



# **Upgrade Express 23.01 to 24.01 User Guide**

## 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. This information is provided AS IS. Unless otherwise agreed, Ex Libris shall not be liable for any damages for use of this document, including, without limitation, consequential, punitive, indirect or direct damages.

Any references in this document to third-party material (including third-party Web sites) are provided for convenience only and do not in any manner serve as an endorsement of that third-party material or those Web sites. The third-party materials are not part of the materials for this Ex Libris product and Ex Libris has no liability for such materials.

## TRADEMARKS

"Ex Libris," the Ex Libris bridge, Primo, Aleph, Alephino, Voyager, SFX, MetaLib, Verde, DigiTool, Preservation, URM, Voyager, ENCompass, Endeavor eZConnect, WebVoyage, Citation Server, LinkFinder and LinkFinder Plus, and other marks are trademarks or registered trademarks of Ex Libris Ltd. or its affiliates.

The absence of a name or logo in this list does not constitute a waiver of any and all intellectual property rights that Ex Libris Ltd. or its affiliates have established in any of its products, features, or service names or logos.

Trademarks of various third-party products, which may include the following, are referenced in this documentation. Ex Libris does not claim any rights in these trademarks. Use of these marks does not imply endorsement by Ex Libris of these third-party products, or endorsement by these third parties of Ex Libris products.

Oracle is a registered trademark of Oracle Corporation.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Ltd.

Microsoft, the Microsoft logo, MS, MS-DOS, Microsoft PowerPoint, Visual Basic, Visual C++, Win32,

Microsoft Windows, the Windows logo, Microsoft Notepad, Microsoft Windows Explorer, Microsoft Internet Explorer, and Windows NT are registered trademarks and ActiveX is a trademark of the Microsoft Corporation in the United States and/or other countries.

Unicode and the Unicode logo are registered trademarks of Unicode, Inc.

Google is a registered trademark of Google, Inc.

Copyright Ex Libris Limited, 2023. All rights reserved.

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

## Table of Contents

<b>1. INTRODUCTION .....</b>	<b>4</b>
<b>2. UPGRADE CONVENTIONS AND REQUIREMENTS .....</b>	<b>5</b>
Conventions .....	5
Prerequisites.....	5
<b>3. USING THE UPGRADE EXPRESS KIT .....</b>	<b>6</b>
<b>4. GUIDELINES FOR THE UPGRADE METHODOLOGY .....</b>	<b>15</b>
4.1 Manual Changes: Store and Restore.....	15
4.3 Not Included in Upgrade Express.....	16
4.4 Cleanup Utilities .....	17
<b>5. INSTALLING THE NEW GUI.....</b>	<b>18</b>
<b>6. STEP DESCRIPTIONS .....</b>	<b>18</b>
6.1 Introduction.....	18
6.2 Changes in Configuration Files .....	18
<b>7. UPGRADE EXPRESS CHECKLIST AFTER UPGRADE .....</b>	<b>28</b>
<b>8. APPENDIX A – COPYING APACHE.....</b>	<b>33</b>
<b>9. APPENDIX B – JBOSS TO TOMCAT UPGRADE.....</b>	<b>34</b>
New Location for Java Classes.....	34

# 1. Introduction

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

The Aleph upgrade process is the process of replacing a version of Aleph with 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. A copy of the 23.01 **utree** and database is added to the new version, Aleph 24.01. This creates an Aleph 24.01 software environment with 23.01 configuration and data.

It is also possible to upgrade from lower versions, such as v.16 to v.24, without installing v.16, v.17, v.18, v.19, v.20, v.21, v.22 and v.23. In this case, a copy of the 16.01 **utree** and database is added to the new version, Aleph 24.01. Minor changes need to be made to allow logging on. After these changes, the upgrade process is executed eight times (v.16 to v.17, v.17 to v.18, v.18 to v.19, v.19 to v.20, v.20 to v.21, v.21 to v.22, v.22 to v.23 and v.23 to v.24). It transforms the configuration and data into the Aleph 24.01 format. Such a big jump is only possible if you are not using the Aleph ILL module. If you are using the Aleph ILL module, you need to install and work with v.18 in order to migrate the old ILL data and to setup the new ILL.

Note that when upgrading from a version lower than 24, all the documentation of the previous versions of the Upgrade Express (such as 18 to 19, 19 to 20, 20 to 21, 21 to 22, and 22 to 23) must be reviewed. One of the main issues to consider is the upgrade of the Oracle version.

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 performed automatically, some configuration has to be performed manually, for example, the implementation of changes to Web files. Some of these changes may 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 23.01 to Aleph 24.01 using Upgrade Express.

## 2. Upgrade Conventions and Requirements

### Conventions

The Upgrade Express documentation refers to the “source” and “target” versions or environments.

The **source version** is the current Aleph version environment that the customer would like to upgrade. In this document, the source version refers to Aleph 23.01.

The **target version** is the new production Aleph version environment that is the target of the upgrade process. In this document, the target version refers to Aleph 24.01

### Prerequisites

#### Installing the Target Version (24.01)

The first step in the actual upgrade process is the preparation of the target environment. The hardware and operating system requirements for version 24.01 and the size of the customer’s database must be taken into account. The installation is scheduled with the local Ex Libris representative. You may be required to complete the *Request for Installation* form.

**Note:** In order to estimate the size of exported data, you can use the "expdp" utility with the "ESTIMATE\_ONLY=y" flag, as follows:

```
expdp $ALEPH_ADMIN ESTIMATE_ONLY=y full=y
```

The result is in the following format:

```
Total estimation using BLOCKS method: 4.403 GB
```

```
Job "ALEPH_ADMIN"."SYS_EXPORT_FULL_01" successfully completed at  
16:50:41
```

### Downloading the Upgrade Express Package

The Upgrade Express Package should be downloaded either to the source or target environment, depending on the Upgrade Express method that you choose to use. The standard Upgrade Express method involves downloading the Upgrade Express Kit to the source environment and following the instructions below.

Download the Upgrade Express package to the target (24.01) environment.

The Upgrade Express package is available for download from the support MFT server. Contact Ex Libris Support to obtain a one-time password for downloading the package from the MFT server.

