ALEPH VERSION 18.01

How to Open a New ALEPH Site

Please note that this document should *only* be used in conjunction with Service Pack 183 and later.

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

1 Overview

This document describes how to open a new ALEPH site with the **open_new_aleph_site (ONAS)** program.

This program can be used to open a complete new ALEPH environment for a new installation. Please note that if you need to open additional libraries in an existing (not new) site, use the **open_new_library** program and not the open_new_aleph_site program.

Opening a new ALEPH site means creating a new working environment for the ALEPH system – opening a suite of libraries and updating as much as possible the setup files. The types, codes and sizes of libraries to be opened are defined by the customer in a setup file.

The libraries of a new ALEPH site are created based on template libraries. The template used can be one of the standard demo libraries (for example, USM, UNI, MAB) or the "vanilla setup" (currently the ATA template libraries are available for the MARC 21 format). Section 7 Using the Vanilla Setup as the Template Libraries for ONAS on page 17 explains how to use the vanilla setup with the ONAS program.

For every library, the ONAS program will create a Unix directory structure with library setup files and an Oracle schema. In addition the program updates some setup files in the alephe directory.

1.1 Supported ALEPH Site Models

The **open_new_aleph_site** program supports the following ALEPH site models:

Single-ADM – a site with a single BIBliographic library with a single linked ADMinistrative library. Optionally the site can also include one or more AUThority libraries, and the ADM can be linked to a single Holdings, ILL and Course Reserves (CR) library.

Multi-ADM – a site with a single BIBliographic library linked to multiple ADMinistrative libraries. Optionally the site can also include one or more AUThority libraries. Every ADM can be linked to an ILL and Course Reserves (CR) library. In terms of the HOLdings library there are two options – all ADM libraries can be linked to the same HOLdings library or every ADM has its own HOLdings library.

BIB-only – a site with a single BIBliographic library with no linked ADMinistrative libraries. Optionally the site can also include one or more AUThority libraries and the BIBliographic library can be linked to a HOLdings library.

Note concerning the USR and PW libraries:

In principle, the USR and PW libraries can each be defined as a separate library (together or each on its own). In some cases, the USR and PW libraries can be defined to be part of another library. In a multi-ADM site it is mandatory to have a separate USR/PW library/ies. In a single-ADM site the ADM library is usually defined to also serve as the USR and PW libraries (another common variation is to define the ADM library also as the USR library and have a separate PW library). In a BIB-only site, the BIB library can also serve as the USR and PW libraries.

2 Before activating the programs

The open_new_aleph_site program has two parameters:

- The three-character code of the template libraries.

  The template libraries either be one of the DEMO library code (for example, USM or UNI) or the "vanilla setup" library code (that is,. ATA)

  In this document a three-character code is indicated by <TEM>.

  A full library code uses the three-character code and two digits. The digits are indicated in this document by a 'XX' – for example, <TEM>XX or ATAXX.

- Input_file name.

  The input file should be in $alephe_scratch

2.1 Input file parameters:

Every library that should be opened for the new site should be defined in the input file, a line per library. For every library the following parameters are defined in five columns:

- Library type

- Library code

- Library size

- Library lib_path

- Library x-path

These parameters are explained below:

LIBRARY TYPE (mandatory parameter)

　　The first parameter defines the library type**.** The valid library types are:

- o BIB – BIBliographic library

- o AUT – AUThority library

- o ADM – ADMinistrative library

- o HOL – Holdings library

- o COURSE – Course Reserve library,

- o ILL – ILL library

- o USR – USR library

- o PW – PW library

　　(Note that the Z105 library is always the ADM or USR library)

　　Conditions:

- o BIB: (mandatory). Only one occurrence of BIB is allowed.

- AUT: (optional). Multiple occurrences are allowed

- ADM: (optional). Multiple occurrences are allowed

- COURSE: (optional). Every ADM can have a linked COURSE. The code of the Course must match those of the ADM both in the three-character code and final digits (refer to the following section on library codes).

  Example:

ADM	COURSE
ABC50	ABC30
XYZ5**0**	XYZ3**0**
XYZ5**1**	XYZ3**1**

- ILL: (optional). Every ADM can have a linked ILL. The code of the ILL must match those of the ADM both in the three-character code and final digits (refer to the following section on library codes).

