



Aleph 22 Extended Topologies

Version 22

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, 2015. All rights reserved.

Document released: December 9, 2015

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

Table of Contents

1	All in One Topology	5
2	Multi-Server Topologies	6
	Motivation	6
	<i>Performance</i>	6
	<i>Scalability</i>	6
	<i>Security</i>	6
	<i>Failover and High Availability</i>	6
	Limitations and Requirements	7
	2 Tier Topology	8
	3 Tier Topology	9
3	Operation Guidelines	11
	Aleph_startup and aleph_shutdown Procedures	11
	<i>Aleph_startup</i>	11
	<i>Aleph_shutdown</i>	11
	<i>Web OPAC Save/Mail Functionality</i>	12
	Apache Startup/Shutdown	12
	Oracle RAC Startup/Shutdown	12
	util-c	13
	util-e	13
	Util W	13
	Batch Services	13
4	Appendix A – Aleph Installation on Multiple Servers	15
5	Appendix C – Soft Link Creation	19
6	Appendix D – Apache Configuration for Multi-Apache Server Installation	20

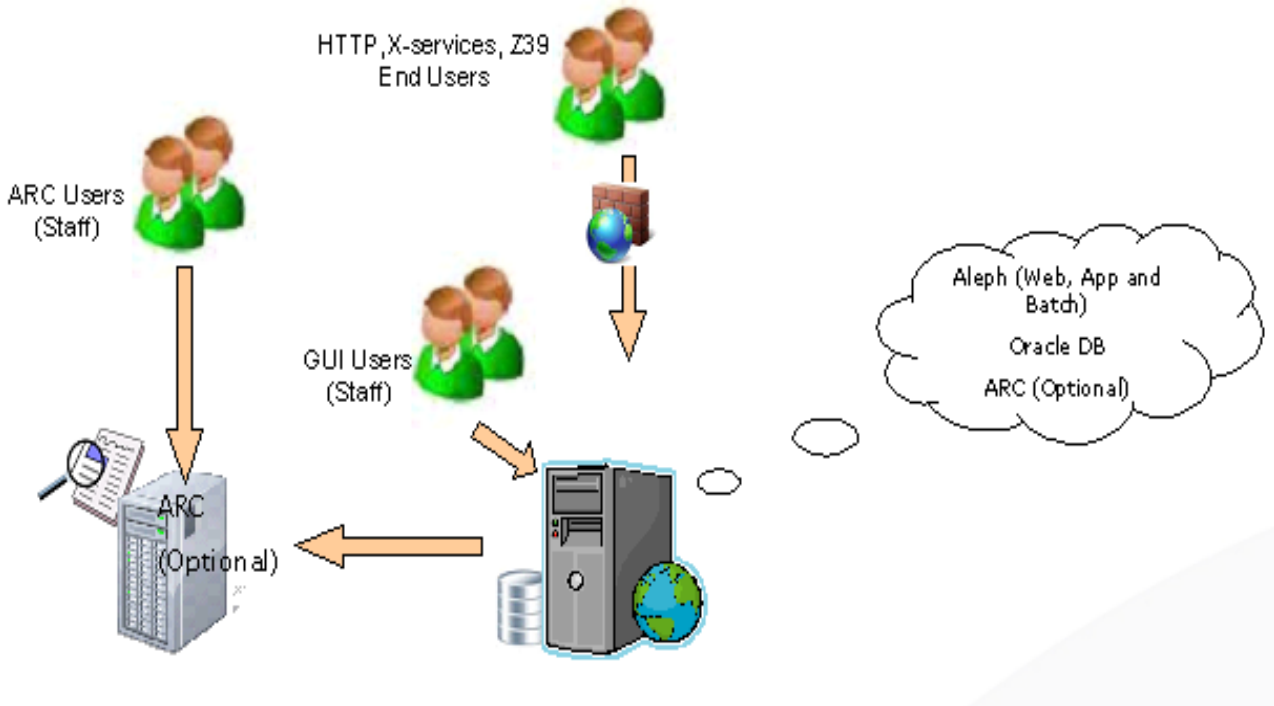
About This Guide

The goals of this document are:

- Describe all existing Aleph topologies
- Describe the motivation and limitation of each topology
- Describe the implementation of each topology
- Describe the high level operation of multi-server Aleph topologies

All in One Topology

The all in one topology consists of a single server containing both the database and the application.