The package is a .tar file that can be extracted anywhere, but it is recommended that you extract it to \$ALEPH\_MOUNT, as follows:

For example:

```
get upgrade_express_2301_2401.tar.1.01.gz
```

To open the kit:

```
>>gzip -d <filename>
```

For example:

```
gzip -d upgrade_express_2301_2401.tar.1.01.gz
>>tar -xf <filename>
```

### 3. Using the Upgrade Express Kit

The following is a detailed description of the steps that can be performed using the Upgrade Express Kit. If you are running two upgrade packages (such as when upgrading from 21 to 24), it is recommended that you run the standard final actions (such as UTIL I/6 and others described under the [Packaging and Final Steps](#) section) only after running the last upgrade.

To access the Upgrade Express menu, upgrade\_express\_2301\_2401 directory by typing the following:

```
cd $ALEPH_MOUNT/upgrade_express_2301_2401
```

Open the Upgrade Express menu by entering the following command:

```
source ./upgrade_util.
```

The Upgrade Express utility identifies your current Aleph version. When running the Upgrade Express from the source version, the following menu is displayed:

```
+-----+
| Upgrade Express Utility 23.01 -> 24.01 Version 1.01 Source |
+-----+

0. Exit
-- 1. Define upgrade parameters
-- 2. Export customer data
-- 3. Transfer customer data to the target location
4. View logfiles
```

If the Upgrade Express utility identifies your current Aleph version as the Target (such as in option 2 described in section [2 Upgrade Conventions and Requirements](#)) the following menu is displayed:

```
+-----+
| Upgrade Express Utility 23.01 -> 24.01 Version 1.01 Target |
+-----+

0. Exit
1. View/Update upgrade parameters
2. Check environment
3. Run upgrade express
4. Perform Post Upgrade Express Actions
5. View logfiles

6. Backup/Restore Oracle Tables to be Upgraded
7. Store/Restore configuration files
9. Transfer data / configuration files between servers
```

## 1. Define upgrade parameters

Define your upgrade parameters using the first option displayed above. In this step, you define the libraries that are upgraded and the e-mail address to which you want to send the results of the export, import, and Upgrade Express utility. Note that defining or updating the parameters without using the utility may cause problems when running the Upgrade Express. The proper way to define or update the upgrade parameters is by using the utility.

### Note:

Although the illsv is defined as a demo library it is used by ILL customers and it is upgraded. Therefore if your library uses illsv you must include it in the definitions so it can be exported.

After performing each step, the (--) symbols change to (++) symbols.

## 2. Pack library data and configuration tables

```
+-----+
| Upgrade Express Utility 23.01 -> 24.01 Version 1.01 Source |
+-----+

  0. Exit
-- 1. Define upgrade parameters
-- 2. Export customer data
-- 3. Transfer customer data to the target location
  4. View logfiles
```

Choose one of the following options to export the data and u-tree.

```
+-----+
| Export of Oracle data and u-tree from source machine. |
+-----+

  0. Exit
  1. Export Oracle data
  2. Pack the u-tree
  3. Export Oracle data & pack the u-tree in one step
```

Each of the options suits a different workflow according to which the library can decide to work (depending on the available servers and the amount of data). Before packing the u-tree, reduce the size of the scratch and print directories as much as possible.

There is an option to exclude tables Z97 and Z98 (since their export/import takes a lot of time). If you want to exclude them, run manage\_01 after performing the upgrade. The data is downloaded into the data/ directory of the Upgrade Express.

The data pump option enables you to extract and upload the data faster.

```
Make sure you have cleaned the u-tree as much as possible:
scratch, print, files directories as well as Apache logs.
```

```
Oracle Data Pump extracts and uploads the data faster the regular
exp/imp
```

```
Note: It works only on Oracle 10 and up.
```

```
Before running it for the first time, do the following:
```

```
s+ ALEPH_ADMIN
```

```
grant JAVA_ADMIN to aleph_admin WITH ADMIN OPTION;
```

You may also choose to pack the library data and configuration tables manually. In this case make sure that Aleph is down on the target version (run `$alephe_root/aleph_shutdown`). Copy the complete utree from the source (23.01) to the target (24.01) using the tar command (in order to copy the symbolic links correctly).

Note that the utree delivered with the target installation (24.01) by Ex Libris must be backed up. Some steps in the upgrade process make use of files in this tree.

It is recommended that you clean the copied utree as much as possible. Remove all unnecessary files from all directories (such as scratch, files, print, and so on).

The following are example of the commands that can be used:

```
cd /exlibris/aleph/
Backup u24_1 tree: mv u24_1/ u24_1.orig/
Tar u23_1 tree: tar -cf u23_1.tar u23_1/
Move u23_1 to u24_1: mv u23_1/ u24_1/
Restore u23_1 tree: tar -xvf u23_1.tar
```

### 3. Transfer data and perform manual changes

```
+-----+
| Upgrade Express Utility 23.01 -> 24.01 Version 1.01 Source |
+-----+

0. Exit
-- 1. Define upgrade parameters
-- 2. Export customer data
-- 3. Transfer customer data to the target location
4. View logfiles
```

If you are performing this step from the target server/slot the following [screen](#) is displayed:

```
+-----+
| Upgrade Express Utility 23.01 -> 24.01 Version 1.01 Target |
+-----+

0. Exit
1. View/Update upgrade parameters
2. Check environment
3. Run upgrade express
4. Perform Post Upgrade Express Actions
5. View logfiles

6. Backup/Restore Oracle Tables to be Upgraded
7. Store/Restore configuration files
9. Transfer data / configuration files between servers
```



Note that option 9 is used in cases where you want to move data/configuration from between 2 servers which are in the same version.

This utility also updates `tab_version`, `license` and `license.www_x` files, and restores `aleph_startup` and `aleph_shutdown` files supplied with the version.

Manually synchronize the new `/alephe/aleph_start` file with the `aleph_start` file supplied with the target installation under the **utree** supplied with the version. It is recommended to merge the files by taking the file supplied with the installation and adding local specifications into the new file.

When the data is uploaded from the target version, the u-tree is unpacked and the following changes are automatically performed:

Apache is restored from `alephe.orig`. (for detailed explanation regarding this function, see [Appendix A – Copying Apache](#).)

- The links under `apache/htdocs` are rebuilt (using the `create_htdocs_links` routine)
- License files and `tab_system_stature` are restored from `alephe.orig/tab`.
- The `aleph_startup` and `aleph_shutdown` are restored from `alephe.orig`.
- The `alephe/tab/tab_version` is updated with the new version number.