  Example:

ADM	ILL
ABC50	ABC40
XYZ5**0**	XYZ**40**
XYZ5**1**	XYZ**41**

- HOL: (optional). If there is a single ADM only one HOL is allowed. If there are multiple ADM libraries then two situations are allowed:

  All ADMs share a single HOL library

  Every ADM can have its own HOL Library. The code of the HOL must match the code of the ADM in the three-character code and final digits (refer to the following section on library codes).

  Example:

ADM	HOL
ABC50	ABC60
XYZ5**0**	XYZ6**0**
XYZ5**1**	XYZ6**1**

- PW: (optional). Only one occurrence of PW is allowed. PW serves for pw_library. If PW is not defined, it will be defaulted to the USR library (if the USR is defined) or the ADM library if the USR library is not defined. If there is no ADM library (in a BIB-only site), the BIB library will be defaulted as the PW library.

o USR: (optional/mandatory). Only one occurrence of USR is allowed. In a multi-ADM site it is mandatory to define a USR library; it is optional in a single-ADM site, in which case the ADM library is defaulted to the USR library; it is also optional in a BIB-only site in which case the BIB library is defaulted as the USR library.

LIBRARY CODE (mandatory parameter):

A five-character code should be used: three characters (xxx) followed by two digits.

Conditions:

The following digits must be used per library types:

o xxx00 – for USR and PW library type. This is true only for standalone USR/PW libraries. Note that if the USR/PW is also the ADM library (which can happen if there is a single ADM library), then the ADM digits will be used.

o xxx01-09 - for BIB library type.

o xxx10-19 – for AUT library type.

o Xxx40-49 – for ILL library type

o xxx30-39 – for COURSE library type.

o xxx50-59 – for ADM library type.

o xxx60-69 – for HOL library type.

As noted above, the codes of the HOL, ILL and COURSE libraries must match (in the three-character code and final digit) that of its linked ADM library.

LIBRARY SIZE (mandatory parameter):

The library size must be specified for every library code. The program accepts the following types:

VSMALL, SMALL, MEDIUM, LARGE, HUGE, DEMO

Choose the appropriate library size based on the expected number of bibliographic records that will be loaded to ALEPH:

o VSMALL - Up to 10,000 docs

o SMALL - Up to 100,000 docs

o MEDIUM - Up to 1,000,000 docs

o LARGE - Up to 5,000,000 docs

o HUGE - Larger than 5,000,000 docs

o DEMO – used by demo libraries only

The program uses the library size defined to create the "file_list_size" parameter (in $data_root/prof_library of the library). This parameter is used to create the appropriate default file_list.

Note that all tablespaces required for the library type/sizes must be defined BEFORE the ONAS script is run.

To check what the default tablespaces are, and if they exist, use UTIL/O/17/3. To create the required tablespaces use UTIL/O/17/1.

LIB_PATH (optional parameter):

LIB_PATH is a logical path to the root directories of the library

You can specify a path for ($data_root) directories of the libraries. If not specified, the program assumes that LIB_PATH should be defined as $alephe_dev.

X-PATH (optional parameter):

You can specify an x-path. If not specified, the default is " " and LIB PATH will be used by the script.

An ALEPH library environment includes a Unix directories tree of setup files, configuration tables directories, and data files directories.

It is recommended that large libraries separate the setup and configuration directories and the data directories. By default, all the directories are under the same root ($data_root) under the "u-tree".

If the directories are separated, it is recommended that setup and configuration directories be on the same device as ALEPH, and that the large data directories, $data_files and $data_scratch, be on another disk and not under $data_root. In such cases, $data_scratch and $data_files directories will be physically placed under the x_path (the name of the path will be supplied by the user), but a symbolic link will be program-generated so that they are logically placed under $data_root of the new library.

2.2 Input File Format

The input file has five columns, which are comma-delimited, for example:

```
!LIBRARY_TYPE    LIBRARY_CODE    LIBRARY_SIZE    LIB_PATH        X_PATH

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

BIB,            TET01,          VSMALL,

AUT,            TET10,          VSMALL,

AUT,            TET12,          VSMALL,

HOL,            TET60,          VSMALL,

ADM,            TET50,          VSMALL,
```

Spaces are not mandatory; the following file is also valid:

```
BIB,TET01,VSMALL,

AUT,TET10,VSMALL,

AUT,TET12,VSMALL,

HOL,TET60,VSMALL,

ADM,TET50,VSMALL,
```

If an X-Path has no lib_path, a comma must be added to indicate the lib_path is missing. For example:

```
BIB,TET01,VSMALL,,X-PATH
AUT,TET10,VSMALL,,X-PATH
AUT,TET12,VSMALL,,X_PATH
HOL,TET60,VSMALL,,X_PATH
ADM,TET50,VSMALL,,X_PATH
```

The file can be created in Excel and exported in comma-delimited format and then ftped to the server.