Note: : ARC can be installed on the same server as Aleph or on a dedicated server.

Implementation: Basic installation using the Aleph Installation Kit.

Multi-Server Topologies

Motivation

Performance

Performance can be improved by doing the following:

- Adding more computing power to the system
- Dividing the system functionality among several computing entities

The Aleph multi-server topologies enable you to perform these solutions by:

- Enabling you to add servers to the system (more CPU's and memory) to increase the computing power of the system
- Enabling the separation of application functionality into several servers to divide the load between the system servers

Scalability

Data size and system load can increase over the years. The option to add servers to the Aleph system increases the system scalability.

Aleph multi-server topologies provide the option to add resources to the system and define the functionality of the new resource (PC server, WWW server, daemons, or batch queue).

Security

Using multi-server topologies increases the system security because:

- Apache can be implemented on a dedicated machine with or without a firewall
- Application and data base servers can be placed in an internal network with access via a firewall only
- There is no direct access to the database from Internet

Failover and High Availability

When an application is installed on two or more servers, failover and high availability of Aleph services can be achieved using the load balancer over application servers. High availability can be implemented for major Aleph services, such as Web OPAC, Z39, and staff services (`pc_server`).

Limitations and Requirements

- Only one application server can be designated as the dedicated machine on which the batch queue is activated.
- Only one application server can be designated as the dedicated machine on which the daemons are activated.

Note:

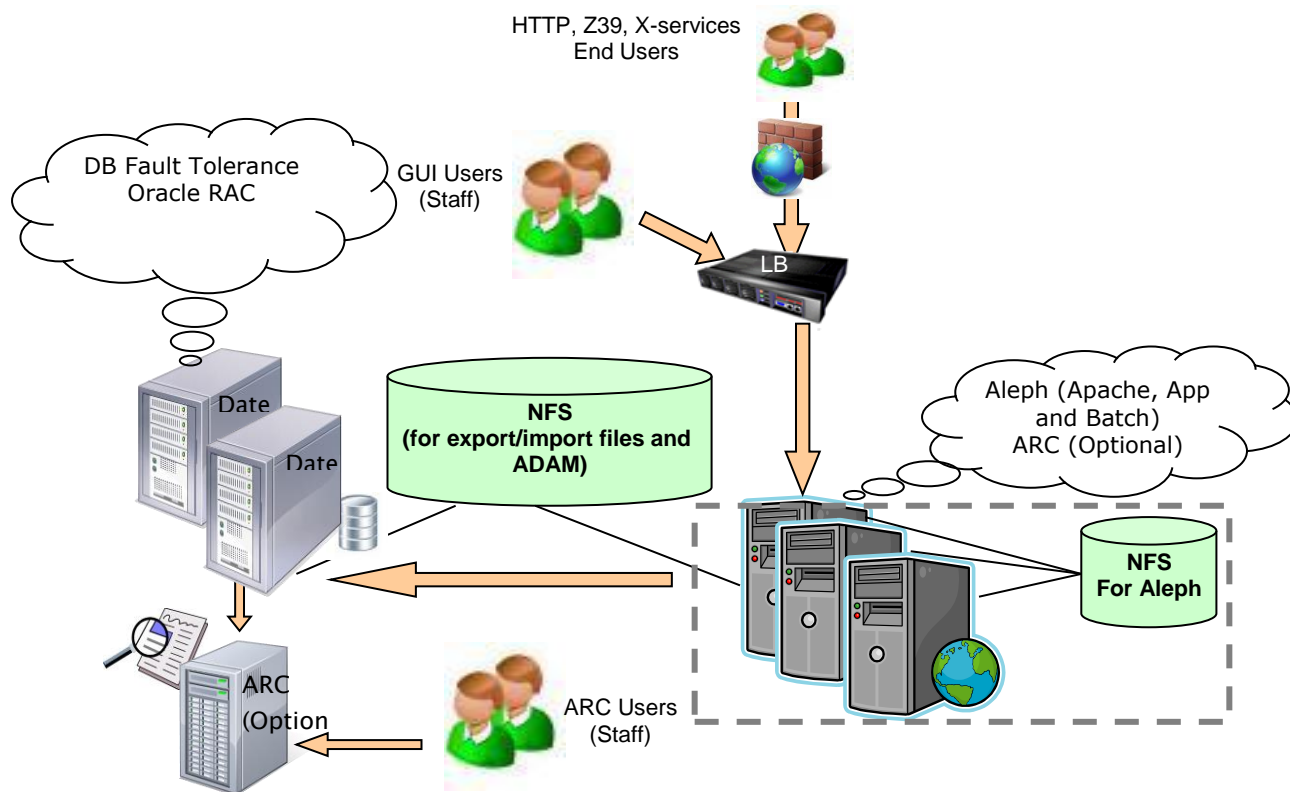
Batch queue and daemons can run on the same machine.