All the steps above are performed automatically when using the Upgrade Express utility. For more information, see [Appendix A – Copying Apache](#).

Shutdown Aleph on the target Aleph 24. Install Upgrade Express kit on the target. (There is no need to configure the kit). Transfer the data from the source to the target location (v.24) using Upgrade Express Util/3 on the source server (the third option in the first screen above) and review the log files (Upgrade Express Util/4, the fourth option in the first screen above):

```
Transfer customer data
=====

Do you want to continue Y/N: Y
Enter transfer method to sftp/ftp:

Enter target server I.P./hostname: demo-server

Enter root directory on demo-server
    [/exlibris/aleph/upgrade_express_2301_2401]:

Enter user name on demo-server [aleph]:

sftp'ng aleph@ demo-server
Enter to continue
Connecting to demo-server...
aleph@ demo-server 's password:
sftp> sftp> sftp> Remote working directory:
/exlibris/aleph/upgrade_express_2301_2401/data
```

Note that when both versions are on the same machine, there is no need for FTP. You just log on to the server in the target version and run the Upgrade Express from the same location.

**Note:**

The location of exported data is under the "data" directory of the Upgrade Express kit.

#### 4. Import Demo Libraries

This section is only relevant for the cloned database.

**Important Note:**

Every installation must have one of the following sets of demo libraries: mab, uni, or usm. This demo data is used as the base library for the util g/3 (Create Initial Library Parameters) and by Ex Libris staff and your staff to reproduce problems occurring with your libraries' tables and data.

At this point, the data of the demo libraries is also in the format of the source database (23.01). To replace this data with the demo data supplied by the installation, run the following commands:

1. Download the aleph24 dump file from the ftp:

The demo libraries are automatically downloaded as part of the Aleph 24 Installation Kit (AIK 24) from Ex Libris MFT servers, using sftp, to the standard destination, usually /exlibris/ftp\_from\_exlibris.

```
cd /exlibris/ftp_from_exlibris

cat aleph24_demo_libs.dmp.gz * > aleph24_demo_libs.dmp.gz
gzip -d aleph24_demo_libs.dmp.gz
chmod 777 aleph24_demo_libs.dmp
rm aleph24_demo_libs.dmp.gz??
```

2. Recreate Oracle schemas:

```
dlib usm01
ap
foreach lib ( $DEMO_LIBS )
    create_ora_user_b $lib
end
```

3. Define dumpdir directory:

```
sqlplus ALEPH_DBA/ALEPH_DBA as sysdba
create or replace directory dumpdir AS '/exlibris/ftp_from_exlibris';
GRANT READ,WRITE ON DIRECTORY dumpdir TO ALEPH_ADMIN;
exit
```

4. Execute the import:

```
set DLIBS=`echo $DEMO_LIBS | sed 's/ /,/g'`
set DD=`date +%y%m%d`
```

```
impdp $ALEPH_ADMIN schemas=$DLIBS DIRECTORY=dumpdir
DUMPFIL=aleph24_demo_libs.dmp LOGFILE=aleph24_libs.$DD.log \
>& /tmp/impdp_demo_libs.log &
```

Note that this process can take around 1-2 hours, depending on the capacity of the server.

#### Notes

- If your `usr_library` and/or `pw_library` in `aleph_start` is set to `USR00`, this process will delete your patron and/or password data. (It is our strong recommendation that the `USR00` not be used as your `usr_library` or `pw_library`.)

## 5 Clean Virtual Library (VIR01) - clear\_vir01 "cold"

VIR01 is normally a DEMO library. If the Demo libraries importing is being done on a server where customer data has already been loaded and is in use, then Aleph (or at least the `pc_server` and `www_server`) should be shut down while the importing is running, in order to avoid conflicts in the use of the `vir01` tables.

A new database table - "Z314" - has been added to VIR01 in Version 24.01.

(It is defined in `file_list.VIR`).

This new table must be created (empty) in VIR01 of the target (24.01).

This is done by: `csch -f $aleph_proc/clear_vir01 vir01 online`

or by util X/ 8 with parameter "cold"

## 6. View/Update Upgrade Parameters and Check Environment

Before performing "6. Backup/Restore Oracle Tables to be Upgraded" and "3. Run upgrade express" confirm that all of your libraries are specified so that Upgrade Express runs only on the libraries specified there. To specify the libraries, select "5. View/Update upgrade parameters".

## 7. Back Up the Oracle Tables

```
+-----+
| Upgrade Express Utility 23.01 -> 24.01 Version 1.01 Target |
+-----+

0. Exit
1. View/Update upgrade parameters
2. Check environment
3. Run upgrade express
4. Perform Post Upgrade Express Actions
5. View logfiles

6. Backup/Restore Oracle Tables to be Upgraded
7. Store/Restore configuration files
9. Transfer data / configuration files between servers
```

You can back up the tables that are modified by the Upgrade Express. Although this step is optional, it is recommended.

To backup the tables:

1. Open the Upgrade Express menu as explained in the previous step.
2. Select **Backup/Restore Oracle Tables to be Upgraded**. The following choices are displayed:
  - Backup
  - Restore
  - List all backed up tables

The **Backup** option enables you to create a backup of the tables that are modified by the upgrade. If you select 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.)

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

The following Oracle tables are modified by the 23.01 – 24.01 Upgrade Express process: Z303, Z304, Z308, Z353, Z700 and Z900.

The **List all backed up 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 if something went 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 parameter stage, a notification e-mail message is sent when the restoration has been completed.

## 8. Upgrade Express Localization

**Note:** For most customers this section is not relevant.

The Upgrade Express supports multi-language environments. The localization process includes the translation of the Upgrade Express to other languages other than English. This translation is usually performed by distributors and offices and does not require customer involvement.

The Upgrade Express can be translated into one or more different languages during the same upgrade. While running, each language-related step is performed for each defined language. The languages are defined during the configuration stage. Each language has a separate directory under `upgrade_express_2301_2401/conf` in which all language-related steps are configured. New languages must be added to this location by copying the original `eng` directory and translating all files within this directory.

Non-language-related configuration files (there are usually only a few of these) are located in the `upgrade_express_2301_2401/conf/general` directory.

## 9. Run the Upgrade Express

All previous steps can be considered prerequisites for the Upgrade Express process. To run the upgrade, open the Upgrade Express menu and select the **Run Upgrade Express** option.

```
Running the Upgrade Express
-----

Upgrade Express runs in the background. At the end of the process you
will get an email to the address specified in the parameters.

0. Exit
1. Run all steps
2. Run a specific step
3. Run all steps >999 (Oracle data only)

Please select [0]:
```

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 the step was successful.

