INSTALLING ADOBE® DIGITAL ENTERPRISE PLATFORM EXPERIENCE SERVICES

Legal notices

Contents

Chapter 1: Adobe Digital Enterprise Platform Experience Services overview

The Adobe® Digital Enterprise Platform (ADEP) Experience Services is based on the Adobe® Digital Enterprise Platform Experience Server. Experience Services include the following components:

Experience Server Is a highly scalable content repository system that manages all the content as defined in the Content Repository for Java Technology API Version 2.0 specification, Java Content Repository (JCR) 2.0.

Foundation Provides the base platform for building Experience Services applications, and includes Data Services, security, and integration with Adobe Digital Enterprise Platform Document Services - User Manager.

Composite Application Framework Provides a framework for modular assembly of Flex and HTML assets into a rich experience for web and Adobe Interactive Runtime (AIR) environments.

Process Management Client Components Provides software components and a user interface for building, running, and managing business processes. It also provides integration with Adobe Digital Enterprise Platform - Process Management 10.0.

Platform SDK is a grouping of technologies that are characteristic of all Customer Experience Management (CEM) applications. The technologies are Security, Content, Data Services, Client Component Framework. Each of these technologies spans the client and server tiers. These capabilities are delivered as a single SDK package (containing SWC and JAR files) placed on the developer's file system and referenced by Adobe Flash Builder.

Experience Services Tooling Includes software and plug-ins for Adobe Flash Builder that enable you to create applications using Flex, Data Services, and Composite Application Framework.

Samples Includes Test Drive samples for Experience Services, Data Services, and Composite Application Framework.

For detailed overview of ADEP platform and its architecture and capabilities, see Adobe Digital Enterprise Platform Overview.

Chapter 2: System requirements

Operating System	Architecture	JDK
Microsoft Windows® Server 2008 R1/R2	64-bit OS / 64-bit JVM	Sun™ Java™ 6 update 26 or later updates to java 6 (64-bit)
Red Hat Enterprise Linux Server & AP 5.5	64-bit OS / 64-bit JVM	Sun™ Java™ 6 update 26 or later updates to java 6 (64-bit)

Note: *Installing ADEP Experience Services on Red Hat Enterprise Linux Server & AP 5.5 requires you to upgrade to ADEP Experience Services 10.0 service pack 1 (10.0.1). For more information about the service pack, visit https://www.adobe.com/cfusion/entitlement/index.cfm?e=adep_sp1.*

Note: *The 32-bit and 64-bit versions of Windows 7 and Windows Vista are supported only for evaluation and development purposes.*

2.1 JAVA_HOME and Path environment variable

Before you install Experience Services, ensure that you have the supported version of Sun JDK installed on your machine. Also, set the JAVA_HOME environment variable to point to the directory where the Sun Java JDK is installed.

2.1.1 Set environment variables on Windows

1. Select **Start** > **Control Panel** > **System**.
2. Click the **Advanced** tab and click **Environment Variables**.
3. In the **System Variables** area, click **New**.
4. In the **New System Variable** box, type `JAVA_HOME` as the variable name.
5. In the **Value** field, specify the path to the directory where you installed the Java JDK and that contains the `bin` directory. For example, `C:\Program Files\Java\jdk1.6.0_26`.
6. Click **Ok**.
7. In the System Variables area, select **Path** and click **Edit**.
8. Append `%JAVA_HOME%\bin;` at the beginning of the value.
9. Click **Ok**.

2.1.2 Set environment variables on Linux

1. Set the `JAVA_HOME` variable for Borne and Bash shells as shown in the following example:

   ```
   JAVA_HOME=/opt/IBM/WebSphere/AppServer/java
   export JAVA_HOME
   ```

   Note: *The specific path varies based on the installation directory you specified and the operating system you are installing on.*

2. Set the PATH variable for Borne and Bash shells as shown in the following example:

```
PATH=$JAVA_HOME/bin:$PATH
export PATH
```

2.1.3 Verify environment variable settings

Open the command prompt and execute the following command:

```
java -version
```

The command returns the JRE version installed on your machine (for example, 1.6.0_26).