- No high availability for batch services and daemons.
- Failover and high availability for the Aleph services listed above can be achieved using the load balancer over application servers.
- All application servers must have a shared disk with full read and write permissions.
An additional shared disk between the application and database servers is absolutely necessary for export and import functionality.

2 Tier Topology

2 tier topology consists of the following:

- **Application server(s) (one or more)** – contains the application and batch server.
Dedicated batch server(s) (one or two) – The batch queue and daemons either both run on the same application server or each runs on a separate server.
- **Database server(s) (one or more)** – accessed using TWO-TASK mode.



Note : ARC can be installed on one of the Aleph application servers or on a dedicated server.

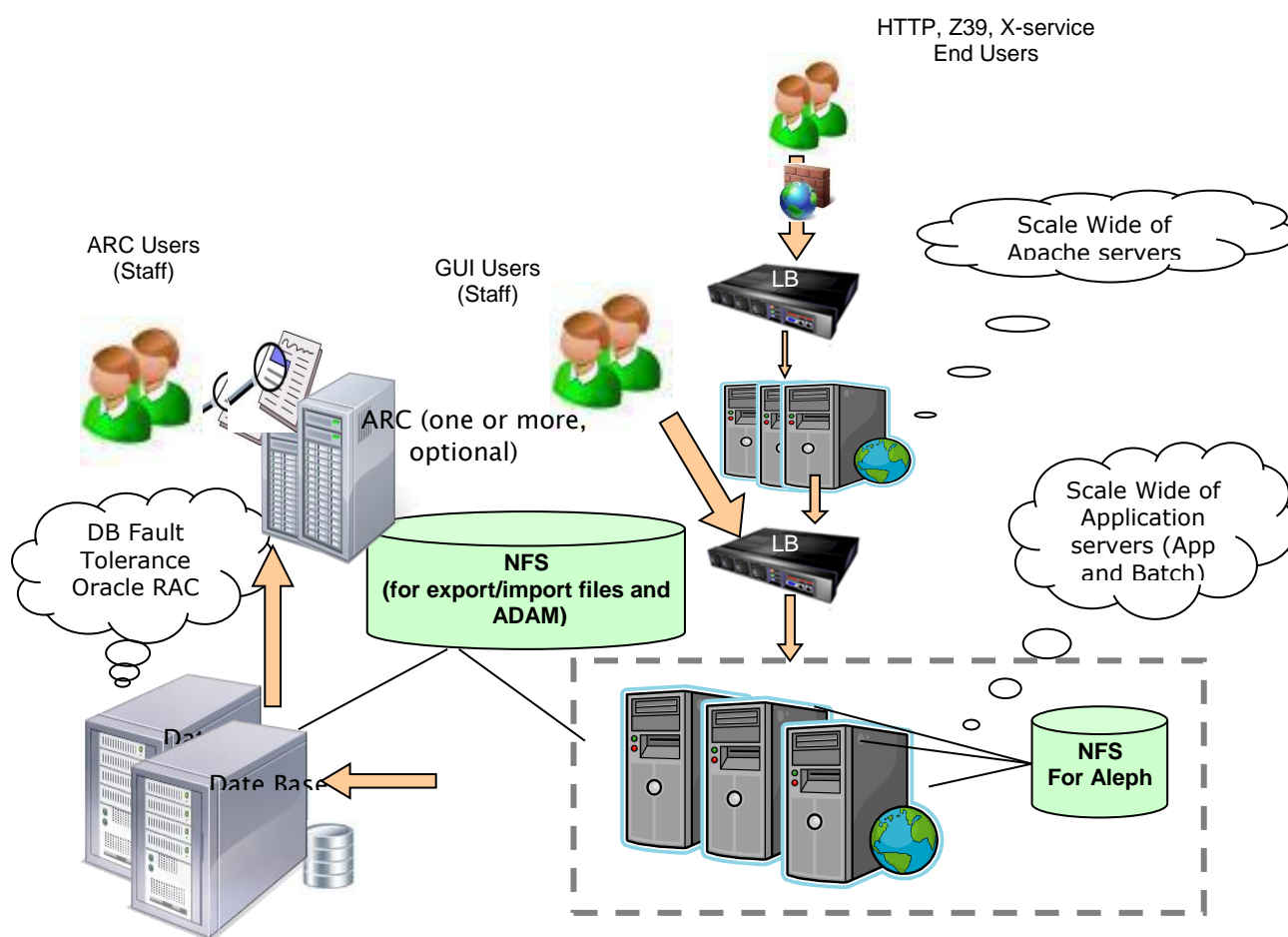
Implementation:

- **For one application server and one database server** – Installation using the Aleph Installation Kit in TWO-TASK mode.
- **For multiple application servers** – See [Appendix A – Aleph Installation on Multiple Servers](#) on page 15 for detailed implementation instructions.
- **For multiple database servers** – The Oracle server and database should be installed by the Ex Libris Installation Team, the customer's DBA, or a system administrator on the DB servers. Install the Oracle client on all Aleph servers and configure it to work in TWO_TASK mode.

3 Tier Topology

3 tier topology consist of the following:

- **Apache server(s) (one or more)**
- **Application server(s) (one or more)**