The **Run Upgrade Express** option enables you to run a single step, the complete set of steps, or the Oracle data steps only. It is highly recommended that you run the complete set of steps the first time around. 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 rerun the entire process. The option to run all steps > 999 should be used to upgrade the Oracle data but not the configuration tables.

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 may arise as a result of actions performed during this time (for example, if you change the initial parameters while the upgrade is running).

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

## 10. Analyzing the Upgrade Express Results

As explained above, the Upgrade Express includes a number of steps and tasks. After running the upgrade, a log file for each of the steps is created in the log directory located under `upgrade_express_2301_2401`. Each task performed has been assigned a numeric identifier and the log file appears in the directory according to the following convention:

```
[step_number].log (for example, 198.log)
```

The log files contain a description of each task performed as well as information pertaining to the success or failure of each task.

As soon as the Upgrade Express concludes, analyze these log files in order to detect whether problems occurred. In the event of failed steps, you may need to perform certain tasks manually, or rerun steps after making certain changes.

You can view the log file by selecting **View Log Files** from the Upgrade Express menu. This option enables you to view the main log file or the log file for a single step. In addition, you can view the log files of previous upgrade runs.

```
View Log Files
-----

0. Exit
1. View Main Log
2. View a Specific Step Log
3. View Older Logs

Please select [0]:
```

Examples:

1. A step adds a new form to the `form_eng` directory of the administrative library. While running the upgrade, the program detects that the file already exists in the target **utree**. In this case, the step's log file states that the file was not added. The user must then decide whether to keep the original file, or manually copy the new file.
2. A step adds changes to a form in the `form_eng` directory. Based on the Upgrade Express methodology, changes to the `form_eng` files are added only if the target file is identical to the source installation's (23.01) demo file. If the files differ, the step's log file contains a message informing the user that the changes could not be made and that they should be implemented manually.

[Section 6 Step Descriptions](#) contains a detailed description of the configuration changes performed by the upgrade kit. If the the error message "The file can not be upgraded automatically. Merge manually!" is displayed, see section [6 Step Descriptions](#) to see what change was introduced to the file in v.21.

For information on how to save modified files so that potential subsequent upgrade runs do not override these files, see section [4](#)

[4.1 Manual Changes: Store and Restore](#) section.

## 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 upgrading. A list of the changes required to these files is included in section [0 6.2 Changes in Configuration Files](#).

## Packaging and Final Steps

The final step of the upgrade includes a set of standard actions. These actions can be performed by selecting **Perform Post Upgrade Express Actions** from the Upgrade Express menu.

```
Perform Post Upgrade Express Actions
```

```
-----  
  
0. Exit  
1. Remove UTF files (util-x-7)  
2. Clear VIR sessions (util-x-8)  
3. Repack Cataloging tables (util-m-7)  
4. Create Print Templates Package for the PC Client (util-i-6)  
  
6. Synchronize pcb files  
7. Synchronize error files
```

The following actions should be performed:

- Remove the UTF files.
- Clear the VIR sessions.
- Repack the cataloging tables (UTIL M/7).
- Create a Print Templates Package for the PC Client (UTIL I/6).
- Run Util H/2 and Util H/3 to synchronize headers. (Although this step is not in the menu above, it is required).

After these steps are completed, restart the servers and run the UE processes. The system is then operational.

**Note:** When upgrading two separate packages, restart the server and run Upgrade Express processes only after the second package upgrade. For example, if you are upgrading from 22.01 to 23.01 and then from 23.01 to 24.01, the **Post Upgrade Express Actions** should be run only after the second upgrade has been performed.

## 4. Guidelines for the Upgrade Methodology

### 4.1 Manual Changes: Store and Restore

```
+-----+
| Upgrade Express Utility 23.01 -> 24.01 Version 1.01 Target |
+-----+

0. Exit
1. View/Update upgrade parameters
2. Check environment
3. Run upgrade express
4. Perform Post Upgrade Express Actions
5. View logfiles

6. Backup/Restore Oracle Tables to be Upgraded
7. Store/Restore configuration files
9. Transfer data / configuration files between servers
```

All configuration files that are modified during the Upgrade Express process must be saved in a separate location; otherwise, subsequent upgrade runs override them. The Upgrade Express package has a mechanism for storing files that have been modified during the process.

```
Store/View/Restore Configuration Files
-----

Storage directory is:
/exlibris/aleph/kit.24.01/upgrade_express_2301_2401/./upgrade_express_2101_2301.store

0. Exit
1. Add files to store
2. View list of stored files
3. Create a report of differences
4. Restore files from
/exlibris/aleph/kit.24.01/upgrade_express_2301_2401/./upgrade_express_2101_2301.store
```

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

### 4.3 Not Included in Upgrade Express

The Upgrade Express upgrades the customer's Aleph system with the aim of maintaining the current setup and functionality. Do not implement new functionality, as this will interfere with the upgrade itself. New functionality should be implemented after the switch to production with the new version. The Upgrade Express does not include automatic setup for implementing new and enhanced functionality, unless it must be implemented to maintain the current functionality. The Upgrade Express from 23.01 to 24.01 does not add new setup files or make changes in setup files unless these changes are mandatory. For instruction regarding changes in setup files that are not mandatory, refer to the Release Notes document.



## 4.4 Cleanup Utilities

Due to architectural changes in Aleph, and for clarity and upgrade purposes, version 17.01 `pc_b_[lng]` and `error_[lng]` directories are located in the `aleph` directory, which means that they are part of the installation of the product. However, the customer can still configure these files within the `alephe` directory, as in previous versions of Aleph. To avoid duplication and errors, you should clean up these files in `alephe`. The Upgrade Express includes two utilities that can help you clean up the `pc_b_[lng]` and `error_[lng]` directories.

### `pc_b_[lng]` File Cleanup Utility

This utility is available from the main menu of the Upgrade Express:

#### 6. Organize pcb files

It compares the `./aleph/pc_b_[lng]` directory to the customer's `./alephe/pc_b_[lng]` directory. A report, which includes the detailed current situation of the directory and advice regarding the action that should be taken, is provided.