Chapter 3: Install ADEP Experience Services

The Experience Services Quickstart is a JAR file, which when run in the GUI or CLI mode, automatically installs the Experience Server and Apache Sling, which is a web framework for developing web applications on Java. In addition, it deploys other Experience Services components described in "Adobe Digital Enterprise Platform Experience Services overview" on page 1.

Before proceeding to install the Experience Services Quickstart, review the *Before you Install ADEP* guide.

3.1 General consideration for installing Experience Services

Pre-installation

- Experience Services is installed in the same directory where the Quickstart JAR file is placed. Therefore, you must download or move the file to the desired installation directory.

  The installation directory for Experience Services is referred to as <*ExperienceServices root*> in ADEP documentation.

- The installation directory path must not contain any space characters.
- Ensure that there is no other JAR file or crx-quickstart directory present in the installation directory. It will cause problems with the installation.

During installation

- If the default or the specified port is not available, Experience Services configures the next available port. The default port number for the Experience Server is **4502**.
- When installed using the GUI mode, a Quickstart window is displayed, which displays an ON button when the Experience Server is running.

  You can click the ON button to stop the running Experience Server. The ON button changes to OFF as the Experience Server stops. To restart the Experience Server, double-click the JAR file.

- The first time you run the JAR file, it extracts all the files contained in it, which may take several minutes. The subsequent startups are quicker as the repository and other associated files have already been extracted and placed in the installation directory.
- The Experience Server is extracted into the <*ExperienceServices root*>/crx-quickstart directory.

Post-installation

- Do not delete the Quickstart from the installation directory as it will cause some operations to fail. Also, the JAR file is required to restart the server.

You can also install a Windows service for your Experience Server as described in http://blogs.adobe.com/ADEPhelp/2011/09/installing-windows-service-for-adep-experience-server.html.

- If you used a port number different from the default **4502**, you can determine the port number for the `-crx.quickstart.server.port` string in the stderr.log file present in the <*ExperienceServices root*>\crx-quickstart\log directory.

3.2 Run the Quickstart

1 Download the Experience Services Quickstart JAR file from https://www.adobe.com/cfusion/tdrc/index.cfm?product=adep_experience_services&loc=en_us.

2 ***(Optional)*** Change the default port number by renaming the JAR file. For example, adep-ria-quickstart-p<*port*>.jar. Replace <*port*> with the desired port number for the Experience Server.

3 ***(Optional)*** Configure the Experience Server security to integrate with Document Services by adding **-ds** to the filename. For example, adep-ria-quickstart-ds-p<*port*>.jar. This configuration is required if your Experience Services applications need to interact with the Document Services components and services.

Note: *You can use the -ds option to configure the Experience Server security only when you are running the Quickstart JAR for the first time. If you attempt to configure it later, you will receive a warning message. In that case, you will need to manually configure the Experience Server security as described in "6.2 Manually configure Experience Server" on page 12.*

4 Do one of the following:

- Run the Quickstart in GUI mode

  Double-click the JAR file to begin the installation program. It launches the Quickstart window, which displays the progress of the installation.

- Run the Quickstart in CLI mode

  Execute the following command to begin the installation program.

  ```
  java -jar adep-ria-quickstart-10-0-all-all.jar
  ```

  There are more options available with the Quickstart file to customize your installation as described in "3.3 CLI Options for the Quickstart" on page 6. You can also use the `-help` option to view the available options.

  Note: *When you fork the process using the `-fork` argument or the process is forked automatically in cases of lesser memory arguments passed, ensure that you specify any VM argument (for example, DataServices clustering argument) with `forkargs` parameter. As you start the Experience Services quickstart using the `java -jar` command, you will find information about processes being forked on System Out.*

When the installation is complete and the Experience Server is running, the Adobe Digital Enterprise Platform - License page opens in a browser window. If the License page does not open automatically, go to http://[host]:[port]/. The default values for host and port are **localhost** and **4502**, respectively.