 - Each application server can activate different services, for example – WWW & PC servers, Z39 & SRU servers, OCLC server, etc.
- **Dedicated batch server(s) (one or two)**– The batch queue and daemons either both run on the same application server or each runs on a separate server.
- **Database server(s) (one or more)** – accessed using TWO-TASK mode.



Notes:

- ARC can be installed on one of the Aleph application servers or on a dedicated server (one or more).
- Customers using ADAM on one server and Apache on separate server should define a shared directory (NFS) between the Apache server and the application server.

The apache_media directory should be shared with the following security restrictions:

- Apache server - Should have read only permissions
 - Application server - Should have read/write permissions
-

Implementation:

- **For Apache servers**
 - 1 Install Aleph on each Apache server using the Aleph Installation Kit in TWO-TASK mode (application part only). Install the service pack on each Apache server.
 - 2 See [Appendix D – Apache Configuration for](#) on page [20](#) for detailed Apache configuration instructions
- **For one application server and one Database server** – Installation using the Aleph Installation Kit in TWO-TASK mode.
- **For multiple application servers** – See [Appendix A – Aleph Installation on Multiple Servers](#) on page [15](#) for detailed implementation instructions.
- **For multiple Database servers** – The **Oracle** server and database should be installed by the Ex Libris Installation Team, your DBA, or your system administrator on the database servers. Install the Oracle client on all Aleph servers and configure it to work in TWO-TASK mode.