For example, the following may be provided:

- Similar files in the two compared directories can be deleted from `./alephe/pc_b_[lng]`.
- Files with differences may be customized and should remain under `alephe`.
- Files that appear only under `./alephe/pc_b_[lng]` and not under `./aleph/pc_b_[lng]` are not in use anymore by Aleph and can be deleted.

The **Delete Obsolete Files** menu option deletes all files with a file name identical to the updated file from `./alephe/pc_b_[lng]`.

### `error_[lng]` File Cleanup Utility

This utility is available from the main menu of the Upgrade Express:

#### 7. Organize error files

It compares the `./aleph/error_[lng]` and `./alephe/error_[lng]` directories. A report, which includes the detailed current situation of the directory and advice regarding the action that should be taken, is provided.

For example, the following may be provided:

- Similar files in the two compared directories can be deleted from `./alephe/error_[lng]`.
- If files differ in one or more error message lines, the files should be merged.

The **Delete Similar Files** menu option deletes all files that have the same name as the new file. The merge actions should be performed manually by the customer.

## 5. Installing the New GUI

The 24.01 GUI installation program includes the option of upgrading your current GUI. This means that setup parameters (\*.ini files) from your older version clients are copied to the new 24.01 clients and synchronized, minimizing the need to customize the new GUI. To apply the upgrade program, install the new GUI in the same directory in which the older version GUI is installed. The old GUI version is automatically saved under the same directory (for example Acq.sav). Install the new GUI in the original or copied directory.

At the end of the installation of the 24.01 GUI, the setup program checks the previous GUI version and prompts you to confirm whether you want to run the upgrade program. The upgrade program synchronizes the 24.01 installation with special changes (for example, formats and names changes) that were done in previous versions.

## 6. Step Descriptions

### 6.1 Introduction

This includes steps 1 – 999. The changes are performed in the relevant configuration tables.

The description relates to the following libraries:

adm_libs	Administrative libraries
bib_libs	Bibliographic libraries
hol_libs	Holding libraries
union_catalog_libs	Union Catalog
crs_libs	Course reading libraries
ill_libs	ILL libraries
pw_libs	password libraries

Changes in Oracle tables relate to steps that are greater than 999. The changes are performed in the ORACLE database.

### 6.2 Changes in Configuration Files

\*\*\*\*\*

Step 1:

Change the following in \$alephe\_error\_lng

1. In the pc\_acq\_c0532 file Change the line that start with "0032",  
From:  
0032 0012 L Invalid material type.

To:  
0032 0011 L Invalid material type.

\*\*\*\*\*  
\*\*\*\*\*

Step 2:

Update \$alephe\_tab/user\_function.LNG with the following changes:

1) Add the line:  
ITEM-UPDATE-S      L Item Updates Services      ITEM-13      L ITEM-13 Create Items for  
Detached Holding Records

\*\*\*\*\*  
\*\*\*\*\*

Step 3:

Add the following new files to alephe/pc\_b\_<lng>:

1) The following form file is new under ./alephe/pc\_b\_lng :  
p-item-13.xml

\*\*\*\*\*  
\*\*\*\*\*

Step 4:

Changed XSL / TRN in BIB libraries:

1) The following files are new in ADM libraries:  
./[adm\_libs]/form\_{lng}/detached-hol-item-report.xml  
./[adm\_libs]/form\_{lng}/detached-hol-item-report.trn

\*\*\*\*\*  
\*\*\*\*\*

Step 5:

Update files (batch services form) under alephe/pc\_b\_<lng>:

1) In the file ./alephe/pc\_b\_<lng>/p-publish-04.xml:

a) After the lines  
    <source>PUBLISH-SET</source>  
    <empty\_line>ALL</empty\_line>  
    </expand\_drop\_down>  
</control>

    Add the lines:

<control>  
    <drop\_down>  
        <argname>F05</argname>  
        <label>\* Extract Records For</label>  
        <default>DOC</default>  
        <cbitem>  
            <value>DOC</value>  
            <display>Range of documents</display>  
        </cbitem>

```

    <cbitem>
      <value>FILE</value>
      <display>Input file</display>
    </cbitem>
  </drop_down>
</control>

```

b) After the lines

```

  <label>To Document Number (up to 9 digits)</label>
  <size>9</size>
  <default>999999999</default>
</edit_num>
</control>

```

Add the lines:

```

<control>
  <edit>
    <argname>F06</argname>
    <label>Input File Name</label>
    <size>100</size>
  </edit>
</control>

```

c) Add at the end of the file the lines:

```

<!-- Param: P-PUBLISH-TYPE           X(10).   F05 -->
<!-- Param: P-INPUT-FILE-NAME       X(100).  F06 -->

```

2) In the file ./alephe/pc\_b\_<lng>/p-publish-05.xml:

a) After the lines

```

  <source>PUBLISH-SET-FROM-DB</source>
  <empty_line>NO_USE</empty_line>
  </expand_drop_down>
</control>

```

Add the lines:

```

<control>
  <drop_down>
    <argname>F05</argname>
    <label>* Extract Records For</label>
    <default>DOC</default>
    <cbitem>
      <value>DOC</value>
      <display>Range of documents</display>
    </cbitem>
    <cbitem>
      <value>FILE</value>
      <display>Input file</display>
    </cbitem>
  </drop_down>
</control>

```

b) After the lines:

```

  <label>To Document Number (up to 9 digits)</label>
  <size>9</size>
  <default>999999999</default>
</edit_num>
</control>

```

Add the lines:

```

<control>

```

```

<edit>
  <argname>F06</argname>
  <label>Input File Name</label>
  <size>100</size>
</edit>
</control>

```

c) Add at the end of the file the lines:

```

<!-- Param: P-PUBLISH-TYPE          X(10).    F05 -->
<!-- Param: P-INPUT-FILE-NAME      X(100).   F06 -->

```

```

*****
*****

```

Step 6:

Update ./[adm\_library]/tab/pc\_tab\_exp\_field.LNG with the following changes:

1) Add the line:

```
SERIAL-PTRN-REG-MODE  L Combined          c
```

2) Add the lines:

```
SERIAL-PTRN-REG-TYPE  L 1st enum level    e1
SERIAL-PTRN-REG-TYPE  L 2nd enum level    e2
```

```

*****
*****

```

Step 7:

Changed XSL / TRN in BIB libraries:

1) In acq-edi-load-log-checkin.xsl, remove the "section-05" parts.