5 On the Adobe Digital Enterprise Platform - License page, click **Click here to obtain a free trial license**.

6 Log in using your Adobe ID and password.

7 Click **Retrieve License Key**. The license key is displayed on the web page as well as sent to the email ID associated with your Adobe ID.

8 Go to the Welcome page or http://[*host*]:[*port*]/. Review and accept the End User License Agreement, specify the customer name and the license key, and click **Register** to register your license key.

The confirmation page with the registration details opens.

9 Click **Close** on the registration details page. The Adobe Digital Enterprise Platform - Sign In page opens.

10 Specify admin/admin as the username/password and click **Sign In**. The ADEP Experience Services Welcome page opens.

3.3 CLI Options for the Quickstart

The following table lists all the options available with the Quickstart file.

Option	Description
(-p,-port) <*port*>	Sets the port number for the Experience Server The default port number is **4502**.
-nobrowser	Indicates not to open browser at startup
-unpack	Unpacks installation files but does not start the server
-fe (-filename-regexp) <expr>	Specifies regular expression used to select part of the jar filename to use for setting system properties. Must contain one (group) used to extract the part of the filename to use. Use the java Pattern class syntax
-v (-verbose)	Indicates not to redirect stdout/stderr to files and do not close stdin
-sp <prop> [<prop> ...]	Sets a system property It overrides any other start up setting.
-nofork	Indicates not to fork the JVM, even if not running on a console
-fork	Enables forced forking the JVM if running on a console, using recommended default memory settings for the forked JVM
-forkargs <args> [<args> ...]	Specifies additional arguments for the forked JVM; defaults to **-Xmx1024M -XX:MaxPermSize=256m** Use -- to specify values starting with a hyphen (-). For example: -forkargs -- -server
-a (--interface) <interface>	Specifies the optional IP address (interface) to bind to
-ss (-shutdown-string) <string>	Specifies the string used to shutdown this application, when received on standard input
-pt <string>	Specifies the process type (main/fork) **Note**: do not use directly, used when forking a process
-r <string> [<string> [<string> [<string> [<string> [<string> [<string> [<string> [<string>]]]]]]]]	Enables you to define the run mode(s)
-b <string>	Enables you to define the path under which the quickstart work folder is created
-low-mem-action <string>	Specifies the low memory action when the memory is insufficient at startup

3.4 Install a Windows Service for Experience Server

You can optionally install a Windows services for your Experience Server to ensure that your Experience Server runs automatically each time your Windows restarts. Also, it will allow you to control the start and stop operations by using the Services control panel.

For more information, see http://blogs.adobe.com/ADEPhelp/2011/09/installing-windows-service-for-adep-experience-server.html.

Chapter 4: Prepare the development environment

You need to prepare the development environment for creating Experience Services applications.

You can install Flash Builder 4.5.0, which is an Eclipse-based Integrated Development Environment (IDE). Flash Builder enables you to create rich Internet applications for various devices using the Flex framework and ActionScript. See "4.1 Install Flash Builder" on page 8.

In addition, the following plug-ins for Flash Builder are available publically. These plug-ins compose the Experience Services tooling.

Adobe Data Model Content Repository Integration Components for Flash Builder Integrates application modeling technology with Experience Server. Facilitates the development of Data Services applications in the Experience Server environment.

Adobe Digital Enterprise Platform Content Repository Integration Components for Flash Builder Integrates data model and Flash Builder with Data Services and Experience Services components installed in the Content Repository.

Adobe Digital Enterprise Platform Data Services Modeler for Flash Builder Provides a framework for developing models using the data model technology. It is a common practice to build the Flex applications on top of an application model.

