

Migration Considerations for XML/SOAP Components

The migration strategy depends on the version of EntireX you have been using. This chapter covers the following topics:

- Migrating XML/SOAP Components for EntireX Version 8 and above
 - Migrating XML/SOAP Components from EntireX Version 7.n
-

Migrating XML/SOAP Components for EntireX Version 8 and above

This section covers the following topics:

- Migrating XML/SOAP RPC Server
- Migrating XML/SOAP Wrapper
- Migrating XML/SOAP Listener

Migrating XML/SOAP RPC Server

Update the start script to set the paths to current installation. If the properties file contains a path to the configuration file, you may need to adjust the path. The properties and configuration files can be used as before. See *Configuration File for the XML/SOAP RPC Server* under UNIX | Windows for more information.

Migrating XML/SOAP Wrapper

Update the start script to set the paths to current installation.

Migrating XML/SOAP Listener

Using the Software AG Web Server based on Apache Tomcat / Software AG Runtime from Software AG Installation

Save your Web services (AAR files) and modified configuration files (i.e. *axis2.xml*). You do not need to re-generate your Web services (AAR files). If the configuration was modified, perform the same modifications for the new installation and restart Software AG Web Server. Redeploy the web services (AAR files).

Note:

The new installation will use another port by default, which means that the Web clients need to be adjusted.

Using another Web Server

Save your Web services (AAR files) and modified configuration files (i.e. *axis2.xml*). You do not need to re-generate your Web services (AAR files). For deployment of file *wsstack.war* and copying file *entirex.jar* to Web Services Stack, see the separate Web Services Stack documentation, also available under <http://documentation.softwareag.com> > *webMethods Product Line* > *Shared Services*.

If configuration was modified, perform the same modifications for new installation and restart the Web server. Redeploy the Web services (AAR files).

Migrating XML/SOAP Components from EntireX Version 7.n

The architecture of the EntireX XML/SOAP Listener (formerly known as XML Servlet) and EntireX XML/SOAP RPC Server was changed with version 8.0. The previous version worked as a Web application running directly in the Web server; the current version works as part of Software AG Common Web Services Stack (WSS), which is a Web application running in a Web server. This change has impact on configuration, packaging and deployment of EntireX Web services.

Note:

This section describes migration from a version earlier than 8.

- Migrating XML/SOAP RPC Server from Version 7.n
- Migrating XML/SOAP Wrapper from Version 7.n
- Migrating XML/SOAP Listener from Version 7.n - Introduction
- Migrating from Version 7.n to one or more Listener Services with new URL(s)
- Migrating from Version 7.n to one Listener Service, Keeping the URL used by Applications

Migrating XML/SOAP RPC Server from Version 7.n

The XML/SOAP RPC Server requires the directory of a Software AG Common Web Services Stack installation. The JAR file *wsstack-client.jar* located in `<WS_STACK_HOME>/lib` must be added to `classpath`. (Note: the JAR file *wsstack-client.jar* must remain in this *lib* directory). File *entirex.jar* from EntireX installation must also be present in `classpath`.

The XML parser required has changed. Therefore, the (optional) properties file (if used) must be modified by redefining the parameter for the XML stream parser in the following way:

```
entirex.sdk.xml.runtime.xmlparserfactory=com.ctc.wstx.stax.WstxInputFactory
```

The configuration file can be used as before. To make use of the latest features, some extensions in the configuration file are required. See *Configuration File for the XML/SOAP RPC Server* under UNIX | Windows for more information.

Migrating XML/SOAP Wrapper from Version 7.n

The used XML parser has changed. XML/SOAP Wrapper uses XMLStreamParser, so the classpath must contain the JAR files of XMLStreamParser. The EntireX installation comes with *stax-api.jar* and *wstx-asl.jar*. The two XMLStreamParser JAR files are located in `<EntireX Home>/classes`.

If you are compiling source using the class `XMLRPCService`, you require an additional JAR file *axiom-api-<version>.jar*. This JAR file is located in `<WebservicesStack>/lib`.