(a) remove the following lines:

```

<xsl:for-each select="//section-05">
  <xsl:call-template name="section-05"/>
</xsl:for-each>

```

(b) remove the following lines:

```

<xsl:template name="section-05">
  <!--Free Section Code -->
  <xsl:call-template name="table-open"/>
  <xsl:call-template name="display-gen">
    <xsl:with-param name="label" select="'Total Lending Records Archived:'"/>
    <xsl:with-param name="value" select="./rec-counter"/>
  </xsl:call-template>
  <xsl:call-template name="table-close"/>
</xsl:template>

```

```

*****
*****

```

Step 8:

Update files (batch services form) under alephe/pc\_b\_<lng>:

1) In the file ./alephe/pc\_b\_<lng>/p-ill-76.xml, change the following line,  
from:

<label>Letter Format</label>

to:

<label>Report Format</label>

\*\*\*\*\*

\*\*\*\*\*

Step 1:

Change the following in \$alephe\_error\_lng

1. In the pc\_acq\_c0532 file Change the line that start with "0032",

From:

0032 0012 L Invalid material type.

To:

0032 0011 L Invalid material type.

\*\*\*\*\*

\*\*\*\*\*

Step 2 :

Update \$alephe\_tab/user\_function.LNG with the following changes:

1) Add the line:

ITEM-UPDATE-S      L Item Updates Services      ITEM-13      L ITEM-13 Create Items for  
Detached Holding Records

\*\*\*\*\*

\*\*\*\*\*

Step 3:

Add the following new files to alephe/pc\_b\_<lng>:

1) The following form file is new under ./alephe/pc\_b\_lng :  
p-item-13.xml

\*\*\*\*\*

\*\*\*\*\*

Step 4:

Changed XSL / TRN in BIB libraries:

1) The following files are new in ADM libraries:

./[adm\_libs]/form\_{lng}/detached-hol-item-report.xsl

./[adm\_libs]/form\_{lng}/detached-hol-item-report.trn

\*\*\*\*\*

\*\*\*\*\*

Step 5:

Update files (batch services form) under alephe/pc\_b\_<lng>:

1) In the file ./alephe/pc\_b\_<lng>/p-publish-04.xml:

a) After the lines

```
<source>PUBLISH-SET</source>
<empty_line>ALL</empty_line>
</expand_drop_down>
</control>
```

Add the lines:

```
<control>
  <drop_down>
    <argname>F05</argname>
    <label>* Extract Records For</label>
    <default>DOC</default>
    <cbitem>
      <value>DOC</value>
      <display>Range of documents</display>
    </cbitem>
    <cbitem>
      <value>FILE</value>
      <display>Input file</display>
    </cbitem>
  </drop_down>
</control>
```

b) After the lines

```
<label>To Document Number (up to 9 digits)</label>
<size>9</size>
<default>999999999</default>
</edit_num>
</control>
```

Add the lines:

```
<control>
  <edit>
    <argname>F06</argname>
    <label>Input File Name</label>
    <size>100</size>
  </edit>
</control>
```

c) Add at the end of the file the lines:

```
<!-- Param: P-PUBLISH-TYPE          X(10).   F05 -->
<!-- Param: P-INPUT-FILE-NAME      X(100).  F06 -->
```

2) In the file ./alephe/pc\_b\_<lng>/p-publish-05.xml:

a) After the lines

```
<source>PUBLISH-SET-FROM-DB</source>
<empty_line>NO_USE</empty_line>
</expand_drop_down>
</control>
```

Add the lines:

```
<control>
  <drop_down>
    <argname>F05</argname>
    <label>* Extract Records For</label>
    <default>DOC</default>
```

```

    <cbitem>
      <value>DOC</value>
      <display>Range of documents</display>
    </cbitem>
    <cbitem>
      <value>FILE</value>
      <display>Input file</display>
    </cbitem>
  </drop_down>
</control>

```

b) After the lines:

```

  <label>To Document Number (up to 9 digits)</label>
  <size>9</size>
  <default>999999999</default>
</edit_num>
</control>

```

Add the lines:

```

<control>
  <edit>
    <argname>F06</argname>
    <label>Input File Name</label>
    <size>100</size>
  </edit>
</control>

```

c) Add at the end of the file the lines:

```

<!-- Param: P-PUBLISH-TYPE          X(10).   F05 -->
<!-- Param: P-INPUT-FILE-NAME      X(100).  F06 -->

```

```

*****
*****

```

Step 6:

Update ./[adm\_library]/tab/pc\_tab\_exp\_field.LNG with the following changes:

1) Add the line:

```

SERIAL-PTRN-REG-MODE   L Combined                               c

```

2) Add the lines:

```

SERIAL-PTRN-REG-TYPE   L 1st enum level                       e1
SERIAL-PTRN-REG-TYPE   L 2nd enum level                       e2

```

```

*****
*****

```

Step 7:

Changed XSL / TRN in BIB libraries:

1) In acq-edi-load-log-checkin.xml, remove the "section-05" parts.

(a) remove the following lines:

```

<xsl:for-each select="//section-05">
  <xsl:call-template name="section-05"/>
</xsl:for-each>

```



(b) remove the following lines:

```
<xsl:template name="section-05">
  <!--Free Section Code -->
  <xsl:call-template name="table-open"/>
  <xsl:call-template name="display-gen">
    <xsl:with-param name="label" select="'Total Lending Records Archived:'"/>
    <xsl:with-param name="value" select="./rec-counter"/>
  </xsl:call-template>
  <xsl:call-template name="table-close"/>
</xsl:template>
```

\*\*\*\*\*  
\*\*\*\*\*

Step 8:

Update files (batch services form) under alephe/pc\_b\_<lng>:

1) In the file ./alephe/pc\_b\_<lng>/p-ill-76.xml, change the following line,  
from:

```
<label>Letter Format</label>
```

to:

```
<label>Report Format</label>
```

\*\*\*\*\*  
\*\*\*\*\*

Step 1:

Change the following in \$alephe\_error\_lng

1. In the pc\_acq\_c0532 file Change the line that start with "0032",

From:

0032 0012 L Invalid material type.

To:

0032 0011 L Invalid material type.