Adobe Digital Enterprise Platform Document Services - Discovery Plug-in for Flash Builder Enables you to quickly build Flex and AIR applications that invoke Document Services through its remoting endpoints. These applications are able to use Document Services to process data, create documents, and initiate workflows.

Adobe Experience Services Tooling for Flash Builder Enables you to create applications using Flex, Data Services, and Composite Application Framework.

4.1 Install Flash Builder

Obtain the Flash Builder 4.5 installer from http://www.adobe.com/go/adep_qstooling_fb450 and install the software as described in Flash Builder 4.5 Release Notes.

4.1.1 Configure Flash Builder (Mac OS X only)

1. Exit Flash Builder if it is running.
2. Open the Flash Builder installation directory in Finder.
3. Navigate to the <*FB_install_dir*>\eclipse\configuration directory, where <*FB_install_dir*> is the Flash Builder installation directory.

   Note: *There is another configuration directory in the <FB_install_dir>\ directory. However, ensure that you navigate to the configuration directory in the <FB_install_dir>\eclipse\ directory.*
4. Open the **config.ini** file in a text editor and remove the following lines:

```
osgi.configuration.area=@user.home/Documents/Adobe Flash Builder
4.5/cascaded/<build_number>/configuration
osgi.shared.configuration.area=file\:configuration
osgi.configuration.cascaded=true
```

5 Save and close the file.

The changes will come into effect the next time Flash Builder is started.

4.2 Install plug-ins for Flash Builder

1 Select **Start > All Programs**, right-click **Adobe Flash Builder 4.5**, and select **Run as administrator** to launch Flash Builder as an Administrator.

Note: *You must launch Flash Builder by using the* ***Run as administrator*** *option even if you are logged in with Administrator credentials.*

2 Click **Help > Install New Software.** The **Available Software** dialog opens.

3 From the **Work with** drop-down, select **Adobe Enterprise Suite extensions for Flash Builder 4.5 - http://aesplugins.adobe.com/aes/tooling/plugin/10.0**.

4 Click **Select All** to select all available plug-ins and click **Next**.

For optimum installation performance, deselect ***Contact all update sites during install to find required software****. However, if you deselect this option, Flash Builder will not receive updates, if any, from the update sites.*

5 Review the installation details and click **Next**.

6 Review the Adobe Systems Incorporated license agreement, accept the terms, and click **Finish**.

7 When the installation is complete, click **Restart Now**.

8 When Flash Builder restarts, click **Yes** when it prompts you to reset the perspective.

Important: *You must reset the perspective, otherwise the menu items contributed by the Experience Services tooling will not be available in the* ***File > New*** *menu in Flash Builder.*

Chapter 5: Deploy Experience Services SDK and Samples

The Experience Services SDK is required to develop applications using Composite Application Framework and Data Services.

5.1 Enable RDS on Experience Server

You must configure Remote Development Services (RDS) in your integrated development environment (IDE), and enable it on your Experience Server. Using RDS, IDE users can securely access development tools and activities, remote files, data sources, and so on.

1. Go to http://[*host*]:[*port*]/system/console and log in using admin/admin as the username/password.
2. Click the **Configuration** tab.
3. Click **Adobe Data Services**.
4. Select **Enable RDS**.
5. Click **Save**.

5.2 Download Experience Services SDK and samples

1. Go to http://[*host*]:[*port*] and log in using admin/admin as the username/password.
2. Click **Package Share** and log in using your Adobe ID user name and password.
3. Select **Public > Adobe > Digital Enterprise Platform > 10.0 > Product**.
4. Click **Download** next to the **es-sdk-pkg-1.0.186** package.
5. Select **Public > Adobe > Digital Enterprise Platform > 10.0 > Samples**.
6. Click **Download** next to the **es-samples-pkg-1.0.186.6** package.