Operation Guidelines

Aleph_startup and aleph_shutdown Procedures

Aleph_startup

It is possible to define on which server a specific process runs. When activating `aleph_startup` from one of the application servers or dedicated batch servers, only the processes defined on that server is executed.

This is done by defining the server names for each process in `aleph_start.private`.

See [Appendix A – Aleph Installation on Multiple Servers](#) on page 15 for detailed implementation instructions.

If a complete startup of Aleph is required, execute `aleph_startup` from each of the machines in the topology.

Note:

The `aleph_startup` procedure starts up only the Aleph application processes. It does not start up Oracle or Apache. Refer to the **Automatic Startup/Shutdown of All Aleph Components** chapter of the *System Administrator's Guide - System Overview* document for detailed information about the startup/shutdown mechanism.

Aleph_shutdown

Activating `aleph_shutdown` on one of the application servers or dedicated batch servers terminates only the processes that are currently running on the server.

Aleph daemons are stopped by creating a file on the disk. These files are created only if `aleph_shutdown` is activated from the dedicated daemon server.

In other words, Aleph daemons stop only when `aleph_shutdown` is activated from the dedicated daemons server. [Appendix A – Aleph Installation on Multiple Servers](#) on page 15 for detailed implementation instructions.

If a complete shutdown of Aleph is required, execute `aleph_shutdown` from each of the machines in the topology.

Note:

The `aleph_shutdown` procedure shuts down only the Aleph application processes. It does not shut down Oracle or Apache. Refer to the **Automatic Startup/Shutdown of All Aleph Components** chapter of the *System Administrator's Guide - System Overview* document for detailed information about the startup/shutdown mechanism.

Web OPAC Save/Mail Functionality

When using a separate Apache server, a special configuration is required for the save/mail functionality in the Web OPAC – to display a file download dialog box and allow users to save the records.

Note that with this topology, the save/mail functionality is not based on the search result, so that an additional search is performed. A sort is not applied.

For more information, refer to the *Aleph System Librarian's Guide - Web OPAC*.

See [Appendix A – Aleph Installation on Multiple Servers](#) on page 15 for detailed implementation instructions.

Apache Startup/Shutdown

Automatic startup/shutdown of Apache is implemented during the initial installation. For more information, refer to the **Automatic Startup/Shutdown of all Aleph Components** section of the *System Administrator's Guide - System Overview*.

Perform the manual startup/shutdown of apache on each necessary Apache server one by one. There is no mechanism for a centralized Apache startup/shutdown.

For more information on Apache startup/shutdown and log files, refer to the **Apache** section of the *System Administrator's Guide - System Overview*.

Oracle RAC Startup/Shutdown

Use the Oracle Enterprise Manager DB console for Oracle RAC data base administration. For instructions, click the following link:

http://download.oracle.com/docs/cd/E11882_01/server.112/e10897/em_manage.htm#ADMOS031

If the database is down on both RAC servers and the EM DB console is not available, the database can be started manually using the following Unix commands:

```
su - oracle
srvctl start database -d <DB name>-n <node name>
```

Note:

Some of Aleph Util O functionality is not suitable for multi-servers topology (with or without Oracle RAC implementation). For more information, refer to the **Util O – Oracle Management** and **Working in a TWO_TASK Environment** sections of the *Aleph 21 System Administration Guide*.

util-c

The dedicated batch queue server is defined in `aleph_start.private`.

Aleph `batch_que` works only on the dedicated batch queue server. Therefore, util-c can run only on this server.

An error message is displayed if it is activated from one of the application servers.

See [Appendix A – Aleph Installation on Multiple Servers](#) for detailed implementation instructions.

util-e

The dedicated daemons server is defined in `aleph_start.private`.

Aleph daemons work only on the dedicated daemon server. Therefore, util-e can run only on this server.