\*\*\*\*\*  
\*\*\*\*\*

Step 2:

Update \$alephe\_tab/user\_function.LNG with the following changes:

1) Add the line:

ITEM-UPDATE-S      L Item Updates Services      ITEM-13      L ITEM-13 Create Items for  
Detached Holding Records

\*\*\*\*\*  
\*\*\*\*\*

Step 3:

Add the following new files to alephe/pc\_b\_<lng>:

1) The following form file is new under ./alephe/pc\_b\_lng :

p-item-13.xml

\*\*\*\*\*  
\*\*\*\*\*

Step 4:

Changed XSL / TRN in BIB libraries:

- 1) The following files are new in ADM libraries:  
./[adm\_libs]/form\_{lng}/detached-hol-item-report.xsl  
./[adm\_libs]/form\_{lng}/detached-hol-item-report.trn

\*\*\*\*\*  
\*\*\*\*\*

Step 5:

Update files (batch services form) under alephe/pc\_b\_{lng}:

- 1) In the file ./alephe/pc\_b\_{lng}/p-publish-04.xml:

a) After the lines

```
<source>PUBLISH-SET</source>  
<empty_line>ALL</empty_line>  
</expand_drop_down>  
</control>
```

Add the lines:

```
<control>  
  <drop_down>  
    <argname>F05</argname>  
    <label>* Extract Records For</label>  
    <default>DOC</default>  
    <cbitem>  
      <value>DOC</value>  
      <display>Range of documents</display>  
    </cbitem>  
    <cbitem>  
      <value>FILE</value>  
      <display>Input file</display>  
    </cbitem>  
  </drop_down>  
</control>
```

b) After the lines

```
<label>To Document Number (up to 9 digits)</label>  
<size>9</size>  
<default>99999999</default>  
</edit_num>  
</control>
```

Add the lines:

```
<control>  
  <edit>  
    <argname>F06</argname>  
    <label>Input File Name</label>  
    <size>100</size>  
  </edit>  
</control>
```

c) Add at the end of the file the lines:  
 <!-- Param: P-PUBLISH-TYPE X(10). F05 -->  
 <!-- Param: P-INPUT-FILE-NAME X(100). F06 -->

2) In the file ./alephe/pc\_b\_<lng>/p-publish-05.xml:

a) After the lines  
 <source>PUBLISH-SET-FROM-DB</source>  
 <empty\_line>NO\_USE</empty\_line>  
 </expand\_drop\_down>  
 </control>

Add the lines:

```
<control>
  <drop_down>
    <argname>F05</argname>
    <label>* Extract Records For</label>
    <default>DOC</default>
    <cbitem>
      <value>DOC</value>
      <display>Range of documents</display>
    </cbitem>
    <cbitem>
      <value>FILE</value>
      <display>Input file</display>
    </cbitem>
  </drop_down>
</control>
```

b) After the lines:

```
<label>To Document Number (up to 9 digits)</label>
<size>9</size>
<default>999999999</default>
</edit_num>
</control>
```

Add the lines:

```
<control>
  <edit>
    <argname>F06</argname>
    <label>Input File Name</label>
    <size>100</size>
  </edit>
</control>
```

c) Add at the end of the file the lines:  
 <!-- Param: P-PUBLISH-TYPE X(10). F05 -->  
 <!-- Param: P-INPUT-FILE-NAME X(100). F06 -->

\*\*\*\*\*  
 \*\*\*\*\*

Step 6:

Update ./[adm\_library]/tab/pc\_tab\_exp\_field.LNG with the following changes:

1) Add the line:  
 SERIAL-PTRN-REG-MODE L Combined c

2) Add the lines:

SERIAL-PTRN-REG-TYPE	L 1st enum level	e1
SERIAL-PTRN-REG-TYPE	L 2nd enum level	e2

\*\*\*\*\*  
 \*\*\*\*\*

Step 7:

Changed XSL / TRN in BIB libraries:

- 1) In acq-edi-load-log-checkin.xsl, remove the "section-05" parts.
  - (a) remove the following lines:

```
<xsl:for-each select="//section-05">
  <xsl:call-template name="section-05"/>
</xsl:for-each>
```

- (b) remove the following lines:

```
<xsl:template name="section-05">
  <!--Free Section Code -->
  <xsl:call-template name="table-open"/>
  <xsl:call-template name="display-gen">
    <xsl:with-param name="label" select="'Total Lending Records Archived:'"/>
    <xsl:with-param name="value" select="./rec-counter"/>
  </xsl:call-template>
  <xsl:call-template name="table-close"/>
</xsl:template>
```

\*\*\*\*\*  
 \*\*\*\*\*

Step 8:

Update files (batch services form) under alephe/pc\_b\_<lng>:

- 1) In the file ./alephe/pc\_b\_<lng>/p-ill-76.xml, change the following line, from:

```
<label>Letter Format</label>
```

to:

```
<label>Report Format</label>
```

\*\*\*\*\*

## 7. Upgrade Express Checklist after Upgrade

This checklist is intended to help you test the conversion of data and configuration tables by checking basic functionality in the various modules of the system. After the upgrade process has been completed, (including the manual parts of the upgrade), go through the following list and test the specified functions.

<b>General</b>	<b>Confirmed by</b>	<b>Date</b>
Test that jobs run correctly in job_list.		

<b>Cataloging and Items</b>	<b>Confirmed by</b>	<b>Date</b>
Create new record using File → New Record / create a new record using a pre-defined template		
Search for created record in Web and confirm that it is retrieved		
Scan the indexes listed in "find" in the cataloging module and make sure they work		
Search for each of the accesses listed in "scan" in the cataloging module and make sure it works		
Load record from server → modify it → send to server		
Ensure character display of foreign characters and diacritics is correct		
Check authority control functionality		
Check that library can download records from its usual supplier. This will test Z39.50 configuration		
Push the above created record to items		
Add new item.		
In item form, confirm that the drop-down boxes appear correctly		

<b>Acquisitions and Serials</b>	<b>Confirmed by</b>	<b>Date</b>
Open module and do a search and scan		
Search Budget file		
Search Vendors file		
Search Budget transactions per active budget		
Confirm that users can view and modify budgets		
Test EDI functionality		
Push record to Serials module from other modules		
Compare subscription information		
Compare publication schedule forms		
Do "arrive" on an expected issue and confirm that it opens		
View serial items in the Web OPAC		

<b>ILL</b>	<b>Confirmed by</b>	<b>Date</b>
Check borrowing/lending requests summary tree – compare requests from previous ILL to new ILL		