Note: *Downloading refers to downloading the SDK and samples packages to Experience Server. They are not downloaded to your machine.*

5.3 Install Experience Services SDK and samples

1. Go to http://[*host*]:[*port*] and click **Packages**. The Package Manager screen opens.
2. Click **Install** next to the downloaded **es-sdk-pkg-1.0.186** package.
3. Click **Install** on the confirmation screen. When the installation is complete, a message is displayed in the Activity Log pane.
4. Go back to the Package Manager screen and click **Install** next to the downloaded **es-samples-pkg-1.0.186.6** package.

5 Click **Install** on the confirmation screen.

Note: *To verify the sample packages are installed, go to http://[host]:[port] and click Getting Started. The installed sample packages are listed under the Framework Samples section.*

5.4 Create a basic Flex project for ADEP

To create a Flex project for Experience Services, you need to first configure the Flash Builder plug-ins.

5.4.1 Configure plug-ins

Note: *Ensure that you restarted Flash Builder and chose to reset the perspective after installing the plug-ins for FlashBuilder. For details, see "4.2 Install plug-ins for Flash Builder" on page 9.*

Note: *Ensure that you have enabled RDS on your Experience Server as described in "5.1 Enable RDS on Experience Server" on page 10.*

1 Start Flash Builder.

2 Select **Windows > Preferences**.

3 In the left panel, select **Adobe > RDS Configuration**, and select **Experience Server (localhost)** in the **Currently Configured RDS Servers** panel.

Note: *If the Experience Server (localhost) option is not available, you must configure an RDS server as described in Configure an RDS server for use with an ADEP Flex project.*

4 Review the server information and click **Test Connection**.

5 Select **Adobe > Experience Services**.

6 In the **SDK Root Directory** field, specify the location where you want the SDKs to be saved.

7 Click **Retrieve SDK**.

8 Click **OK**.

5.4.2 Create and test a Flex project for ADEP

1 In Flash Builder, select **File > New > Flex Project for ADEP - Experience Services**.

2 Specify a project name and click **Next**.

3 Review the server information and click **Finish**. The project opens in Flash Builder.

4 Add elements and save the project.

5 Select **Run > Run** or press Ctrl+F11 to deploy the application on the Experience Server. It also launches the application in a browser.

Chapter 6: Integrating Experience Server with Document Server

For your Experience Services applications to interact with Document Server and leverage Document Services User Management, you must do the following configurations:

6.1 Integrate Experience Server with Document Server

1. Go to http://<*host*>:<*port*> and click **OSGi Console**. Log in using admin/admin as the username/password.
2. Click the **Document Services Settings** tab.
3. Specify the following information:
   - In the **Document Server Url** field, specify the fully qualified URL to the machine where the Document Server is installed.
   - In the **Username** and **Password** field, specify the Super Administrator credentials for the Document Server. The default username and password are administrator and password, respectively.
   - In the **Experience Server Url** field, specify the fully qualified URL to the machine where the Experience Server is installed.
   - In the **System user for accessing Experience Server**, use the default value, **crxuserfordsc,** or specify a Experience Server user for accessing the Experience Server from within the Document Server.
   - In the **System user for accessing Document Server**, use the default, **dscuserforcrx**, or specify a Document Server user for accessing the Document Server from within the Experience Server.

     Important: *Deselect* ***Reset password for Document Server System User*** *if you do not want to reset the password.*
4. Click **Configure**.

Important: *If your Experience Server and Document Server are on different machines, ensure that you synchronize their system clocks. Otherwise, Document Server may not recognize the session from the Experience Server.*

6.2 Manually configure Experience Server

Important: *Perform these steps only if you want Document Services users to log on to the Experience Server.*

Important: *You can skip this configuration if you renamed the Quickstart JAR filename to include* ***-ds*** *before installing Experience Services.*

1 Open the <*ExperienceServices root*>/crx-quickstart/repository/repository.xml file, and comment out the following lines of code:

```
<LoginModule class="com.day.crx.core.CRXLoginModule">
    <param name="anonymousId" value="anonymous"/>
    <param name="adminId" value="admin"/>
</LoginModule>
```