An error message is displayed if it is activated from one of the application servers.

See [Appendix A – Aleph Installation on Multiple Servers](#) for detailed implementation instructions.

Util W

Util-w manages and monitors processes (WWW, PC, etc.) only on the server from which it is activated.

Batch Services

Batch services are normally executed on the dedicated batch queue server by activating them from the GUI.

It is possible to run a service on one of the other application servers by performing one of the following options:

- Choosing the Online option in the GUI service window
- Manually executing the service on the server (from the UNIX command line level)

Until now, running services that lock the library created stop files that caused the daemons to stop. The following files were created:

- `$data_scratch/util_e_01_stop`
- `$data_scratch/util_e_06_stop`
- `$data_scratch/util_e_08_stop`

- \$data_scratch/util_e_11_stop
- \$data_scratch/util_e_13_stop
- \$data_scratch/util_e_21_stop

After the library was unlocked, the daemons started on the server from which the service had been executed. This prevented executing batch services that lock the library from servers other than the dedicated batch server.

A new mechanism now enables you to run batch services that lock the library on the dedicated batch queue server by activating them from the GUI and on one of the other application servers either by choosing the Online option in the GUI service window or by running them manually.

When executing services that lock the library, a new file is created – util_e_nn_idle. The existence of this file stops the daemons from processing, but they are still active.

After the library is unlocked, these files are removed and the daemons resume processing on the dedicated daemon server regardless of the server from which the service has been executed.

The new files are -

- \$data_scratch/util_e_01_idle
- \$data_scratch/util_e_06_idle
- \$data_scratch/util_e_08_idle
- \$data_scratch/util_e_11_idle
- \$data_scratch/util_e_13_idle
- \$data_scratch/util_e_21_idle

Appendix A – Aleph Installation on Multiple Servers

1 Install Aleph in a shared /exlibris directory (NFS):

- Create local directories /exlibris_local on all machines to store local configurations.
- Copy all local configuration files to the /exlibris_local directory of all shared servers (see [Appendix B – List of Unique Files](#) on page 18 for a list of the local configuration files).
- Create links from the shared version in /exlibris/aleph to the local configuration files in /exlibris_local (see [Appendix C – Soft Link Creation](#) on page 19 for the exact commands).

2 The following files are created by soft links or configuration files on the local copy of each machine. (The files are located in /exlibris_local) with the relative path under:

- \$salephe_tab/server_info – created empty on the local dir and soft linked.
- \$salephe_tab/server_info_childs – created empty on the local dir and soft linked.
- \$httpd_root/logs/httpd.pid – pointed to the local dir by the httpd.conf file.
- \$salephe_root/apache/htdocs/.index.html – only needs to be soft linked when load balancer is not operational.

3 For each application server, create LOGDIR and TMPDIR:

```
mkdir $LOGDIR.`hostname` $TMPDIR.`hostname`
```

4 Update the following configuration files to allow sharing by several application servers:

- \$salephe_root/apache/conf/httpd.conf
Comment out the server name directory.
- \$salephe_root/aleph_start
 - ORA_HOST is hard coded to the database machine name. If Oracle RAC is implemented, use the TNS entry name as defined in tnsnames.ora
 - WWW_HOST is set to <hostname> -i until the load balancer is up. After the load balancer is up, it is hard coded to the load balancer IP.
 - Z39_HOST is set to <hostname> -s
 - Since external access to PDS is through the balancer, direct access from Aleph to the PDS (get attributes) should be done internally:
 - setenv PDS_HOST_IN `hostname -s`
 - setenv PDS_PORT_IN 899\${ALEPH_SUB_PORT}

- \$alephe_root/aleph_start.private
 - **Define the batch queue server:**
 - setenv BATCH_HOST_NAME <Batch_hostname>
 - **Define the daemons server:**
 - setenv DAEMONS_HOST_NAME <Daemons_hostname>
 - **For each application server, define LOGDIR and TMPDIR as a server specific directory:**

```
setenv TMPDIR ${ALEPH_MOUNT}/a${ALEPH_VERSION}_1/log.`hostname`
setenv LOGDIR ${ALEPH_MOUNT}/a${ALEPH_VERSION}_1/log.`hostname`
```

Note:

A single WWW or PC server session can be logged in different log files.

