

Configuring ADOBE® LIVECYCLE® ES4 Application Server Clusters using JBOSS®

Legal notices

For legal notices, see http://help.adobe.com/en_US/legalnotices/index.html.

Contents

Chapter 1: About This Document

1.1 Who should read this document?	1
1.2 Conventions used in this document	1
1.3 Additional information	2

Chapter 2: Introduction to Installation, Configuration, and Deployment Process

2.1 Installation, configuration, and deployment overview	3
2.2 Selecting tasks for configuring and deploying	3
2.3 LiveCycle installation and deployment list	4

Chapter 3: Configuring JBoss in a Cluster

3.1 Preparing to install	5
3.2 Installing JBoss Application Server software	6
3.3 Running JBoss in a cluster	7
3.4 Modifying the JBoss run.conf file	7
3.5 Configuring LiveCycle database connectivity	10
3.6 Testing the JBoss Application Server cluster	16

Chapter 4: Installing LiveCycle modules

4.1 Before you begin	17
4.2 Installation considerations	18
4.3 Installing LiveCycle	19
4.4 Configuring the caching locators in clusters (caching using TCP only)	20
4.5 Global Document Storage Directory (GDS)	23
4.6 Installing the font directories in cluster nodes	23
4.7 Next steps	23

Chapter 5: Configuring LiveCycle for deployment

5.1 Considerations when configuring and deploying LiveCycle	24
5.2 LiveCycle pre-configuration tasks	25
5.3 Configuring and deploying LiveCycle	26

Chapter 6: Post-deployment tasks

6.1 General tasks	32
6.2 Verify the LiveCycle cluster	36
6.3 Verify the CRX Cluster	36
6.4 Accessing module web applications	37
6.5 Configure Correspondence Management Solution	39
6.6 Configuring PDF Generator	45
6.7 Final setup for Rights Management	52
6.8 Configuring LDAP access	52
6.9 Enabling FIPS mode	53
6.10 Configuring HTML digital signature	54
6.11 Configuring Connector for EMC Documentum	54

Contents

6.12 Configuring the Connector for IBM Content Manager	59
6.13 Configuring the Connector for IBM FileNet	62
6.14 Isolating JBoss Clusters	66
6.15 (Optional)Enable JMX console security	66
 Chapter 7: Configuring Load Balancing	
7.1 HTTP Connector Based Load Balancing	68
7.2 AJP Connector Based Load Balancing	70
 Chapter 8: Advanced Production Configuration	
8.1 Configuring pool size for Output and Forms	73
8.2 PDF Generator	73
8.3 Enabling CIFS on Windows	74
 Chapter 9: Appendix - Install Command Line Interface	
9.1 Overview	76
9.2 Install LiveCycle	76
9.3 Error logs	77
9.4 Uninstalling LiveCycle in console mode	78
 Chapter 10: Appendix - Configuration Manager Command Line Interface	
10.1 Order of operations	79
10.2 Command Line Interface property file	79
10.3 General configuration properties	80
10.4 Examples Usage	90
10.5 Configuration Manager CLI Logs	90
10.6 Next steps	90
 Chapter 11: Appendix - Configuring JBoss as a Windows Service	
11.1 Download the Web Native Connector	91
11.2 Install the Windows service	91
11.3 Start and stop JBoss Application Server as a Windows service	92
11.4 Verify the installation	92
11.5 Additional configuration	92
 Chapter 12: Appendix - Manually Configuring JBoss	
12.1 Installing the JDK for JBoss	94
12.2 Manually installing JBoss	95
12.3 Starting and stopping JBoss	95
12.4 Modifying the JBoss configuration	96
12.5 Copying jar files	100
12.6 LiveCycle database connectivity for manually installed JBoss	100
12.7 Next steps	108

Chapter 1: About This Document

LiveCycle is an enterprise server platform that helps you automate and streamline business processes. LiveCycle comprises the following components:

- J2EE-based Foundation provides server capabilities and runtime environment
- Tools to design, develop, and test LiveCycle Applications
- Modules and Services are deployed on LiveCycle Server and provide functional services

For more information about the LiveCycle architecture and capabilities, see [LiveCycle Overview](#).

This document is part of a larger documentation set available at [LiveCycle Documentation page](#). It is advised that you start with the preparing guide and then move on to installation and configuration guide depending on whether you are performing a fresh installation (single server or cluster setup) or upgrading your existing LiveCycle deployment. For Turnkey deployment, which is only for evaluation purposes, see [Installing and Deploying LiveCycle using JBoss Turnkey](#).

1.1 Who should read this document?

This guide provides information for administrators or developers who are responsible for installing, upgrading, configuring, administering, or deploying LiveCycle components. The information provided is based on the assumption that anyone reading this guide is familiar with J2EE application servers, operating systems, database servers, and web environments.

1.2 Conventions used in this document

The installation and configuration documentation for LiveCycle uses the following naming conventions for common file paths.

Name	Default value	Description
<i>[LiveCycle root]</i>	Windows: C:\Adobe\Adobe LiveCycle ES4 Linux and Solaris: /opt/adobe/adobe_livecycle_es4	The installation directory that is used for all LiveCycle modules. The installation directory contains subdirectories for LiveCycle Configuration Manager. This directory also includes directories related to third-party products.
<i>[appserver root]</i>	These installation locations are examples. Your installation location may be different. JBoss Application Server on Windows: C:\jboss\ JBoss Application Server on Linux: /opt/jboss/	The home directory of the application server that runs the services that are part of LiveCycle.

Name	Default value	Description
[dbserver root]	Depends on the database type and your specification during installation.	The location where the LiveCycle database server is installed.
[lc_temp_dir]	On Windows: C:\Adobe\Adobe LiveCycle ES4\tmp On Linux, UNIX and AIX: /opt/adobe/adobe_lifecycle_es4/tmp	The temporary directory for LiveCycle server.
[CRX_home]	On Windows: C:\Adobe\Adobe LiveCycle ES4\crx-repository On Linux, UNIX and AIX /opt/adobe/adobe_lifecycle_es4/crx-repository	The directory that is used for installing the CRX repository.

Most of the information about directory locations in this guide is cross-platform (all file names and paths are case-sensitive on non-Windows operating systems). Any platform-specific information is indicated as required.

1.3 Additional information

The resources in this table can help you learn more about LiveCycle.

For information about	See
General information about LiveCycle and the modules	LiveCycle Overview
LiveCycle modules	LiveCycle Modules
Other services and products that integrate with LiveCycle	Adobe Developer Connection
Installing Adobe® LiveCycle® Workbench 11	Installing Adobe LiveCycle Workbench 11
Preparing to Install LiveCycle	Preparing to Install LiveCycle (Server Cluster)
LiveCycle ES4 Upgrade Checklist and Planning	LiveCycle ES4 Upgrade Checklist and Planning
Troubleshooting LiveCycle	Troubleshooting LiveCycle
Performing administrative tasks for LiveCycle	LiveCycle Administration Help
All the documentation available for LiveCycle	LiveCycle documentation
Patch updates, technical notes, and additional information about this product version	Adobe Enterprise Support

Chapter 2: Introduction to Installation, Configuration, and Deployment Process

2.1 Installation, configuration, and deployment overview

Installing, configuring, and deploying LiveCycle involves the following processes:

- **Installing:** Install LiveCycle by running the installation program. Installing LiveCycle places all of the required files onto your computer, within one installation directory structure. The default installation directory is C:\Adobe\Adobe LiveCycle ES4 (Windows) or /opt/adobe/adobe_livecycle_es4 (non-windows); however, you can install the files to a different directory.
- **Configuring:** Configuring LiveCycle modifies various settings that determine how LiveCycle works. Assembling the product places all of the installed components into several deployable EAR and JAR files, according to your configuration instructions. Configure and assemble the components for deployment by running Configuration Manager. You can configure and assemble multiple LiveCycle modules at the same time.
- **Deploying:** Deploying the product involves deploying the assembled EAR files and supporting files to your application server on which you plan to run your LiveCycle. If you have configured multiple modules, the deployable components are packaged within the deployable EAR files. Components and LiveCycle archive files are packaged as JAR files.

***Note:** LiveCycle archive file use .lca file extension.*

- **Initializing the LiveCycle database:** Initializing the database to be used with LiveCycle creates tables for use with User Management and other components. Deploying any module that connects to the LiveCycle database requires you to initialize the LiveCycle database after the deployment process.

Before you begin to install and configure LiveCycle, ensure that you have prepared your environment as described in the applicable Preparing guides.

2.2 Selecting tasks for configuring and deploying

After you have installed LiveCycle, you can run Configuration Manager to:

- Configure LiveCycle modules in an EAR file for deploying to the application server or cluster of application servers
- Initialize LiveCycle database
- Deploy LiveCycle components
- Validate LiveCycle component deployment
- Configure LiveCycle components

2.3 LiveCycle installation and deployment list

The following list includes the steps that are required for installing LiveCycle by using the manual method. Your application server or cluster must be installed and configured before you perform the installation.

- Ensure that the required software is installed on each machine and configured in the target environment.
- Ensure that you created and configured the application server cluster in the target environment. You can choose to manually configure JBoss or use the Adobe pre-configured one.
- Run the installation program only on one machine.
- Run Configuration Manager and select the Configure LiveCycle EARs task. This task configures and assembles LiveCycle.

Run the manual configuration steps on the machines where LiveCycle is not installed. For example, copy content repository.

- Deploy the EAR files to the application server or cluster. You must do it manually.
- Run Configuration Manager to initialize the LiveCycle database and deploy LiveCycle component files.
- Access Administration Console and User Management.
- (Optional) Configure LDAP access.

Chapter 3: Configuring JBoss in a Cluster

The JBoss Application Server configuration is defined by a number of configuration files in several directories. To configure JBoss for use in a cluster, you must modify a number of configuration files. You can use any text editor to modify them.

Perform the following tasks to configure your JBoss cluster environment:

- Ensure that you properly prepared all computers in the cluster. (See “[3.1 Preparing to install](#)” on page 5.)
- Install JBoss Application Server software. (See “[3.2 Installing JBoss Application Server software](#)” on page 6.)
- Modify the JBoss run file. (See “[3.4 Modifying the JBoss run.conf file](#)” on page 7.)
- Configure LiveCycle database connectivity. (See “[3.5 Configuring LiveCycle database connectivity](#)” on page 10.)
- Test your JBoss cluster configuration. (See “[3.6 Testing the JBoss Application Server cluster](#)” on page 16.)

3.1 Preparing to install

Before you install JBoss Application Server on the computers of your cluster, ensure that your system meets the following configuration requirements:

Disk space: Ensure that the partition that will hold the application server has a minimum of 10 GB of free disk space. In addition to the space required to install the product, your environment variable `TEMP` or `TMP` must point to a valid temporary directory with at least 500 MB of free disk space. The downloadable executable requires approximately 500 MB, plus an additional 1.0 GB to unpack the images.

IP address settings: All the computers must have a fixed IP address that is managed through a single DNS.

IP multicast: All the computers must fully support IP multicast packet propagation, which means that all routers and other tunneling technologies must be configured to propagate multicast messages to clustered server instances. The network latency must be low enough to ensure that most multicast messages reach their final destination within 200 to 300 milliseconds. Also, the multicast time-to-live (TTL) value for the cluster must be high enough to ensure that routers do not discard multicast packets before they reach their final destination.

Versions: All the computers in the cluster must have the same version and same service pack of JBoss Application Server software.

Horizontal clustering: If your configuration is horizontally clustered (that is, instances of JBoss Application Server are installed on separate computers), ensure that all computers are on the same network subnet and that the computer clocks are synchronized. (See [Preparing to Install LiveCycle \(Server Cluster\)](#).)

Account privileges: (Only for PDF Generator on Windows) You must install and run JBoss Application Server under a user account that has administrator privileges.

Shared network drive: You must have a secure shared network drive created that all computers in the cluster can access with read and write permissions. (See [Preparing to Install LiveCycle \(Server Cluster\)](#))

J2SE SDK version: For information on J2SE SDK version, see [Supported Platform Combinations](#) document.

Clocks of all the systems on the cluster might be synchronized to a common time server. In Windows domain, clock synchronization is done automatically. You must set-up `Network Time Protocol` on non-windows systems.

3.2 Installing JBoss Application Server software

Install and configure JBoss Application Server on each computer of the cluster. You can install multiple instances on any computer where you plan to implement vertical clustering. The *Preparing to Install LiveCycle (Server Cluster)* document describes the versions of JBoss Application Server that are supported for LiveCycle ES4.

Install the Adobe-preconfigured JBoss Application Server that is provided on the LiveCycle installation medium under the `third_party` directory. When you extract the `third_party\jboss.zip` file, the following sub-directories are created under the `[appserver root]/server/` directory:

- (Single server) `lc_oracle`
- (Single server) `lc_sqlserver`
- (Single server) `lc_mysql`
- (Cluster) `lc_sqlserver_cl`
- (Cluster) `lc_oracle_cl`
- (Cluster) `lc_mysql_cl`

You can safely remove the directories that are not relevant to your configuration. For example, if you plan to use Oracle for Adobe-preconfigured JBoss in a clustering configuration, retain `lc_oracle_cl` and delete the other directories.

Note: The `[appserver root]/server/all` directory is relevant only for manually-configured JBoss. For Adobe-preconfigured JBoss, you can use a relevant database-specific directory mentioned above instead of the `/all` directory.

Important: Install only the Adobe-preconfigured JBoss Application Server described above, and then see the following sections of this document to configure the nodes for your cluster. Do not follow the JBoss configuration instructions that are described in *Preparing to Install LiveCycle (Single server)*; they apply to a stand-alone configuration and are not appropriate for a clustered configuration.

3.2.1 Installing JBoss Application Server for a horizontal cluster

Install the Adobe-preconfigured JBoss Application Server by extracting the contents of the `JBoss.zip` directory to the location where you intend to install JBoss Application Server on each computer of the cluster. This installation is fully configured for a horizontal cluster.

3.2.2 Configuring Windows services for JBoss Application Servers

If the JBoss Application Servers of your cluster run on a Windows operating system, you may optionally install Windows services to manage them. The Windows service provides a GUI that simplifies starting and stopping of the application servers of your cluster.

You must install JBoss Application Server before you create the Windows service to manage the application server. You must create a separate Windows service to manage each JBoss Application Server of the cluster. See “[Appendix - Configuring JBoss as a Windows Service](#)” on page 91 for information about using the JBoss Web Native Connector to configure JBoss as a Windows service.

To start JBoss Application Server as a Windows service:

- ❖ On a JBoss Application Server of the cluster, select **Start > Control Panel > Administrative Tools > Services**, then select the Windows service for JBoss Application Server and click **Start**.

Note: When starting JBoss Application Server as a Windows service, the console output is redirected to the file `run.log`. You can inspect the file to discover any errors that occur during service startup.

To stop JBoss Application Server as a Windows service:

- ❖ On a JBoss Application Server of the cluster, select **Start > Control Panel > Administrative Tools > Services**, then select the Windows service for JBoss Application Server and click **Stop**.

***Note:** When stopping JBoss Application Server as a Windows service, the console output is redirected to the file run.log. You can inspect the file to discover any errors that occur during service shutdown.*

3.3 Running JBoss in a cluster

Start the JBoss

Start the JBoss Application server by entering the following command.

```
./run.sh -b <ipaddress> -c <server_profile>
```

***Note:** This will start JBoss instance using default cluster partition name, multicast address and multicast port.*

Changing the Cluster Partition name

```
./run.sh -g <partition_name> -b <ipaddress> -c <server_profile>
```

***Note:** This command will use default multicast port and address. Every node of the cluster should have same value for the <partition name> and ensure that this value is unique for your JBoss cluster.*

You can also mention partition name as a JVM argument as mentioned in “[3.4 Modifying the JBoss run.conf file](#)” on page 7 section.

Changing the Multicast address

Some users have a specific set of multicast addresses. To change the default Multicast address to the address of your choice enter the following command:

```
/run.sh -u <UDP group Ip address> -g <partition_name> -b <ipaddress> -c <server_profile>
```

***Note:** In the above command, <UDP group Ip address> represents UDP Multicast address.*

Changing the MultiCast port

To change the Multicast port provide the following jvm argument in the run.conf file as mentioned in “[3.4 Modifying the JBoss run.conf file](#)” on page 7 s:

```
-Djboss.jgroups.udp.mcast_port=<port_number>
```

***Note:** By default, JBoss cluster is configured to use UDP based clustering. You can configure cluster to use TCP based clustering. Refer JBoss Clustering, for detailed information about configuring cluster to use TCP based clustering.*

3.4 Modifying the JBoss run.conf file


Modify the JBoss run file of each JBoss Application Server instance in the LiveCycle cluster to add LiveCycle options.

Before you start this procedure, determine how your LiveCycle cluster implements cluster caching so that you can correctly configure an argument for cluster caching. You can implement cluster caching by using either TCP or UDP, but not both. The following factors may affect your choice:

- (Recommended) Use TCP if your cluster is either IPv4-based or IPv6-based. On an IPv6-based cluster, you must use TCP to be IPv6-compliant.

If you implement cluster caching by using TCP, also ensure that you configure the TCP locators correctly. (See “Configuring the caching locators (caching using TCP only)” .)

- Option to use UDP is available only for IPv4 based cluster.

 *It is recommended to use TCP instead of UDP multicasting for production systems because of the inherent reliability of the TCP protocol.*

To modify the JBoss run.conf file:

- 1 Open the following file in a text editor:

- (Windows) [appserver root]/bin/run.conf.bat
- (UNIX) [appserver root]/bin/run.conf

- 2 In the JAVA_OPTS line, add or change the following argument:

```
-Djboss.partition.name=<partition_name>
```

Note: The value for <partition_name> can be any value that is unique to your LiveCycle cluster. Configure the same <partition_name> value on every node of the LiveCycle cluster, as in this example:

```
-Djboss.partition.name=livecycle_cluster
```

Note: You can also pass this value as a command line argument while starting JBoss server. The value should be same for all nodes in the JBoss cluster. The value should be passed either as a JVM argument in run.conf file or command line argument '-g <partition_name>.

- 3 In the JAVA_OPTS line, add or change the following argument:

```
-Dadobeidp.serverName=<server name>
```

Note: The value for <server name> can be any value; however, you must configure a unique <server name> value for each node of the LiveCycle cluster, as in this example:

- On one node of the cluster, configure the -Dadobeidp.serverName=server1 argument
- On another node of the cluster, configure the -Dadobeidp.serverName=server2 argument.

You can configure additional nodes for the LiveCycle cluster in a similar manner but with unique <server name> values.

- 4 In the JAVA_OPTS line, the following argument might already be set for IPv4. If not, then set the argument:

```
-Djava.net.preferIPv4Stack=true
```

For IPv6, Remove -Djava.net.preferIPv4Stack=true and add the following arguments:

```
-Djava.net.preferIPv6Addresses=true
```

```
-Djava.net.preferIPv6Stack=true
```

- 5 Configure a JVM argument for cluster caching. In the JAVA_OPTS line, add or change one of the following arguments:

Caching using UDP discovery

- Configure the multicast port argument in the following format:

```
-Dadobe.cache.multicast-port=<port number>
```

Note: The value for <port number> can be any available port between 1025 and 65535. The multicast port must be unique to the LiveCycle cluster (that is, the port must not be used by any other cluster on the same network, any attempt to use the same port by any other cluster on the same network would result in bootstrap failure). It is recommended that you configure the same <port number> on all nodes in the LiveCycle cluster, as in this example:

```
-Dadobe.cache.multicast-port=33456
```

- Setting multicast address argument is optional. Default muticast addresses for IPv4 and IPv6 are as following:

```
IPv6 - FF38::1234
IPv4 - 239.192.81.1
```

If you have restriction on multicast addresses in your network, use following argument to set multicast addresses:

```
-Dadobe.cache.multicast-address=<ip address>
```

The value for <ip address> is the IP address used for multicast networking. The IP address is ignored if adobe.cache.multicast-port is zero.

The multicast address must be unique to the LiveCycle cluster and must not be used by any other cluster on the same network. It is recommended that you configure the same <ip address> on all nodes in the LiveCycle cluster. For example:

```
-Dadobe.cache.multicast-address=239.192.81.1
```

- For machines with multiple Network Interfaces

Some machines may be connected to multiple networks via multiple Network Interface Cards (NICs). For such machines, set the JVM property -Dadobe.cache.bind-address to the IP address of the network interface card that you are using for LiveCycle Server.

```
-Dadobe.cache.bind-address=<IP Address>
```

Note: It is recommended to set JVM property -Dadobe.cache.bind-address for machines with one Network Interface Card, also.

Caching using TCP only

- For IPv4, configure the cluster locators argument in the following format:

```
-Dadobe.cache.cluster-locators=<IPaddress>[<port number>],<IPaddress> [ <port number>]
```

For IPv6, configure the cluster locators argument in the following format:

```
-Dadobe.cache.cluster-locators=<hostname>@<IPv6 address>[<port number>],<hostname>@<IPv6 address>[<port number>]
```

Note: Configure, as a comma-separated list, the locators for all nodes of the cluster. The value for <IPaddress> is the IP address of the computer that is running the locator. The value for <port number> is any unused port between 1025 and 65535. It is recommended that you configure the same <port number> on all nodes in the LiveCycle cluster, as in this example:

```
-Dadobe.cache.cluster-locators=10.20.30.5[22345],10.20.30.6[22345]
```

Note: Do not run TCP locators on all the nodes of the LiveCycle Cluster. It is good practice to run minimum two locators. One TCP locator serves as a primary locator and other TCP locator serves as a secondary locator to handle failover issues. You can add more than two TCP locators as backup locators but it is not mandatory. For information on configuring TCP locators, see “4.4 Configuring the caching locators in clusters (caching using TCP only)” on page 20.

- 6 To prevent application server from Denial of Service attacks configure the following JVM argument:

```
-DentityExpansionLimit=10000
```

- 7 Save the edited file.
- 8 Repeat steps 1 to 6 for each node in the cluster.

3.5 Configuring LiveCycle database connectivity

You must enable database connectivity from each JBoss Application Server in the cluster to the LiveCycle database by performing the following tasks:

- Ensure that the correct JDBC driver exists on each instance of JBoss Application Server in the cluster.
- Create a data source file and deploy it to each instance of JBoss Application Server in the cluster. The adobe-ds.xml file configures the data source that is used by LiveCycle, including parameters such as the host name of the computer where the database resides, the database name, port number, and the user name and password for the database.

You can simplify this task by following these steps:

- 1 Copy the necessary files from your LiveCycle installation medium to any computer.
- 2 Edit the files as described in the following subsections.
- 3 Save the edited files to each node of the cluster.

See one of the following sections for instructions that are relevant to your database:

- “3.5.1 Configuring Oracle for Adobe-preconfigured JBoss” on page 10
- “3.5.2 Configuring SQL Server for Adobe-preconfigured JBoss” on page 12
- “3.5.3 Configuring MySQL for Adobe pre-configured JBoss” on page 14

3.5.1 Configuring Oracle for Adobe-preconfigured JBoss

To enable JBoss to connect to the Oracle database that stores LiveCycle data, you need the following files supplied with Adobe-preconfigured JBoss:

- Oracle JDBC driver file at `[appserver_root]/server/lc_oracle_cl/lib`
- Adobe data source file at `[appserver_root]/server/lc_oracle_cl/deploy`
- Oracle data source file at `[appserver_root]/server/lc_oracle_cl/deploy`

Note: Encrypt the password in the data source files using one of the methods described at <http://community.jboss.org/wiki/EncryptingDataSourcePasswords>. You can also use the instructions available on http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_clearte.html.

3.5.1.1 Configuring the data source files

Before you configure the Oracle data source, you must have already created the database on Oracle. (See Preparing to Install LiveCycle (Server Cluster).)

To modify the Adobe data source file:

- 1 Open the `[appserver root]/server/lc_oracle_cl/deploy/adobe-ds.xml` file in a text editor and locate the following lines:

```
<connection-url>jdbc:oracle:thin:@localhost:1521:adobe</connection-url>
<user-name>adobe</user-name>
<password>adobe</password>
```

- 2 Replace the following text with values that are specific to your database:
 - *localhost*: The name, IP address, or fully qualified path of the computer that hosts the database. The default is *localhost*.
 - *1521*: The port that is used to access the database. The default port is *1521*.
 - *adobe*: The System ID (SID) of the Database instance that stores the LiveCycle data. You will need to update the default value *adobe* with your database System ID.
- 3 In the `<user-name>` and `<password>` tags, specify the user name and password that the application server uses to access the database. You will need to update the default values *adobe* and *adobe* with the credentials for your database.
- 4 Repeat steps 1 to 3 for the remaining elements in *IDP_DS* and *EDC_DS*.
- 5 Save the file.

To modify the Oracle data source file:

- 1 Open the `[appserver root]/server/lc_oracle_cl/deploy/oracle-ds.xml` file in a text editor and locate these lines:

```
<connection-url>jdbc:oracle:thin:@localhost:1521:adobe</connection-url>
<user-name>adobe</user-name>
<password>adobe</password>
```

- 2 Replace the following text with values that are specific to your database:
 - *localhost*: The name, IP address, or fully qualified path of the computer that hosts the database. The default is *localhost*.
 - *1521*: The port that is used to access the database. The default port is *1521*.
 - *adobe*: The System ID (SID) of the Database instance that stores the LiveCycle data. You will need to update the default value *adobe* with your database System ID.
- 3 In the `<user-name>` and `<password>` tags, specify the user name and password that the application server uses to access the database. You will need to update the default values *adobe* and *adobe* with the credentials for your database.
- 4 Save the file.

3.5.1.2 Editing the login-config.xml file

- 1 Open the `[appserver root]/server/lc_oracle_cl/conf/login-config.xml` file in a text editor and modify the following code within the `<policy>` element:

```
<application-policy name="OracleDbRealm">
  <authentication>
    <login-module
      code="org.jboss.resource.security.ConfiguredIdentityLoginModule" flag
      = "required">
      <module-option name="principal">adobe</module-option>
      <module-option name="userName">adobe</module-option>
      <module-option name="password">adobe</module-option>
      <module-option
        name="managedConnectionFactoryName">jboss.jca:service=LocalTxCM,
        name=Default DS </module-option>
      </login-module>
    </authentication>
  </application-policy>
```

- 2 Replace the **bold text** (for parameters principal,userName and password) with values that are specific to your database so that the application server can access your database.
- 3 Save and close the file.
- 4 Restart JBoss.

3.5.2 Configuring SQL Server for Adobe-preconfigured JBoss

To enable JBoss to connect to the SQL Server database that stores LiveCycle data, you need the following files supplied with Adobe-preconfigured JBoss:

- SQL Server JDBC driver file at `[appserver_root]/server/lc_sqlserver_cl/lib`
- Adobe data source file at `[appserver_root]/server/lc_sqlserver_cl/deploy`
- SQL Server data source file at `[appserver_root]/server/lc_sqlserver_cl/deploy`

Note: Encrypt the password in the data source files using one of the methods described at <http://community.jboss.org/wiki/EncryptingDataSourcePasswords>. You can also use the instructions available on http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_clearite.html.

3.5.2.1 Configuring the data source files

Before you configure the SQL Server data source, you must have already created the LiveCycle database on SQL Server. (See [Preparing to Install LiveCycle \(Server Cluster\)](#).)

To modify the Adobe data source file:

- 1 Open the `[appserver_root]/server/lc_sqlserver_cl/deploy/adobe-ds.xml` file in a text editor and locate these lines:

```
<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>
<user-name>adobe</user-name>
<password>adobe</password>
```

- 2 Replace the following text with values that are specific to your database:
 - **localhost**: The name, IP address, or fully qualified path of the computer that hosts the database. The default is `localhost`.
 - **1433**: The port that is used to access the database. The default port is `1433`.
 - **adobe**: The name of the database that stores the LiveCycle data. You will need to update the default value `adobe` with your database name.

- 3 In the `<user-name>` and `<password>` tags, specify the user name and password that the application server uses to access the database. You will need to update the default values `adobe` and `adobe` with the credentials for your database.
- 4 Repeat steps 1 to 3 for the remaining elements in `IDP_DS` and `EDC_DS`.
- 5 Save the file.

