replace this data with the demo data supplied with the installation, run the following command from the **ap** directory (get to this directory by typing ap at the command line):

```
imp_demo_libraries
```

```
This process make take up to a couple of hours, depending on the
capacity of your server.
Beware that if your usr_library or pw_library in aleph_start is set
to usr00, this process will override your user password data (z66 and
z67).
```

Configuring Upgrade Express

In order to perform the initial upgrade configuration, perform the following:

1. Go to the **upgrade_express_1602_1701** directory by entering the following:

   ```
   cd $ALEPH_MOUNT/upgrade_express_1602_1701
   ```

2. Invoke the Upgrade Express menu by entering the following command:

   ```
   source ./upgrade_util
   ```

3. From the Upgrade Express menu, select the first option: Define Upgrade Parameters.

4. Select the View Current Parameters option. This option will display a default set of parameters:

   - The following libraries are not in demo and they will be processed

   - Main BIB library

   - Course libraries (must be populated even if you don't use a course library)

   - Notification email address

   The defaults can be overridden if necessary. Selecting the Update parameters option from the Define Upgrade Parameters submenu does this. This option enables you to update the parameters above.

Backing Up Oracle Tables

You can back up the tables that are modified by the Upgrade Express. Note that this step is optional but recommended. Carry out the backup as follows:

1. Invoke the Upgrade Express menu as explained in the previous step.

2. Select the **Backup/Restore Oracle Tables to be Upgraded** option. The following choices are displayed:

   - Backup

   - Restore

   - List all backuped tables

The **Backup** option enables you to create a backup of the tables that are modified by the upgrade. When using this option, you are prompted to supply a directory for export and the table name of the tables to be backed up (ALL can be used for all tables involved in the upgrade). When you specify the directory make sure it is not one already in use.

If you supplied an e-mail address at the parameters stage, a notification e-mail message will be sent when the export is complete.

The following Oracle tables are modified by the 16.02 – 17.01 Upgrade Express process: Z104, Z16, Z30, Z30H, Z303, Z304, Z305, Z308, Z325, Z36, Z36H, Z37, Z37H, Z38, Z67, Z68, Z70 and Z52

The following new Oracle tables are added by the 16.02 – 17.01 Upgrade Express process: Z00E, Z07A, Z129, Z309, Z324, Z327, Z328, Z69, Z91 and Z403.

The **List all backuped tables** option enables you to view the list of tables that you have backed up.

The **Restore** option enables you to restore the backed up Oracle tables in the event that something has gone wrong with the upgrade. As with the **Backup** option, you are prompted to supply the directory of the exported files to be imported and the table name of the tables to be imported (ALL can be used for all tables involved in the upgrade).

If you supplied an e-mail address at the parameters stage, a notification e-mail message will be sent when the import is complete.

Running the Upgrade Express

All previous steps can be considered prerequisites for the Upgrade Express process. In order to run the upgrade, invoke once again the Upgrade menu and select the **Run Upgrade Express** option.

The upgrade program runs a number of steps that include conversion of data and setup. Every step is listed in a log file that explains what action is included in the step and whether it was successful or not.

The **Run Upgrade Express** option enables you to run a single step or the complete set of steps. It is highly recommended to run the complete set of steps. The option to run a single step should only be used if you find that a particular step failed and you do not want to restart the complete process.

The upgrade program may take several hours to run, depending on the size of the database and server resources.

Note that while the upgrade is running all menu options are disabled in order to avoid conflicts that might arise by, for example, changing the initial parameters.

If you supplied an e-mail address at the parameters stage, a notification e-mail message will be sent as soon as the upgrade is complete.

Analyzing the Upgrade Express Results

As explained above, the Upgrade Express includes a number of steps/tasks. After running the upgrade, a log file for each of the steps is created under the **log** directory located under the **upgrade_express_1602_1701** directory. Each task performed has been assigned a numeric identifier; the log for the file appears in the directory under the following convention: [step_number].log (for example, 198.log). These files contain a description of the task performed and success/failure messages.

The first task to be performed as soon as the Upgrade Express concludes is the analysis of these log files in order to detect any problems as soon as possible. In the event of failed steps, you might need to perform the tasks manually or re-run the step after making necessary changes. If you need to re-run a step make sure you back up the main log file first.

You can view the log file by invoking the Upgrade menu and selecting the **View Logfiles** option. This option enables you to view the main log or the log for a single step. In addition, you can view the logs of previous Upgrade runs.

Following is an example of a manual change that might be required:

Our sample-step adds a new form to the form_eng directory of the administrative library. While running the upgrade, the program has detected that the file already exists in the target **utree**. In this case, the log of the step will state that the file was not added since it found that the file already exists.

The user's task is to analyze if he should keep the original file or manually copy the new file.

Following is another example:

Our sample-step adds changes to a form in the form_eng directory. Based on the Upgrade Express methodology, changes to the form_eng files are only added if the target file is identical to the demo file of the source installation (16.02). If the files differ, the log of the step contains a message informing the user that the change could not be made and that the changes should be implemented manually.

Please refer to the *Manual Changes: Store and Restore* section on page 13. This section describes how to save modified files so that potential subsequent upgrade runs do not override them.

Implementation Notes

In addition to failed steps, which need to be handled manually, there are a number of configuration files (mainly HTML files) that require manual upgrade. A list of these changes is included in the **Upgrade Express 16.02 – 17.01 - Implementation Notes** documents distributed with the Upgrade Express programs.

Please refer to the *Manual Changes: Store and Restore* section on page 13. This section describes how to save modified files so that potential subsequent upgrade runs do not override them.

Packaging and Final Steps

If you are going on to do a further upgrade e.g. 17-18, then wait until the end of the v18 upgrade before performing these steps.

The final step of the upgrade is to perform a set of standard actions. These actions can be performed by invoking the Upgrade menu and selecting the **Perform Post Upgrade Express Actions** option. Following actions should be performed:

- Remove UTF files
- Clear VIR sessions
- Repack Cataloging tables (UTIL M/7)
- Synchronize Headers (UTIL H/2-3)
- Create Print Templates Package for the PC Client (UTIL I/6)

After these steps are completed, re-start servers and UE processes. The system should be operational.

Some Guidelines on the Upgrade Methodology

Manual Changes: Store and Restore

All configuration files that are modified during the Upgrade Express process must be saved to the side – otherwise subsequent upgrade runs will override them. Each time a file is modified use util d:

D. Online Store/Restore Utilities

0. Exit Procedure
1. Store Library Data
2. Store Library Configuration
3. Restore Library Data
4. Restore Library Configuration

All saved files can be restored after a subsequent run of the Upgrade Express.

What is Not Included in Upgrade Express

Upgrade Express upgrades the customer's ALEPH system with the aim of maintaining current setup and functionality. The upgrade process is not the time for implementing new functionality as this will interfere with the upgrade itself. New

functionality should be implemented after the switch to production with the new version. Upgrade Express does not include automatic setup for implementing new and enhanced functionality, unless it must be implemented to maintain current functionality. Upgrade Express from 16.02 to 17.01 will not even add new setup files or changes in setup files unless they are mandatory.