2.2.1 Examples of input_file:

Example 1

LIBRARY TYPE	LIBRARY CODE	LIBRARY SIZE	LIB-PATH	X-PATH
BIB	ABC01	MEDIUM		
AUT	ABC10	MEDIUM		
AUT	ABC12	SMALL		
ADM	ABC50	MEDIUM		
HOL	ABC60	MEDIUM		
COURSE	ABC30	SMALL		
ILL	ABC40	SMALL		

* Single ADM setup, where ADM library is USR and PW library, there is no X-PATH and LIB-PATH is $alephe_dev

Example 2

LIBRARY TYPE	LIBRARY CODE	LIBRARY SIZE	LIB-PATH	X-PATH
BIB	ABC01	MEDIUM	/exlibris2/aleph/u18_1	
AUT	ABC10	MEDIUM	/exlibris2/aleph/u18_1	
AUT	ABC12	SMALL	/exlibris2/aleph/u18_1	
ADM	ABC50	MEDIUM	/exlibris2/aleph/u18_1	
ADM	XYZ50	MEDIUM	/exlibris2/aleph/u18_1	
HOL	ABC60	MEDIUM	/exlibris2/aleph/u18_1	
COURSE	ABC30	SMALL	/exlibris2/aleph/u18_1	

LIBRARY TYPE	LIBRARY CODE	LIBRARY SIZE	LIB-PATH	X-PATH
COURSE	XYZ30	SMALL	/exlibris2/aleph/u18_1	
ILL	ABC40	SMALL	/exlibris2/aleph/u18_1	
ILL	XYZ40	SMALL	/exlibris2/aleph/u18_1	
USR	ABC00	MEDIUM	/exlibris2/aleph/u18_1	
PW	PWL00	SMALL	/exlibris2/aleph/u18_1	

* Multi-ADM setup, where there is a single HOL for all; every ADM has its own COURSE and ILL; there is no X-PATH and LIB-PATH is /exlibris2/aleph/u18_1;

Example 3

LIBRARY TYPE	LIBRARY CODE	LIBRARY SIZE	LIB-PATH	X-PATH
BIB	ABC01	MEDIUM		/exlibris3/aleph/u18_1
AUT	ABC10	MEDIUM		/exlibris3/aleph/u18_1
AUT	ABC12	SMALL		/exlibris3/aleph/u18_1
ADM	ABC50	MEDIUM		/exlibris3/aleph/u18_1
ADM	XYZ50	MEDIUM		/exlibris3/aleph/u18_1
HOL	ABC60	MEDIUM		/exlibris3/aleph/u18_1
HOL	XYZ60	MEDIUM		/exlibris3/aleph/u18_1
COURSE	ABC30	MEDIUM		/exlibris3/aleph/u18_1
COURSE	XYZ30	SMALL		/exlibris3/aleph/u18_1
ILL	ABC40	SMALL		/exlibris3/aleph/u18_1
ILL	XYZ40	SMALL		/exlibris3/aleph/u18_1
USR	ABC00	MEDIUM		/exlibris3/aleph/u18_1

* Multiple ADM setup, where there is a separate HOL and separate COURSE and ILL for different ADMs;

LIB-PATH is $alephe_root; $data_file and $data_scratch directories will be located under /exlibris3/aleph/u18_1;

USR library is a PW library

<Please note that a setup where PW library is defined and USR library is not defined is invalid>

3 Activation

3.1 Syntax

The syntax of the 'open_new_aleph_site' command is as follows:

csh -f $aleph_proc/ open_new_aleph_site

There are two activation modes for opening the library:

3.1.1 Batch Mode

Batch mode – provide parameter values. If you enter all parameters, the system will not prompt you to enter values.

csh -f $aleph_proc/ open_new_aleph_site <input_file_name><template code>

Examples:

csh -f $aleph_proc/ open_new_aleph_site param_files USM

csh -f $aleph_proc/ open_new_aleph_site param_files ATA

param_files – input_file located in $alephe_scratch

USM/ATA – template libraries to be used

3.1.2 Interactive Mode

If you do not enter any of above parameters, the program will work in interactive mode and you will be prompted for each of the missing parameters.