2 Add the following module:

```
<Module class="com.adobe.livecycle.usermanager.crx.clientsdk.LCClientSDKModule">
    <param name="configWspName" value="crx.default"/>
    <param name="configPath" value="/apps/docservices/config" />
</Module>
```

3 Verify the following entry in the <*ExperienceServices root*>/crx-quickstart/server/etc/jaas.cofig file:

```
com.day.crx {
com.adobe.livecycle.usermanager.crx.loginmodule.LCAwareCRXLoginModule sufficient
principalProvider="com.adobe.livecycle.usermanager.crx.loginmodule.UMPrincipalProvider"
;
com.adobe.livecycle.usermanager.crx.loginmodule.LCLoginModule required
principalProvider="com.adobe.livecycle.usermanager.crx.loginmodule.UMPrincipalProvider"
;
};
```

4 Open the server.bat file present in the <*ExperienceServices root*>/crx-quickstart/server/ directory in a text editor.

5 Change the values for the `JVM_MAXHEAP` and `JVM_PERMGEN` properties according to the recommended memory settings for JVM, if required.

6 Save and close the server.bat file.

7 Restart the server using the server.bat file.

Important: *To retain the security settings, you must use the server.bat file for any subsequent server restart.*

6.3 Verify Experience Server security configuration

To verify that the Experience Services security is configured to authenticate users against the Document Server:

1 Go to http://[*host*]:[*port*]/ and click **Sign Out**.

2 Go to the Experience Services repository, http://[*host*]:[*port*]/crx/, and log in using the administrator credentials for the Document Server. The default user name and password are administrator and password, respectively.

3 Log out and log in again using admin/admin as the username/password.

4 Click **User Administration**.

5 Expand the folder **Users > l > lc**.

If you see **lcu::DefaultDom::SuperAdmin** in the list of users, your Experience Services security is properly configured.

Important: *Users will have limited access to the protected resources on the Experience Server unless they are assigned to some group and given appropriate rights. For information about how to administering user, groups, and rights, see Administering ADEP Experience Services.*

6.4 Enable Single Sign-On authentication on Experience Server

To configure header-based Single Sign-On (SSO) authentication on Experience Server:

Modify the jaas.config file

1 Open the <*ExperienceServices root*>/crx-quickstart/server/etc/jaas.config file in a text editor.

2 Locate the following lines and add the text in bold.

```
com.day.crx {
com.adobe.livecycle.usermanager.crx.loginmodule.LCAwareCRXLoginModule sufficient
principalProvider="com.adobe.livecycle.usermanager.crx.loginmodule.UMPrincipalProvider"
trust_credentials_attribute="attribute_name"
;
com.adobe.livecycle.usermanager.crx.loginmodule.LCLoginModule required
principalProvider="com.adobe.livecycle.usermanager.crx.loginmodule.UMPrincipalProvider"
trust_credentials_attribute="attribute_name"
;};
```

Note: *Replace* ***atrtribute_name*** *with any attribute name of your choice. However, ensure that you specify the same attribute name when configuring the SSO authentication handler.*

3 Save and close the file.

4 Restart the Experience Server.

Configure SSO authentication handler

1 Go to http://<*host*>:<*port*>/system/console/configMgr and log in using admin/admin as the username/password.

2 In the Configuration tab, click **Adobe Granite SSO Authentication Handler**.

3 Specify the following information:

- In the **Header Names** field, specify the name of the header that contains the user ID.
- In the **Format** field, type **AsIs** or **Basic**, based on your requirement:

  AsIs Configures the SSO solution to send the user ID as the value of the header in plain text or any regular expression

  Basic Configures the SSO solution to send the user ID as the value of the header encoded with Base64 encoding in the HTTP Basic Authentication format

- In the **Trusted Credential Attribute** field, specify the attribute name you added in the jaas.config file.

4 Click **Save**.

6.5 Next Steps

For information about setting up a developer environment for the Experience Services platform and key workflows, such as building a simple model-driven application, see Getting Started for Developers guide.