Migrating XML/SOAP Listener from Version 7.n - Introduction

The XML/SOAP Listener requires a Software AG Common Web Services Stack (WSS) installation. The WSS Web application is the container of XML/SOAP Listener. In previous EntireX versions (lower than version 8) the XML/SOAP Listener was referred to as "XML Servlet" and was a Web application itself. The file *entirex.jar* containing the XML/SOAP Runtime is placed in directory *lib* of the WSS Web application. An EntireX Web service is created using the XML Mapping Editor as in previous versions, and using a Packaging Wizard to build an AAR file. The AAR file contains all information for this Web service and must be deployed to the Web application.

The following new terms are applicable:

- **AAR Files**

EntireX packs the XMM file(s) and some more files in a Web service archive (.aar) that is deployed in the Web application WSSStack. To put one or more XMM files in one Web service archive, select the XMM files and from the context menu choose **Web Service > Generate Web Service....** All the files you select must have the same mapping type (i.e. SOAP or non-SOAP mapping); do not mix mapping types.

- **File *entirex-forward.war***

The servlet `entirex-forward` only forwards the message to the new URL. This means you can continue using the old URL and migrate the service to a new infrastructure. The servlet does not process the payload, so this Web application does not contain an *entirex.jar* file and does not contain a mapping file (XMM).

The following scenarios are available for migrating a service offered by XML Servlet of earlier EntireX versions (lower than version 8):

- migrating to one or more services, using new URL(s); see *Migrating from Version 7.n to one or more Listener Services with new URL(s)*
- migrating to one service only, keeping the URL; see *Migrating from Version 7.n to one Listener Service, Keeping the URL used by Applications*

Migrating from Version 7.n to one or more Listener Services with new URL(s)

You can migrate a service offered by XML Servlet of earlier EntireX versions (lower than version 8) to one or more services in the context of XML/SOAP Listener with new URL(s). That means you can split the original service into different services, each with its own URL. However, this means you need to update the address in the Web client.

➤ **To migrate a service with new URLs**

1. Copy all the IDL-XML mapping files (XMMs) listed in file *xml-init.xml* to one Eclipse project.
2. Select multiple XMM files that are to be bundled in one service, and from the context menu choose **Web Service > Generate Web Service...**
3. Modify the service name if required.
4. Clear **Use of defaults**.
5. Check **General service parameters (xml-init.xml)** if you have any XMM-independent settings in file *xml-init.xml*.
6. Check **Set connection and security parameters in mapping file**.
7. Follow the wizard.
8. Modify the setting if required. The included XMM files can be selected with combo box at the top.
9. Deploy the service.
10. Change the target URL of calling instances (Web clients).

Migrating from Version 7.n to one Listener Service, Keeping the URL used by Applications

The first step is to generate one service archive with all XMM files listed in file *xml-init.xml*. The second step is the configuration and usage of *entirex-forward.war*. This section covers the following topics:

- Step 1: Generate Service Archive
- Step 2: Use File *entirex-forward.war*
- Example of Migrating with Servlet *entirex-forward*

Step 1: Generate Service Archive

➤ To generate a service archive

1. Copy all IDL-XML mapping files (XMMs) listed in *xml-init.xml* to one Eclipse project.
2. Select all XMM files, and from the context menu choose **Generate Web Service from EntireX Mapping ...**
3. Modify the service name if required. This service name is used for configuring of servlet *entirex-forward* later.
4. Clear **Use of defaults**.
5. Check **General service parameters (xml-init.xml)** if you have any XMM-independent settings in file *xml-init.xml*.
6. Check **Set connection and security parameters in mapping file**.
7. Follow the wizard.
8. Modify the setting if required. The included XMM files can be selected with combo box at the top.
9. Deploy the service.

Step 2: Use File *entirex-forward.war*

This Web application delivered as *entirex-forward.war* allows you to forward a SOAP/XML request from an endpoint used in an older EntireX version (e.g. *http://hostname:port/entirex/xmlrt*) to an endpoint used in the current EntireX version (e.g. *http://hostname:port/wsstack/services/example*).

The steps below explain how to forward, for example, from the endpoint *http://localhost:10010/entirex/xmlrt* to the endpoint *http://localhost:10010/wsstack/services/example*.

Important:

The servlet *entirex-forward* forwards the calls to exactly one service and replaces the old EntireX Web application; it does not run in parallel (old EntireX XML Servlet and the new servlet *entirex-forward*).