- \$alephe_root/www_server.conf

Replace the IP address or host name of the WWW_HOST with \${WWW_HOST}

5 Update \$alephe_root/aleph_start.private to define on which application servers each Aleph process will be executed:

- setenv WWW_START_HOST <WWW server(s)>
- setenv PC_START_HOST <PC server(s)>
- setenv Z39_START_HOST <Z39 server(s)>
- setenv Z39_GATE_HOST <Z39 Gate server(s)>
- setenv SRU_START_HOST <SRU server(s)>
- setenv OCLC_START_HOST <OCLC server(s)>
- setenv VST_START_HOST <VST server(s)>
- setenv NCIP_HOST <NCIP server(s)>
- setenv SLNP_HOST <SLNP server(s)>
- setenv SIP2_HOST <SIP2 server(s)>
- setenv SC_STARTUP_HOST <SC server(s)>

6 Set the preview_required attribute to Y in all forms that use the Save/Mail option to allow save/mail with separate apache machine by adding the following line:

```
<input type=hidden name=func value=myshelf-mail>
```

In the following files:

- ./alephe/www_f_lng/myshelf-mail
- ./alephe/www_f_lng/myshelf-course
- ./alephe/www_f_lng/user-info-mail
- ./alephe/www_f_lng/short-mail


```
./alephe/www_f_lng/full-mail
```

- 7 Define for all profiles (Z61) an appropriate maximum limit of the number of save/mail files for all profiles (the default is 999999). This is in order to allow save/mail with separate Apache machines.

Only install the service pack for multi-server installations with a shared disk once. If there are any implementation notes related to changes in unique files, implement them on each server.

See [Appendix B – List of Unique Files](#) on page [18](#) for a list of unique files.

Note:

When running two JBoss servers on two different machines that share a central disk space, the `deploy` directory (where the `rest-dlf.war` file is located) and the `lib` directory can reside in a central location. However, the `work`, `data`, `conf`, `log`, and `tmp` directories must reside locally, by using symbolic links.

Appendix B – List of Unique Files

Copy the list of files and directories to the /exlibris_local directory of each server:

```
$alephe_root/apache/logs  
$LOGDIR  
$TMPDIR  
$alephe_root/tab/server_info  
$alephe_root/tab/server_info_childs  
$http_root/htdocs/.index.html *only when the load balancer is off
```

Appendix C – Soft Link Creation

```
mkdir -p /exlibris_local/alephe/tab
cd $alephe_tab
rm server_info server_info_childs
ln -sf /exlibris_local/alephe/tab/server_info $alephe_tab/
ln -sf /exlibris_local/alephe/tab/server_info_childs $alephe_tab/
```

Some directories are created with a <hostname> suffix if the application configuration is too complicated. The local directory is replaced by a link to the local disk. It links to the suffix dir.

For example -

```
cd $aleph_dev
mkdir log.`hostname`
mv log log.$$
ln -s /exlibris_local/aleph/a21_1/log .
ln -s log.`hostname` /exlibris_local/aleph/a21_1/log
```

This is relevant for the log, tmp, Apache logs, and jboss logs.

Appendix D – Apache Configuration for Multi-Apache Server Installation

This section describes how to configure Apache to work with the Load Balancer or dedicated application server:

- 1 Change the following section in the `$httpd_root/httpd.conf` file:

```
<IfModule mod_aleph_22.c>  
    ServerAddr 127.0.0.1  
    ServerPort 4991  
</IfModule>
```

- 2 Change the value of `ServerAddr` from `127.0.0.1` to the load balancer or dedicated application server address.