Examples:

csh -f open_new_aleph_site

Enter input file (needs to be in /exlibris/aleph/u17_2/alephe/scratch): param_files

Enter three-character code of the template library: USM

4 Open New ALEPH Site Updates

In addition to opening the libraries, the ONAS program updates the following setup files:

- In the $alephe_root/**aleph_start** file**:**

  UPDATE www_**base** parameter:

  o -if DEMO libraries are used as a template – BIB library code is inserted

  (for example, www_base   "ABC01")

  o -if VANILLA libraries are used as a template – BIB library code with PUB extension is inserted

(for example, www_base   "ABC01PUB")

- In the $alephe_root/aleph_start.private file:
  - -UPDATE pw_library, usr_library, z105_library based on the libraries in the input file
  - -ADD new libraries with their LIB_PATH to
  - setenv xxxnn_dev {USER_MOUNT}/u${ALEPH_VERSION}_$ALEPH_COPY
  - -ADD new libraries to ALEPH_LIBS and QUE_STARTUP_LIBS
  - -ADD new libraries of type BIB or ILL or COURSE to WORD_STARTUP_LIBS
  - -ADD new libraries of type BIB to ACC_AUT_STARTUP_LIBS
  - -ADD new libraries of type ADM to REQUEST_STARTUP_LIBS

- UPDATE /alephe/tab/tab_system_status

- ADD new libraries at the end of /alephe/tab/library_relation

- ADD new libraries at the end of /alephe/tab/tab_library_group

- ADD new libraries at the end of /alephe/tab/tab_oclc

- ADD at the end of /alephe/tab/tab_base.lng:

  If DEMO libraries are used as a template:
  - Col. 1 <BIB library>
  - Col. 2 blank
  - Col. 3 <BIB library> OPAC
  - Col. 4 <BIB library>
  - Col. 5 blank
  - Col. 6 blank
  - Col. 7 <BIB library>
  - Col. 8 N
  - Col. 9 blank

  For example

```
!        1          2           3            4      5      6      7     8
9
!!!!!!!!!!!!!!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!-
!!!!!!!!!>
TET01                TET01 OPAC          TET01            TET01 N
```

  If VANILLA libraries are used as a template:
  - Col. 1 <BIB library>PUB
  - Col. 2 blank
  - Col. 3 Public Catalog

- o Col. 4 <BIB library>

- o Col. 5 blank

- o Col. 6 blank

- o Col. 7 <BIB library>

- o Col. 8 N

- o Col. 9 wrd=alldocuments not wst=(suppressed or circ-created)

```
!          1          2          3          4     5     6     7     8
9
!!!!!!!!!!!!!!!!!!!!!-!-!!!!!!!!!!!!!!!!!!!!!-!!!!!-!!!!!-!!!!!-!!!!!-!-
!!!!!!!!!>
TET01PUB           Public Catalog      TET01           TET01 N
wrd=alldocuments not wst=(supressed or circ-created)
```

Adds all libraries as defined in the input file

- REPLACE in lib01/pc_tab/catalog/scancode.dat :

  - o Puts matching AUT libraries from the input file into LOCAL line

  - o Activates UTIL/M/7 to Update Tables Package

- REPLACE in lib01/tab/tab_locate/COL.1:

  - o Puts matching AUT library from the input file into col1

- REPLACE in lib01/tab/tab_aut/COL.3/4:

  - o Puts matching AUT library from the input file into cols 3, 4,5, 6

- REPLACE in aut/tab/tab_z105/COL.3:

  - o Puts BIB library from the input file into col. 3

- REPLACE in lib01/tab/tab_rlin:

  - o Puts matching AUT libraries from the input file into col.2 if COL.2 = AUT,

- CREATE in lib01/tab/

  - o Creates 'edit_doc_999' files for every AUT library, defined in the input file.

- ACTIVATE UTIL/H/2 in every new library

- UPDATE in alephe/tab/tab100:

  ILL-MODE = 2

- If the vanilla setup (ATA) was used as the template libraries then the script will also update certain parameters in $alephe_root/pc_server_defaults and www_server.conf. It will also copy the Z39 server and gate related directories to the active alephe directory.