To modify the SQL Server data source file:

- 1 Open the `[appserver root]/server/lc_sqlserver_cl/deploy/mssql-ds.xml` file in a text editor and locate these lines:

```
<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>  
<user-name>adobe</user-name>  
<password>adobe</password>
```
- 2 Replace the following text with values that are specific to your database:
 - `localhost`: The name, IP address, or fully qualified path of the computer that hosts the database. The default is `localhost`.
 - `1433`: The port that is used to access the database. The default port is `1433`.
 - `adobe`: The name of the database that stores the LiveCycle data. You will need to update the default value `adobe` with your database name.
- 3 In the `<user-name>` and `<password>` tags, specify the user name and password that the application server uses to access the database. You will need to update the default values `adobe` and `adobe` with the credentials for your database.
- 4 Save the file.

To configure Integrated Security on Windows:

- 1 Modify the `adobe-ds.xml` file, located in `[appserver root]/server/all/deploy`, to add `integratedSecurity=true` to the connection URL, as in this example:

```
jdbc:sqlserver://<serverhost>:<port>;databaseName=<dbname>;integratedSecurity=true.
```
- 2 Add the `sqljdbc_auth.dll` file to the Windows systems path (C:\Windows) on the computer that is running JBoss. The `sqljdbc_auth.dll` file is located with the Microsoft SQL JDBC 3.0 driver installation (default is `<InstallDir>/sqljdbc_3.0/enu/auth/x86`).
- 3 Open the properties for the JBoss for Adobe LiveCycle ES4 service and click the **Log On** tab.
- 4 Select **This Account** and type the value of a valid user account. This change is not required if you are running JBoss from the command line.
- 5 Change SQL Server's Security from Mixed mode to Windows Authentication only.

3.5.2.2 Editing the login-config.xml file

- 1 Open the `[appserver root]/server/lc_sqlserver_cl/conf/login-config.xml` file in a text editor and modify the following code within the `<policy>` element:

```
<application-policy name="MSSQLDbRealm">
  <authentication>
    <login-module
      code="org.jboss.resource.security.ConfiguredIdentityLoginModule" flag
      = "required">
      <module-option name="principal">adobe</module-option>
      <module-option name="userName">adobe</module-option>
      <module-option name="password">adobe</module-option>
      <module-option
        name="managedConnectionFactoryName">jboss.jca:service=LocalTxCM,
        name=Default DS </module-option>
      </login-module>
    </authentication>
  </application-policy>
```

- 2 Replace the **bold text** (for parameters principal,userName and password) with values that are specific to your database so that the application server can access your database.
- 3 Save and close the file.
- 4 Restart JBoss.

3.5.3 Configuring MySQL for Adobe pre-configured JBoss

Note: Encrypt the password in the data source files using one of the methods described at <http://community.jboss.org/wiki/EncryptingDataSourcePasswords>. You can also use the instructions available on http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_clearte.html.

3.5.3.1 Edit adobe-ds.xml file

Before configuring the MySQL data source, you must have already created the database on MySQL.

- 1 Open the `[appserver root]/server/lc_mysql_cl/deploy/adobe-ds.xml` file in a text editor and locate this line for both IDP_DS and EDC_DS:

```
<connection-url>jdbc:mysql://localhost:3306/adobe</connection-url>
<user-name>adobe</user-name>
<password>adobe</password>
```

- 2 Replace the following text in the file with values that are specific to your database:
 - **localhost**: The name, IP address, or fully-qualified path of the computer that hosts the database. The default is localhost.
 - **3306**: The port used to access the database. The default port is 3306.
 - **adobe**: The name of the database that stores the data. Replace the default value, `adobe`, with your database name.
- 3 In the lines that follow the `<connection-url>` settings, locate the `user-name` and `password` settings and replace the default values with the user name and password that the application server uses to access your database.
- 4 Ensure that the minimum and maximum values for the data source connections are set as follows:
 - For IDP_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```
 - For EDC_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>20</max-pool-size>
```

Note: If your LiveCycle Server handles heavy load, increase the maximum number of JDBC connections to ensure that all jobs are processed. In such cases, increase `<max-pool-size>` to 50 or more for both IDP_DS and EDC_DS.

- 5 Save and close the file.

3.5.3.2 Edit mysql-ds.xml file

If you are running LiveCycle with a MySQL database, you must set MySQL as the default data source for JBoss. This procedure assumes that the MySQL JDBC driver is installed in the `[appserver root]/server/lc_mysql/lib` directory.

- 1 Open the `[appserver root]/server/lc_mysql_cl/deploy/mysql-ds.xml` file in a text editor and modify the `<local-tx-datasource>` element with your MySQL connection settings:

```
<jndi-name>DefaultDS</jndi-name>
<connection-url>jdbc:mysql://localhost:3306/adobe</connection-url>
<user-name>adobe</user-name>
<password>adobe</password>
```

- 2 Replace the following text in the file with values that are specific to your database:
 - **localhost**: Replace this value with the name of the server hosting the database.
 - **3306**: Replace this value with the port number for your database server.
 - **adobe**: Replace this value with the database that will connect with LiveCycle.
- 3 In the lines that follow the `<connection-url>` settings, locate the `user-name` and `password` settings and replace the default values with the user name and password that the application server uses to access your database.
- 4 Save and close the file.

3.5.3.3 Edit login-config.xml file

- 1 Open the `[appserver root]/server/lc_mysql/conf/login-config.xml` file in a text editor and modify the following code within the `<policy>` element:

```
<application-policy name="MySqlDbRealm">
  <authentication>
    <login-module
      code="org.jboss.resource.security.SecureIdentityLoginModule" flag
      = "required">
      <module-option name="principal">adobe</module-option>
      <module-option name="userName">adobe</module-option>
      <module-option name="password">adobe</module-option>
      <module-option
        name="managedConnectionFactoryName">jboss.jca:service=LocalTxCM,
        name=Default DS </module-option>
    </login-module>
  </authentication>
</application-policy>
```

- 2 Replace the **bold text** (for parameters principal, userName and password) with values that are specific to your database so that the application server can access your database.
- 3 Save and close the file.
- 4 Start JBoss.

3.6 Testing the JBoss Application Server cluster

You can test the JBoss Application Server cluster to ensure that all members are active and that the cluster operates according to your design. You should ensure that the JBoss Application Server cluster functions correctly before you proceed with installing and configuring LiveCycle.

To test the JBoss Application Server cluster:

- 1 Start all JBoss Application Server instances of the cluster by entering the appropriate command:

(For Windows) `run.bat -g <partition_name> -b <ipaddress or hostname> -c <server_profile>`

(For Unix based environment) `run.sh -g <partition_name> -b <ipaddress or hostname> -c <server_profile>`

Note: For IPv6, in the commands above, use the IPv6 address or a host name mapped to an IPv6 address in the host file of the system.

Note: When you start JBoss Application Server instances, to bind to all addresses on the computer (including the local host), you can specify `-b 0.0.0.0` instead of the IP address or host name.

For clusters, it is recommended that you bind to a particular IP address and not all IP addresses.

- 2 View the `server.log` file located in `[appserver root]/server/<server_profile>/log`. Messages such as this one confirm the active members of the cluster:

```
INFO [org.jboss.ha.framework.interfaces.HAPartition.DefaultPartition]
Number of cluster members: 2
INFO [org.jboss.ha.framework.server.DistributedReplicantManagerImpl.
DefaultPartition] All Members : 2
([<IPAddress1>:<Port1>], [<IPAddress2>:<Port2>])
```

Chapter 4: Installing LiveCycle modules

4.1 Before you begin

4.1.1 Installation overview

Before you install the modules, ensure that your environment includes the software and hardware that is required to run LiveCycle. You should also understand the installation options and have the environment prepared as required. For more information, see the Preparing to Install (Singer Server or Server Cluster) or Preparing to Upgrade guide. The complete LiveCycle documentation is available at http://www.adobe.com/go/learn_lc_documentation_11.

LiveCycle also provides a command line interface (CLI) for the installation program. See “[Appendix - Install Command Line Interface](#)” on page 76 for instructions on using the CLI. There is also a CLI for Configuration Manager. See “[Appendix - Configuration Manager Command Line Interface](#)” on page 79. These CLIs are intended to be used by advanced users of LiveCycle, in server environments that do not support the use of the graphical user interface of the installation program or of Configuration Manager, or for users who wish to implement batch (non-interactive) installation capabilities.

4.1.2 Checking the installer

Observe the following best practices with the installer files before you begin the installation process.

Check the DVD installation media

Ensure that the installation media that you received is not damaged. If you copy the installation media contents to the hard disk of your computer where you are installing LiveCycle, ensure that you copy the entire DVD contents on to the hard disk. To avoid installation errors, do not copy the DVD install image to a directory path that exceeds the Windows maximum path length limit.

Install LiveCycle either by using a local copy of the installation files or directly from the DVD. The installation could fail when LiveCycle is installed over the network. Also, do not use special characters in the local path (for example, the character ‘#’).

Check the downloaded files

If you downloaded the installer from the Adobe web site, verify the integrity of the installer file using the MD5 checksum. Do one of the following to calculate and compare the MD5 checksum of the downloaded file with the checksum published on the Adobe download web page:

- **Linux:** Use the `md5sum` command
- **Solaris:** Use the `digest` command
- **Windows:** Use a tool such as WinMD5

Expand the downloaded archive files

If you downloaded the ESD from the Adobe web site, extract the entire `lces_server_11_0_0_jboss_all_win.zip` (Windows) or `lces_server_11_0_0_jboss_all_unix.tar.gz` (Linux or Solaris) archive file to your computer. For Solaris, use the `gunzip` command to extract the `.gz` file.

Note: Be sure to keep the directory hierarchy unchanged from the original ESD file.

4.2 Installation considerations

4.2.1 Installation paths

To successfully install, you need read, write, and execute permissions on the installation directory. The following installation directories are the defaults; however, you can specify a different directory as required:

- (Windows) C:\Adobe\Adobe LiveCycle ES4
- (Linux or Solaris) /opt/adobe/adobe_lifecycle_es4

Important: When installing LiveCycle, do not use double byte or extended latin characters (such as àâçèèëïïòùûÄÖßÜ) in the installation path.

Important: (Windows only) The LiveCycle installation directory path must not contain any non-ASCII characters (for example, international characters such as é or ñ), otherwise the JBoss Service for Adobe LiveCycle ES4 will fail to start.

When you are installing the modules on UNIX-based systems, you must be logged in as the root user to successfully install the modules to the default location, which is /opt/adobe/adobe_lifecycle_es4. If you are logged in as a non-root user, change the installation directory to one that you have permissions (read-write-execute privileges) for. For example, you can change the directory to /home/[username]/adobe/adobe_lifecycle_es4.

Note: On a UNIX-like system, when you copy/download files from the source (installation media), install.bin might lose the executable permissions. Ensure that you restore the write-execute permissions after copying/downloading the files.

On Windows, you must have administrator privileges to install LiveCycle.

4.2.2 Temporary directories

Temporary files are generated in the temp directory. In certain instances, the generated temporary files may remain after the installer is closed. You can remove these files manually.

When installing on Linux, the installation program uses the logged-in user's home directory as a temporary directory for storing files. As a result, messages such as the following text may appear in the console:

```
WARNING: could not delete temporary file /home/<username>/ismp001/1556006
```

When you complete the installation, you must manually delete the temporary files from the following directories:

- (Windows) TMP or TEMP path as set in the environment variables
- (Linux or Solaris) Logged-in user's home directory

On UNIX-based systems, a non-root user can use the following directory as the temporary directory:

- (Linux) /var/tmp or /usr/tmp
- (Solaris) /var/tmp or /usr/tmp

4.2.3 Installing on a Windows staging platform for Linux or UNIX

LiveCycle can be installed and configured on Windows for deployment on a Linux or UNIX platform. You can use this functionality for installing on a locked-down Linux or UNIX environment. A locked-down environment does not have a graphical user interface installed. For the Linux or UNIX platform, the installation program installs binaries that are used by Configuration Manager to configure the product.

The computer running Windows can then be used as a staging location for the deployable objects, which can be copied to a Linux or UNIX computer for deployment to the application server. The application server on the Windows-based computer, and the Linux or UNIX target computer on which you want to install LiveCycle must be the same.

4.2.4 Configuring the JAVA_HOME environment variable

The JAVA_HOME environment variable must point to the Java SDK for your application server as outlined in the preparing guide. See [Preparing to Install LiveCycle \(Single Server\)](#) or [Preparing to Install LiveCycle \(Server Cluster\)](#) for more information

4.2.5 General installation notes

- On Windows, improve the speed of installation by disabling any on-access virus scanning software during installation.
- If you are installing on UNIX-based systems and are not installing directly from a release DVD, set executable permissions on the installation file.
- To avoid permission issues during deployment, ensure that you run the LiveCycle installer and Configuration Manager as the same user who will run the application server.
- If you are installing on UNIX-based computers, the installation directory you specify should not contain any spaces.
- If errors occur during installation, the installation program creates the install.log file, which contains the error messages. This log file is created in the *[LiveCycle root]/log* directory.
- Ensure that the JAVA_HOME environment variable points to the directory that contains a compatible JDK. See the [supported platform combinations](#) for more information.

4.3 Installing LiveCycle

1 Start the installation program:

- (Windows) Navigate to the \server\Disk1\InstData\Windows_64\VM directory on the installation media or folder on your hard disk where you copied the installer. Right-click the install.exe file and select Run as administrator.

***Note:** 32-bit version of LiveCycle is also available. Navigate to the corresponding directory and choose the installation file to launch the installer. However, note that the 32-bit version is supported only for development or evaluation purposes, and not for a production environment.*

- (Non-Windows) Navigate to the appropriate directory, and from a command prompt, type ./install.bin.
 - (Linux) /server/Disk1/InstData/Linux/NoVM
 - (Solaris) /server/Disk1/InstData/Solaris/NoVM

2 When prompted, select the language for the installation to use and click **OK**.

3 On the Introduction screen, click **Next**.

4 If you have a previous version of LiveCycle ES2, ADEP, or LiveCycle ES3 installed on the computer where you are running the installer, the Preparation for Upgrade screen appears.

***Note:** If you are performing an out-of-place upgrade on a new machine, this screen is not shown.*

- **Prepare to upgrade existing installation to Adobe LiveCycle ES4:**
Do not select this option if you are performing a fresh installation.
- **Install Adobe LiveCycle ES4:** Installs LiveCycle afresh.

Select **Next** to continue.

- 5 On the Choose Install Folder screen, accept the default directory or click **Choose** and navigate to the directory where you intend to install LiveCycle, and then click **Next**. If you type the name of a directory that does not exist, it is created for you.
Click Restore Default Folder to restore the default directory path.
- 6 On the Choose Installation Type screen, select **Custom > Manual**, and click **Next**.
- 7 **(Windows only)** On the Manual Installation Options screen, select the target deployment option and click **Next**:
 - **Windows (Local)**: Select this option if you are installing and deploying LiveCycle on the local server.
 - **Staged (Installed on Windows, targeting remote systems)**: Select this option if you plan to use Windows as a staging platform for your deployment and then select the target operating system on the remote server. You can select a UNIX operating system as the target for deployment even if you are installing on Windows. (See “[4.2.3 Installing on a Windows staging platform for Linux or UNIX](#)” on page 18.)
- 8 Read the Adobe LiveCycle ES4 License Agreement, select **I accept** to accept the terms of the license agreement and then click **Next**. If you do not accept the license agreement, you cannot continue.
- 9 On the Pre-Installation Summary screen, review the details and click **Install**. The installation program displays the progress of the installation.
- 10 Review the Release Notes information and click **Next**.
- 11 Review the details on the Install Complete screen.
- 12 The **Start LiveCycle Configuration Manager** checkbox is selected by default. Click **Done** to run the Configuration Manager.

Note: (Adobe® LiveCycle® PDF Generator 11 for Windows only) If Acrobat is not installed on all nodes in the cluster, install it now. Then complete the steps in “[6.6 Configuring PDF Generator](#)” on page 45.

Note: To run Configuration Manager later, deselect the **Start LiveCycle Configuration Manager** option before you click **Done**. You can start Configuration Manager later using the appropriate script in the [LiveCycle root]/configurationManager/bin directory. See the *Configuring LiveCycle For Deployment* chapter in this guide.

4.4 Configuring the caching locators in clusters (caching using TCP only)

If you implement caching for your LiveCycle cluster by using TCP, configure the TCP locators to find other members of the LiveCycle cluster.

Note: This section does not apply if you implement caching for your LiveCycle cluster by using UDP (See “[3.4 Modifying the JBoss run.conf file](#)” on page 7 to configure caching for your LiveCycle cluster using UDP.)

Do the following to enable LiveCycle cluster caching using TCP:

- Ensure that the TCP locators are installed and configured. TCP locators are installed in the [LiveCycle root]/lib/caching directory, with a default configuration, when you install LiveCycle. You can change the default configuration. (See *Modifying the TCP locators*.)
- Configure each node in the LiveCycle cluster to use the locators. (See “[3.4 Modifying the JBoss run.conf file](#)” on page 7)
- Ensure that TCP locators are running.

4.4.1 Modifying TCP locators

The LiveCycle installer creates a default configuration of the TCP locators that is ready to use without modification. You can move the locators to any computer on your network and run them on that computer. The locators do not have to reside on a computer that is a member of the LiveCycle cluster. You can also create additional failover locators to support high availability in your cluster. (See To install the TCP locators:.)

You can also modify the TCP locators to use a port other than the default port (22345). (See To modify the default locator port (Windows): or To modify the default locator port (UNIX):.)

4.4.2 Install TCP locators

- 1 Log on to the computer where you installed LiveCycle and navigate to the `[LiveCycle root]/lib/` caching directory.
- 2 Copy the caching directory and its contents to the computer on which you want to run the locators.

You can start TCP locator from default location. You should copy the caching directory to another location, only if:

- You want to run TCP locator on a machine that does not have LiveCycle.
- You do not want to start TCP locator from default location.

Note: Do not run TCP locators on all the nodes of the LiveCycle Cluster. It is good practice to run minimum two locators. One TCP locator serves as a primary locator and other TCP locator serves as a secondary locator to handle failover issues. You can add more than two TCP locators as backup locators but it is not mandatory.

4.4.3 Modify the default locator port (Windows)

- 1 Open the `startlocator.bat` file in a text editor. The `startlocator` file for a default installation is on the computer where you installed LiveCycle, in the `[LiveCycle root]/lib/caching` directory.
- 2 Change the default port number (22345) to your preferred port number in the following properties:

```
set port=22345
```

The port number can be any available port between 1025 and 65535.

Note: Ensure that the port number that is configured here matches the port number that is configured in the JVM argument of each node of the LiveCycle cluster. The port number can be any available port between 1025 and 65535. See “3.4 Modifying the JBoss `run.conf` file” on page 7 to complete the configuration.

- 3 If you are using more than one locator for failover, assign all of such locators to JVM argument `-Dlocators` listed at the end of the `startlocator.bat` file.

```
-Dlocators=localhost [22345]
```

- 4 (Computers with multiple network cards only) If the computer hosting the locator has multiple network cards, set the following properties in the script:

```
set bindaddr=<bind IP address>
```

Where `<bind IP address>` is the IP address that the locator will listen on. You must specify the `<bind IP address>` for the JVM argument `adobe.cache.cluster-locators` on each node in your LiveCycle cluster.

Note: If you do not specify the bind address and the bind port in the `startlocator` script, you will be prompted to input these values when you execute the script. However, for IPv6, you must specify the bind address and the bind port in the `startlocator` script itself.

- 5 Save the edited file.
- 6 Repeat steps 1 to 4 on any additional locators for your LiveCycle cluster.

4.4.4 Modify the default locator port (UNIX)

- 1 Open the startlocator.sh file in a text editor. The startlocator file for a default installation is located on the computer where you installed LiveCycle, in the [LiveCycle root]/lib/caching directory.
- 2 Change the default port number (22345) to your preferred port number in the following properties:

```
GF_PORT=22345
```

The port number can be any available port between 1025 and 65535.

Important: Ensure that the port number that is configured here matches the port number that is configured in the JVM argument of each node of the LiveCycle cluster.

- 3 If you are using more than one locator for failover, assign all of such locators to JVM argument -Dlocators listed at the end of the startlocator.sh file.

```
-Dlocators=localhost [22345]
```

- 4 (Computers with multiple network cards only) If the computer hosting the locators has multiple network cards, modify the following argument:

```
GF_BIND_ADDRESS="<bind IP address>"
```

Where <bind IP address> is the IP address that the locator will listen on. You must specify the <bind IP address> for the JVM argument adobe.cache.cluster-locators on each node in your LiveCycle cluster.

Note: For IPv6, it is recommended that you specify the bind address and the bind port in the startlocator script itself.

- 5 Save the edited file.
- 6 Repeat steps on any additional locators for your LiveCycle cluster.

4.4.5 Start the TCP locators

To use TCP-based caching for LiveCycle Cluster using TCP locator, you must start the TCP locators before you start your cluster. If the TCP locators are not running when you start the members of the LiveCycle cluster, the LiveCycle cluster will not function.

- 1 On the computer where the TCP locators are installed, navigate to the caching directory. For a default installation, the TCP locators are installed on the computer where you installed LiveCycle, in the [LiveCycle root]/lib/caching directory.
- 2 (IPv6 only) Modify startlocator.bat (Windows) or startlocator.sh (UNIX) and add the following JVM arguments:

```
-Djava.net.preferIPv6Stack=true  
-Djava.net.preferIPv6Addresses=true
```

- 3 Run the appropriate file:

- (Windows) startlocator.bat
- (UNIX) startlocator.sh

- 4 Repeat above steps on any additional locators for your LiveCycle cluster.

Note: (Windows Only) On running startlocator script, you would be prompted to change the default value. You can choose to keep the default values provided in the script or you can provide new values.

4.4.6 Stop TCP locators

- 1 On the computer where the TCP locators are installed, navigate to the caching directory. For a default installation, the TCP locators are installed on the computer where you installed LiveCycle, in the *[LiveCycle root]/lib/caching* directory.
- 2 Run the appropriate file:
 - (Windows) `stoplocator.bat`
 - (UNIX) `stoplocator.sh`
- 3 Repeat steps 1 to 2 on any additional locators for your LiveCycle cluster.

Note: If you are not using the default values in the `startlocator` script and mentioned specific IP address and port values, specify the same values in the `stoplocator` script. Otherwise, the `stoplocator` script may fail to stop the locators.

4.5 Global Document Storage Directory (GDS)

On cluster configurations, create a shared file system for GDS directory accessible to each node on the LiveCycle cluster. This shared file system can be a local storage on a computer or a share on the dedicated network storage system. Ensure that all the nodes of the cluster have read and write permissions on the shared storage. The GDS directory should have low access time, high up time, and should be accessible as UNC style path. For example, `\\storagename\shared\GDS`.

4.6 Installing the font directories in cluster nodes

You must install the font directories for each node in the cluster, including the LiveCycle fonts that are installed in the *[LiveCycle root]/fonts* directory.

The fonts must exist in the same path on each node, and the directory must have identical contents on all nodes in the cluster. To ensure this, use one of the following options:

- Use a shared directory that all nodes in the cluster can access. Copy fonts from *[LiveCycle root]/fonts* to the shared directory. Using shared fonts directories can slow down the access to the fonts and induces performance issues.
- Copy the *[LiveCycle root]/fonts* directory to each node in the cluster in an identical path.

Record the location where you create these shared directories for later use when you configure LiveCycle using Configuration Manager.

Note: The font directories must be distinct from the GDS directory. However, they may be distinct sibling subdirectories of a single shared parent directory. The requirements and steps mentioned above are applicable to the custom font directories.

4.7 Next steps

You must now configure LiveCycle for deployment. You can also choose to run Configuration Manager later by using the `ConfigurationManager.bat` or `ConfigurationManager.sh` file located in *[LiveCycle root]/configurationManager/bin*.

Chapter 5: Configuring LiveCycle for deployment

5.1 Considerations when configuring and deploying LiveCycle

5.1.1 General Considerations

- For IPv6, run IPv6 LiveCycle Configuration Manager. For detailed information, see LiveCycle IPv6 support section in [prepare to install guide](#).
- You can override the default font for the Configuration Manager by adding the following JVM argument in [LiveCycle root]\configurationManager\bin\ConfigurationManager.bat (Windows) or [LiveCycle root]\configurationManager\bin\ConfigurationManager.sh (Linux, UNIX):

```
-Dlcm.font.override=<FONT_FAMILY _NAME>
```

For example:

```
-Dlcm.font.override=SansSerif
```

Restart the Configuration Manager after adding the JVM argument.

- During configuration, if you must reset the data in Configuration Manager by using the Reset to Default option, ensure that you relaunch the Configuration Manager. Otherwise, you may encounter some missing configuration screens.
- During configuration, you must provide the location of the JDBC drivers for your database. The Oracle, SQL Server, and DB2 drivers are in the [LiveCycle root]/lib/db/[database] directory. For MySQL, download and install the necessary JDBC drivers.

If you have manually configured JBoss, the database drivers have to be downloaded and copied in the [appserver root]/server/<profile_name>/lib

- Temporary directory: Do not specify a shared network directory as your temporary directory on cluster configurations. It is recommended to use local directory as a temporary directory. The temporary directory should exist on every node of the cluster and the path of the temporary directory should be same for every node of the cluster.
- Global Document Storage (GDS) directory: Specify the GDS directory that meets the requirements outlined in the Preparing to Install (Single Server or Server Cluster). For latest documentation, see http://www.adobe.com/go/learn_lc_documentation_11.
- On cluster environments, several steps need to be performed manually in addition to the automatic configuration that Configuration Manager performs.

5.1.2 CLI versus GUI versions of Configuration Manager

This section describes the GUI version of Configuration Manager. For instructions about using the command line interface (CLI) version of Configuration Manager, see “[Appendix - Configuration Manager Command Line Interface](#)” on page 79.

LiveCycle configuration task	Configuration Manager GUI	Configuration Manager CLI	Manual
Configure LiveCycle	Yes	Yes	No
Initialize LiveCycle database	Yes	Yes	No
Validate LiveCycle Server connection	Yes	Yes	No
Deploy LiveCycle components	Yes	Yes	No
Validate LiveCycle component deployment	Yes	Yes	Yes
Configure LiveCycle components	Yes	Yes	Yes

5.1.3 Considerations for JBoss application server

The **Configure Application Server**, **Validate Application Server Configuration**, and **Deploy LiveCycle EARs** tasks are not available for JBoss.

You must configure JBoss and deploy the LiveCycle EARs manually as described in the Deploying LiveCycle to JBoss.

5.1.4 Considerations while configuring LiveCycle Server Clusters

- You cannot configure settings for IPv6-based clusters using Configuration Manager.
- It is recommended that you have local server fonts and customer fonts directories at the same path on each node in the cluster. Having shared fonts directories instead of local fonts directories may cause performance issues.

5.1.5 Set the date, time, and time zone

Setting the date, time, and time zone on all servers connected to your LiveCycle environment will ensure that time-dependent modules, such as Adobe® LiveCycle® Digital Signatures 11 and Reader Extensions 11, function correctly. For example, if a signature appears to have been created in the future, it will not validate.

Servers that require time synchronization are database servers, LDAP servers, HTTP servers and J2EE servers (Application servers).

Note: Ensure that all the machines used in a LiveCycle cluster are time synchronized.

5.2 LiveCycle pre-configuration tasks

Note: Press **F1** in Configuration Manager to view Help information for the screen you are viewing. You can view the configuration progress at any time by clicking View Progress Log.

- 1 If you did not start Configuration Manager automatically from the installation program, navigate to the [LiveCycle root]/configurationManager/bin directory and run the ConfigurationManager.bat/sh script.
- 2 If prompted, select a language for Configuration Manager to use and click **OK**.
- 3 On the Welcome screen, click **Next**.
- 4 Do not select any option on the Upgrade Task Selection screen and click **Next**.
- 5 On the Modules screen, select Adobe LiveCycle ES4 modules you wish to configure and click **Next**.

Note: Some modules have technical dependencies on other modules for proper configuration and functioning. Configuration Manager displays a dialog and does not allow to proceed further if mutually dependent modules are not selected. For example, you must select Adobe LiveCycle Forms, Adobe LiveCycle Output, and Content Repository modules if you are configuring Correspondence Management Solution.

- 6 On the Task Selection screen, select all the tasks you want to perform and click **Next**.

Note: Tasks Configure Application Server, Validate Application Server, and Deploy LiveCycle ES4 EAR files are not available for JBoss application server.

5.3 Configuring and deploying LiveCycle

Note: Press **F1** in Configuration Manager to view Help information for the screen you are viewing.

Configuring LiveCycle

- 1 On the Configure LiveCycle ES4 (1 of 5) screen, click **Configure** and click **Next** when done.
- 2 On the Configure LiveCycle ES4 (2 of 5) screen, set the directories that LiveCycle will use to access fonts and then click **Next**.



Click **Edit configuration** to change any values on this screen. This button is not available when the Configuration Manager is run for the first time, but is available on the second and subsequent runs of the Configuration Manager.

- (Optional) To change the default location of the **Adobe server fonts directory**, type the path or browse to the directory.
- To change the default location of the **Customer fonts directory**, click **Browse** or specify a new location for your customer fonts.

Note: Your right to use fonts provided by parties other than Adobe is governed by the license agreements provided to you by such parties with those fonts, and is not covered under your license to use Adobe software. Adobe recommends that you review and ensure that you are in compliance with all applicable non-Adobe license agreements before using non-Adobe fonts with Adobe software, particularly with respect to use of fonts in a server environment.

- (Optional) To change the default location of the **System fonts directory**, type the path or browse to the directory. To add more directories to the list, click **Add**.
 - (Optional) To enable FIPS, ensure that **Enable FIPS** is selected. Select this option only if you require the Federal Information Processing Standards (FIPS) to be enforced.
- 3 Click **Browse** on the Configure LiveCycle ES4 (3 of 5) screen to specify the **Location of the temporary directory**.

Note: Ensure that the temporary directory is on the local file system. LiveCycle does not support a temporary directory at a remote location.

Note: If you do not specify the temporary directory, the default system-configured temp location is used.

- 4 On the Configure LiveCycle ES4 (4 of 5) screen, click **Browse** to specify the path for the Global Document Storage (GDS) directory.

Note: If you leave the GDS directory field empty, LiveCycle will create the directory in a default location in the application server directory tree. After you finish the configuration steps, you can access the location from Administration Console > Settings > Core System Settings > Configurations.

***Note:** Ensure that GDS directory is accessible from all the nodes of the cluster. For Cluster, do not leave the directory field empty.*

- 5 On the Configure Persistent Document Storage (5 of 5) screen, select the option for persistent document storage in addition to the GDS directory. Select one of the following:
- **Use GDS:** Use the file system-based GDS for all persistent document storage. This option provides the best performance, and a single location for GDS.
 - **Use database:** Use the LiveCycle database for storing the persistent documents and long-lived artifacts. However, the file-system based GDS is also required. Using the database simplifies backup and restore procedures.

Click **Configure** to configure the LiveCycle EARs with this directory information and, after the configuration is complete, click **Next**.

Configuring Acrobat for PDF Generator

- ❖ **(Windows only)** On the Configure Acrobat For LiveCycle PDF Generator screen, click **Configure** to run the script that will configure Adobe Acrobat and required environment settings. Click **Next** when complete.

***Note:** This screen will perform the desired configuration only when Configuration Manager is running locally. You must have Adobe Acrobat XI Pro already installed or this step will fail.*

***Note:** You should manually configure Acrobat for PDF generator on all the other nodes of the cluster. See “[6.6 Configuring PDF Generator](#)” on page 45 in the Post Deployment chapter.*

LiveCycle Configuration Summary

- ❖ On the Configure LiveCycle ES4 Summary screen, click **Next**. Configured archives are placed in the `[LiveCycle root]/configurationManager/export` directory.

***Note:** Stop each JBoss Application Server instance in the cluster.*

Configure CRX

- ❖ The CRX Configuration screen allows you to configure the CRX repository and install it into the LiveCycle Core EAR file. On this screen, specify the path to the repository and click **Configure** to create the required repository files at the specified location. If you are using a custom path for the repository, ensure that the file system already contains custom directory.

***Note: (Non-turnkey only)** If your LiveCycle server is running remotely, select **Server is running on remote host**, and specify the path to the repository on the remote host.*

Click **Next** to continue.

***Note:** Ensure that the CRX repository path does not contain spaces and the content repository is available on all the nodes of the cluster. After the configuration is complete, copy the content repository from local node to all the nodes on the same location (as specified on the CRX configuration screen) .*

***Note:** Once the packages are configured, you cannot remove them by re-running the Configuration Manager. For clean uninstallation of deployed packages, you need to uninstall and delete the packages using Package Manager.*

(Remote host only) CRX Configuration Summary

- ❖ For a remote deployment, copy the content from the `[LiveCycle root]/configurationManager/export/crx-quickstart/` directory to the location on the remote host you specified on the CRX Configuration screen.

Copy CRX Content

Copy all the content of the `[LiveCycle root]/configurationManager/export/crx-quickstart/` directory to the location specified on the CRX Configuration screen on all the cluster nodes.

Deploying LiveCycle EARs

❖ Without exiting Configuration Manager, manually deploy the LiveCycle EAR files to JBoss by copying the following files from the `[LiveCycle root]/configurationManager/export` directory to directories as specified:

- `adobe-livecycle-native-jboss-[OS].ear`
- `adobe-livecycle-jboss.ear`
- `adobe-workspace-client.ear` (Adobe® LiveCycle® Process Management 11 only)

Manually-configured JBoss on Cluster `[appserver root]/server/all/deploy`

Adobe-preconfigured JBoss on Cluster `[appserver root]/server/lc_<db-name>_cl/deploy`.

These files must be copied in each cluster node.

You can optionally deploy the Adobe® LiveCycle® Forms Standard 11, Adobe® LiveCycle® Output 11, Adobe® LiveCycle® Mobile Forms, and Assembler IVS EARs as well.

To create publish instance for Correspondence Management, deploy `adobe-livecycle-cq-publish.ear`. Ensure that `adobe-livecycle-cq-publish.ear` is deployed on a different server. Do not deploy `adobe-livecycle-cq-publish.ear` on LiveCycle server. For detailed information about configuring the publish instance, see “[6.5.2 Configure the Publish instance](#)” on page 39

Note: The data source definition files have to be modified to point to the database server and database. For more information, see *Appendix - Manually Configuring Data Sources*.

Important: Deploying the IVS EAR files to a production environment is not recommended.

Start JBoss to ensure the LiveCycle applications start successfully and return to Configuration Manager.

Initializing LiveCycle database

- 1 On the LiveCycle ES4 Database Initialization screen, verify that the hostname and port number provided for your application server is correct and then click **Initialize**. The database initialization task creates tables in the database, adds default data to the tables, and creates basic roles in the database. When the initialization has completed successfully, click **Next**. Restart the application server manually when you are prompted to do so.

Note: Initialize the database against only one server of the cluster. Subsequent steps are performed only on initialized server.

- 2 On the LiveCycle ES4 Information screen, enter **LiveCycle ES4 User ID** and **Password** whose default values are *administrator* and *password* respectively.

Click **Verify Server Connection**, and when complete, click **Next**.

Note: The server information that appears on this screen represents default values for the deployment.

Verifying the server connection helps narrow troubleshooting in case failures occur in the deployment or validation. If the connection test passes but deployment or validation fails in the next few steps, connectivity issues can be eliminated from the troubleshooting process.

Deploying Central Migration Bridge Service

- ❖ On the Central Migration Bridge Service Deployment Configuration screen, if applicable, select the **Include Central Migration Bridge Service** in deployment option and then click **Next**.

Deploying LiveCycle components

- 1 On the LiveCycle ES4 Component Deployment screen, click **Deploy**. The components that are deployed at this time are Java archive files that plug into the LiveCycle service container for purposes of deploying, orchestrating, and executing services. When the deployment has completed successfully, click **Next**.
- 2 On the LiveCycle ES4 Component Deployment Validation screen, click **Validate**. When the validation has completed successfully, click **Next**.

Configuring LiveCycle components

- ❖ On the Configure LiveCycle Components screen, select the tasks to run with Configuration Manager, and click **Next**.

LiveCycle Server JNDI information

- ❖ On the LiveCycle ES4 Server JNDI Information screen, enter the host name, port number, and JBoss client jar location for the JNDI server. Press F1 for more information. Click **Verify Server Connection** to ensure that Configuration Manager can connect to the JNDI server. Click **Next** to continue.

Connector for EMC Documentum

Note: In case of a remote LiveCycle deployment, you cannot configure the Connector for EMC Documentum using Configuration Manager.

- 1 On the Specify Client for EMC Documentum screen, select **Configure Connector for EMC Documentum Content Server**, and specify the following settings. Enter the details, click **Verify**, and when complete, click **Next** to continue.
 - **Choose EMC Documentum Client Version:** Select the client version to use with the EMC Documentum Content Server.
 - **EMC Documentum Client Installation Directory Path:** Click **Browse** to select the directory path.

Note: Configure Documentum 6.7 manually, LCM does not contains support for Documentum 6.7.
- 2 On the Specify EMC Documentum Content Server Settings screen, enter the EMC Documentum Server details, and then click **Next**. Press F1 for information about the details you need to enter.
- 3 On the Configure Connector for EMC Documentum screen, click **Configure Documentum Connector**. When completed, click **Next**.
- 4 On the Required Manual Configurations for Connector for EMC Documentum screen, review and perform the manual steps listed and then click **Next**.

Connector for IBM Content Manager

Note: In case of a remote LiveCycle deployment, you cannot configure the Connector for IBM Content Manager using Configuration Manager.

- 1 On the Specify Client for IBM Content Manager screen, select **Configure Connector for IBM Content Manager**, and enter a value for the IBM Content Manager Client Installation Directory Path. Click **Verify** and when complete, click **Next** to continue.
- 2 On the Specify IBM Content Manager Server Settings screen, enter the details of the IBM Content Manager Server, and click **Next**.
- 3 On the Configure Connector for IBM Content Manager screen, click **Configure IBM Content Manager Connector**. When complete, click **Next**.
- 4 On the Required Manual Configurations for Connector for IBM Content Manager screen, review and perform the manual steps listed and then click **Next**.

Connector for IBM FileNet

Note: In case of a remote LiveCycle deployment, you cannot configure the Connector for IBM FileNet using Configuration Manager.

- 1 On the Specify Client for IBM FileNet screen, select **Configure Connector for IBM FileNet Content Manager**, and specify the following settings.

- **Choose IBM FileNet Client Version:** Select the client version that you want to use with the IBM FileNet Content Server.
- **IBM FileNet Client Installation Directory Path:** Click **Browse** to select the directory path.

Note: The validation of IBM FileNet may fail if there are special characters, such as hyphen (-), underscore (_), comma (,), or dot (.) in the directory name containing the IBM FileNet client.

Click **Verify**, and when complete, click **Next** to continue.

- 2 On the Specify IBM FileNet Content Server Settings screen, enter the required details, and click **Next**.
- 3 On the Specify Client for IBM FileNet Process Engine screen, enter the required details, and click **Verify**. When complete, click **Next**.
- 4 On the Specify IBM FileNet Process Engine Server Settings screen, enter the required details and click **Next**.
- 5 On the Configure Connector for IBM FileNet screen, click **Configure FileNet Connector**. When complete, click **Next**.
- 6 On the Required Manual Configurations for Connector for IBM FileNet screen, review and perform the manual steps listed and then click **Next**.

Connector for Microsoft SharePoint

Note: In case of a remote LiveCycle deployment, you cannot configure the Connector for Microsoft SharePoint using Configuration Manager.

On the Configure Adobe LiveCycle ES4 Connector for Microsoft SharePoint screen, do one of the following tasks:

- Deselect the **Configure Adobe LiveCycle ES4 Connector for Microsoft SharePoint** option to manually configure Microsoft Sharepoint later, and then click **Next**.
- Leave the **Configure Adobe LiveCycle ES4 Connector for Microsoft SharePoint** option selected. Enter the required values, and then click **Configure SharePoint Connector**. When complete, click **Next**.

Note: You can skip this step if you want to configure the Connector for Microsoft SharePoint later using Administration Console.

Configuring LiveCycle Server for native file conversions

- ❖ (PDF Generator only) On the **Admin user credentials for native PDF conversions** screen, enter the user name and password of a user with administrative privileges on the server computer, and then click **Add user**.

Note: You must add at least one administrative user for Windows 2008 Server. On Windows 2008 Server, User Account Control (UAC) must be disabled for the users you add. To disable UAC, click **Control Panel > User Accounts > Turn User Account Control on or off** and deselect **Use User Account Control (UAC) to help protect your computer**, then click **OK**. Restart the computer to apply these changes.

System readiness test for PDF Generator

- ❖ On the **Document Services PDF Generator System Readiness Test** screen, click **Start** to validate if the system has been appropriately configured for PDF Generator. Review the System Readiness Tool Report and click **Next**. Note that the system readiness test fails if LiveCycle is deployed on a remote machine.

Configuring Reader Extensions

- ❖ On the Reader Extensions Credential Configuration screen, specify the details that are associated with the Reader Extensions credential that activates the module services:

***Note:** You can skip this step at this time by selecting **Configure later using Administration Console**. You can configure the Reader Extensions credential by using Administration Console after you complete the deployment. (After logging in to Administration Console, click **Home** > **Settings** > **Trust Store Management** > **Local Credentials**.)*

Click **Configure** and then click **Next**.

Summary, and Next Steps

- ❖ Review the Configuration Manager task summary list and choose the appropriate options:
 - Select Launch Next Steps to view information about LiveCycle users and administrative interfaces to launch an html page containing step-by-step instructions to start and use LiveCycle.

Click **Finish** to exit the Configuration Manager

Chapter 6: Post-deployment tasks

6.1 General tasks

6.1.1 Configure Allowed Referers

When you run Configuration Manager, the default host, IPv4 address, IPv6 address, loopback address, and localhost address are added to the Allowed Referer list. These addresses are added only for the machine where LCM is executed. For a LiveCycle cluster, manually add all the other cluster nodes to the list:

- 1 In Administration Console, click **Settings > User Management > Configuration > Configure Allowed Referer URL's**. The Allowed Referer list appears at the bottom of the page.
- 2 To add an allowed referer:
 - a Type a host name or IP address in the Allowed Referers box. To add more than one allowed referer at a time, type each host name or IP address on a new line. Provide hostName and IP address of all the cluster nodes/load balancer.
 - b In the HTTP Port and HTTPS Ports boxes, specify which ports to allow for HTTP, HTTPS, or both. If you leave those boxes empty, the default ports (port 80 for HTTP and port 443 for HTTPS) are used. If you enter 0 (zero) in the boxes, all ports on that server are enabled. You can also enter a specific port number to enable only that port.
 - c Click Add.
- 3 Click **Save**.

If the Allowed Referer List is empty, the CSRF feature stops working and the system becomes insecure.

- 4 After changing the Allowed Referer list, restart the LiveCycle cluster.

6.1.2 Configure CRX Repository Clustering

Note: The Configure CRX Repository Clustering section described in this article is updated for LiveCycle ES4 Service Pack 1. If you are on the LiveCycle ES4 base release, follow the steps described in [Key Distinctions in LiveCycle ES4 from LiveCycle ES4 Service Pack 1](#).

Perform the following steps to configure CRX repository clustering:

- 1 Go to `http://[Host]:[Port]/lc/libs/granite/cluster/content/admin.html`. Login as an administrator.
- 2 Change the master URL to `http://[Master_host]:[port]/lc`. Provide hostname and port of the node that should act as a master.
- 3 Change the username/password to admin/admin and click Join. It may take some time to complete the configuration. Do not press refresh or back. On completion of configuration, a success message appears.
- 4 To connect more slave nodes, repeat steps 1-4 on each slave node. For each slave, provide the master URL mentioned in the step 4.

Note: Do not perform above steps on the master node.

On starting a cluster, ensure that the master node is started before all the slave nodes. On stopping the cluster, stop all slaves before stopping the master node. In some specific scenarios, Master node and Slave nodes can switch roles; ensure your master before stopping the cluster.

The particular start /stop cluster order is enforced for CRX clustering but since it is embedded in LiveCycle, ensure that you follow above procedure while starting and stopping LiveCycle cluster.

A slave node waits for the specified number of seconds for the master node to be up and running. If the master node is not up in specified seconds, the slave node stops its repository. To join the slave node in the cluster, restart the slave node. The default wait time for a node is 60 seconds. Use the following JVM argument to configure the number of seconds for the slave nodes:

```
-Dcom.day.crx.core.cluster.WaitForMasterRetries=<value>
```

If you start all the nodes of the cluster at once, the start order dependency fails and slave nodes of a cluster fail to start. To avoid such issues, ensure that the wait time for a node is 300 seconds or more.

Note: Restart the slave instance to avoid stale sessions.

Important: All author instances in the cluster should be time synchronized. You can use an NTP (Network Time Protocol) server to ensure time synchronization.

6.1.3 Perform a system image backup

After LiveCycle is installed and deployed into production areas and before the system is live, it is recommended that you perform a system image backup of the servers on which LiveCycle is implemented. Also take backup of CRX repository.

The LiveCycle database, GDS directory, and application servers must be part of this backup. This is a complete system backup that you can use to restore the contents of your computer if your hard drive or entire computer stops working. See the LiveCycle Backup and Recovery topic in [Administration Help](#).

6.1.4 Restart the application server

When you first deploy LiveCycle, the server is in a deployment mode in which most modules are in memory. As a result, the memory consumption is high and the server is not in a typical production state. You must restart the application server to get the server back into a clean state.

Note: You may skip to restart the LiveCycle server, if you have restarted the server after configuring CRX clustering or after updating Allowed Referer list

Note: When you upgrade the LiveCycle Server or deploy a Service pack, ensure that you delete the `[Jboss_root]\server\<server name>\work` and `[Jboss_root]\server\<server name>\tmp` folders before restarting the application server.

6.1.5 Verify the deployment

You can verify the deployment by logging in to Administration Console. If you log in successfully, then LiveCycle is running on the application server and the default user is created in the database. To verify the CRX repository deployment, access the CRX welcome page.

You can review the application server log files to ensure that components were deployed correctly or to determine the cause of any deployment issues you may encounter.

6.1.5.1 Accessing LiveCycle Administration Console

Administration Console is the web-based portal for accessing a variety of configuration pages where you can set run-time properties that control the way LiveCycle operates. When you log in to Administration Console, you can access User Management, Watched Folder, and Email client configuration, and administrative configuration options for other services. Administration Console also provides access to Applications and Services, which administrators use for managing archives and deploying services to a production environment.

The default user name and password for logging in is *administrator* and *password*. After you log in the first time, access User Management and change the password.

Before you access Administration Console, LiveCycle must be deployed and running on your application server.

For information about using Administration Console, see [Administration Help](#).

- 1 Type the following URL in a web browser:

`http://[hostname]:[port]/adminui`

For example: `http://localhost:8080/adminui`

- 2 If you have upgraded to LiveCycle, enter the same administrator user name and password as that of your previous LiveCycle installation. In case of a fresh installation, enter the default user name and password.
- 3 After you log in, click **Services** to access the service administration pages or click **Settings** to access the pages on which you can administer settings for different modules.

6.1.5.2 Change the default password of LiveCycle Administrator

LiveCycle creates one or more default users during the installation. The password for these users is in the product documentation and is publicly available. You must change this default password, depending on your security requirements.

The LiveCycle administrator user password is set to “password” by default. You must change it in Administration Console > Settings > User Management.

Also, it is recommended to change the default password for CRX Administrator.

For detailed information, see [Change the default administrator password](#).

6.1.5.3 Accessing CQ Welcome Page

CQ welcome page is the web-based portal for accessing various CQ components, administration, deployment and development tools. The default user name and password for logging in is administrator and password (same as LiveCycle Administrator).

Access the welcome page using the following steps:

- 1 Type the following URL in a web browser:

`http://[hostname]:[port]/lc/welcome`

- 2 Enter the same administrator user name and password as mentioned above.
- 3 After you log in, you can access various components, administration, deployment and development UIs.

6.1.5.4 Accessing OSGi Management Console

In CQ, components are in form of OSGi bundles which are deployed to Apache Felix OSGi container. OSGi console provides a way to manage OSGi bundles and services configurations. The default user name and password for logging in is admin and admin (same as CRX Administrator).

Access the OSGi Management console using the following steps:

- 1 Type the following URL in a web browser:

`http://[hostname]:[port]/lc/system/console`

- 2 Enter the same administrator username and password as mentioned above.
- 3 After you log in, you can access various components, services, bundles and other configurations.

6.1.5.5 Change CQ Administrator default password

CQ embedded within LiveCycle has two administrator users as mentioned below.

- **Super Administrator (administrator):** The Super Administrator user can access various CQ/CRX UIs and perform admin operations. The default username and password are same as LiveCycle Administrator, **administrator/password**. This user doesn't have access to OSGi Management Console. The default password of this user can be changed using LiveCycle Administrator console only as mentioned in section **Change default LiveCycle password**. The changed password will be applicable for both LiveCycle and CQ.
- **Administrator (admin):** This user can access to OSGi console in addition to CQ/CRX UIs and has administrator privileges. The default username and password for the user are **admin/admin**. To change the default password follow the below mentioned steps:

- 1 Type the following URL in a web browser.

`http://[hostname]:[port]/lc/libs/granite/security/content/admin.html`

- 2 Login using following credential:

Username: admin

Password: admin.

- 3 Search for user **Administrator**.
- 4 Click on the user in left pane, the user details is displayed in the right pane.
- 5 Click on **Edit** icon in the right pane.
- 6 On the edit page in the right pane, provide new password in the **New Password** field and current password in **Your Password** field.
- 7 Click **Save** icon in the right pane.
- 8 Re-login using the changed password to verify.

6.1.5.6 View the log files

Events, such as run-time or startup errors, are recorded to the application server log files. If you have problems deploying to the application server, you can use the log files to help you find the problem. You can open the log files by using any text editor.

Log files, in case of manually-configured JBoss, are located at:

- **(Standalone JBoss)**`[appserver root]/server/standard/logs` directory
- **(Cluster)**`[appserver root]/server/all/logs` directory

Log files, in case of Adobe-preconfigured JBoss, are located at:

- **(Standalone)**`[appserver root]/server/lc_<dbname>/logs` directory
- **(Cluster)**`[appserver root]/server/lc_<dbname>_cl/logs` directory

The log files are:

- `server.log`
- `boot.log`

Following CRX log files are located at `[CRX_home]/`

- `error.log`
- `audit.log`
- `access.log`
- `request.log`
- `update.log`

6.2 Verify the LiveCycle cluster

- 1 Ensure that all application server instances of the cluster are started.
- 2 View the Gemfire.log file, located in the directory appropriate to your application server:
 - `Jboss: [lc_temp_dir]/adobejb_[idp_server_name]/caching`

Note: `idp_server_name` is the value of the JVM argument `-Dadobeidp.serverName` passed to the JBoss instance.

- 3 Messages such as the following confirm that the cache is connected to all servers of the cluster:

```
[info 2008/01/22 14:24:31.109 EST GemfireCacheAdapter <UDP mcast
receiver> nid=0x5b611c24] Membership: received new view
[server-0:2916|1] [server-0:2916/2913, server-1:3168/3165]
[info 2008/01/22 14:24:31.125 EST GemfireCacheAdapter <View Message
Processor> nid=0x7574d1dc] DMMembership: admitting member
<server-1:3168/3165>; now there are 2 non-admin member(s)
```

Note: Ensure that the number of non-admin members (two in the example log entry above) matches the number of members in your cluster. A discrepancy indicates that some members of the cluster are not connected to the cache.

6.3 Verify the CRX Cluster

- 1 Go to `http://<authorHost>:<authorPort>/lc/system/console`. Login with OSGi Management Console user credentials. The default credential is `admin/admin`
- 2 Navigate to `Main>JMX`, locate the row with domain: `com.adobe.granite` and type: `Repository`.
- 3 Click `Repository` and locate `Attribute Name: ClusterNodes`.

The `ClusterNodes` attribute contains a table. Each row in the table represents a node in the cluster. Each row contains `Operating System`, `hostname`, `id`, and `repositoryHome` of a cluster node.

On all the slave node, value of `crx.cluster.master` is `false` and on the master node value of `crx.cluster.master` is `true`.

6.4 Accessing module web applications

After LiveCycle is deployed, you can access the web applications that are associated with the following modules:

- Reader Extensions
- Adobe® LiveCycle® Workspace 11
 - Note:** The Flex Workspacce is deprecated for AEM forms. It is available for the LiveCycle ES4 release.*
- HTML Workspace
- User management
- Correspondance management
- PDF Generator web application
- Adobe® LiveCycle® PDF Generator 11
- Adobe® LiveCycle® Rights Management 11

After accessing the web applications by using the default administrator permissions to ensure that they are accessible, you can create additional users and roles so that others can log in and use the applications. (See [Administration Help](#).)

6.4.1 Access the Reader Extensions web application

***Note:** You must apply a Reader Extensions credential and apply the user roles for a new user. (See “Configuring credentials for use with Reader Extensions” in LiveCycle Administration Help.)*

- 1 Open a web browser and enter this URL:

```
http://[hostname]:[port]/ReaderExtensions
```

- 2 Log in using the user name and password for LiveCycle.

***Note:** You must have administrator or superuser privileges to log in. To allow other users to access the Reader Extensions web application, you must create the users in User Management and grant them the Reader Extensions Web Application role.*

6.4.2 Access Workspace

- 1 Open a web browser and enter this URL:

```
http://[hostname]:[port]/workspace
```

- 2 Log in using the user name and password for LiveCycle.

***Note:** The Flex Workspacce is deprecated for AEM forms. It is available for the LiveCycle ES4 release.*

6.4.3 Access HTML Workspace

- 1 Open a web browser and enter this URL:

```
http://[hostname]:[port]/lc/ws
```

- 2 Log in using the user name and password for LiveCycle.

6.4.4 Access Forms Manager

- 1 Open a web browser and enter this URL:

`http://[hostname]:[port]/lc/fm`

- 2 Log in using the user name and password for LiveCycle.

6.4.5 Access PDF Generator Web Application

- 1 Open a web browser and enter this URL:

`http://[hostname]:[port]/pdfgui`

- 2 Log in using the user name and password for LiveCycle.

6.4.6 Access Rights Management

You must create a user with the Rights Management End User role in User Management and log in to the Rights Management administrator or end-user applications by using the login information that is associated with that user.

***Note:** The default administrator user cannot access the Rights Management end-user web application but you can add the appropriate role to its profile. You can create a new user or modify an existing user through Administration Console.*

Access the Rights Management end-user web application

- ❖ Open a web browser and enter this URL:

`http://[hostname]:[port]/edc`

Access the Rights Management administration web application

- 1 Open a web browser and enter this URL:

`http://[hostname]:[port]/adminui`

- 2 Click **Services > LiveCycle Rights Management 11**.

For information about setting up users and roles, see Administration Help.

Assign the Rights Management End User role

- 1 Log in to Administration Console. (See “[6.1.5.1 Accessing LiveCycle Administration Console](#)” on page 34.)
- 2 Click **Settings > User Management > Users and Groups**.
- 3 In the **Find** box, type **all** and, in the **In** list, select **Groups**.
- 4 Click **Find** and, for the required domains, click **All Principals** in the list that appears.
- 5 Click the **Role Assignments** tab and click **Find Roles**.
- 6 In the list of roles, select the check box next to **Rights Management End User**.
- 7 Click **OK** and then click **Save**.

6.4.7 Accessing User Management

By using User Management, administrators can maintain a database of all users and groups, synchronized with one or more third-party user directories. User Management provides authentication, authorization, and user management for LiveCycle modules, including Reader Extensions, Workspace, Rights Management, Adobe® LiveCycle® Process Management 11, Adobe® LiveCycle® Forms Standard 11 and PDF Generator.

- 1 Log in to Administration Console.
- 2 On the home page, click **Settings > User Management**.

***Note:** For information about configuring users with User Management, click **User Management Help** in the upper-right corner of the User Management page.*

6.4.8 Access Correspondence Management Solution template

You can verify the Correspondence Management Solution deployment by visiting `http://[hostname]:[port]/lc/cm` and logging in using LiveCycle administrator credentials. Solution template is a reference implementation of Correspondence Management Solution.

***Note:** In a non-turnkey deployment, if you encounter an error while accessing the solution template, you must integrate LiveCycle with Correspondence Management Solution. For more information, see the **Configure Publish nodes to Integrate with LiveCycle** section of this document.*

6.5 Configure Correspondence Management Solution

Perform the following tasks to configure Correspondence Management Solution.

For a recommended setup for the Correspondence Management Solution, see Correspondence Management Solution Topology.

6.5.1 Configure the Author instance

The Author instance is embedded within the LiveCycle server. This implies that you do not need to make any configuration updates to the Author instance. The instance inherits all the configuration settings from the LiveCycle server.

6.5.2 Configure the Publish instance

You must run separate Author and Publish instances for Correspondence Management Solution. However, you can configure the two instances on the same or on different machines. An author instance is embedded in the LiveCycle Core application and runs on the LiveCycle server. For publish instance, LCM configures a publish ear (`adobe-lifecycle-cq-publish.ear`). Deploy the publish ear on a separate server instance.

***Note:** Before configuring the Publish instance, ensure that your author instance is configured and deployed. You can verify by successfully logging in to the solution template for Correspondence management Solution. For more information, see the **Access Correspondence Management Solution template** section of this document.*

- 1 Create a new server for the publish instance. Perform JBoss installation and use standard profile.
- 2 Copy the `[LiveCycle root]/crx-repository` directory from the author instance to the publish instance machine.

- 3 From the crx-repository directory on the publish instance machine, open the crx-repository/install folder. Keep the following packages and delete all other packages from the install folder:

- dataservices-pkg.zip
- platform-common-pkg.zip
- platform-content-pkg.zip
- platform-security-pkg.zip
- solution-correspondencemanagement-pkg.zip

- 4 Start the Publish server with `-Dcom.adobe.livecycle.crx.home=<location for crx-repository>` parameter, where `<location for crx-repository>` is the location where you copied the crx-repository directory for the Publish instance.

For details on how to configure generic JVM arguments for WebSphere and WebLogic, see the **Configuring the JVM arguments** section for [WebSphere](#) and [WebLogic](#).

- 5 Copy the adobe-livecycle-cq-publish.ear file and deploy it to the appserver profile created in step 1.

Note: If author and Publish instances are on the same machine, ensure that you start the Publish instance using a different port.

Now that the Publish instance is up and running, you need to configure the two instances to communicate with each other.

6.5.3 Configure Publish nodes to Integrate with LiveCycle

Perform these steps on all publish instances. To enable communication between publish instance and LiveCycle Server:

- 1 Go to `http://[publishhost]:[publishport]/lc/system/console/configMgr` and Login with OSGi Management Console user credentials. The default credential are admin/admin.
- 2 Search and click **Edit** next to the **Adobe LiveCycle Client SDK Configuration** setting.
- 3 In the Server Url field, ensure that `http://[lchost]:[lcport]` is specified.

Important: Ensure that the LiveCycle server is listening on the specified host and port combination. The following three scenarios are possible in the case of a LiveCycle server cluster:

- All LiveCycle server instances are running on *localhost* and the same port. In this case use `localhost:[port]`.
- All LiveCycle server instances are running on *localhost* but on different ports. In this case, use a load balancer host name and port combination—`[loadbalancer_host]:[loadbalancer_port]`.
- All LiveCycle server instances are running on a particular host name (not *localhost*) and different/same ports. In this case, use a load balancer host name and port—`[loadbalancer_host]:[loadbalancer_port]`.

If you need to use a load balancer URL to access the LiveCycle server cluster (as mentioned above), ensure that the required communication ports between Author instances and the load balancer are open.

- 4 Specify LiveCycle administrator credentials in the Username as Password fields.
- 5 Click **Save**.

6.5.4 Communicating between the Author and Publish instances

You need to perform certain configuration changes to enable two-way communication between the Author and Publish instances.

6.5.4.1 Configure Replication Agents (Define publish instance URL)

On the Author instance, you need to configure replication agents for each Publish instance. These agents replicate content from the Author instances to all the Publish instances.

1 Log in to Tools UI at **http://<authorHost>:<authorPort>/lc/miscadmin**

2 Select **Replication**, then **Agents on author** in the left panel.

On the right panel, you see various agents configured for the Author instance.

3 On the right panel, Select **New....** and click **New Page**.

The **Create Page** dialog displays.

4 Set the **Title** and **Name**, then select **Replication Agent**.

5 Click **Create** to create new agent.

6 Double-click the new agent item to open the configuration panel.

7 Click **Edit** - the **Agent Settings** dialog displays.

a In the **Settings** tab:

- Enter a **Description**.
- Check **Enabled**.
- Select **Serialization Type** as **Default**.
- Set the **Retry Delay** to **60000**.
- Set the **Log Level** as **Info**.

b In the **Transport** tab:

- Enter the required URI for the Publish instance
`http://<publishHost>:<publishPort>/lc/bin/receive?sling:authRequestLogin=1`
- Set **User** and **Password**. The default credential are admin/admin.

8 Click **OK** to save the settings.

9 On the agent configuration panel, click **Test Connection**.

Successful connection ensures that the configuration is done correctly.

Note: : In case, you have only one Publish instance you can use the default Replication Agent named as publish. You need to edit it for specifying Publish URI in the Transport tab as mentioned in the step b(i). In this case, you do not need to create a new replication agent.

Note: : In case, you have a publish farm(multiple non-clustered publish instances), you need to create a replication agent for each Publish instance as mentioned in Steps 1-9. For each such replication agent, Title and Name should be significant and unique, so the identification of the corresponding Publish instance can be simpler. Each such replication agent has a different URI in the Transport tab pointing to a particular Publish instance. For multiple publish instances, you can also create replication agents by copying the default agent publish and then editing Name and URI in transport tab of the created agent. If you are not using the default Replication Agent, disable it, so an unnecessary replication attempt can be avoided.

Note: For Author clusters, these steps need to be performed on one Author instance (preferably a master instance).

6.5.4.2 Define Publish instance URL for ActivationManagerImpl

1 Go to `http://<authorHost>:<authorPort>/lc/system/console/configMgr`. Login with OSGi Management Console user credentials. The default credential is admin/admin.

- 2 Find and click the Edit icon next to the
`com.adobe.livecycle.content.activate.impl.ActivationManagerImpl.name` setting.
- 3 In the ActivationManager Publish URL field, specify the URL for accessing the Publish instance ActivationManager. You can provide the following URLs.
 - a **Load Balancer URL (Recommended):** Provide load balancer URL, If you have a webserver acting as load balancer in front of publish farm (multiple non-clustered publish instances).
 - b **Publish instance URL:** Provide any publish instance URL, If you have a single publish instance or the webserver fronting the publish farm is not accessible from the author environment due to any restrictions. In case, the specified publish instance is down, there is a fallback mechanism to deal with on the author side.

URL string: `http://<hostname>:<port>/lc/bin/remoting/lc.content.remote.activate.activationManager`
- 4 Click **Save**.

6.5.4.3 Configure reverse replication queue

On the Author instance, you need to configure reverse replication agents for each Publish instance. These agents replicate content from the Publish instance to the Author instance.

- 1 Log in to Tools UI at `http://<authorHost>:<authorPort>/lc/miscadmin`
- 2 Select **Replication**, then **Agents on author** in the left panel.

On the right panel, you see various agents configured for the Author instance.
- 3 On the right panel, Select **New**, and click **New Page**.

The **Create Page** dialog appears.
- 4 Set the **Title** and **Name**, then select **Reverse Replication Agent**.
- 5 Click **Create** to create new agent.
- 6 Double-click the new agent item to open the configuration panel.
- 7 Click **Edit** - the **Agent Settings** dialog displays.
 - a In the **Settings** tab:
 - Enter a **Description**
 - Check **Enabled**.
 - Set the **Retry Delay** to **60000**.
 - Set the **Log Level** as **Info**.
 - b In the **Transport** tab:
 - Enter the required URI for the Publish instance -
`http://<publishHost>:<publishPort>/lc/bin/receive?sling:authRequestLogin=1`
 - Set **User** and **Password** - admin/admin
 - c In the **Extended** tab: Set HTTP Method as GET
- 8 Click **OK** to save the settings.
- 9 On the agent configuration panel, click **Test Connection**.

Successful connection ensures that the configuration is done correctly.

Note : In case, you have only one Publish instance you can use the default Reverse Replication Agent named as **publish_reverse**. You need to edit it for specifying Publish URI in the **Transport** tab as mentioned in the step b(i). In this case, you do not need to create a new reverse replication agent.

Note: In case, you have a publish farm(multiple non-clustered publish instances), you need to create a reverse replication agent for each Publish instance as mentioned in Steps 1-9. For each such replication agent, **Title** and **Name** should be significant and unique, so the identification of the corresponding Publish instance can be simpler. Each such replication agent has a different URI in the **Transport** tab pointing to a particular Publish instance. For multiple publish agents, you can also create reverse replication agents by copying the default agent **publish_reverse** and then editing **Name** and **URI** in transport tab of the created agent. If you are not using the default Reverse Replication Agent, disable it, so an unnecessary replication attempt can be avoided.

Note: For Author clusters, these steps need to be performed on one Author instance (preferably a master instance).

6.5.4.4 Define author instance URL for VersionRestoreManagerImpl

- 1 Go to `http://<publishHost>:<publishPort>/lc/system/console/configMgr`. Login with OSGi Management Console user credentials. The defaults credential are admin/admin.
- 2 Find and click the Edit icon next to the `com.adobe.livecycle.content.activate.impl.VersionRestoreManagerImpl.name` setting.
- 3 In the VersionRestoreManager Author URL field, specify the URL of the author instance VersionRestoreManager.
URL string: `http://<hostname>:<port>/lc/bin/remoting/lc.content.remote.activate.versionRestoreManager`

Note: If there are multiple author instances(Clustered) fronted by a Load Balancer, specify the URL to the load balancer in the VersionRestoreManager Author URL field.

- 4 Click **Save**.

6.5.5 Install sample users and assets

You can install sample users with predefined user permissions to further explore the solution template, which you can customize to build your own solution.

- 1 Go to `http://<authorHost>:<authorPort>/lc/crx/explorer/index.jsp`.
- 2 Log in using LiveCycle administrator credentials and click **Package Manager**.
- 3 In **Package Manager**, upload the `samples-correspondencemanagement-pkg-<version>.zip` package from `<LC_HOME>/deploy/crx`.
- 4 Click **Install** once the package uploads successfully.
- 5 Click **Install** on the confirmation dialog to install the sample users and assets.

Correspondence Management Sample Users

Correspondence Management Solution Accelerator contains the following sample users. These user are expected to participate in the activities leading to generation of interactive customer communication.

The following roles mentioned are automatically assigned to the users during package Installation:

User name	Assigned role	Responsibilities
Todd Goldman	Correspondence Management Administrator	This user is the general system administrator. This role enables the user to modify all assets. This role also lets define the categories.
Heather Douglas	Correspondence Management Subject Matter Expert	This persona has the role enabling him to CRUD texts and images.
Caleb Lopez	Correspondence Management Application Specialist	This user defines the letter template by judicious usage of the text, picture, condition, list objects. With this role, the user can CRUD the letter templates, layouts, lists, conditions, texts and images.
Gloria Rios	Correspondence Management Claim Adjustor	An agent user would use the letter template defined by the business user to produce the letter communication to deliver to the customer.
Jocelyn Robinson	Correspondence Management Form Designer	This user has the skills to design form layouts using LiveCycle Designer. Having equipped with the necessary know-how to design form layouts for use in correspondence management, this user will use LiveCycle Designer and design the XDP templates, which would serve as the boilerplate for the letter.
Frank Kricfalusi	Correspondence Management Developer	This user has the knowledge about XSD schema and data modelling concepts and is responsible for creation and maintenance of Data Dictionaries.

For more information about the sample users and guidelines to implement a solution using the solution template, see [Correspondence Management Solution Guide](#).

Note: For author clusters, these steps need to be performed on one author instance (preferably a master instance).

6.5.6 Configure IPv6 implementation

Note: Perform these steps only if Correspondence Management Solution is running on a machine that uses an IPv6 address.

To map the IPv6 address to a hostname on the server and client machines:

- 1 Navigate to the C:\Windows\System32\drivers\etc directory.
- 2 Open the `hosts` file in a text editor.
- 3 Add a mapping for the IPv6 address to a host name. For example:
`2001:1890:110b:712b:d1d:9c99:37ef:7281 <ipv6_hostname>`
- 4 Save and close the file.

Ensure that you use the mapped host name instead of the IPv6 address to access Correspondence Management Solution.

6.5.7 Install Japanese fonts for Adobe Reader

If your Correspondence Management assets use Japanese fonts, you must install the Japanese Language Support Package for Adobe Reader. Otherwise, your letters and forms will not render and function properly. For installing language packs, visit the downloads page for Adobe Reader.

6.6 Configuring PDF Generator

If you installed PDF Generator as part of your LiveCycle, complete the following tasks:

6.6.1 Environment variables

If you installed the PDF Generator module and configured it to convert files to PDF, for some file formats, you must manually set an environment variable that contains the absolute path of the executable that is used to start the corresponding application. The table below lists the environment variables for the native applications that you have installed.

Note: Ensure that the required applications are installed on all nodes in the cluster.

Note: All environment variables and respective paths are case-sensitive.

Application	Environment variable	Example
Adobe Acrobat	Acrobat_PATH	C:\Program Files (x86)\Adobe\Acrobat 11.0\Acrobat\Acrobat.exe
Adobe FrameMaker®	FrameMaker_PATH	C:\Program Files (x86)\Adobe\FrameMaker8.0\FrameMaker.exe
Notepad	Notepad_PATH	C:\WINDOWS\notepad.exe You can leave the Notepad_PATH variable blank.
OpenOffice	OpenOffice_PATH	C:\Program Files (x86)\OpenOffice.org 3.3
Adobe PageMaker®	PageMaker_PATH	C:\Program Files (x86)\Adobe\PageMaker 7.0.2\PageMaker.exe
WordPerfect	WordPerfect_PATH	C:\Program Files (x86)\WordPerfect Office 12\Programs\wpwin12.exe
Adobe Photoshop®	Photoshop_PATH	C:\Program Files (x86)\Adobe\Adobe Photoshop CS4\Photoshop.exe

Note: These environment variables must be set for all nodes in the cluster.

Note: The environment variable `OpenOffice_PATH` is set to the installation folder instead of the path to the executable.

You do not need to set up the paths for Microsoft Office applications such as Word, PowerPoint, Excel, Visio, and Project, or for AutoCAD. The Generate PDF service starts these applications automatically if they are installed on the server.

Create a new Windows environment variable

- 1 Select **Start > Control Panel > System**.
- 2 Click the **Advanced** tab and click **Environment Variables**.
- 3 In the System variables section, click **New**.
- 4 Enter the environment variable name you need to set (for example, enter `Photoshop_PATH`). This folder is the one that contains the executable file. For example, type the following path:

```
D:\Program Files\Adobe\Adobe Photoshop CS4\Photoshop.exe
```

Set the PATH variables on Linux or UNIX (OpenOffice only)

Execute the following command:

```
export OpenOffice_PATH=/opt/openoffice.org3.3
```

6.6.2 Configuring the application server to use HTTP proxy server

If the computer that LiveCycle is running on uses proxy settings to access external web sites, the application server should be started with the following values set as Java virtual machine (JVM) arguments:

```
-Dhttp.proxyHost=[server host]  
-Dhttp.proxyPort=[server port]
```

Complete the following procedure to start your application server with HTTP proxy host setting.

1 From a command line, edit the run script in the *[appserver root]/bin/* directory:

- (Windows)
 - run.conf.bat
- (Linux, UNIX)
 - run.conf

2 Add the following text to the script file:

```
Set JAVA_OPTS=%JAVA_OPTS%  
-Dhttp.proxyHost=[server host]  
-Dhttp.proxyPort=[server port]
```

3 Save and close the file.

6.6.3 Setting the Adobe PDF Printer as the default printer

You must set the Adobe PDF Printer to be the default printer on the server. If the Adobe PDF Printer is not set as the default, PDF Generator cannot convert files successfully.

For clusters, you must set Adobe PDF Printer as the default printer on all nodes.

Set the default printer

- 1 Select **Start > Printers and Faxes**.
- 2 In the Printers and Faxes window, right-click **Adobe PDF** and select **Set as Default Printer**.

6.6.4 Configuring Acrobat Professional (Windows-based Computers Only)

***Note:** This procedure is required only if you upgraded to or installed Acrobat after you completed the LiveCycle installation. Upgrading Acrobat can be completed after you run Configuration Manager and deploy LiveCycle to the application server. Acrobat Professional root directory is designated as *[Acrobat root]*. Typically, the root directory is *C:\Program Files\Adobe\Acrobat 11.0\Acrobat*.*

Configure Acrobat for use with PDF Generator

- 1 If an earlier version of Acrobat is installed, uninstall it by using Add or Remove Programs in the Windows Control Panel.
- 2 Install Acrobat XI Pro by running the installer.

- 3 Navigate to the additional\scripts folder on the LiveCycle installation media.
- 4 Run the following batch file.

```
Acrobat_for_PDFG_Configuration.bat [LiveCycle root]/pdfg_config
```

Note: On clusters, you must run the command on the cluster node where LiveCycle is installed.
- 5 On other cluster nodes on which you do not run LiveCycle Configuration Manager, do the following:
 - Add a new registry DWORD entry named SplWOW64TimeOut at HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Print. Set its value to 60000.
 - Copy PDFGen.api from the [LiveCycle root]/plugins/x86_win32 directory on the node where LiveCycle is installed to the [Acrobat root]/plug_ins directory on the node being currently configured.
- 6 Open Acrobat and select **Help > Check for updates > Preferences**.
- 7 Deselect **Automatically check for Adobe updates**.

Validate the Acrobat installation

- 1 Navigate to a PDF file on your system and double-click it to open it in Acrobat. If the PDF file opens successfully, Acrobat is installed correctly.
- 2 If the PDF file does not open correctly, uninstall Acrobat and reinstall it.

Note: Ensure that you dismiss all the Acrobat dialog boxes that are displayed after the Acrobat installation is completed and disable the automatic updates for Acrobat. Set the `Acrobat_PATH` environment variable to point to Acrobat.exe (For example, `C:\Program Files\Adobe\Acrobat 11.0\Acrobat\Acrobat.exe`).

Configure native application support

- 1 Install and validate Acrobat as described in the previous procedure.
- 2 Set Adobe PDF printer as the default printer.

Add temporary directories to trusted directories list in Acrobat

The OptimizePDF service uses Adobe Acrobat and mandates that LiveCycle temporary directory and PDF Generator temporary directory are listed in the trusted directories list of Acrobat.

If LiveCycle temporary directory and PDF Generator temporary directory are not listed in the trusted directories list, the OptimizePDF service fails to run. Perform the following steps to add directories to the temporary directory list:

- 1 Open Acrobat, Choose Edit > Preferences.
- 2 From the Categories on the left, select Security (Enhanced), and then select the Enable Enhanced Security option.
- 3 To add LiveCycle temporary directory and PDF Generator temporary directory to the trusted directories list, click **Add Folder Path**, select directories and click **OK**.

6.6.5 Adding fonts to PDF Generator

LiveCycle provides a central repository of fonts, which is accessible to all LiveCycle modules. Make the extra fonts available to non-LiveCycle applications on the server so that PDF Generator can use these fonts to create PDF documents that are created with these applications.

Note: Restart the application server after adding new fonts to the specified fonts folder.

6.6.5.1 Non-LiveCycle applications

The following list contains non-LiveCycle applications that PDF Generator can use for PDF generation on the server side:

Windows-only Applications

- Microsoft Office Word
- Microsoft Office Excel
- Microsoft Office PowerPoint
- Microsoft Office Project
- Microsoft Office Visio
- Microsoft Office Publisher
- AutoDesk AutoCAD
- Corel WordPerfect
- Adobe Photoshop CS
- Adobe FrameMaker
- Adobe PageMaker
- Adobe Acrobat Professional

Multiplatform applications

- OpenOffice Writer
- OpenOffice Calc
- OpenOffice Draw
- OpenOffice Impress

***Note:** In addition to these applications, your list may include additional applications that you added.*

Of the above applications, the OpenOffice Suite (which includes Writer, Calc, Draw, and Impress) is available on Windows, Solaris, and Linux platforms, whereas other applications are available on Windows only.

6.6.5.2 Adding new fonts to Windows applications only

All the Windows-only applications that are mentioned above can access all the fonts that are available in the C:\Windows\Fonts (or equivalent) folder. In addition to C:\Windows\Fonts, each of these applications may have its own private fonts folders.

Therefore, if you plan to add any custom fonts to the LiveCycle fonts repository, ensure that the same fonts are available to the Windows-only applications also by copying these fonts to either C:\Windows\Fonts or to an equivalent folder.

Your custom fonts must be licensed under an agreement that allows you to use them with the applications that have access to these fonts.

6.6.5.3 Adding new fonts to other applications

If you added support for PDF creation in other applications, see the Help for these applications to add new fonts. In Windows, copying your custom fonts to the C:\Windows\Fonts (or equivalent) folder should be sufficient.

6.6.6 Configuring HTML to PDF conversions

The HTML-to-PDF conversion process is designed to use the settings from Acrobat XI Pro that override the settings from PDF Generator.

***Note:** This configuration is required to enable the HTML-to-PDF conversion process, otherwise this conversion type will fail.*

6.6.6.1 Configure the HTML-to-PDF conversion

- 1 Install and validate Acrobat as described in “[6.6.4 Configuring Acrobat Professional \(Windows-based Computers Only\)](#)” on page 46.
- 2 Locate the pdfgen.api file in the `[LiveCycle root]\plugins\x86_win32` directory and copy it to `[Acrobat root]\Acrobat\plug_ins` directory.

6.6.6.2 Enable support for Unicode fonts in HTML to PDF conversions

Important: The HTML-to-PDF conversion fails if a zipped input file contains HTML files with double-byte characters in filenames. To avoid this problem, do not use double-byte characters when naming HTML files.

- 1 Copy the Unicode font to any of the following directories as appropriate for your system:

- Windows

`[Windows root]\Windows\fonts`

`[Windows root]\WINNT\fonts`

- UNIX

`/usr/lib/X11/fonts/TrueType`

`/usr/openwin/lib/X11/fonts/TrueType`

`/usr/share/fonts/default/TrueType`

`/usr/X11R6/lib/X11/fonts/ttf`

`/usr/X11R6/lib/X11/fonts/truetype`

`/usr/X11R6/lib/X11/fonts/TrueType`

`/usr/X11R6/lib/X11/fonts/TTF`

`/Users/cfqauser/Library/Fonts`

`/System/Library/Fonts`

`/Library/Fonts`

`/Users/ + System.getProperty(<user name>, root) + /Library/Fonts`

`System.getProperty(JAVA_HOME) + /lib/fonts`

`/usr/share/fonts (Solaris)`

Note: Ensure that the directory `/usr/lib/X11/fonts` exists. If it does not, create a symbolic link from `/usr/share/X11/fonts` to `/usr/lib/X11/fonts` using the `ln` command.

- 2 Modify the font-name mapping in the `cffont.properties` file located in the `[LiveCycle root]/deploy/adobe-generatepdf-dsc.jar` file:


- Extract this archive, and locate the `cffont.properties` file and open it in an editor.
- In the comma-separated list of Java font names, add a map to your Unicode system font for each font type. In the example below, `kochi mincho` is the name of your Unicode system font.

`dialog=Arial, Helvetica, kochi mincho`

`dialog.bold=Arial Bold, Helvetica-Bold, kochi mincho ...`

- Save and close the properties file, and then repackage and redeploy the `adobe-generatepdf-dsc.jar` file.

Note: On a Japanese operating system, specify the font mapping in the `cffont.properties.ja` file as well, which takes precedence over the standard `cffont.properties` file.

 Fonts in the list are searched from left to right, using the first font found. HTML-to-PDF conversion logs return a list of all the font names that are found in the system. To determine the font name you need to map, add the font to one of the directories above, restart the server, and run a conversion. You can determine from the log files the font name to use for mapping.

To embed the font in the generated PDF files, set the `embedFonts` property in the `cffont.properties` file to `true` (the default is `false`).

6.6.7 Modify Microsoft Visio default macro settings

When a Microsoft Visio file containing macros is submitted for conversion, the resultant Microsoft Office Visio Security Notice dialog causes the conversion to time out. To successfully convert files that contain macros, the default macro settings in Visio must be changed.

- ❖ In Visio, click **Tools > Trust Center > Macro Settings** and select either of the following options and then click **OK**:
 - Disable all macros without notification
 - Enable all macros

6.6.8 Installing the Network Printer Client

PDF Generator includes an executable file to install the PDF Generator network printer on a client computer. After the installation is complete, a PDF Generator printer is added to the list of existing printers on the client computer. This printer can then be used to send documents for conversion to PDF.

Note: The Network Printer Client installation wizard available in the Administration Console is supported only on Windows operating system. Ensure that you use a 32-bit JVM to launch the Network Printer Client installation wizard. You will encounter an error if you use a 64-bit JVM.

If the PDFG Network Printer fails to install on Windows or if you want to install the printer on UNIX or Linux platforms, use the operating system's native Add Printer utility and configure it as described in [“6.6.8.2 Configure PDFG Network Printer on Windows using the native Add Printer wizard”](#) on page 51

6.6.8.1 Install the PDF Generator Network Printer Client

Note: Before installing the PDF Generator network printer client on Windows Server 2008, Ensure that you have the Internet Printing Client feature installed on your Windows Server 2008. For installing the feature, see *Windows Server 2008 Help*.

- 1 Ensure that you successfully installed PDF Generator on your server.
- 2 Do one of the following:
 - From a Windows client computer, enter the following URL in your web browser, where `[host]` is the name of the server where you installed PDF Generator and `[port]` is the application server port used:
`http://[host]:[port]/pdfg-ipp/install`
 - In Administration Console, click **Home > Services > PDF Generator > PDFG Network Printer**. In the **PDFG Network Printer Installation** section, click **Click here** to launch the PDFG Network Printer Installation.

- 3 On the Configure Internet Port screen, select **Use the specified user account** option, and provide the credentials of a LiveCycle user who has the PDFG Administrator/User role. This user must also have an email address that can be used to receive the converted files. To have this security setting apply to all users on the client computer, select **Use the same security options for all users**, and then click **OK**.

Note: If the user's password changes, then users will need to reinstall the PDFG Network Printer on their computers. You cannot update the password from Administration Console.

Upon successful installation, a dialog box appears, indicating that "The Printer Adobe LiveCycle PDF Generator 11 has been successfully installed."

- 4 Click **OK**. You will now have a printer named *Adobe LiveCycle PDF Generator 11* in your list of available printers.

6.6.8.2 Configure PDFG Network Printer on Windows using the native Add Printer wizard

- 1 Click **Start > Printers and Faxes** and double-click **Add Printer**.
- 2 Click **Next**, select **A network printer, or a printer attached to another computer**, and then click **Next**.
- 3 Select **Connect to a printer on the internet or on a home or office network** and type the following URL for the PDFG printer, where *[host]* is the server name and *[port]* is the port number where the server is running:

`http://[host]:[port]/pdfg-ipp/printer`

- 4 On the Configure Internet Port screen, select **Use the specified user account** and provide valid User credentials.
- 5 In the **Printer Driver Select** box, choose any standard PostScript-based printer driver (for example, HP Color LaserJet PS).
- 6 Complete the installation by choosing appropriate options (for example, setting this printer as default).

Note: The user credentials used while adding the printer must have a valid email ID configured in User Management to receive the response.

- 7 Configure the email service's sendmail service. Provide a valid SMTP server and authentication information in the service's configuration options.

6.6.8.3 Install and configure the PDF Generator Network Printer Client using Proxy server port forwarding

- 1 Configure port forwarding on the CC Proxy server on a particular port to the LiveCycle Server, and disable the authentication at proxy server level (because LiveCycle uses its own authentication). If a client connects to this Proxy server on the forwarded port, then all the requests will be forwarded to the LiveCycle Server.
- 2 Install PDFG Network Printer using the following URL:
`http://[proxy server]:[forwarded port]/pdfg-ipp/install.`
- 3 Provide the necessary credentials for authentication of the PDFG Network Printer.
- 4 The PDFG Network Printer will be installed on the client machine which you can use for PDF conversion using the firewall protected LiveCycle Server.

6.6.9 Changing File Block Settings

Change Microsoft Office trust center settings to enable PDFG to convert older versions of Microsoft office documents.

- 1 Click the **File tab** in any Office 2010 application. Under **Help**, click **Options**; the Options dialog box appears
- 2 Click **Trust Center**, and then click **Trust Center Settings**.
- 3 In the **Trust Center settings**, click **File Block Settings**.

- 4 In the File Type list, uncheck open for the file type that you want to be converted by PDFG.

6.6.10 Watched folder performance parameters

To avoid `java.io.IOException` error messages indicating that not enough disk space is available to perform PDF conversions by using a watched folder, you can modify the settings for PDF Generator in Administration Console.

Set performance parameters for PDF Generator

- 1 Log in to Administration Console and click **Services > Applications and Services > Service Management**.
- 2 In the list of services, navigate to and click **PDFGConfigService**, and then set the following values:
 - **PDFG Cleanup Scan Seconds:** 1800
 - **Job Expiration Seconds:** 6000
 - **Server Conversion Timeout:** Change the default of 270 to a higher value, such as 450.
- 3 Click **Save** and restart the server.

6.6.11 Enable PDF Conversion for Microsoft Word document containing protected fields

The PDF Generator supports Microsoft Word documents containing protected fields. To enable PDF Conversion for Microsoft Word document containing protected fields, change the file type settings:

- 1 In the **Administration Console**, navigate to **Services > PDF Generator > File Type Settings**, and open your file type settings profile.
- 2 Expand the **Microsoft Word** option and select the **Preserve document markup in Adobe PDF (for Microsoft Office 2003 or later)** option.
- 3 Click **Save As**, specify name of the file type setting, and click **OK**.

6.7 Final setup for Rights Management

Rights Management requires the application server to be configured to use SSL. (See [Administration Help](#).)

6.8 Configuring LDAP access

6.8.1 Configure User Management (Local Domain)

- 1 Open a web browser, navigate to `http://[host]:[port]/adminui`, and log in. (See “[6.1.5.1 Accessing LiveCycle Administration Console](#)” on page 34 .)
- 2 Click **Settings > User Management > Domain Management**, and then click **New Local Domain**.
- 3 In the appropriate boxes, enter the domain ID and name. (See “Adding local domains” in [Administration help](#).)
- 4 (Optional) Disable account locking by deselecting the **Enable Account Locking** option.
- 5 Click **OK**.

6.8.2 Configure User Management with LDAP (Enterprise Domain)

- 1 Open a web browser, navigate to `http://[host]:[port]/adminui` and log in. (See “[6.1.5.1 Accessing LiveCycle Administration Console](#)” on page 34.)
- 2 Click **Settings > User Management > Domain Management**, and then click **New Enterprise Domain**.
- 3 In the **ID** box, type a unique identifier for the domain and, in the **Name** box, type a descriptive name for the domain.
Note: When using MySQL for your LiveCycle database, use only single-byte (ASCII) characters for the ID. (See “Adding enterprise domains” in [Administration Help](#).)
- 4 Click **Add Authentication** and, in the **Authentication Provider** list, select **LDAP**.
- 5 Click **OK**.
- 6 Click **Add Directory** and, in the **Profile Name** box, type a name for your LDAP profile.
- 7 Click **Next**.
- 8 Specify values in the **Server**, **Port**, **SSL**, and **Binding** boxes, and in the **Populate Page with** box, select a directory settings option such as **Default Sun ONE values**. Also, specify values in the **Name** and **Password** box that would be used to connect to the LDAP database when anonymous access is not enabled. (See “Directory settings” in [Administration Help](#).)
- 9 (Optional) Test your configuration:
 - Click **Test**. The screen displays a message indicating either a successful server test or any configuration errors that exist.
- 10 Click **Next** and configure the **User Settings** as required. (See “Directory settings” in [Administration Help](#).)
- 11 (Optional) Test your configuration:
 - Click **Test**.
 - In the Search Filter box, verify the search filter or specify a new search filter, and then click **Submit**. The screen displays a list of entries that match the search criteria.
 - Click **Close** to return to the User Settings screen.
- 12 Click **Next** configure the **Group Settings** as required. (See “Directory settings” in [Administration Help](#).)
- 13 (Optional) Test your configuration:
 - Click **Test**.
 - In the Search Filter box, verify the search filter or specify a new search filter, and then click **Submit**. The screen displays a list of entries that match the search criteria.
 - Click **Close** to return to the Group Settings screen.
- 14 Click **Finish** to exit the New Directory page and then click **OK** to exit.

6.9 Enabling FIPS mode

LiveCycle provides a FIPS mode to restrict data protection to Federal Information Processing Standard (FIPS) 140-2 approved algorithms using the RSA BSAFE Crypto-C 2.1 encryption module.

If you did not enable this option by using Configuration Manager during LiveCycle configuration or if you enable it but want to turn it off, you can change this setting through Administration Console.

Modifying FIPS mode requires you to restart the server.

FIPS mode does not support Acrobat versions earlier than 7.0. If FIPS mode is enabled and the Encrypt With Password and Remove Password processes include the Acrobat 5 setting, the process fails.

In general, when FIPS is enabled, the Assembler service does not apply password encryption to any document. If this is attempted, a `FIPSMODEException` is thrown, indicating that “Password encryption is not permitted in FIPS mode.” Additionally, the `PDFsFromBookmarks` element is not supported in FIPS mode when the base document is password-encrypted.

Turn FIPS mode on or off

- 1 Log in to Administration Console.
- 2 Click **Settings > Core System Settings > Configurations**.
- 3 Select **Enable FIPS** to enable FIPS mode or deselect it to disable FIPS mode.
- 4 Click **OK** and restart the application server.

Note: LiveCycle software does not validate code to ensure FIPS compatibility. It provides a FIPS operation mode so that FIPS-approved algorithms are used for cryptographic services from the FIPS-approved libraries (RSA).

6.10 Configuring HTML digital signature

To use the HTML digital signature feature of Forms, complete the following procedure.

- 1 Manually deploy the `[LiveCycle root]/deploy/adobe-forms-ds.ear` file to your application server.
- 2 Log in to Administration Console and click **Services > LiveCycle Forms ES4**.
- 3 Select **HTML Digital Signature Enabled** and then click **Save**.

6.11 Configuring Connector for EMC Documentum

Note: LiveCycle supports EMC Documentum, versions 6.0, 6.5, 6.7 SP1, and 7.0 and minor updates only. Make sure your ECM is upgraded accordingly.

Note: Ensure that installing client for the connectors, copying of JAR's file and configuration changes tasks are performed on all the nodes of the cluster.

If you installed Connector for EMC Documentum as part of your LiveCycle, complete the following procedure to configure the service to connect to the Documentum repository.

Configure Connector for EMC Documentum

- 1 Locate the `adobe-component-ext.properties` file in the `[appserver root]/bin` folder (if the file does not exist, create it).
- 2 Add a new system property that provides the following Documentum Foundation Classes JAR files:
 - `dfc.jar`
 - `aspectjrt.jar`
 - `log4j.jar`
 - `jaxb-api.jar`

- (For Connector for EMC Documentum 6.5 only)
 - configservice-impl.jar,
 - configservice-api.jar

The new system property should take on this form:

```
[component id].ext=[JAR files and/or folders]
```

For example, using default Content Server and Documentum Foundation Classes installations, add to the file one of the following system properties on a new line, with no line breaks, and end the line with a carriage return:

- Connector for EMC Documentum 6.0 only:

```
com.adobe.livecycle.ConnectorforEMCDocumentum.ext=  
C:/Program Files/Documentum/Shared/dfc.jar,  
C:/Program Files/Documentum/Shared/aspectjrt.jar,
```

- Connector for EMC Documentum 6.5 only:

```
com.adobe.livecycle.ConnectorforEMCDocumentum.ext=  
C:/Program Files/Documentum/Shared/dfc.jar,  
C:/ProgramFiles/Documentum/Shared/aspectjrt.jar,  
C:/Program Files/Documentum/Shared/log4j.jar,  
C:/Program Files/Documentum/Shared/jaxb-api.jar,  
C:/Program Files/Documentum/Shared/configservice-impl.jar,  
C:/Program Files/Documentum/Shared/configservice-api.jar
```

Note: The above text contains formatting characters for line breaks. If you copy and paste this text, you must remove the formatting characters.

- Connector for EMC Documentum 6.7 SP1 and 7.0 only:

```
com.adobe.livecycle.ConnectorforEMCDocumentum.ext=  
C:/Program Files/Documentum/Shared/dfc.jar,  
C:/ProgramFiles/Documentum/Shared/aspectjrt.jar,  
C:/Program Files/Documentum/Shared/log4j.jar,  
C:/Program Files/Documentum/Shared/jaxb-api.jar,  
C:/Program Files/Documentum/Shared/configservice-impl.jar,  
C:/Program Files/Documentum/Shared/configservice-api.jar  
C:/Program Files/Documentum/Shared/commons-codec-1.3.jar  
C:/Program Files/Documentum/Shared/commons-lang-2.4.jar
```

Note: The above text contains formatting characters for line breaks. If you copy and paste this text, you must remove the formatting characters.

3 Repeat previous steps on each application server instance of the cluster.

4 Open a web browser and enter this URL:

```
http://[host]:[port]/adminui
```

5 Log in using the default user name and password:

User name: administrator

Password: password

6 Navigate to **Services > LiveCycle 11 Connector for EMC Documentum > Configuration Settings** and perform these tasks:

- Type all the required Documentum repository information.

- To use Documentum as your repository provider, under Repository Service Provider Information, select **EMC Documentum Repository Provider**, and then click **Save**. For more information, click the Help link in the upper-right corner of the page in the [Administration](#) Help.
- 7 (Optional) Navigate to **Services > LiveCycle 11 Connector for EMC Documentum > Repository Credentials Settings**, click **Add**, specify the Docbase information, and then click **Save**. (For more information, click **Help** in the upper-right corner.)
 - 8 If the application server is not currently running, start the server. Otherwise, stop and then restart the server.
 - 9 Open a web browser and enter this URL.
`http://[host]:[port]/adminui`
 - 10 Log in using the default user name and password:
User name: administrator
Password: password
 - 11 Navigate to **Services > Applications and Services > Service Management** and select these services:
 - EMCDocumentumAuthProviderService
 - EMCDocumentumContentRepositoryConnector
 - EMCDocumentumRepositoryProvider
 - 12 Click **Start**. If any of the services do not start correctly, check the settings you completed earlier.
 - 13 Do one of the following tasks:
 - To use the Documentum Authorization service (EMCDocumentumAuthProviderService) to display content from a Documentum repository in the Resources view of Workbench, continue with this procedure. Using the Documentum Authorization service overrides the default LiveCycle authorization and must be configured to log in to Workbench using Documentum credentials.
 - To use the LiveCycle repository, log in to Workbench by using the LiveCycle super administrator credentials (by default, *administrator* and *password*).

You have now completed the required steps for this procedure. Use the credentials provided in step 19 for accessing the default repository in this case and use the default LiveCycle authorization service.
 - 14 Restart the application server.
 - 15 Log in to Administration Console and click **Settings > User Management > Domain Management**.
 - 16 Click **New Enterprise Domain**, and type a domain ID and name. The domain ID is the unique identifier for the domain. The name is a descriptive name for the domain.
***Note:** When using MySQL for your LiveCycle database, use only single-byte (ASCII) characters for the ID. (See “Adding enterprise domains” in LiveCycle Administration Help.)*
 - 17 Add a custom authentication provider:
 - Click **Add Authentication**.
 - In the Authentication Provider list, select **Custom**.
 - Select **EMCDocumentumAuthProvider** and then click **OK**.
 - 18 Add an LDAP authentication provider:
 - Click **Add Authentication**.
 - In the Authentication Provider list, select **LDAP**, and then click **OK**.

19 Add an LDAP directory:

- Click **Add Directory**.
- In the Profile Name box, type a unique name, and then click **Next**.
- Specify values for the **Server**, **Port**, **SSL**, **Binding**, and **Populate page with** options. If you select User for the Binding option, you must also specify values for the **Name** and **Password** fields.
- (Optional) Select **Retrieve Base DN** to retrieve base domain names, as required.
- Click **Next**, configure the user settings, click **Next**, configure group settings, as required, and then click **Next**.

For details about the settings, click **User Management Help** in the upper-right corner of the page.

20 Click **OK** to exit the Add Directory page and then click OK again.

21 Select the new enterprise domain and click **Sync Now**. Depending on the number of users and groups in your LDAP network and the speed on your connection, the synchronization process may take several minutes.

(Optional) To verify the status of the synchronization, click **Refresh** and view the status in the Current Sync State column.

22 Navigate to **Settings > User Management > Users and Groups**.

23 Search for users that were synchronized from LDAP and perform these tasks:

- Select one or more users and click **Assign Role**.
- Select one or more LiveCycle roles and click **OK**.
- Click **OK** a second time to confirm the role assignment.

Repeat this step for all users that you assign roles to. For more information, click **User Management Help** in the upper-right corner of the page.

24 Start Workbench and log in by using the credentials for the Documentum repository:

Username: `[username]@[repository_name]`

Password: `[password]`

After you log in, the Documentum repository appears in the Resources view within Workbench. If you do not log in using the `username@repository_name`, Workbench attempts to log in to the default repository.

25 (Optional) To install the LiveCycle Samples for Connector for EMC Documentum, create a Documentum repository named Samples, and then install the samples in that repository.

After you configure the Connector for EMC Documentum service, see *LiveCycle Administration Help* for information about configuring Workbench with your Documentum repository.

6.11.1 Creating the XDP MIME format in a Documentum repository

Before users can store and retrieve XDP files from a Documentum repository, you must do one of these tasks:

- Create a corresponding XDP format in each repository where users will access XDP files.
- Configure the Connector for EMC Documentum service to use a Documentum Administrator account when accessing the Documentum repository. In this case, the Connector for EMC Documentum service uses the XDP format whenever it is required.

Create the XDP format on Documentum Content Server using Documentum Administrator

1 Log in to Documentum Administrator.

- 2 Click **Formats** and then select **File > New > Format**.
- 3 Type the following information in the corresponding fields:
Name: xdp
Default File Extension: xdp
Mime Type: application/xdp
- 4 Repeat steps 1 to 3 for all other Documentum repositories where users will store XDP files.

Configure the Connector for EMC Documentum service to use a Documentum Administrator

- 1 Open a web browser and enter this URL:
`http://[host]:[port]/adminui`
- 2 Log in using the default user name and password:
User name: administrator
Password: password
- 3 Click **Services > LiveCycle 11 Connector for EMC Documentum > Configuration Settings**.
- 4 Under Documentum Principal Credentials Information, update the following information and then click **Save**:
User Name: *[Documentum Administrator user name]*
Password: *[Documentum Administrator password]*
- 5 Click **Repository Credentials Settings**, select a repository from the list or, if none exist, click **Add**.
- 6 Provide the appropriate information in the corresponding fields and then click **Save**:
Repository Name: *[Repository Name]*
Repository Credentials User Name: *[Documentum Administrator user name]*
Repository Credentials Password: *[Documentum Administrator password]*
- 7 Repeat steps 5 and 6 for all repositories where users will store XDP files.

6.11.2 Add support for multiple connection brokers

LiveCycle Configuration Manager supports configuring only one connection broker. Use LiveCycle Administrator Console to add support for multiple connection brokers:

- 1 Open LiveCycle Administrator Console.
- 2 Navigate to **Home > Services > LiveCycle 11 Connector for EMC Documentum > Configuration Settings**.
- 3 In the **Connection broker Host Name or IP Address**, enter comma separated list of hostnames of different connection brokers. For example, host1, host2, host3.
- 4 In the **Port Number of Connection broker**, enter comma separated list of the ports of corresponding connection brokers. For example, 1489, 1491, 1489.
- 5 Click **Save**.

6.12 Configuring the Connector for IBM Content Manager

Note: AEM forms supports IBM Content Manager. See the [Supported Platform Combinations](#) document and make sure your ECM is upgraded to the supported version.

Note: Ensure that installing client for the connectors, copying of JAR's file and configuration changes tasks are performed on all the nodes of the cluster.

If you installed the Connector for IBM Content Manager as part of your LiveCycle, complete the following procedure to configure the service to connect to the IBM Content Manager datastore.

Configure Connector for IBM Content Manager

1 Locate the adobe-component-ext.properties file in the `[appserver root]/bin` folder. If the file does not exist, create it.

2 Add a new system property that provides the location of the following IBM II4C JAR files:

- cmb81.jar
- cmbcm81.jar
- cmbicm81.jar
- cmblog4j81.jar
- cmbsdk81.jar
- cmbutil81.jar
- cmbutilicm81.jar
- cmbview81.jar
- cmbwas81.jar
- cmbwcm81.jar
- cmgmt

Note: cmgmt is not a JAR file. On Windows, by default, this folder is at `C:/Program Files/IBM/db2cmv8/`.

- common.jar
- db2jcc.jar
- db2jcc_license_cisuz.jar
- db2jcc_license_cu.jar
- ecore.jar
- ibmjgssprovider.jar
- ibmjsseprovider2.jar
- ibmpkcs.jar
- icmrm81.jar
- jcache.jar
- log4j-1.2.8.jar
- xerces.jar
- xml.jar

- xsd.jar

The new system property looks similar to the following:

[component id].ext=[JAR files and/or folders]

For example, using a default DB2 Universal Database Client and II4C installation, in the file, add the following system property on a new line, with no line breaks, and end the line with a carriage return:

```
C:/Program Files/IBM/db2cmv8/cmgmt,  
C:/Program Files/IBM/db2cmv8/java/jre/lib/ibmjsseprovider2.jar,  
C:/Program Files/IBM/db2cmv8/java/jre/lib/ibmjgssprovider.jar,  
C:/Program Files/IBM/db2cmv8/java/jre/lib/ibmpkcs.jar,  
C:/Program Files/IBM/db2cmv8/java/jre/lib/xml.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbview81.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmb81.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbcm81.jar,  
C:/Program Files/IBM/db2cmv8/lib/xsd.jar,  
C:/Program Files/IBM/db2cmv8/lib/common.jar,  
C:/Program Files/IBM/db2cmv8/lib/ecore.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbicm81.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbwcm81.jar,  
C:/Program Files/IBM/db2cmv8/lib/jcache.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbutil81.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbutilicm81.jar,  
C:/Program Files/IBM/db2cmv8/lib/icmrm81.jar,  
C:/Program Files/IBM/db2cmv8/lib/db2jcc.jar,  
C:/Program Files/IBM/db2cmv8/lib/db2jcc_license_cu.jar,  
C:/Program Files/IBM/db2cmv8/lib/db2jcc_license_cisuz.jar,  
C:/Program Files/IBM/db2cmv8/lib/xerces.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmblog4j81.jar,  
C:/Program Files/IBM/db2cmv8/lib/log4j-1.2.8.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbstdk81.jar,  
C:/Program Files/IBM/db2cmv8/lib/cmbwas81.jar
```

- 3 If the application server is not currently running, start the server; otherwise, stop and then restart the server.

You can now connect to the IBM Content Manager datastore from the IBMCMConnectorService Property Sheets by using the Use User Credentials as the login mode.

You have now completed the required steps for this procedure.

(Optional) If you want to connect to IBM Content Manager datastore from IBMCMConnectorService Property Sheets by using the Use Credentials From Process Context as the login mode, complete the following procedure.

Connect using Use Credentials from process context login mode

- 1 Open a web browser and enter this URL:

`http://[host]:[port]/adminui`

- 2 Log in using the super administrator credentials. Default values set during installation are:

User name:*administrator*

Password:*password*

- 3 Click **Services > LiveCycle 11 Connector for IBM Content Manager**
- 4 Type all of the required repository information and click **Save**. For more information about the IBM Content Manager repository information, click the **Help** link in the upper-right corner of the page.

5 Do one of these tasks:

- To use the IBM Content Manager Authorization service IBMCMAuthProvider to use content from an IBM Content Manager datastore, in the Processes view of Workbench, continue with this procedure. Using the IBM Content Manager Authorization service overrides the default LiveCycle authorization and must be configured to log in to Workbench by using IBM Content Manager credentials.
- To use the System Credentials provided in step 4 to use content from an IBM Content Manager datastore, in the Processes view of Workbench, log in to Workbench by using the LiveCycle super administrator credentials (by default, *administrator* and *password*). You have now completed the required steps for this procedure. The System Credentials that are provided in step 4 use the default LiveCycle authorization service for accessing the default repository in this case.

6 Log in to the Administration Console, and click **Settings > User Management > Domain Management**.

7 Click **New Enterprise Domain** and type a domain ID and name. The domain ID is the unique identifier for the domain. The name is a descriptive name for the domain.

***Note:** When using MySQL for your LiveCycle database, use only single-byte (ASCII) characters for the ID. (See Adding enterprise domains in [Administration Help](#).)*

8 Add a custom authentication provider:

- Click **Add Authentication**.
- In the **Authentication Provider** list, select **Custom**, and then select **IBMCMAuthProviderService** and click **OK**.

9 Add an LDAP authentication provider:

- Click **Add Authentication**.
- In the **Authentication Provider** list, select **LDAP** and then click **OK**.

10 Add an LDAP directory:

- Click **Add Directory**.
- In the **Profile Name** box, type a unique name, and then click **Next**.
- Specify values for the **Server**, **Port**, **SSL**, **Binding**, and **Populate page with** options. If you select **User** for the **Binding** option, you must also specify values for the **Name** and **Password** fields. (Optional) Select **Retrieve Base DN** to retrieve base domain names, as required. When finished, click **Next**.
- Configure the user settings, click **Next**, configure group settings as required, and then click **Next**.

For details about the above settings, click the **Help** link in the upper-right corner of the page.

11 Click **OK** to exit the Add Directory page and click **OK** again.

12 Select the new enterprise domain and click **Sync Now**. Depending on the number of users and groups in your LDAP network and the speed on your connection, the synchronization process may take several minutes.

13 To verify the status of the synchronization, click **Refresh** and view the status in the **Current Sync State** column.

14 Navigate to **Settings > User Management > Users and Groups**.

15 Search for users that were synchronized from LDAP and do these tasks:

- Select one or more users and click **Assign Role**.
- Select one or more LiveCycle roles and click **OK**.
- Click **OK** a second time to confirm the role assignment.

Repeat this step for all users that you want to assign roles to. For more information, click the **Help** link in the upper-right corner of the page.

16 Start Workbench and log in using the following credentials for IBM Content Manager datastore:

Username: [username]@[repository_name]

Password: [password]

The IBM Content Manager datastore can now be used in the Processes view within Workbench when the login mode for IBMCMConnectorService orchestrable components is selected as **Use Credentials from process context**.

6.13 Configuring the Connector for IBM FileNet

LiveCycle supports IBM FileNet, versions 4.0, 4.5, 5.0, and 5.2 only. Make sure your ECM is upgraded accordingly.

***Note:** LiveCycle supports FileNet 5.2 Content Engine; FileNet 5.2 Process Engine is not supported. Moreover, the LiveCycle Configuration Manager (LCM) does not support/configure FileNet 5.2, perform manual configuration for FileNet 5.2.*

If you installed Connector for IBM FileNet as part of your LiveCycle, you must configure the service to connect to the FileNet object store.

***Note:** Ensure that installing client for the connectors, copying of JAR's file and configuration changes tasks are performed on all the nodes of the cluster.*

Complete the following procedure to configure Connector for IBM FileNet.

Configure Connector for IBM FileNet using FileNet 4.x or FileNet 5.x and CEWS transport

1 Open the application server run file in a text editor. The run file is as follows:

- (Windows) [appserver root]/bin/run.conf
- (Non-Windows) [appserver root]/bin/run.conf

2 **(Only for FileNet 4.x)** Add the location of the FileNet Configuration files as a Java option to the application server start command, and then save the file.

***Note:** If JBoss is running as a service, add the Java option in the registry where other JVM arguments are defined.*

```
-Dwaspl.location= <configuration files location>
```

For example, using a default FileNet Application Engine installation on a Windows operating system, add this Java option:

```
-Dwaspl.location=C:/Program Files/FileNet/AE/CE_API/wsi
```

3 If your deployment uses the Process Engine Connector service, copy the file [appserver root]/client/logkit.jar to the following directory:

- **(Manually-configured JBoss, cluster)** [appserver root]/server/all/lib
- **(Manually-configured JBoss, single server)** [appserver root]/server/standard/lib
- **(Adobe-preconfigured JBoss, cluster)** [appserver root]/server/lc_<db-name>_cl/lib
- **(Adobe-preconfigured JBoss, single server)** [appserver root]/server/lc_<db-name>/lib

4 Locate the adobe-component-ext.properties file in the [appserver root]/bin folder (if the file does not exist, create it).

5 Add a new system property that provides the location of these FileNet Application Engine JAR files:

For FileNet 4.x add following JAR files.

- javaapi.jar

- soap.jar
- wasp.jar
- builtin_serialization.jar (FileNet 4.0 only)
- wsdl_api.jar
- jaxm.jar
- jaxrpc.jar
- saaj.jar
- jetty.jar
- runner.jar
- p8cjares.jar
- Jace.jar
- (optional) pe.jar

For FileNet 5.x add following JAR files

- Jace.jar
- javaapi.jar
- log4j.jar
- pe.jar
- stax-api.jar
- xlsxScanner.jar
- xlsxScannerUtils.jar

Note: Add the *pe.jar* file only if your deployment uses the *IBMFileNetProcessEngineConnector* service. The new system property should reflect this structure:

```
[component id].ext=[JAR files and/or folders]
```

For example, using a default FileNet Application Engine installation on a Windows operating system, add the following system property on a new line with no line breaks and end the line with a carriage return:

Note: The following text contains formatting characters for line breaks. If you copy this text to a location outside this document, remove the formatting characters when you paste it to the new location.