(numbers, status, bib info, etc).		
Check ILL-Admin-Partner list, potential suppliers, customer ID, APDU Exceptions (number of records, record info, etc)		
Check borrowing requests basic flow for returnable material (request creation, send to supplier, register arrival, return to supplier)		
Check lending requests basic flow for returnable material (request creation, supply material, check-in material)		
Check OPAC-Library card-ILL info (compare previous ILL to new ILL)		

<b>Circulation</b>	<b>Confirmed by</b>	<b>Date</b>
Search for borrowers		
On lower version server create a Loan report (cir-04) or Overdue and lost billing notices (cir-50) report. Compare these results with searches in the Circulation module. A recent report may already exist		
Open new borrower and inform library of number		
Loan the new item you entered to the new borrower		
Test renewal, renew all, return, whatever fine regime is used		
Test overdue/courtesy notice generation		
Compare cash transactions between test server data and production server data of the previous version.		
Activate and check self-check server transactions		
Run p_cir_60 and p_cir_65 in order to check advanced booking function. (Only on test server)		

<b>Web OPAC</b>	<b>Confirmed by</b>	<b>Date</b>
Sign in as the user you entered above		
Confirm that your loan appears correctly as “user” in Web		
Confirm that your information appears in “user” correctly		
Search for the record you created, then click “items” and confirm that you see that it is on loan		
Try keyword and browse searches, in full and in logical bases		
Use basket and e-shelf, e-mail results to yourself		
Check customization generally, menus and navigation links		
Connect to a remote link		
Verify that Course Reading works		
Ensure character display of foreign characters and diacritics is correct		
Verify that related searches (tab_service) are working		
Verify that ‘limit’ functionality works OK		
Check the sorting of records		
Check analytical records for connections between main analytical records and locations		
Test interaction between the Web OPAC and staff client; create an ILL request on the Web OPAC and check the data fed into the GUI, place a hold request, check display of Advanced Booking slots.		

<b>Course Reading</b>	<b>Confirmed by</b>	<b>Date</b>
Open the module and see that the data appears: courses, reading lists, items.		
Make sure that course that had a proxy have the proxy's course number on the course note.		
Compare the existence of courses in the old CR and new CR- use course-06.		
Test Course Management basic flow (new course, update, delete, etc.)		
Compare reading lists in old CR and new CR – use course-01.		
Add document to reading list (by search, catalog new, load file), make sure that it appears in the list.		
Test reading list management basic flow (suppress, remove, edit notes, etc.).		
Connect to OPAC, Course Reading base. Compare Search results to results in old CR: the results should be the same BUT not duplicated.		

<b>Search in GUI Modules</b>	<b>Confirmed by</b>	<b>Date</b>
Search for records via various fields		
Browse for records via various fields		
Add records to navigation map and send to different modules		
Ensure character display of foreign characters and diacritics is correct: Font.ini must have appropriate settings		
Verify that related searches (tab_service) are working		

<b>Z39.50</b>	<b>Confirmed by</b>	<b>Date</b>
Check all Z39.50 targets for standard searches		
Do tables cater correctly for translation of diacritics and foreign characters?		

<b>MetaLib/SFX</b>	<b>Confirmed by</b>	<b>Date</b>
Need to check that ALEPH resource in MetaLib works. Need to check interactions between ALEPH/SFX/MetaLib. Do deep links work correctly? ALEPH as an SFX source. ALEPH as an SFX target.		



	<b>Confirmed by</b>	<b>Date</b>
<b>X-server</b>		
If library is using X-server, verify that links via X-server are OK		
When all is complete		
Cleanup / Delete data added for testing		
Ask installation team to switch Web port to new version (production server only)		
Verify that archiving is on when going live with version 6 (production server only)		
Verify that library knows how to do backup and that it is working properly.		
Send official letter to library that version 22 data has been uploaded and indexed		

Customers that used Publishing in the former version should check the following:

	<b>Confirmed by</b>	<b>Date</b>
<b>Publishing</b>		
Configure new set in xxx01/tab/tab_publish		
Run p_publish_04 with the new set.		
Run p_publish_05 on part of the new set		
Run p_publish_06 on new set		

## 8. Appendix A – Copying Apache

The utility copies the Apache directory supplied with the version (under the backed up **utree**) to the new **utree** that is to be upgraded.

You may choose to copy apache manually:

1. Copy the directory by using the `tar` command to preserve the links. Avoid copying the links as directories. Note that the links might point to locations that do not exist. In this case, it is necessary to recreate the links.

To locate the `tar` Apache directory in the original `u22_1/ tree`:

```
cd /exlibris/aleph/u23_1.orig/alephe
tar -cf apache.tar apache/
```

2. Move `apache.tar` to the "real" ver 23 tree:

```
mv apache.tar /exlibris/aleph/u23_1/alephe
cd /exlibris/aleph/u23_1/alephe
```

3. "Untar" using the following command:

```
tar -xvf apache.tar
```

4. After copying the Apache directory, perform the following:
  - a) If necessary, add any local configuration (such as local files) that was a part of your previous Apache configuration.
  - b) Clean up the `logs` directory (`./alephe/apache/logs`) by removing all files (`rm *`).
  - c) Recreate the Apache links by running the following command from the `apm` directory (get to this directory by typing `apm` at the command line):
    - d) `create_htdocs_links`
  - e) Restart the Apache server.

## 9. Appendix B – JBoss to Tomcat upgrade

Aleph 22 and later versions are delivered with Apache Tomcat - the Servlet and Java Server Pages container, as a replacement to JBoss.

The configuration files under "`$alephe_root/jboss_conf`" are still relevant and used for Tomcat - in the same way as it was in JBoss.

For backing up and restoring the configuration files, see the instructions below.

### New Location for Java Classes

Aleph 24 is installed with new Java WAR files:

```
rest-dlf.war, aleph-api.war, core-sms-proxy.war.
```

These files are located in the new Aleph 23 "jdeploy" directory:

```
$aleph_dev/ng/aleph/home/system/thirdparty/tomcat/aleph-webapps
```

For manual deployment - you should copy the files:

```
rest-dlf.war, aleph-api.war, core-sms-proxy.war
```

- from the v. 23 "jdeploy" directory:

```
$aleph_dev/ng/aleph/home/system/thirdparty/openserver/server/default/  
deploy
```

- into V. 24 "jdeploy" directory:

```
$aleph_dev/ng/aleph/home/system/thirdparty/tomcat/aleph-webapps.
```