5 Checking Your New Libraries

The following steps are intended to ensure that the new libraries were s opened correctly. It is recommended to check at least one of the newly opened libraries

To check your new libraries

1. Log out, and then log in. This implements all the changes / definitions that were made in the configuration tables.

2. From the command line, enter:

```
dlib <lib_name>
dr
ls
```

3. Check if the library contains the directories copied from the template library.

4. Check if all occurrences of the template library in prof_library were replaced with the current one.

5. If you have selected to create an Oracle user, enter the following in the command line:

```
s+ <lib_name>
```
You should get the prompt:

```
SQL-<lib-name>
```
Then enter:

```
SQL-XXXNN>desc z00;
```
You should get:

```
Name                                    Null?    Type
--------------------------------------- -------- ------------
----------------
Z00_DOC_NUMBER                                   NOT NULL CHAR(9)
Z00_NO_LINES                                     NUMBER(4)
Z00_DATA_LEN                                     NUMBER(6)
Z00_DATA
```
6. Exit by entering:

```
SQL-<lib-name> > exit
```

6 Additional Manual Actions following Activation of ONAS

You must log out and then log in before you proceed (unless you already did so due to section 5 Checking Your New Libraries above) with the following actions in order to implement all the changes and definitions that were made in the environment, and to perform:

dlib xxxnn

6.1 Oracle Tables and Tablespace (file_list and prof_library)

The ONAS script creates an empty $data_root/file_list (that is, the file is created with no content). In this case all Oracle objects will have the default attributes. If the library requires a different value for a specific Oracle object, it must be added manually to the file_list to override the default value.

Open file_list for editing by entering the following in the command line:

```
dlib xxxnn
dr
vi $alephe_file_list
```
If you want to use tablespaces other than the default, you must edit prof_library.

Open prof_library for editing by entering the following in the command line:

```
dlib xxxnn
dr
vi prof_library
```

and modify the following commented lines accordingly:

```
#       setenv   file_list_TS1D       ts_XXXXX_1D
#       setenv   file_list_TS2D       ts_XXXXX_2D
#       setenv   file_list_TS3D       ts_XXXXX_3D
#       setenv   file_list_TS4D       ts_XXXXX_4D
#       setenv   file_list_TS1X       ts_XXXXX_1X
#       setenv   file_list_TS2X       ts_XXXXX_2X
#       setenv   file_list_TS3X       ts_XXXXX_3X
#       setenv   file_list_TS4X       ts_XXXXX_4X
```

XXXXX can be any valid name. Please note that the same tablespace can be shared by several libraries.

Do not forget to remove the # in the beginning of the line.

In addition, you must ensure that these tablespaces exist (using UTIL/A/17/12) and, if necessary, create them (using UTIL/O/17/1).

If you have modified the file_list, or the tablespaces to be used by the library, you must activate UTIL/A/17/1/all to make the changes effective.

6.2 loop_length

The program sets all the loop_length variables (defined in prof_library) to 50,000.

If you want to use different values, you must edit prof_library.

Open prof_library for editing by entering the following in the command line:

```
dlib xxxnn
dr
vi prof_library
```

6.3 Updating Configuration Tables

The following tables should be modified as appropriate:

6.3.1 tab_sub_library.lng

The tab_sub_library.lng table defines the sublibraries of all the ADM libraries on the server.

Open tab_sub_library.eng for editing by entering the following in the command line:

```
vi $alephe_tab/tab_sub_library.eng
```

To consult vanilla set up, see tab_sub_library.eng under alephe_ata/tab, which appears under your U tree.

If your system is configured for languages other than English, the additional copies of the tab_sub_library (for example, tab_sub_library.fre) must also be updated.

Example of the tab_sub_library.eng table:

```
ALEPH 2        L General Patron                    ALEPH       ALEPH
ATA50 2 ATA50 L Express ADM  Library               ATA50       ALEPH
SUBL1 1 ATA50 L Main Library            ATA15 ATA16 ATA17 ALEPH
SUBL2 1 ATA50 L Special Collection      ATA15 ATA16 ATA17 ALEPH
```

6.3.2 tab_base.lng

The tab_base.lng table defines the logical and physical databases that can be accessed by the user through the Web OPAC and search functions.

The program adds all customer libraries codes as defined in input_file.

You can edit col.3 and optionally add logical bases.