Chapter 7: Appendix - Uninstalling Experience Services Packages

Uninstalling an Experience Services package refers to restoring the state of the Experience Services repository to just before the package was installed. You can uninstall a package from the Package Manager.

7.1 Uninstall Experience Services Packages

1 Go to http://[*host*]:[*port*] and click **Packages**. The Package Manager screen opens.

2 Click the package that you want to uninstall.

3 From the More drop-down list, select **Uninstall**.

4 Click **Uninstall** on the confirmation dialog to begin uninstallation.

Note: *Uninstalling the package doesn't remove it from Experience Server. You can reinstall the package later. To completely remove the package, click* ***Delete*** *from the* ***More*** *drop-down.*

7.2 Uninstall ADEP Experience Services

The Experience Services Quickstart installs all the ADEP components in the installation directory. Therefore, you must delete the installation directory to uninstall ADEP. It removes all components including the repository.

Note: *To keep the repository data before deleting, make a copy of the <ExperienceServices root>/repository directory before deleting the installation directory.*

Chapter 8: Appendix - Install mosaic release package for Composite Application Framework

In a developer environment, you must continue to use the debug variant of the mosaic package for Composite Application Framework as this package is required for developing composite applications. However, on a production server, it is recommended to use the release variant of the mosaic package.

Before uninstalling the debug mosaic from the production server, you must preserve your custom applications and catalogs. Otherwise, you may lose them.

The entire workflow of uninstalling the mosaic debug package and installing the release mosaic package is as follows:

1 "8.1 Preserve your applications and catalogs" on page 16
   - Create a package
   - Add applications and catalogs to the package
   - Build the package
2 "8.2 Uninstall debug mosaic and install release mosaic" on page 17
3 "8.3 Install custom applications and catalogs packages" on page 17

8.1 Preserve your applications and catalogs

1 Go to http://[*host*]:[*port*] and click **Packages**. The Package Manager screen opens.
2 Click **Create Package**.
3 In the New Package dialog, specify the package name and version number for your package. For example, **mypackage** and **1.0**, respectively.
4 Click **OK**. The package you created is displayed as **mypackage-1.0.zip**.
5 Click **Edit** on mypackage-1.0.zip.
6 Add a description for mypackage-1.0.zip.
7 In the **Filters** tab, click **Add Filter**.
8 In the **Root path** field, specify **/content/mosaic/applications/<*custom-app*>**, where <*custom-app*> is the application you want to preserve.
9 Click **Done**.
10 Click **Add Filter**.
11 In the **Root path** field, specify **/content/mosaic/catalogs/<*custom-catalog*>**, where <*custom-catalog*> is the catalog you want to preserve.
12 Click **Done**.
13 Click **Save**.

14 Click **Build** next to mypackage-1.0.zip. The Activity Log shows when the package is built.

8.2 Uninstall debug mosaic and install release mosaic

1 Go to http://[*host*]:[*port*] and click **Packages**. The Package Manager screen opens.

2 Click **mosaic-debug-pkg-2.0.82**.

3 From the **More** drop-down, click **Uninstall**.

4 Click **Uninstall** on the confirmation dialog to start the uninstallation process.

5 (Optional) When the uninstallation is complete, click **Delete** from the **More** drop-down.

6 Go to http://[*host*]:[*port*] and click **Package Share**.

7 Select **Public > Adobe > Digital Enterprise Platform > 10.0 > Product**.

8 Click **Download** next to the **mosaic-pkg-2.0.82** package.

9 Click **Install**.

10 Click **Install** on the confirmation dialog to begin the installation process.

8.3 Install custom applications and catalogs packages

1 Go to http://[*host*]:[*port*] and click **Packages**. The Package Manager screen opens.

2 Click the <*custom_app*> package you created in "8.1 Preserve your applications and catalogs" on page 16.

3 Click **Install**.

4 Repeat steps 2 and 3 for your package that contains custom catalogs.