```
com.adobe.livecycle.ConnectorforIBMFileNet.ext=  
C:/Program Files/FileNet/AE/CE_API/lib2/javaapi.jar,  
C:/Program Files/FileNet/AE/CE_API/lib2/log4j-1.2.13.jar
```

6 (FileNet Process Engine Connector only) Configure the connection properties for the process engine as follows:

- Using a text editor, create a file with the following content as a single line and end the line with a carriage return:
(FileNet 4.x and 5.0 only)

```
RemoteServerUrl = cemp:http://[contentserver_IP]:[contentengine_port]/wsi/FNCEWS40DIME/
```

(FileNet 5.2 only)

```
RemoteServerUrl = cemp:http://[contentserver_IP]:[contentengine_port]/wsi/FNCEWS40MTOM/
```

- Save the file as *WcmApiConfig.properties* in a separate folder, and add the location of the folder that contains the *WcmApiConfig.properties* file to the *adobe-component-ext.properties* file.

For example, if you save the file as `c:/pe_config/WcmApiConfig.properties`, add the path `c:/pe_config` to the `adobe-component-ext.properties` file.

Note: The filename is case-sensitive.

- 7 Locate the `login-config.xml` file in the following folder and add the following application policy as a child of the `<policy>` node:

- **(Manually-configured JBoss, single server)** `[appserver root]/server/standard/conf`
- **(Manually-configured JBoss, cluster)** `[appserver root]/server/all/conf`
- **(Adobe-preconfigured JBoss, single server)** `[appserver root]/server/lc_<dbname>/conf`
- **(Adobe-preconfigured JBoss, cluster)** `[appserver root]/server/lc_<dbname>_cl/conf`

```
<application-policy name = "FileNetP8WSI">
  <authentication>
    <login-module code = "com.filenet.api.util.WSILoginModule" flag =
      "required" />
  </authentication>
</application-policy>
```

- 8 (FileNet Process Engine Connector only) If your deployment uses the process engine, add the following node to the `login-config` file:

```
<application-policy name = "FileNetP8">
  <authentication>
    <login-module code = "com.filenet.api.util.WSILoginModule" flag =
      "required" />
  </authentication>
</application-policy>
```

- 9 If the application server is not currently running, start the server. Otherwise, stop and then restart the server.

- 10 If JBoss runs as a service, start (or restart) the JBoss for Adobe LiveCycle ES4 Service.

- 11 (**Cluster only**) Repeat all previous steps on each instance on the cluster.

- 12 Open a web browser and enter this URL:

`http://[host]:[port]/adminui`

- 13 Log in using the default user name and password:

User name: administrator

Password: password

- 14 Click **Services > LiveCycle 11 Connector for IBM FileNet**.

- 15 Provide the Content Engine URL. For example,

`cemp:http://ContentEngineHostNameorIP:port/wsi/FNCEWS40MTOM?jaasConfigurationName=FileNetP8WSI`

- 16 Provide all of the required FileNet repository information and, under Repository Service Provider Information, select **IBM FileNet Repository Provider**.

If your deployment uses the optional process engine service, under Process Engine Settings, select **Use Process Engine Connector Service** and specify the process engine settings. For more information, click the **Help** link in the upper-right corner of the page.

Note: The credentials that you provide in this step are validated later when you start the IBM FileNet repository services. If the credentials are not valid, an error is thrown and the services will not start.

- 17 Click **Save** and navigate to **Services > Applications and Services > Service Management**.

18 Select the check box next to each of these services and then click **Start**:

- IBMFileNetAuthProviderService
- IBMFileNetContentRepositoryConnector
- IBMFileNetRepositoryProvider
- IBMFileNetProcessEngineConnector (if configured)

If any of the services do not start correctly, verify the Process Engine settings.

19 Do one of the following tasks:

- To use the FileNet Authorization service (IBMFileNetAuthProviderService) to display content from a FileNet object store in the Resources view of Workbench, continue with this procedure. Using the FileNet Authorization service overrides the default LiveCycle authorization and must be configured to log in to Workbench by using FileNet credentials.
- To use the LiveCycle repository, log in to Workbench by using the LiveCycle super administrator credentials (by default, *administrator* and *password*). The credentials provided in step 16 use the default LiveCycle authorization service for accessing the default repository in this case.

20 Restart your application server.

21 Log in to Administration Console and click **Settings > User Management > Domain Management**.

22 Click **New Enterprise Domain** and then type a domain ID and name. The domain ID is the unique identifier for the domain. The name is a descriptive name for the domain.

When using MySQL for your LiveCycle database, use only single-byte (ASCII) characters for the ID. (See “Adding enterprise domains” in [LiveCycle Administration](#) Help)

23 Add a custom authentication provider:

- Click **Add Authentication**.
- In the **Authentication Provider** list, select **Custom**.
- Select **IBMFileNetAuthProviderService** and then click **OK**.

24 Add an LDAP authentication provider:

- Click **Add Authentication**.
- In the **Authentication Provider** list, select **LDAP** and then click **OK**.

25 Add an LDAP directory:

- Click **Add Directory** and, in the **Profile Name** box, type a unique name, and then click **Next**.
- Specify values for the **Server**, **Port**, **SSL**, **Binding**, and **Populate page with** options. If you select **User** for the **Binding** option, you must also specify values for the **Name** and **Password** fields.
- (Optional) Select **Retrieve Base DN** to retrieve base domain names, as required. When finished, click **Next**.
- Configure the user settings, click **Next**, configure group settings as required, and then click **Next**.

For details about the settings, click **Help** link in the upper-right corner of the page.

26 Click **OK** to exit the Add Directory page, and then click **OK** again.

27 Select the new enterprise domain and click **Sync Now**. Depending on the number of users and groups in your LDAP network and the speed on your connection, the synchronization process may take several minutes.

(Optional) To verify the status of the synchronization, click **Refresh** and view the status in the **Current Sync State** column.

28 Navigate to **Settings > User Management > Users and Groups**.

29 Search for users that were synchronized from LDAP and perform these tasks:

- Select one or more users and click **Assign Role**.
- Select one or more LiveCycle roles and click **OK**.
- Click **OK** a second time to confirm the role assignment.

Repeat this step for all users you want to assign roles to. For more information, click the **Help** link in the upper-right corner of the page.

30 Start Workbench and log in using the following credentials for the IBM FileNet repository:

User name: `[username]@[repository_name]`

Password: `[password]`

The FileNet object store should now be visible in the Resources view within Workbench. If you do not log in using the `username@repository name`, Workbench attempts to log in to the default repository specified in step 16.

31 (Optional) If you intend to install the LiveCycle Samples for Connector for IBM FileNet, create a FileNet object store named *Samples* and install the samples in that object store.

After you configure Connector for IBM FileNet, it is recommended that you see LiveCycle Administration Help for information about configuring Workbench functions properly with your FileNet repository.

6.14 Isolating JBoss Clusters

There are a lot of JBoss services that create multiple JGroup channels services. These channels should only communicate with specific channels.

To isolate JGroups clusters from other clusters on the network, ensure that

- The channels in the various clusters use different group names. Use `./run.sh -g QAPartition -b <ipaddress> -c all` to create unique groups.
- The channels in the various clusters use different multicast addresses. Use `/run.sh -u <UDP group Ip address> -g QAPartition -b <ipaddress> -c all` to control the multicast address.
- The channels in each cluster use different multicast ports. Use `/run.sh -u <UDP group Ip address> -g QAPartition -b <ipaddress> -c all \\-Djboss.jgroups.udp.mcast_port=12345 -Djboss.messaging.datachanneludpport=23456` to control the muticast sockets.

See, Isolating JGroups Channels in jbossclustering guide at <http://docs.jboss.org/> for detailed information to isolate JBoss Clusters

6.15 (Optional)Enable JMX console security

In the default setup of LiveCycle, JBoss JMX console security is disabled. To enable the security, follow the steps mentioned below:

- 1 Shut down the application server.
- 2 Navigate to the `[appserver root]/server/<profile_name>/deploy` directory and open the `jmx-invoker-service.xml` file in a text editor.

Post-deployment tasks

- 3 Ensure that the following line is not commented out in the `invoke` section:

```
<interceptor code="org.jboss.jmx.connector.invoker.AuthenticationInterceptor"
securityDomain="java:/jaas/jmx-console"/>
```

- 4 Save and close the file.
- 5 Create a new file, `work-manager.properties`, at `[appserver root]/server/`.
- 6 Open `work-manager.properties` file in a text editor and add following code:

```
adobe.work-
manager.jboss.jmx.lookup.java.naming.factory.initial=org.jboss.security.jndi.JndiLo
ginInitialContextFactory
adobe.work-manager.jboss.jmx.lookup.java.naming.provider.url=jnp://localhost:1099/
adobe.work-manager.jboss.jmx.lookup.java.naming.security.credentials=<password>
adobe.work-manager.jboss.jmx.lookup.java.naming.security.principal=<username>
adobe.work-manager.jboss.jmx.lookup.java.naming.security.protocol=jmx-console
```

Note: Ensure that identical credentials are mentioned in `jmx-console-users.properties` file and in `work-manager.properties` file. The default credentials are `admin/admin`.

- 7 Save and close the file.
- 8 Navigate to the `[appserver root]/server/<profile_name>/conf/props` directory and open `jmx-console-users.properties` file in a text editor.
- 9 Ensure that an entry that contains credentials used in the `work-manager.properties` file is not commented out.
- 10 Save and close the file.
- 11 Navigate to `[appserver root]/bin` and open `run.conf.bat` file in a text editor and add following code:

```
set "JAVA_OPTS=%JAVA_OPTS% -Dadobe.workmanager.properties = <path of the work-
manager.properties file>
```
- 12 Save and close the file.
- 13 Start the application server.

Chapter 7: Configuring Load Balancing

You can configure your JBoss cluster to provide load-balancing functionality. You can use a load balancer to distribute the workload evenly across all nodes of your cluster. Apache web server and various plug-in are used to implement load balancing for the cluster. LiveCycle Cluster supports only stickiness enabled load balancer.

Obtain the Apache web server software that is applicable to your operating system:

- For Windows, download the Apache web server from the Apache HTTP Server Project site.
- For Solaris 64 bit, download the Apache web server from the Sunfreeware for Solaris Website.
- For Linux, the Apache web server is preinstalled on a Linux system.

Apache can communicate to JBoss using HTTP or AJP protocol. The following are configurations for Load balancing using both protocols.

7.1 HTTP Connector Based Load Balancing

7.1.1 Using mod_proxy

This configuration uses Apache proxy balancer module along with proxy module to load balance JBoss cluster on HTTP protocol.

Uncomment the following module configurations in `APACHE_HOME/conf/httpd.conf` file

```
LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
LoadModule proxy_balancer_module modules/mod_proxy.so
LoadModule proxy_balancer_module modules/mod_proxy_http.so
```

Note: For Linux, the default `APACHE_HOME` is `/etc/httpd/`

There are two methods to implement mod_proxy based load balancing:

- Using mod_headers
- Using JSESSIONID

7.1.1.1 Using mod_headers

This configuration provides a way for load balancing with stickiness using mod_headers.

Add following configuration in `APACHE_HOME/conf/httpd.conf` configuration file.


```
LoadModule headers_module modules/mod_headers.so
<VirtualHost *:80>
ServerName lb.xyz.com
ServerAlias lb.xyz.com
ProxyRequests Off
ProxyPreserveHost On
Header add Set-Cookie "ROUTEID=.%{BALANCER_WORKER_ROUTE}e; path=/" env=BALANCER_ROUTE_CHANGED
<Proxy balancer://cluster>
BalancerMember http://abc.com:8080 route=1
BalancerMember http://xyz.com:8080 route=2
ProxySet stickysession=ROUTEID
</Proxy>
ProxyPass / balancer://cluster/
ProxyPassReverse / balancer://cluster/

</VirtualHost>
```

7.1.1.2 Using JSESSIONID

This configuration provides a way for load balancing using JSESSIONID.

In JBoss application server, tomcat container adds the name of the instance to the end of its session id cookie, separated with a dot (.) from the session id. Thus if the Apache web server finds a dot in the value of the session persistence cookie, it only uses the part behind the dot to search for the route. For Tomcat Server instance to be aware of its instance name, set the attribute `jvmRoute` inside the JBoss configuration file `deploy/jbossweb.sar/server.xml` to the value of the route of the worker that connects to the respective JBoss. The name of the session cookie used here is JSESSIONID.

Add following configuration to `APACHE_HOME/conf/httpd.conf` file.

```
<VirtualHost *:80>
ServerName lb.xyz.com
ServerAlias lb.xyz.com
ProxyRequests Off
ProxyPreserveHost On
<Proxy balancer://cluster>
BalancerMember http:// abc.com:8080 route=node1
BalancerMember http:// xyz.com:8080 route=node2
</Proxy>
ProxyPass / balancer://cluster/ stickysession=JSESSIONID|jsessionid
ProxyPassReverse / balancer://cluster/
</VirtualHost>
```

Server-Side Configuration

- 1 For each node in the cluster, open the `server.xml` file in a text editor from this location: `[appserver root]/server/<profile_name>/deploy/jbossweb.sar`
- 2 Search the `server.xml` file for the `Engine` name element and add a `jvmRoute` attribute. For example, on a node named `node1`, edit the element to read as follows: `<Engine name="jboss.web" defaultHost="localhost" jvmRoute="node1">`
- 3 Save the edited `server.xml` file.

Note : For each cluster node, the value of `jvmRoute` must be unique and match to the value of the route of the worker that connects to the respective JBoss Server(See `BalancerMember` Configuration in `httpd.conf`).

7.2 AJP Connector Based Load Balancing

7.2.1 Using mod_jk

This configuration uses Apache mod_jk plug-in to load balance JBoss cluster on AJP protocol. It provides support for the Apache JServ Protocol (AJP).

Perform following steps to configure:

- 1 From the Apache Tomcat Connector site, download the mod_jk plug-in file for your operating system.
Note: Ensure that your Apache server supports downloaded mod_jk plug-in file.
- 2 Rename the downloaded file to mod_jk.so and save it in the APACHE_HOME/modules/ directory.
- 3 In a text editor, open the httpd.conf file located in APACHE_HOME/conf and add the following line at the end of the file: Include conf/mod-jk.conf
- 4 Create a new file APACHE_HOME/conf/mod-jk.conf with the following content :

```
# Load mod_jk module
# Specify the filename of the mod_jk lib
LoadModule jk_module modules/mod_jk.so
# Where to find workers.properties
JkWorkersFile conf/workers.properties
# Where to put jk logs
JkLogFile logs/mod_jk.log
# Set the jk log level [debug/error/info]
JkLogLevel info
# Select the log format
JkLogStampFormat "[%a %b %d %H:%M:%S %Y]"
# JkOptions indicates to send SSK KEY SIZE
JkOptions +ForwardKeySize +ForwardURICompat -ForwardDirectories
# JkRequestLogFormat
JkRequestLogFormat "%w %V %T"
# Mount your applications
JkMount /* loadbalancer
# You can use external file for mount points.
# It will be checked for updates each 60 seconds.
# The format of the file is: /url=worker
# /examples/*=loadbalancer
#JkMountFile conf/uriworkermap.properties
# Add shared memory.
# This directive is present with 1.2.10 and
# later versions of mod_jk, and is needed
# for load balancing to work properly
JkShmFile logs/jk.shm
# Add jkstatus for managing run-time data
<Location /jkstatus/>
JkMount status
Order deny,allow
Deny from all
Allow from 127.0.0.1
</Location>
```

- 5 Create a file at conf/workers.properties with following content

```
# for mapping requests
worker.list=loadbalancer,status
# Define Node1
# modify the host as your host IP or DNS name.
worker.node1.port=8009
worker.node1.host=node1.mydomain.com
worker.node1.type=ajp13
worker.node1.lbfactor=1
worker.node1.cachesize=10
# Define Node2
# modify the host as your host IP or DNS name.
worker.node2.port=8009
worker.node2.host= node2.mydomain.com
worker.node2.type=ajp13
worker.node2.lbfactor=1
worker.node2.cachesize=10
# Load-balancing behavior
worker.loadbalancer.type=lb
worker.loadbalancer.balance_workers=node1,node2
worker.loadbalancer.sticky_session=1
#worker.list=loadbalancer
# Status worker for managing load balancer
worker.status.type=status
```

6 In the file, define following items:

- Each node of the cluster (in this example, two nodes named node1 and node2)
- Add all the nodes defined in the file to the **worker.loadbalancer.balance_workers** entry.

7 For each node in the cluster, open the server.xml file located at [appserver root]/server/<profile_name>/deploy/jbossweb.sar for editing.

8 Search the server.xml file for the Engine name element and add a jvmRoute attribute. For example, on a node named node1 should be:<Engine name="jboss.web" defaultHost="localhost" jvmRoute="node1">

9 Save the edited server.xml file.

Note: For each cluster node value of jvmRoute must be unique and match to the value of the worker that connects to the respective JBoss.

7.2.2 Using mod_proxy_ajp

This configuration uses Apache mod_proxy_ajp plugin to load balance JBoss cluster on AJP protocol. This module requires the service of [mod_proxy](#). Thus, in order to get the ability of handling AJP protocol, mod_proxy and mod_proxy_ajp should be present in the server.

This module is used to reverse proxy to a backend application server using the AJP protocol. The usage is similar to an HTTP reverse proxy, but uses the ajp:// prefix:

Uncomment the following module configurations in APACHE_HOME/conf/httpd.conf file

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_balancer_module modules/mod_proxy_balancer.so
LoadModule proxy_ajp_module modules/mod_proxy_ajp.so
```

Add following configuration to the APACHE_HOME/conf/httpd.conf file.

```
<VirtualHost *:80>
ServerName lb.xyz.com
ServerAlias lb.xyz.com
ProxyRequests Off
ProxyPreserveHost On
<Proxy balancer://cluster>
BalancerMember ajp://abc.com:8009 route=node1
BalancerMember ajp://xyz.com:8009 route=node2
ProxySet lbmethod=byrequests
</Proxy>
ProxyPass / balancer://cluster/ stickysession=JSESSIONID|jsessionid
</VirtualHost>
```

Usually, no ProxyPassReverse directive is necessary. The AJP request includes the original host header given to the proxy, and the application server can be expected to generate self-referential headers relative to this host, so no rewriting is required.

Server-Side Configuration:

- 1 For each node in the cluster, open the server.xml file located at [appserver root]/server/<profile_name>/deploy/jbossweb.sar for editing.
- 2 Search the server.xml file for the Engine name element and add a jvmRoute attribute. For example, on a node named node1, edit the element to read as follows: <Engine name="jboss.web" defaultHost="localhost" jvmRoute="node1">
- 3 Save the edited server.xml file.

Note: For each cluster, the node value of jvmRoute must be unique and match to the value of the route of the worker that connects to the respective JBoss (See BalancerMember Configuration in httpd.conf).

Chapter 8: Advanced Production Configuration

This section describes advanced tuning for Adobe® LiveCycle® Output 11, Adobe® LiveCycle® Forms Standard 11, and PDF Generator. This section should be completed only on a production system by an advanced application server administrator.

8.1 Configuring pool size for Output and Forms

The current default value for PoolMax is 4. The actual value to set depends on the hardware configuration and the expected usage in your environment.

For optimal use, we recommend that the lower limit of PoolMax not be less than the number of CPUs that are available. The upper limit must be determined by the load pattern on your server. Generally, the upper limit should be set to twice the number of CPUs cores on your server.

Modify the existing PoolMax value

- 1 Using a text editor, edit the JBoss startup script.
- 2 Add the following properties for `ConvertPdf`:
 - `com.adobe.convertpdf.bmc.POOL_MAX=[new value]`
 - `com.adobe.convertpdf.bmc.MAXIMUM_REUSE_COUNT=5000`
 - `com.adobe.convertpdf.bmc.REPORT_TIMING_INFORMATION=true`
 - `com.adobe.convertpdf.bmc.CT_ALLOW_SYSTEM_FONTS=true`
- 3 Add the following properties for `XMLFM`:
 - `com.adobe.xmlform.bmc.POOL_MAX=[new value]`
 - `com.adobe.xmlform.bmc.MAXIMUM_REUSE_COUNT=5000`
 - `com.adobe.xmlform.bmc.REPORT_TIMING_INFORMATION=true`
 - `com.adobe.xmlform.bmc.CT_ALLOW_SYSTEM_FONTS=true`

8.2 PDF Generator

PDF Generator is capable of doing multiple PDF conversions simultaneously for some types of input files. This is enforced through the use of stateless session beans.

8.2.1 Configuring EJB Pool Size

Four different stateless session beans exist for enforcing independent pool sizes for the following types of input files:

- Adobe PostScript® and Encapsulated PostScript (EPS) files
- Image files, such as BMP, TIFF, PNG, and JPEG files

- OpenOffice files
- All other file types (except HTML files), such as Microsoft Office, Photoshop®, PageMaker®, and FrameMaker® files

The pool size for HTML-to-PDF conversions is not managed through the use of stateless session beans.

The default pool size for PostScript and EPS files and for image files is set to 3, and the default pool size for OpenOffice and other file types (except HTML) is set to 1.

You can configure the PS/EPS and image pool size to a different value based on your server hardware configuration, such as the number of CPUs, the number of cores within each CPU, and so on. However, it is mandatory that the pool size for the OpenOffice and other file types be left unchanged at 1 for proper functioning of PDF Generator.

This section describes how the pool size for PS2PDF and Image2PDF can be configured for each of the supported application servers.

The text that follows assumes that the following two LiveCycle application EARs are deployed on the application server:

- adobe-livecycle-jboss.ear
- adobe-livecycle-native-jboss-[platform].ear

where [platform] should be replaced with one of the following strings, depending on your operating system:

- (Windows) x86_win32
- (Linux) x86_linux
- (SunOS™) sparc_sunos

Configure the pool size for PS2PDF and Image2PDF

Refer to Distiller service settings and Generate PDF service settings under “Managing services” in the LiveCycle Administration Help.

8.3 Enabling CIFS on Windows

You will need to manually configure the Windows Server machine that host LiveCycle.

Note: Ensure that the server has a static IP address.

On Windows machines, you need to do the following:

8.3.1 Enable NetBIOS over TCP/IP

You need to enable NetBIOS over TCP/IP so that clients connecting to the LiveCycle Server can have their requests revolved for the server host name.

- 1 In the **Local Area Connection Properties** dialog box, on the **General** tab, select **Internet Protocol**, and then click **Properties**.
- 2 In the **General** tab of the **Internet Protocol (TCP/IP) Properties** dialog box, ensure that the server has a static IP address. Click **Advanced**.
- 3 In the **Advanced TCP/IP Settings** dialog box, select the **WINS** tab and select **Enable NetBIOS over TCP/IP**.

8.3.2 Add additional IP addresses

- 1 In the **Local Area Connection Properties** dialog box, on the **General** tab, select **Internet Protocol**, and then click **Properties**.
- 2 In the **General** tab of the **Internet Protocol (TCP/IP) Properties** dialog box, ensure that the server has a static IP address. Click **Advanced**.
- 3 In the **Advanced TCP/IP Settings** dialog box, select the **IP Settings** tab and click **Add**.
- 4 Specify a static IP address and click **Add**.

8.3.3 Disable File and Printer Sharing (Windows Server 2008 only)

- Go to **Network Settings**, deselect **File and Printer Sharing for Microsoft Clients**, and click **Apply**.

Chapter 9: Appendix - Install Command Line Interface

9.1 Overview

LiveCycle provides a command line interface (CLI) for the installation program. The CLI is intended to be used by advanced users of LiveCycle or in server environments which do not support the use of the Graphical User Interface (GUI) of the installation program. The CLI runs in console mode with one interactive session for all install operations.

Before you install the modules using the CLI install option, ensure that you have prepared your environment required to run LiveCycle according to the Preparing guide for fresh single server installation, cluster setup, or upgrade, as appropriate. The completed LiveCycle documentation is available at http://www.adobe.com/go/learn_lc_documentation_11.

For an overview of the installation process, see “4.1 Before you begin” on page 17.

After you start the installation process, follow the on-screen instructions to choose your installation options. Respond to each prompt to proceed to the next step in the installation.

Note: If you want to change a choice that you made on a previous step, type *back*. You can cancel the installation at any time by typing *quit*.

9.2 Install LiveCycle

1 Open a command prompt and navigate to the folder in the installation media or your hard disk that contains the installer executable:

- (Windows) server\Disk1\InstData\Windows_64\VM
- (Linux) server/Disk1/InstData/Linux/NoVM
- (Solaris) server/Disk1/InstData/Solaris/NoVM

2 Open a command prompt and run the following command:

- (Windows) `install.exe -i console`
- (Non-Windows) `./install.bin -i console`

Note: Entering the command without the `-i console` option launches the GUI-based installer.

3 Respond to the prompts as described in the following table:

Prompt	Description
Choose Locale	Select the locale for the installation to use by entering a value between 1 and 3. You can select the default value by pressing Enter . The options are Deutsch, English, and Français. English is the default language.
Choose Install Folder	On the Destination screen, press Enter to accept the default directory or type the new installation directory location. Default install folders are: (Windows): C:\Adobe\Adobe LiveCycle ES4 (Non-Windows): /opt/adobe/adobe_lifecycle_es4 Do not use accented characters in the directory name. Otherwise, the CLI will ignore the accents and create a directory after modifying the accented characters.
Choose Operating System	(Windows only) Select the operating system that you want to install LiveCycle to. The options are Windows and Linux, Solaris. Windows (Local) is the default. Select a different target operating system so that you can use the installation on Windows as the staging platform to deploy LiveCycle on to another operating system.
LiveCycle Server License Agreement	Press Enter to read through the pages of the license agreement. If you agree to the agreement, type Y and press Enter .
Pre-Installation Summary	Review the installation choices you have made and press Enter to continue installation with the choices you have made. Type back to go back to previous steps and change any of the settings.
Ready To Install	Installer displays the installation directory. Press Enter to start the installation process.
Installing	During the installation process, the progress bar advances to indicate the progress of installation.
Configuration Manager	Press Enter to complete the installation of LiveCycle. You can run the Configuration Manager in GUI mode by invoking the following script: (Windows): C:\Adobe\Adobe LiveCycle ES4\configurationManager\bin\ConfigurationManager.bat (Non-Windows): /opt/adobe/adobe_lifecycle_es4/configurationManager/bin/ConfigurationManager.sh
Installation Complete	The installation completion screen displays the status and the location of install. Press Enter to exit the installer.

9.3 Error logs

If an error occurs, you can review the install.log in the log directory of your installation:

- (Windows) [LiveCycle root]\log
- (Linux, Solaris) [LiveCycle root]/log

For information about errors that may occur during the installation, see the appropriate troubleshooting guide.

9.4 Uninstalling LiveCycle in console mode

Note: If you had installed LiveCycle using the command line option, you can uninstall LiveCycle ES4 only by running the uninstaller from the command line. If you want a silent uninstallation, omit the “-i console” flag.

- 1 Open a command prompt, and navigate to the directory which contains the uninstall script:

Note: On UNIX systems, you should manually navigate to the directory that contains the uninstall script because the directory name contains spaces.

- (Windows) `cd C:\Adobe\Adobe LiveCycle ES4\Uninstall_Adobe LiveCycle ES4`
- (UNIX-like systems) `cd /opt/adobe/adobe_lifecycle_es4/Uninstall_Adobe Livecycle ES4`

- 2 Type the following command at the prompt and press Enter:

- (Windows) `Uninstall Adobe LiveCycle ES4 -i console`
- (Linux, Solaris) `./Uninstall Adobe Livecycle ES4 -i console`

- 3 Follow the on-screen instructions.

Prompt	Description
Uninstall LiveCycle ES4	Press Enter to continue uninstallation. Enter quit to close the uninstall program.
Uninstalling...	After the uninstallation starts, the rest of the uninstallation process is completed and the cursor returns to the prompt.
Uninstall Complete	Note that some items may not be removed. Also, any folder created after installing LiveCycle are not removed. You must remove these files and folders manually.

Chapter 10: Appendix - Configuration Manager Command Line Interface

LiveCycle provides a Command Line Interface (CLI) for the Configuration Manager. The CLI is intended to be used by advanced users of LiveCycle, for example in server environments which do not support the use of the Graphical User Interface (GUI) of the Configuration Manager.

10.1 Order of operations

The Configuration Manager CLI must follow the same order of operations as the GUI version of the Configuration Manager. Ensure that you use the CLI operations in this order:

- 1 Configure LiveCycle.
- 2 Configure CRX
- 3 Manually deploy the configured EAR files.
- 4 Initialize LiveCycle.
- 5 Validate LiveCycle.
- 6 Deploy the LiveCycle modules.
- 7 Validate the LiveCycle module deployment.
- 8 Check system readiness for PDF Generator.
- 9 Add administrator user for PDF Generator.
- 10 Configure Connector for IBM Content Manager.
- 11 Configure Connector for IBM FileNet.
- 12 Configure Connector for EMC Documentum.
- 13 Configure Connector for SharePoint.

Important: You must restart each of your cluster nodes after you complete Configuration Manager CLI operations.

10.2 Command Line Interface property file

You should create the property file according to your installation. Use one of the following methods.

- Create a property file and populate the values according to your installation and configuration scenarios.
- Use the property file `cli_propertyFile_template.txt` file as a template and edit the values based on the Configuration Manager operations you intend to use.
- Use the GUI of the Configuration Manager and then use the property file created by the GUI version as the CLI version property file. When you run the `[LiveCycle root]/configurationManager/bin/ConfigurationManager.bat/sh` file, the `userValuesForCLI.properties` file is created in the `[LiveCycle root]/configurationManager/config` directory. You can use this file as input for the Configuration Manager CLI.

Note: In the CLI properties file, you must use the escape character (\) for Windows paths directory separator (\). For example, if the Fonts folder to be mentioned is C:\Windows\Fonts, in the Configuration Manager CLI script, you should enter it as C:\Windows\Fonts.

Note: The following modules depend on ALC-LFS-ContentRepository. If you are using the cli_propertyFile_template.txt as template then either remove the ALC-LFS-ContentRepository from excludedSolutionComponents list or add the following LFS in excludedSolutionComponents list:

- ALC-LFS-ProcessManagement
- ALC-LFS-CorrespondenceManagement
- ALC-LFS-ContentRepository
- ALC-LFS-MobileForms
- ALC-LFS_FormsManager

10.3 General configuration properties

10.3.1 Common properties

Common properties are:

LiveCycle Server specific properties: Required for the Initialize LiveCycle and Deploy LiveCycle Components operations.