Open tab_base.eng for editing by entering the following in the command line:

```
vi $alephe_tab/tab_base.eng
```

To consult vanilla set up, see tab_base.eng under alephe_ata/tab, which appears under your U tree.

If your system is configured for languages other than English, the additional copies of the tab_base (for example, tab_base.fre) must also be updated.

Example of the tab_base.lng table:

```
ATA01                  Express BIB         ATA01 USM01      ATA01 N
ATA01-NOEMPTYHEAD      Express BIB         ATA01 USM01      ATA01 N
ATA10                  Express LC          ATA10 USM10      ATA10 N
ATA12                  Express MeSH        ATA12 USM10      ATA12 N
ATA30                  Express Course R.   ATA30 USM30      ATA30 N
ATA50                  Express ADM                          ATA50 N
ATA60                  Express Holdings                     ATA60 N
SUBL1                   Main Library       ATA01 USM01       ATA01 Y
wsl=subl1
SUBL2                   Special Collection ATA01 USM01       ATA01 Y
wsl=subl2
ATA01PUB                Public Catalog     ATA01 USM01       ATA01 N
wrd=alldocuments not  wst=(suppressed or circ-created)
MONOGRAPHS             Monographs          ATA01             ATA01 Y
wfm=BK
JOURNAL               Serials             ATA01             ATA01 Y
wfm=SE
```

```
ARCHIVE                    Archives Collection  ATA01                ATA01 Y
wcl=ARCHV
GRADUATE                   Graduate Seminar     ATA01                ATA01 Y
wcl=GSC
5771  AUT ATA10    3          OCLC       OCLC
```

6.3.3 base-list (relevant only for BIB libraries)

This HTML file contains the list of databases available to the Web OPAC user. It is located in $alephe_root/www_f_eng/base-list.

Open base-list for editing by entering the following in the command line:

> `wf` (this puts you into the www_f_eng directory)
> `vi base-list`

If your system is configured for languages other than English, the additional copies of the base-list (for example, www_f_fre/base-list) must also be updated.

Example of a library in the base-list file:

```
<td class=text3>
   <img src="&icon_path/f-small-dot.gif" border="0" alt="">
   <a href="&server_f?func=find-b-0&local_base=LOC">
   Library Of Congress - (Z39.50) </a>
</td>

<td class=text3>
   <img src="&icon_path/f-small-dot.gif" border="0" alt="">
   <a href="&server_f?func=find-b-0&local_base=monographs">
   Monographs Catalog (USM01)</a>
</td>
```

7 Using the Vanilla Setup as the Template Libraries for ONAS

The "Vanilla setup" libraries can be used for sites that use MARC 21 as their bibliographic format.

The vanilla setup includes the following libraries:

```
ATA01 – BIB
ATA10 – AUT for LCSH
ATA12 – AUT for MeSH
ATA30 – Course Reserves
ATA40 – ILL
ATA50 – ADM.
ATA60 – HOL
ATA00 – USR/PW
```

Refer to the *"Guide to ALEPH Vanilla Setup"* document for more details on the content of the vanilla setup.

In order to use ATAXX libraries as the template libraries for the **open_new_aleph_site** script, first download the vanilla setup package from the FTP server. The file is called ata_pack_<version>_<service_pack) – for example ata_pack_v17_435 - that is, the vanilla setup is for version 17 and has been updated to SP 435.

Put the file under the U tree in your installation under /uXX_X/alephe/vanilla (for example /exlibris/aleph/u17_7/alephe/vanilla

Steps

1. Go to $alephe_root directory under U tree (for example, /exlibris/aleph/u50_5/ alephe).

2. Create the directory /vanilla under $alephe_root.

   ```
   >> mkdir vanilla
   ```

3. Move to the /vanilla directory:

   ```
   >> cd vanilla
   ```

4. Connect to FTP server/directory outgoing and download ata_pack_v17_<SP>.Z (compressed tar file)

5. Uncompress the file

   ```
   >> uncompress ata_pack.Z
   ```

6. Untar ata_pack as following:

   ```
   >> tar –xvf ata_pack
   ```

You receive the following directory tree structure now under your uXX_X/alephe/vanilla:

```
alephe/      ata00/  ata10/  ata30/  ata50/   ext02/
alephe_ata/  ata01/  ata12/  ata40/  ata60/  ext01/
```

Note that the alephe_ata directory contains some files that are used to update the standard alephe tables.