These properties are required for the following operations:

- Initialize LiveCycle
- Deploy LiveCycle components.

Property	Values	Description
<i>LiveCycle Server specific properties</i>		
LCHost	String	The hostname of the server where LiveCycle will be deployed. For cluster deployments, hostname of any one of the cluster nodes where the application server is running.
LCPort	Integer	The web port number where LiveCycle will be deployed.

Property	Values	Description
excludedSolutionComponents	String. Values include: ALC-LFS-Forms, ALC-LFS-ConnectorEMCDocumentum, ALC-LFS-ConnectorIBMFileNet, ALC-LFS-ConnectorIBMContentManager, ALC-LFS-DigitalSignatures, ALC-LFS-DataCapture, ALC-LFS-Output, ALC-LFS-PDFGenerator, ALC-LFS-ProcessManagement, ALC-LFS-ReaderExtensions, ALC-LFS-RightsManagement ALC-LFS-CorrespondenceManagement, ALC-LFS-ContentRepository, ALC-LFS-MobileForms, ALC-LFS_FormsManager	(Optional) List the LiveCycle modules you do not want to configure. Specify the excluded modules in a comma separated list.
includeCentralMigrationService	true: to include service false: to exclude service	The property to include or exclude Central Migration Bridge Service.
CRX Content repository The following properties are specified in the cli_propertyFile_crx_template.txt file.		
contentRepository.rootDir		Path of the CRX repository.

10.3.2 Configure LiveCycle properties

These properties only apply to the configure LiveCycle operation.

Property	Values	Description
AdobeFontsDir	String	Location of the Adobe server fonts directory. This path must be accessible from all cluster nodes being deployed to.
customerFontsDir	String	Location of the customer fonts directory. This path must be accessible from all cluster nodes being deployed to.
systemFontsDir	String	Location of the system fonts directory. This path must be accessible from all cluster nodes being deployed to.

Property	Values	Description
LCTempDir	String	Location of the temporary directory. This path must be accessible from all cluster nodes being deployed to.
LCGlobalDocStorageDir	String	The global document storage root directory. Specify a path to an NFS shared directory used to store long-lived documents and to share them among all cluster nodes. This path must be accessible from all cluster nodes being deployed to.
EnableDocumentDBStorage	true or false Default: false	Enables or disables document storage in database for persistent documents. Even if you enable document storage in database, you will need the file system directory for GDS.

10.3.3 Configure or validate application server properties

10.3.3.1 Configure JBoss properties

If you are installing LiveCycle with a JBoss application server, you must manually configure JBoss. See Manually configuring JBoss section in, [Preparing to Install LiveCycle \(Single Server\)](#) Guide. Use the Adobe preconfigured JBoss provided on the LiveCycle DVD, download from the internet or use the JBoss turnkey option.

10.3.4 Initialize LiveCycle properties

These initialize LiveCycle properties only apply to the initialize LiveCycle operation.

Property	Values	Description
For more information, see "10.3.1 Common properties" on page 80		

10.3.5 Deploy LiveCycle Components properties

These properties apply to the following operations:

- Deploy LiveCycle Components
- Validate LiveCycle Component Deployment
- Validate LiveCycle Server.

Property	Values	Description
You must configure the LiveCycle Server Information section. For more information, see Common properties		
LCAdminUserID	String	The user ID to assign to the LiveCycle Administrator user. This User ID is used to login to the Administrator Console.
LCAdminPassword	String	The password to assign to the LiveCycle Administrator user. This password is used to login to the Administrator Console.

10.3.6 Add administrator user for PDF Generator

These properties apply only to the adding administrator user for PDF Generator operation. These properties are present in cli_propertyFile_pdfg_template.txt

Property	Values	Description
LCHost	String	Hostname where LiveCycle Server is installed.
LCPort	Integer	Port number where LiveCycle application server is configured
LCAdminUserID	String	The user ID to assign to the LiveCycle Administrator user. This User ID is used to login to the Administrator Console.
LCAdminPassword	String	The password to assign to the LiveCycle Administrator user. This password is used to login to the Administrator Console.
LCServerMachineAdminUser	String	The user ID of the Administrator user of the Operation System hosting LiveCycle
LCServerMachineAdminUserPasswd	String	The password of the Administrator user of the Operation System hosting LiveCycle

10.3.7 Configure Connector for IBM Content Manager

Property	Values	Description
LCHost	String	Hostname where LiveCycle Server is installed.
LCPort	Integer	Port number where LiveCycle application server is configured
LCAdminUserID	String	The user ID to assign to the LiveCycle Administrator user. This User ID is used to login to the Administrator Console.
LCAdminPassword	String	The password to assign to the LiveCycle Administrator user. This password is used to login to the Administrator Console.
jndiPortNumber	String	JNDI port corresponding to LiveCycle application server.
jboss.clientjar.location	String	The location of the jbossall-client.jar file (JBoss only)
CDVTopology.appserverrootdir	String	The root directory of the application server instance that you are configuring on a remote server (on which you plan to deploy LiveCycle)
ConfigureIBMCM	true or false	Specify true to configure Connector for IBM Content Manager
IBMCMClientPathDirectory	String	Location of IBM Content Manager client installation directory.
DataStoreName	String	Name of the DataStore of IBM Content Manager Server that you want to connect to

Property	Values	Description
IBMCUsername	String	The user name assign to the IBM Content Manager Administrator user. This User ID is used to login to the IBM Content Manager.
IBCMPassword	String	The password to assign to the IBM Content Manager Administrator user. This password is used to login to the IBM Content Manager.
ConnectionString	String	Additional arguments used in the connection string to connect to IBM Content Manager(Optional).

10.3.8 Configure Connector for IBM FileNet

Property	Values	Description
LCHost	String	Hostname where LiveCycle Server is installed.
LCPort	Integer	Port number where LiveCycle application server is configured
LCAdminUserID	String	The user ID to assign to the LiveCycle Administrator user. This User ID is used to login to the Administrator Console.
LCAdminPassword	String	The password to assign to the LiveCycle Administrator user. This password is used to login to the Administrator Console.
jndiPortNumber	String	JNDI port corresponding to LiveCycle application server.
jboss.clientjar.location	String	The location of the jbossall-client.jar file (JBoss only)
CDVTopology.appserverrootdir	String	The root directory of the application server instance that you are configuring on a remote server (on which you plan to deploy LiveCycle)
ConfigureFileNetCE	true or false	Specify true to configure Connector for IBM FileNet
FileNetConfigureCEVersion	String	The FileNet client version to configure. Specify FileNetClientVersion4.0 or FileNetClientVersion5.0
FileNetCEClientPathDirectory	String	Location of IBM FileNet Content Manager client installation directory.
ContentEngineName	String	Hostname or IP address of the machine where IBM FileNet Content Engine is installed
ContentEnginePort	String	The port number used by IBM FileNet Content Engine
CredentialProtectionSchema	CLEAR or SYMMETRIC	Specify the level of protection.
EncryptionFileLocation	String	Location of the encryption file. This is required only when you select SYMMETRIC option for CredentialProtectionSchema attribute. Use a forward slash (/) or double backward slashes (\\) as a path separator.

Property	Values	Description
DefaultObjectStore	String	Name of the ObjectStore for the Connector for IBM FileNet Content Server.
FileNetContentEngineUsername	String	The user ID to connect to the IBM FileNet Content server. The user ID with read-access privileges would be allowed to connect to the Default object Store.
FileNetContentEnginePassword	String	The password to assigned to the IBM FileNet user. This password is used to connect to Default object Store.
ConfigureFileNetPE	true or false	Specify true to configure Connector for IBM FileNet
FileNetPEClientPathDirectory	String	Location of IBM FileNet client installation directory
FileNetProcessEngineHostname	String	Hostname or IP address of the process router.
FileNetProcessEnginePortNumber	Integer	Port number for IBM FileNet Content Server
FileNetPERouterURLConnectionPoint	String	Name of the process router.
FileNetProcessEngineUsername	String	The user ID to connect to the IBM FileNet Content Server
FileNetProcessEnginePassword	String	The password to connect to the IBM FileNet Content Server

10.3.9 Configure Connector for EMC Documentum

Property	Values	Description
LCHost	String	Hostname where LiveCycle Server is installed.
LCPort	Integer	Port number where LiveCycle application server is configured
LCAdminUserID	String	The user ID to assign to the LiveCycle Administrator user. This User ID is used to login to the Administrator Console.
LCAdminPassword	String	The password to assign to the LiveCycle Administrator user. This password is used to login to the Administrator Console.
jndiPortNumber	String	JNDI port corresponding to LiveCycle application server.
jboss.clientjar.location	String	The location of the jbossall-client.jar file (JBoss only)
CDVTopology.appserverrootdir	String	The root directory of the application server instance that you are configuring on a remote server (on which you plan to deploy LiveCycle)
ConfigureDocumentum	true or false	Specify true to configure Connector for EMC Documentum

Property	Values	Description
DocumentumClientVersion	String	The EMC Documentum client version to configure. Specify DocumentumClientVersion6.0 or DocumentumClientVersion6.5
DocumentumClientPathDirectory	String	Location of EMC Documentum client installation directory
ConnectionBrokerHostName	String	Hostname or IP address of the EMC Documentum Content Server.
ConnectionBrokerPortNumber	String	Port number for EMC Documentum Content Server
DocumentumUsername	String	The user ID to connect to the EMC Documentum Content Server.
DocumentumPassword	String	The password ID to connect to the EMC Documentum Content Server.
DocumentumDefaultRepositoryName	String	Name of the default repository of MC Documentum Content Server

10.3.10 Configure Connector for Microsoft SharePoint

Property	Values	Description
LCHost	String	Hostname where LiveCycle Server is installed.
LCPort	Integer	Port number where LiveCycle application server is configured
LCAdminUserID	String	The user ID to assign to the LiveCycle Administrator user. This User ID is used to login to the Administrator Console.
LCAdminPassword	String	The password to assign to the LiveCycle Administrator user. This password is used to login to the Administrator Console.
jndiPortNumber	String	JNDI port corresponding to LiveCycle application server.
jboss.clientjar.location	String	The location of the jbossall-client.jar file (JBoss only)
CDVTopology.appserverrootdir	String	The root directory of the application server instance that you are configuring on a remote server (on which you plan to deploy LiveCycle)
ConfigureSharePoint	true or false	Specify true to configure Connector for Microsoft SharePoint
SharePointServerAddress	String	Hostname or IP address of the Sharepoint Server
SharePointUsername	String	The user ID to connect to the Sharepoint Server

Property	Values	Description
SharePointPassword	String	The password to connect to the Sharepoint Server
SharePointDomain	String	The Domain Name of the Sharepoint Server
ConnectionString	String	Additional arguments used in the connection string to connect to the Sharepoint Server(optional)

10.3.11 Command Line Interface Usage

Once you have configured your property file, you must navigate to the *[LiveCycle root]/configurationManager/bin* folder.

To view a complete description of the Configuration Manager CLI commands, type: `ConfigurationManagerCLI help <command name>`.

10.3.11.1 Configure CRX CLI Usage

The Configure CRX Repository requires the following syntax:

```
configureCRXRepository -f <propertyFile>
```

Where:

- `-f <propertyFile>`: A property file containing the required arguments. For more information on creating a property file, see Command Line Interface property file.

10.3.11.2 Manually deploy configured EAR files

For detailed instructions to manually deploy configured ear files, see Deploying to JBoss Application Server

10.3.11.3 Initialize LiveCycle CLI Usage

The initialize LiveCycle operation requires the following syntax:

```
initializeLiveCycle -f <propertyFile>
```

Where:

- `-f <propertyFile>`: A property file containing the required arguments. For more information on creating a property file, see Command Line Interface property file.

10.3.11.4 Validate LiveCycle Server CLI Usage

The Validate LiveCycle Server operation is optional and requires the following syntax:

```
validateLiveCycleServer -f <propertyFile> -LCAdminPassword <password>
```

Where:

- `-f <propertyFile>`: A property file containing the required arguments. For more information on creating a property file, see Command Line Interface property file.
- `-LCAdminPassword <password>`: Allows you to set the Admin password on the command line. If this argument is present, it will override the `targetServer.adminPassword` property in the property file.

10.3.11.5 Deploy LiveCycle Components CLI Usage

The Deploy LiveCycle Components operation requires the following syntax:

```
deployLiveCycleComponents -f <propertyFile> -LCAdminPassword <password>
```

Where:

- -f <propertyFile>: A property file containing the required arguments. For more information on creating a property file, see Command Line Interface property file.
- -LCAdminPassword <password>: Allows you to set the Admin password on the command line. If this argument is present, it will override the targetServer.adminPassword property in the property file.

10.3.11.6 Validate LiveCycle Component Deployment CLI Usage

The Validate LiveCycle Component Deployment operation is optional and requires the following syntax:

```
validateLiveCycleComponentDeployment -f <propertyFile> -LCAdminPassword <password>
```

Where:

- -f <propertyFile>: A property file containing the required arguments. For more information on creating a property file, see Command Line Interface property file.
- -LCAdminPassword <password>: Allows you to set the Admin password on the command line. If this argument is present, it will override the targetServer.adminPassword property in the property file.

10.3.11.7 Check system readiness for PDF Generator

The Checking system readiness for PDF Generator operation requires the following syntax:

```
pdfg-checkSystemReadiness
```

10.3.11.8 Adding administrator user for PDF Generator

The adding administrator user for PDF Generator operation requires the following syntax:

```
pdfg-addAdminUser -f <propertyFile>
```

Where:

- -f <propertyFile>: A property file containing the required arguments. For more information on creating a property file, see Command Line Interface property file.

10.3.11.9 Configure Connector for IBM Content Manager

The Configure Connector for IBM Content Manager operation is optional and requires the following syntax:

```
IBMCM-configurationCLI -f <propertyFile>
```

Where:

- -f <propertyFile>: A property file containing the required arguments. For more information on creating a property file, see Command Line Interface property file.

Important: Modify the <propertyFile> called *cli_propertyFile_ecm_ibmcm_template.txt* located in the [LiveCycle root]\configurationManager\bin\ directory.

- 1 Copy the *adobe-component-ext.properties* file from [LiveCycle root]/configurationManager/configure-ecm/jboss to the following [appserver root]/bin directory.
- 2 Restart the Application Server.

3 Start the following services from LiveCycle Administration Console

- IBMCMAuthProviderService
- IBMCMConnectorService

10.3.11.10 Configure Connector for IBM FileNet

The Configure Connector for IBM FileNet operation is optional and requires the following syntax:

filenet-configurationCLI -f <propertyFile>

Where:

- -f <propertyFile>: A property file containing the required arguments. For more information on creating a property file, see Command Line Interface property file.

Important: Modify the <propertyFile> called `cli_propertyFile_ecm_filenet_template.txt` located in the `[LiveCycle root]\configurationManager\bin\` directory.

Perform the following steps manually to complete the configuration for Connector for IBM Content Manager.

- 1 Copy the `adobe-component-ext.properties` file from `[LiveCycle root]/configurationManager/configure-ecm/jboss` to the following `[appserver root]/bin` directory.

- 2 Locate the `login-config.xml` file in the `[appserver root]/server/[profile]/conf` folder and add to it contents of `login-config.xml` file available in `[LiveCycle root]/configurationManager/configure-ecm/jboss` directory.

Default jboss setup comes up with a `[profile]` value as "all". However, for Adobe Configured Jboss use `[lc_DatabaseName]` (e.g. `lc_mysql`, `lc_oracle`).

- 3 Copy the `logkit.jar` file from `[appserver root]/client` to the following `[appserver root]/server/[profile]/lib` directory. Default jboss setup comes up with a `[profile]` value as "all". However, for Adobe Configured Jboss use `[lc_DatabaseName]` (e.g. `lc_mysql`, `lc_oracle`).

- 4 **(Only for FileNet 4.x)** Add the Java option `-Dwasp.location=[FileNetClient root]/wsi` to the Application Server startup options.

- 5 Restart the Application Server.

- 6 Start the following services from LiveCycle Administration Console

- IBMFileNetAuthProviderService
- IBMFileNetContentRepositoryConnector
- IBMFileNetRepositoryProvider
- IBMFileNetProcessEngineConnector(If configured)

10.3.11.11 Configure Connector for EMC Documentum

The Configure Connector for EMC Documentum operation is optional and requires the following syntax:

documentum-configurationCLI -f <propertyFile>

Where:

- -f <propertyFile>: A property file containing the required arguments. For more information on creating a property file, see Command Line Interface property file.

Important: Modify the <propertyFile> called `cli_propertyFile_ecm_documentum_template.txt` located in the `[LiveCycle root]\configurationManager\bin\` directory.

Perform the following steps manually to complete the configuration for Connector for EMC Documentum.

- 1 Copy the *adobe-component-ext.properties* file from *[LiveCycle root]/configurationManager/configure-ecm/jboss* to the following *[appserver root]/bin* directory.
- 2 Restart the Application Server.
- 3 Start the following services from LiveCycle Administration Console
 - EMCDocumentumAuthProviderService
 - EMCDocumentumRepositoryProvider
 - EMCDocumentumContentRepositoryConnector

10.3.11.12 Configure Connector for Microsoft SharePoint

The Configure Connector for Microsoft SharePoint operation is optional and requires the following syntax:

`sharepoint-configurationCLI -f <propertyFile>`

Where:

- `-f <propertyFile>`: A property file containing the required arguments. For more information on creating a property file, see Command Line Interface property file.

Important: Modify the `<propertyFile>` called *cli_propertyFile_ecm_sharepoint_template.txt* located in the *[LiveCycle root]/configurationManager/bin* directory.

10.4 Examples Usage

From the *C:\Adobe\Adobe LiveCycle ES4\configurationManager\bin*, type:

```
ConfigurationManagerCLI configureLiveCycle -f cli_propertyFile.txt
```

Where *cli_propertyFile.txt* is the name of the property file you created.

10.5 Configuration Manager CLI Logs

If an error occurs, you can review the CLI logs located here in the *[LiveCycle root]/configurationManager/log* folder. The log file generated will have a naming convention such as *lcmCLI.0.log* where the number in the filename (0) will increment when the log files are rolled over.

10.6 Next steps

If you used Configuration Manager CLI to configure and deploy LiveCycle, you can now do the following tasks:

- Verify the deployment. (See [“6.1.5 Verify the deployment”](#) on page 33.)
- Access Administration Console. (See [“6.1.5.1 Accessing LiveCycle Administration Console”](#) on page 34.)
- Configure LiveCycle modules to access LDAP. (See [“6.8 Configuring LDAP access”](#) on page 52.)

Chapter 11: Appendix - Configuring JBoss as a Windows Service

This appendix describes how you can configure the JBoss application server to run as a Windows service using the JBoss Web Native Connectors. Use this procedure on Windows Server 2008, both 32-bit and 64-bit versions.

11.1 Download the Web Native Connector

- 1 Download the JBoss Web Native Connector for Windows from the *JBoss Web Native Connectors - Current packages* download page. Depending upon your Windows version, download either of the following files:
 - (64-bit): <http://download.jboss.org/jbossweb/2.0.8.GA/jboss-native-2.0.8-windows-x64-ssl.zip>
 - (32-bit): <http://download.jboss.org/jbossweb/2.0.8.GA/jboss-native-2.0.8-windows-x86-ssl.zip>
- 2 Extract the ZIP file and copy all contents of the \bin folder (except the \native folder) to the \bin folder of your JBoss installation folder.
- 3 Open the `service.bat` file in a text editor and update the variables.

You should update the variables for Service Name (SVCNAME), Service Display (SVCDISP) and Service Description (SVCDESC) with values that reflect your JBoss environment. For example, if your JBoss version is 5.1, enter the following:

```
set SVCNAME=JBAS51SVC

set SVCDISP=JBossAS 5.1 for Adobe LiveCycle ES4

set SVCDESC=JBoss Application Server 5.1 GA/ Platform: Windows x64
```

- 4 In the `:cmdStart` section, locate and edit the `call run.bat` line to add the configuration name and bind IP address (0.0.0.0 for binding to all IP addresses of the server) such as follows:

```
call run.bat -c <profilename> -b 0.0.0.0 < .r.lock >> run.log 2>&1
```

- 5 Repeat the edits in step 4 for the `:cmdRestart` section:

```
call run.bat -c <profilename> -b 0.0.0.0 < .r.lock >> run.log 2>&1
```

- 6 Save and close the file.

Note: Specify JBoss cluster arguments in Step 4 and 5 to include the JBoss instance in the cluster. For detailed information about JBoss cluster arguments, see “3.3 Running JBoss in a cluster” on page 7

11.2 Install the Windows service

- 1 From the \bin folder of JBoss, create the Windows service using the following command:

```
service.bat install
```

If the command is successful, you will get a response such as:

```
Service JBossAS 5.1 for Adobe LiveCycle ES4 installed
```

- 2 Check the Services applet in Windows Control Panel for a new service listed as *JBossAS 5.1 for Adobe LiveCycle ES4* which is the value of the `SVCDISP` variable in the `service.bat` file.
- 3 Using the Services applet in Windows Control Panel, set the *Startup type* to `Automatic`.
- 4 (Optional) In the *Recovery* tab, set the *First failure* and *Second failure* recovery options such as *Restart the Service* and *Restart the Computer* respectively.

Note: If necessary, you can change the Logon as value from the default Local System account to another user or service account.

11.3 Start and stop JBoss Application Server as a Windows service

Start JBoss as a Windows service

- ❖ On the Windows server, select **Start > Control Panel > Administrative Tools > Services**, then select the Windows service for JBoss Application Server and click **Start**.

Note: When starting JBoss Application Server as a Windows service, the console output is redirected to the file `run.log`. You can inspect the file to discover any errors that occur during service startup.

Stop JBoss as a Windows service

- ❖ On the Windows server, select **Start > Control Panel > Administrative Tools > Services**, then select the Windows service for JBoss Application Server and click **Stop**.

Note: When stopping JBoss Application Server as a Windows service, the console output is redirected to the file `run.log`. You can inspect the file to discover any errors that occur during service shutdown.

11.4 Verify the installation

- 1 Start the service from the Services applet in Windows Control Panel.
- 2 Watch (tail) the `[appserver root]\<profile_name>\logs\server.log` file to make sure that the service starts successfully.
- 3 Shutdown the service from the Services applet in Windows Control Panel and verify that it is shut down successfully.
- 4 Make sure that you are able to restart the service from the Services applet in Windows Control Panel.

11.5 Additional configuration

In addition to these steps, you can also perform additional configuration steps using either the Services applet in Windows Control Panel or by using the built-in Windows Service Configuration utility (`sc`).

For example, if you have a Microsoft SQL Server as the database, and the database service runs on the same machine instance, you can create a dependency on that service with the following command:

```
sc config JBAS51SVC depend= MSSQL$MYSERVER
```


Update the `MSSQL$MYSERVER` variable with service name of the Microsoft SQL Server 2005 service running on the same server instance.

Note: *Ensure that there is no space before the = sign but after the = sign.*

If the command is successful, you will get a response such as follows:

```
[SC] ChangeServiceConfig SUCCESS
```

Chapter 12: Appendix - Manually Configuring JBoss

This appendix describes the configuration that is required for JBoss EAP that you can download from Red Hat. This option should be considered for advanced installations only. Advanced knowledge of JBoss is typically required.

LiveCycle ES3 runs on JBoss Windows Server 2008 R1/R2 (Enterprise or Standard Edition), Red Hat Linux ES/AS 5.5 or 6, SUSE Linux ES 11 platforms, and Solaris 10.

12.1 Installing the JDK for JBoss

You must download and install Oracle JDK 6.0 update 31 or later updates to 6.0 versions from

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Create or set the `JAVA_HOME` environment variable to point to the location where the JDK is installed.

12.1.1 Set the `JAVA_HOME` environment variable (Windows)

- 1 Select **Start > Control Panel > System**.
- 2 Click the **Advanced** tab.
- 3 Click **Environment Variables** and, under System Variables, click **New**.
- 4 In the **New System Variable** box, type `JAVA_HOME` as the variable name and enter the directory where you installed the JDK. This directory is the directory that contains the `/bin` subdirectory. For example, type the following path:

```
C:\Program Files\Java\jdk1.6.0_31
```

12.1.2 Set the `PATH` environment variable (Windows)

- 1 Select **Start > Control Panel > System**.
- 2 Click the **Advanced** tab and click **Environment Variables**.
- 3 In the System Variables area, select the `PATH` variable and then click **Edit**.
- 4 Append the following text to the beginning of the variable value:

```
%JAVA_HOME%\bin;
```

12.1.3 Set the `JAVA_HOME` environment (Linux and Solaris)

- It is recommended that you set the `JAVA_HOME` variable for Bourne and Bash as shown in the following example:

```
JAVA_HOME=/usr/java
export JAVA_HOME
```

12.1.4 Set the `PATH` environment variable (Linux and Solaris)

- Set the `PATH` variable for Bourne and Bash as shown in the following example:

```
PATH=$JAVA_HOME/bin:$PATH
export PATH
```

12.1.5 Verify JAVA_HOME environment variable setting (Windows, Linux, or Solaris)

(Optional) Open a command prompt and run the following command:

```
java -version
```

The command returns the Java version installed on your system.

12.2 Manually installing JBoss

You can download and install JBoss EAP from <http://www.jboss.org/jbossas/downloads/>.

12.3 Starting and stopping JBoss

Several procedures in this appendix require you to stop and start the instance of JBoss where you want to deploy the product.

All JBoss start configurations are located in the *[appserver root]/server* directory. For JBoss obtained directly from Red Hat, either use one of the JBoss provided start configurations (all, default or minimal) or a custom configuration.

12.3.1 Start JBoss

1 From a command prompt, navigate to *[appserver root]/bin*.

2 Start the application server by typing the following command:

- (Windows) `run.bat -c [profile_name] -b [server_IP_Address]`
- (Linux and Solaris) `./run.sh -c [profile_name] -b [server_IP_Address]`

where *[profile_name]* is the configuration required for your database and *[server_IP_Address]* is the IP address of the server.

12.3.2 Stop JBoss

1 From a command prompt, navigate to *[appserver root]/bin*.

2 Stop the application server by typing the following command:

- (Windows) `shutdown.bat -s [server_IP_Address]:<jndi -port>`
- (Linux and Solaris) `./shutdown.sh -s [server_IP_Address]:<jndi -port>`

12.4 Modifying the JBoss configuration

The JBoss Application Server is configured using various XML configuration files. JBoss must be shut down before editing any of these configuration files. If JBoss is running and these files are changed, JBoss will probably crash. JBoss also has a few configuration files that are formatted as .property files. You must ensure that the .property files are saved as UNIX text files on Linux or Solaris if you edit these files on Windows environments at any time.

For single-server installations, you may use jboss profile located at `[appserver root]\server\standard\` as a template. For cluster installations, use jboss profile located at `[appserver root]\server\all\` as a template.

It is recommended that you make a copy of the profile (all or standard) and make changes to the copied profile.

12.4.1 Remove JMS and configuration files (optional)

Because LiveCycle does not use JMS configuration, you can delete the following files and directories that are part of JBoss. Ensure that you delete the entire directory and its contents listed below.

Note: Skip this step if you need to enable LiveCycle Foundation JMS services or deploy other services or applications that depend on the JMS service.

- `[appserver root]\server\<profile_name>\deploy\messaging`
- `[appserver root]\server\<profile_name>\deploy\jms-ra.rar`

12.4.2 Modify the JBoss configuration

Perform the following steps to modify the JBoss configuration to customize JBoss for LiveCycle.

- Update the `jacorb.properties` file
- Update the URI Encoding in the JBoss `server.xml` file
- Modify EAR file class-loading isolation
- Modify the `run.conf.bat` file (Windows)
- Modify `run.conf` (Linux and Solaris)
- Modify `log4j.xml`
- Modify the `jbosssts-properties.xml` file
- Change path of the session cookie

12.4.2.1 Update the `jacorb.properties` file

- 1 Open the `[appserver root]/server/<profile_name>/conf/jacorb.properties` file in a text editor.
- 2 Open the `[appserver root]/server/<profile_name>/conf/jacorb.properties` file in a text editor.
- 3 Locate the `jacorb.poa.thread_pool_max` setting and change the value to 16.
- 4 Save and close the file.

12.4.2.2 Update the URI Encoding in the JBoss `server.xml` file

- 1 Open the `[appserver root]/server/<profile_name>/deploy/jbossweb.sar/server.xml` file in a text editor.
- 2 Locate the following lines:

```
<Connector protocol="HTTP/1.1" port="8080" address="${jboss.bind.address}"
connectionTimeout="20000" redirectPort="8443" />
```

3 Append URIEncoding="UTF-8":

```
<Connector protocol="HTTP/1.1" port="8080" address="${jboss.bind.address}"  
connectionTimeout="20000" redirectPort="8443" URIEncoding="UTF-8" />
```

4 Save and close the file.

12.4.2.3 Modify EAR file class-loading isolation

1 Open the [appserver root]/server/<profile_name>/deployers/ejb-deployer-jboss-beans.xml file in an editor.

2 Locate <property name="CallByValue"> and change the value to true.

3 Save and close the file.

4 Open the [appserver root]/server/<profile_name>/deployers/ear-deployer-jboss-beans.xml file in an editor.

5 Locate <property name="isolated"> and change the value to true.

6 Locate <property name="CallByValue">false</property> and change the value to true.

7 Save and close the file.

8 Open the [appserver root]/server/<profile_name>/conf/jboss-service.xml file in an editor.

9 Locate <attribute name="CallByValue">false</attribute> and change the value to true.

10 Save and close the file.

12.4.2.4 Modify the run.conf.bat file (Windows only)

1 Open the [appserver root]/bin/run.conf.bat file in an editor.

2 Delete the text in bold in the following line and add the memory arguments for 32-bit and 64-bit JVM, as applicable.

```
set "JAVA_OPTS=-Xms1303m -Xmx1303m -XX:MaxPermSize=256m -Dorg.jboss.resolver.warning=true -  
Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000 -  
Dsun.lang.ClassLoader.allowArraySyntax=true"
```

- Memory argument for 32-bit JVM:

```
set "JAVA_HEAP_ARGS=-Xms1024m -Xmx1024m -XX:PermSize=128m -XX:MaxPermSize=192m"
```

- Memory argument for 64-bit JVM:

```
set "JAVA_HEAP_ARGS=-Xms1024m -Xmx2048m -XX:PermSize=256m -XX:MaxPermSize=512m -  
XX:+UseCompressedOops"
```

3 Add the following line.

```
set "JAVA_OPTS=%JAVA_OPTS% %JAVA_HEAP_ARGS%"
```

4 Add the following argument for 32-bit and 64-bit JVM:

```
set "JAVA_OPTS=%JAVA_OPTS% -Dadobeidp.serverName=server1 -Dfile.encoding=utf8 -  
Djava.net.preferIPv4Stack=true"  
set "JAVA_OPTS=%JAVA_OPTS% -DentityExpansionLimit=10000"  
set "JAVA_OPTS=%JAVA_OPTS% -XX:+HeapDumpOnOutOfMemoryError"
```

5 (Optional) Modify JBoss Application Server to run in IPv6 mode as follows:

- Locate and modify -Djava.net.preferIPv4Stack=false
- Insert the string -Djava.net.preferIPv6Stack=true

Note: If the application server log contains the following error on startup, remove the value for the IPv6 stack and set the IPV4 value back to true:

```
"13:37:44,488 WARN [HANamingService] Failed to start AutomaticDiscovery java.net.SocketException: bad
argument for IP_MULTICAST_IF: address not bound to any interface at
java.net.PlainDatagramSocketImpl.socketSetOption(Native Method)at
java.net.PlainDatagramSocketImpl.setOption(PlainDatagramSocketImpl.java:260)"
```

6 Save and close the file.

12.4.2.5 Modify the run.conf file (JBoss with Solaris 10, Red Hat 5.5, 64-bit only)

Solaris JDKs from Sun require an additional argument to use 64-bit features. Without this configuration change, the Sun JDK defaults to 32-bit support only.

Note: If you're running JBoss as a non-root user, use `-Djava.io.tmpdir="location"` to set the location of the temporary directory to a directory to which you have access.

1 Open the `[appserver root]/bin/run.conf` file in an editor.

2 Locate the section starting with:

```
if [ "x$JAVA_OPTS" = "x" ]; then
```

Modify the section to look like:

```
if [ "x$JAVA_OPTS" = "x" ]; then
    #JAVA_OPTS="-Xms1303m -Xmx1303m -XX:MaxPermSize=256m -Dorg.jboss.resolver.warning=true -
Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000 -
Dsun.lang.ClassLoader.allowArraySyntax=true"
    JAVA_OPTS="$JAVA_OPTS -Xms1024m -Xmx2048m -XX:MaxPermSize=512m -
Dorg.jboss.resolver.warning=true -Dsun.rmi.dgc.client.gcInterval=3600000 -
Dsun.rmi.dgc.server.gcInterval=3600000 -Dsun.lang.ClassLoader.allowArraySyntax=true"
    JAVA_OPTS="$JAVA_OPTS -Dadobeidp.serverName=server1 -Dfile.encoding=utf8 -
Djava.net.preferIPv4Stack=true"
    JAVA_OPTS="$JAVA_OPTS -DentityExpansionLimit=10000"
    JAVA_OPTS="$JAVA_OPTS -XX:+UseCompressedOops -XX:+HeapDumpOnOutOfMemoryError"
```

Note: Ensure that this entry appears as a single line in the `run.conf` file.

3 (optional) Modify JBoss Application Server to run in IPv6 mode as follows:

- Locate and modify `-Djava.net.preferIPv4Stack=false`
- Add `-Djava.net.preferIPv6Stack=true`

4 Save and close the file.

12.4.2.6 Modify the log4j.xml file

1 Open the `[appserver root]/server/<profile_name>/conf/jboss-log4j.xml` file in an editor.

2 Locate the following text in the `FILE` appender section and add the line/change the value that appears in bold:

```
<appender name="FILE" class="org.jboss.logging.appender.DailyRollingFileAppender">
  <errorHandler class="org.jboss.logging.util.OnlyOnceErrorHandler"/>
  <param name="File" value="{jboss.server.home.dir}/log/server.log"/>
  <param name="Append" value="false"/>
```

3 Locate the Limit categories section and add the following:

```
<category name="jacorb.config">
<priority value="ERROR"/>
</category>
<category name="com.adobe">
<priority value="INFO"/>
</category>
<category name="org.springframework">
<priority value="WARN"/>
</category>
<!-- Limit the verbose MC4J EMS (lib used by admin-console) categories -->
<category name="org.mc4j.ems">
<priority value="WARN"/>
</category>
```

- 4 Save and close the file.

12.4.2.7 Modify the jbossts-properties.xml file:

- 1 Ensure that transaction management works as expected by locating the jbossts-properties.xml file in the [appserver root]/server/<profile_name>/conf directory and opening the file in an editor.
- 2 Locate and modify the properties element as follows (modification in bold). Add the property in bold if it doesn't already exist in the jbossts-properties.xml file.

```
<properties depends="arjuna" name="jta">
<property name="com.arjuna.ats.jta.allowMultipleLastResources" value="true"/>
<!-- ... other properties ... -->
</properties>
```

- 3 If you are not using messaging, locate and comment out following lines

```
<property
name="com.arjuna.ats.jta.recovery.XAResourceRecovery.JBMESSAGING1" value="org.jboss.jms.ser
ver.recovery.MessagingXAResourceRecovery;java:/DefaultJMSProvider"/>
```

- 4 Save and close the file.

Note: For more information, see the JBoss article 11443 at <http://www.jboss.org/community/docs/DOC-11443>.

12.4.2.8 Modify jmx-invoker-service.xml file

- 1 Open the [appserver root]/server/<profile_name>/deploy/jmx-invoker-service.xml file for editing and comment out the following line:

```
<interceptor code="org.jboss.jmx.connector.invoker.AuthenticationInterceptor"
securityDomain="java:/jaas/jmx-console"/>
```

- 2 Save and close the file.

12.4.2.9 Change path of the session cookie

- 1 Locate the context.xml file in the following location and open it in an editor.

[appserver root]/server/all/deploy/jbossweb.sar

- 2 Locate following line:

```
<InstanceListener>org.jboss.web.tomcat.security.RunAsListener</InstanceListener>
```

- 3 Add following text after above line:

```
<SessionCookie path="/" />
```

- 4 Save and close the file.

12.5 Copying jar files

Copy the following JAR files from `[DVD root]/third_party/jboss.zip/server/lc_<db-name>_cl/lib` to the `[appserver root]/server/<profile_name>/lib` directory of your downloaded JBoss.

- certjFIPS.jar
- cglib.jar
- hibernate-annotations.jar
- hibernate-entitymanager.jar
- hibernate3.jar
- jsafeFIPS.jar
- jsafeJCEFIPS.jar

12.6 LiveCycle database connectivity for manually installed JBoss

To configure the LiveCycle database connectivity, you must complete the following tasks:

- Configure the LiveCycle data source.
- Configure JBoss to use your database as the default data source.

You must install database drivers to the installation directories of the application server. Drivers are required to enable Configuration Manager and the application server to connect to the LiveCycle database. Install the drivers for the type of database that you use for the database.

You must configure the data source to connect to the database. For JBoss, you can configure an MySQL, Oracle, or SQL Server data source.

Note: Before proceeding with the following tasks, ensure that JBoss is not running.

12.6.1 Configuring MySQL for manually installed JBoss

To enable JBoss to connect to the MySQL database that stores LiveCycle data, you must complete these tasks.

- Obtain and copy the MySQL JDBC driver to the instance of JBoss where you will deploy LiveCycle.
- Create a data source file and deploy it to the instance of JBoss where you will deploy LiveCycle.
- Encrypt the password in the data source files (`adobe-ds.xml` and `mysql-ds.xml`) and the `login-config.xml` file using one of the methods described at <http://community.jboss.org/wiki/EncryptingDataSourcePasswords>. You can also use the instructions available on http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_clearite.html.

12.6.1.1 Configuring the MySQL data source

Before you configure the MySQL data source, you must have the database created on MySQL. (See Creating a MySQL database.)

12.6.1.2 Install the MySQL database driver:

- Copy the mysql-connector-java-5.1.6-bin.jar driver file from the `[DVD_root]/third_party/db/mysql` directory on the installation DVD to the `[appserver root]/server/<profile_name>/lib` directory.
- Copy the mysql-connector-java-5.1.6-bin.jar driver file from the `[DVD_root]/third_party/db/mysql` directory on the installation DVD to the `[appserver root]/server/<profile_name>/lib` directory.

12.6.1.3 Edit adobe-ds.xml file

- 1 Copy the adobe-ds.xml file from the `[DVD_root]/third_party/additional/datasources/lc_mysql/deploy` directory on the installation DVD to the `[appserver root]/server/<profile_name>/deploy` directory.

- 2 Open the adobe-ds.xml file in a text editor and locate this line:

```
<connection-url>jdbc:mysql://localhost:3306/adobe</connection-url>
<driver-class>com.mysql.jdbc.Driver</driver-class>
<user-name>adobe</user-name>
<password>adobe</password>
```

- 3 Replace the following values with values that are specific to your database:

- **localhost:** The name, IP address, or fully-qualified path of the computer that hosts the database. The default is localhost.
- **3306:** The port used to access the database. The default port is 3306.
- **adobe:** The name of the database that stores the LiveCycle data. You will need to update the default value, adobe, with your database name.

- 4 In the lines that follow the `<connection-url>` settings, locate the `user-name` and `password` settings and replace the default values with the user name and password that the application server uses to access your database.

- 5 Modify the minimum and maximum values for the data source connections:

- IDP_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

- EDC_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>20</max-pool-size>
```

Note: If your LiveCycle Server handles heavy load, increase the number of maximum JDBC connections to ensure that all jobs are processed. In such cases, increase `<max-pool-size>` to 50 or more for both IDP_DS and EDC_DS.

- 6 Save and close the file.

12.6.1.4 Set MySQL as the data source

- 1 Navigate to the `[appserver root]/server/<profile_name>/deploy` directory and delete the `hsqldb-ds.xml` file.
- 2 Navigate to the `[appserver root]/server/<profile_name>/deploy` directory and delete the `hsqldb-ds.xml` file.
- 3 Navigate to the `[appserver root]/docs/examples/jca` directory and copy the `mysql-ds.xml` file to the `[appserver root]/server/<profile_name>/deploy` directory.
- 4 Navigate to the `[appserver root]/docs/examples/jca` directory and copy the `mysql-ds.xml` file to the `[appserver root]/server/<profile_name>/deploy` directory.

- 5 Open the `[appserver root]/server/<profile_name>/deploy/mysql-ds.xml` file in a text editor and modify the `<local-tx-datasource>` element with your MySQL connection settings:

```
<jndi-name>MySQLDS</jndi-name>
<connection-url>jdbc:mysql://mysql-hostname:3306/jbossdb</connection-url>
<driver-class>com.mysql.jdbc.Driver</driver-class>
<user-name>x</user-name>
<password>y</password>
```

- 6 Replace the bold values with values that are specific to your database:
 - **MySQLDS**: Change to `DefaultDS`.
 - **mysql-hostname**, **3306**, **jbossdb**, **x**, and **y**: The database values that the application server uses to access the database.
- 7 Add the following line to the `<local-tx-datasource>` section, if it does not already exist.

```
<transaction-isolation>TRANSACTION_READ_COMMITTED</transaction-isolation>
```
- 8 Save and close the file.

12.6.1.5 Edit the login-config.xml file

- 1 Open the `[appserver root]/server/<profile_name>/conf/login-config.xml` file in a text editor and add the following text within the `<policy>` element:

```
<application-policy name="MySqlDbRealm">
  <authentication>
    <login-module
      code="org.jboss.resource.security.ConfiguredIdentityLoginModule" flag
      = "required">
      <module-option name="principal">adobe</module-option>
      <module-option name="userName">adobe</module-option>
      <module-option name="password">adobe</module-option>
      <module-option
        name="managedConnectionFactoryName">jboss.jca:service=LocalTxCM,
        name=DefaultDS </module-option>
      </login-module>
    </authentication>
  </application-policy>
```

- 2 Replace the bold values with values that are specific to your database.
- 3 Save and close the file.
- 4 Start JBoss.

12.6.2 Configuring Oracle for manually installed JBoss

To enable JBoss to connect to the Oracle database that stores LiveCycle data, you must complete the following tasks if you are manually deploying LiveCycle:

- Obtain and copy the Oracle JDBC driver to the instance of JBoss where you will deploy LiveCycle.
- Create a data source file and deploy it to the instance of JBoss where you will deploy LiveCycle.
- Encrypt the password in the data source files (`adobe-ds.xml` and `oracle-ds.xml`) and the `login-config.xml` file using one of the methods described at <http://community.jboss.org/wiki/EncryptingDataSourcePasswords>. You can also use the instructions available on http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_clearite.html.

12.6.2.1 Install the Oracle 11g database driver

Copy the ojdbc6.jar for JDK 1.6 driver file from the *[LiveCycle root]/lib/db/oracle* directory to the *[appserver root]/server/<profile_name>/lib* directory. You can also download the Oracle 11g driver from the JDBC Driver Downloads site, see [Supported Platform Combinations](#) for supported versions Oracle 11g driver.

12.6.2.2 Edit adobe-ds.xml file

- 1 Copy the adobe-ds.xml file from the *[DVD_root]/third_party/additional/datasources/lc_oracle/deploy* directory to the *[appserver root]/server/<profile_name>/deploy* directory.

- 2 Open the adobe-ds.xml file in a text editor and locate this line:

```
<connection-url>jdbc:oracle:thin:@localhost:1521:adobe</connection-url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
<user-name>adobe</user-name>
<password>adobe</password>
```

- 3 Replace the following values with values that are specific to your database:

- **localhost:** The name, IP address, or fully-qualified path of the computer that hosts the database. The default is localhost.
- **1521:** The port used to access the database. The default port is 1521.
- **adobe:** Change the default value, adobe, with your database SID.

- 4 In the lines that follow the <connection-url> settings, locate the user-name and password settings and replace the default values with the user name and password that the application server uses to access your database.

- 5 (Only for Oracle RAC) Replace the connection URL mentioned in step 2 with the following connection URL:

```
jdbc:oracle:thin:@(DESCRIPTION=(ENABLE=broken) (ADDRESS_LIST=(ADDRESS= (PROTOCOL=TCP)
(HOST=yourhost1) (PORT=1521)) (ADDRESS=(PROTOCOL=TCP) (HOST=yourhost2) (PORT=1521))
(LOAD_BALANCE=on) (FAILOVER=on)) (CONNECT_DATA=(SERVER=dedicated)
(SERVICE_NAME=service.yourcompany.com) (FAILOVER_MODE=(TYPE=session) (METHOD=basic)
(RETRIES=10) (DELAY=3))))
```

Note: Ensure that this entry appears as a single line in the adobe-ds.xml file.

- 6 (Only for Oracle RAC) Replace the following text from the connection URL in step 5 with values that are specific to your database:

- **yourhost1:** The name, IP address, or fully-qualified domain name of the first node in the cluster that hosts the database.
- **yourhost2:** The name, IP address, or fully-qualified domain name of the second node in the cluster that hosts the database.

Note: The cluster hosting the database could have *n* nodes. **yourhost1** and **yourhost2** are examples in the case of a two-node cluster.

- **service.yourcompany.com:** The service name for the Oracle RAC database.

- 7 Modify the minimum and maximum values for the data source connections:

- IDP_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```

- EDC_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>20</max-pool-size>
```

***Note:** If your LiveCycle Server handles heavy load, increase the number of maximum JDBC connections to ensure that all jobs are processed. In such cases, increase `<max-pool-size>` to 50 or more for both IDP_DS and EDC_DS.*

- 8 Save and close the file.

12.6.2.3 Set Oracle as the data source

If you are running LiveCycle with a Oracle database, you must set Oracle to be the default data source for JBoss. This procedure assumes that the Oracle JDBC driver is installed in the `[appserver root]/server/<profile_name>/lib` directory.

- 1 Navigate to the `[appserver root]/server/<profile_name>/deploy` directory and delete the `hsqldb-ds.xml` file.
- 2 Copy the `oracle-ds.xml` file from the `[appserver root]/docs/examples/jca` directory to the `[appserver root]/server/<profile_name>/deploy` directory.
- 3 Open the `[appserver root]/server/<profile_name>/deploy/oracle-ds.xml` file in a text editor and modify the `<local-tx-datasource>` element with your Oracle connection settings:

```
<jndi-name>OracleDS</jndi-name>
<connection-url>jdbc:oracle:thin:@youroraclehost:1521:yoursid</connection-url>
<driver-class>oracle.jdbc.driver.OracleDriver</driver-class>
<user-name>x</user-name>
<password>y</password>
```

- 4 Replace the bold values with values that are specific to your database:
 - **OracleDS**: Change this value to `DefaultDS`.
 - **youroraclehost**: Replace this value with the host name of your Oracle server.
 - **1521**: If Oracle is not using the default port, replace this value with the appropriate port number.
 - **yoursid**: Replace this value with your Oracle System Identifier.
- 5 In the lines that follow the `<connection-url>` settings, locate the `user-name` and `password` settings and replace the default values with the user name and password that the application server uses to access your database.
- 6 (Only for Oracle RAC) Replace the connection settings mentioned in step 3 with the following connection URL:

```
jdbc:oracle:thin:@(DESCRIPTION=(ENABLE=broken) (ADDRESS_LIST=(ADDRESS= (PROTOCOL=TCP)
(HOST=yourhost1) (PORT=1521)) (ADDRESS=(PROTOCOL=TCP) (HOST=yourhost2) (PORT=1521))
(LOAD_BALANCE=on) (FAILOVER=on)) (CONNECT_DATA=(SERVER=dedicated)
(SERVICE_NAME=service.yourcompany.com) (FAILOVER_MODE=(TYPE=session) (METHOD=basic)
(RETRIES=10) (DELAY=3))))
```

***Note:** Ensure that this entry appears as a single line in the `oracle-ds.xml` file.*

- 7 (Only for Oracle RAC) Replace the following text from the connection URL in step 6 with values that are specific to your database:
 - **yourhost1**: The name, IP address, or fully-qualified domain name of the first node in the cluster that hosts the database.
 - **yourhost2**: The name, IP address, or fully-qualified domain name of the second node in the cluster that hosts the database.

***Note:** The cluster hosting the database could have *n* nodes. **yourhost1** and **yourhost2** are examples in the case of a two-node cluster.*

 - **service.yourcompany.com**: The service name for the Oracle RAC database.
- 8 Save and close the file.

12.6.2.4 Edit the login-config.xml file

- 1 Open the `[appserver root]/server/<profile_name>/conf/login-config.xml` file in a text editor and add the following text within the `<policy>` element:

```
<application-policy name = "OracleDbRealm">
  <authentication>
    <login-module code =
      "org.jboss.resource.security.ConfiguredIdentityLoginModule" flag =
      "required">
      <module-option name = "principal">adobe</module-option>
      <module-option name = "userName">adobe</module-option>
      <module-option name = "password">adobe</module-option>
      <module-option name = "managedConnectionFactoryName">
        jboss.jca:service=LocalTxCM,name=DefaultDS</module-option>
    </login-module>
  </authentication>
</application-policy>
```

- 2 Replace the bold values with values that are specific to your database.
- 3 Save and close the file.
- 4 Start JBoss.

12.6.3 Configuring SQL Server for manually installed JBoss

To enable JBoss to connect to the SQL Server database that stores LiveCycle data, you must complete the following tasks:

- Obtain and copy the SQL Server JDBC driver files to the instance of JBoss where you will deploy LiveCycle.
- Create a SQL Server data source file and deploy it to the instance of JBoss where you will deploy LiveCycle, such as `[appserver_root]/server/<profile_name>/deploy`.
- Encrypt the password in the data source files (`adobe-ds.xml` and `mssql-ds.xml`) and the `login-config.xml` file using one of the methods described at <http://community.jboss.org/wiki/EncryptingDataSourcePasswords>. You can also use the instructions available on http://blogs.adobe.com/livecycle/2009/10/livecycle_-_encrypting_clearite.html.

12.6.3.1 Configuring the SQL Server database connectivity

Before you configure the SQL Server data source, you must have the LiveCycle database created on SQL Server. (See Creating a SQL Server database.)

12.6.3.2 Install the SQL database driver for JBoss

- 1 Obtain the SQL Server JDBC 3.0 database driver from the Microsoft website.

Note: Use SQL Server JDBC Driver 3.0 for both Microsoft SQL Server 2005 SP2 and Microsoft SQL Server 2008.

- 2 (Windows) Download the *.exe file and run it, and then extract the files to a temporary directory (referred to as the `[SQL_root]` directory in the remainder of this section).
- 3 (Linux) Extract the *.tar.gz files to a temporary directory (referred to as the `[SQL_root]` directory in the remainder of this section).
- 4 Copy the `sqljdbc.jar` file from the `[SQL_root]/sqljdbc_3.0/enu` directory to the `[appserver root]/server/<profile_name>/lib` directory.

- 5 Delete the mysql-connector-java-3.1.12-bin.jar file located in the `[appserver root]/server/<profile_name>/lib` directory.

12.6.3.3 Edit adobe-ds.xml file

- 1 Copy the adobe-ds.xml file from the `[DVD_root]/third_party/additional/datasources/lc_sqlserver/deploy` directory to the `[appserver root]/server/<profile_name>/deploy` directory.
- 2 Open the adobe-ds.xml file in a text editor and modify the `<local-tx-datasource>` element with your SQL Server connection settings:

```
<connection-url>jdbc:sqlserver://localhost:1433;DatabaseName=adobe</connection-url>
<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>
<user-name>adobe</user-name>
<password>adobe</password>
```

- 3 Replace the following values with values that are specific to your database:
 - `localhost`: The name, IP address, or fully-qualified path of the computer that hosts the database. The default is `localhost`.
 - `1433`: The port used to access the database.
 - `adobe`: The name of the database that stores the LiveCycle data. You will need to update the default value, `adobe`, with your database name.

- 4 Change the `<driver-class>` element as follows:

```
<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver</driver-class>
```

- 5 In the lines that follow the `<driver-class>` settings, locate the `user-name` and `password` settings and replace the default values with the user name and password that the application server uses to access your database. Modify the minimum and maximum values for the data source connections:

- IDP_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>30</max-pool-size>
```
- EDC_DS:

```
<min-pool-size>1</min-pool-size>
<max-pool-size>20</max-pool-size>
```

Note: If your LiveCycle Server handles heavy load, increase the number of maximum JDBC connections to ensure that all jobs are processed. In such cases, increase `<max-pool-size>` to 50 or more for both IDP_DS and EDC_DS.

- 6 Save and close the file.

12.6.3.4 Edit the mssql-ds.xml file

- 1 Navigate to the `[appserver root]/server/<profile_name>/deploy` directory and delete the `hsqldb-ds.xml` file.
- 2 Navigate to the `[appserver root]/docs/examples/jca` directory and copy the `mssql-ds.xml` file to the `[appserver root]/server/<profile_name>/deploy` directory.
- 3 Open the `mssql-ds.xml` file in a text editor and change the `<local-tx-datasource>` element with your SQL Server connection settings (not necessarily on consecutive lines):

```
<jndi-name>MSSQLDS</jndi-name>
<datasource-mapping>MS SQLSERVER2000</datasource-mapping>
<connection-url>jdbc:sqlserver://localhost:1433; DatabaseName=MyDatabase</connection-url>
<driver-class>com.microsoft.sqlserver.jdbc.SQLServerDriver </driver-class>
<user-name>x</user-name>
<password>y</password>
```

- The <jndi-name> element to DefaultDS
- The <datasource-mapping> element to MS SQLSERVER2000
- The Database name *MyDatabase* to your database name
- The <user-name>x</user-name> element to your user name
- The <password>y element to your password

Note: The <datasource-mapping> element should point to MS SQLSERVER2000, even if you are using MS SQL Server 2005.

- 4 Save and close the file.

12.6.3.5 Edit the login-config.xml file

- 1 Open the [appserver root]/server/<profile_name>/conf/login-config.xml file in a text editor and add the following lines within the <policy> element:

```
<application-policy name = "MSSQLDbRealm">
  <authentication>
    <login-module code =
      "org.jboss.resource.security.ConfiguredIdentityLoginModule"  flag = "required">
      <module-option name = "principal">adobe</module-option>
      <module-option name = "userName">adobe</module-option>
      <module-option name = "password">adobe</module-option>
      <module-option name =
        "managedConnectionFactoryName">jboss.jca:service=LocalTxCM,name= DefaultDS </module-
option>
    </login-module>
  </authentication>
</application-policy>
```

- 2 Replace the bold values with values that are specific to your database.
- 3 Save and close the file.
- 4 Start JBoss.

12.6.3.6 Configure Integrated Security on Windows

- 1 Modify the adobe-ds.xml and mssql-ds.xml files, located in [appserver root]/server/<profile_name>/deploy, to add integratedSecurity=true to the connection URL, as shown in this example:

```
<connection-url>jdbc:sqlserver://<serverhost>:<port>;
  databaseName=<dbname>;integratedSecurity=<true></connection-url>
```

- 2 Add the sqljdbc_auth.dll file to the Windows systems path (C:\Windows) on the computer that is running JBoss. The sqljdbc_auth.dll file is located within the Microsoft SQL JDBC 3.0 driver installation. The default location is [SQL_root]/sqljdbc_3.0/enu/auth/x86 for 32-bit operating systems and [SQL_root]/sqljdbc_3.0/enu/auth/x64 for 64-bit operating systems.
- 3 Open the properties for the JBoss for Adobe LiveCycle ES3 service or the JBoss service that you configured, and click the **Log On** tab.

- 4 Select **ThisAccount** and type the value of a valid user account. This change is not required if you are running JBoss from the command line.
- 5 Change SQL Server Security from Mixed mode to Windows Authentication only.

12.7 Next steps

Configure LiveCycle on JBoss Cluster by following the instructions provided in [Configuring LiveCycle Application Server Clusters Using JBoss](#)