Note:

You *cannot* use the file *entirex-forward.war* to forward a call from one Web application server to another.

➤ To forward a SOAP/XML request

1. Rename *entirex-forward.war* to *entirex.war*.
2. Deploy *entirex.war* into the *webapps* folder of your application server (e.g. Tomcat).
3. Make sure that the Web Services Stack Web application (*wsstack.war*) is still deployed in the same *webapps* folder.
4. Make sure that a service with the name "example" is deployed into the Web Services Stack Web application. See Step 4 under *Step 1: Generate Service Archive*.
5. Make sure that your application server allows cross-context access. Set the `crossContext` attribute of all `Context` implementations to "true". In Apache Tomcat you can modify the file `$CATALINA_HOME/conf/context.xml`:

Set

```
<Context>
```

to

```
<Context crossContext="true">
```

6. Make sure that the file *web.xml* (located in *webapps/entirex/WEB-INF*) contains the following entries:

```
<servlet>
  <description>
    This is the EntireX ForwardServlet</description>
  <display-name>
    forward</display-name>
  <servlet-name>forward</servlet-name>
  <servlet-class>
    com.softwareag.entirex.xml.forward.ForwardServlet</servlet-class>
  <init-param>
    <description>
      This is the targetURI</description>
    <param-name>targetURI</param-name>
    <!--replace the servicename with service name defined in aar -->
    <param-value>/services/servicename</param-value>
  </init-param>
  <init-param>
    <description>
      This is the target context i.e. /wsstack</description>
    <param-name>targetContext</param-name>
    <param-value>/wsstack</param-value>
  </init-param>
</servlet>
<servlet-mapping>
  <servlet-name>forward</servlet-name>
  <url-pattern>/xmlrt</url-pattern>
</servlet-mapping>
```

The old endpoint `http://hostname:port/entirex/xmlrt` is built from the name of the Web application ("entirex.war") and the `url-pattern` of the servlet `entirex-forward` ("/xmlrt"). The new endpoint is built from the `targetContext` (/wsstack) and the `targetURI` ("services/servicename").

You can modify these parameters to adapt your Web service.

Example of Migrating with Servlet `entirex-forward`

This section describes how a migrated service interacts with servlet `entirex-forward`.

For this example, the Tomcat installation that comes with the EntireX installation (Version 8.2) is used. The assumed endpoint from the previous XML Servlet installation is `http://localhost:10010/entirex/xmlrt`. The endpoint of the migrated service will be `http://localhost:10010/wsstack/services/exampleXML`.

The start point is file `xml-init.xml` (below), with its mapping file `example.xmm`.

Follow the steps above for generating a Web service archive file (*aar) with service name "exampleXML" (see *Step 1: Generate Service Archive*). The installation contains the generated `exampleXML.aar` in directory `<installation home>/EntireX/examples/RPC/basic/example/XMLClient`. File `exampleXML.aar` is also deployed during installation if the Software AG Common Platform was selected.

The last step is to configure and deploy the servlet `entirex-forward`.

➤ To configure and deploy the servlet `entirex-forward`

1. Rename `entirex-forward.war` to `entirex.war`.
2. Deploy `entirex.war` to the `webapps` folder of your application server (e.g. Tomcat).
3. Make sure that the Web Services Stack Web application (`wsstack.war`) is still deployed in the same `webapps` folder.
4. Make sure that a service with the name "exampleXML" is deployed into the Web Services Stack Web application. If the Web server is running, start a browser session and enter URL `http://localhost:10010/wsstack/services`. The service "exampleXML" should be listed.
5. Make sure that your application server allows cross-context access. See the `crossContext` attribute of all `Context` implementations to "true". In Apache Tomcat you can modify file `$CATALINA_HOME/conf/context.xml`:

Set

```
<Context>
```

to

```
<Context="true">
```

6. Make sure that file `web.xml` (located in `webapps/entirex/WEB-INF`) contains the following entries:

```
<servlet>
  <description>
    This is the EntireX ForwardServlet</description>
  <display-name>
    forward</display-name>
  <servlet-name>forward</servlet-name>
  <servlet-class>
    com.softwareag.entirex.xml.forward.ForwardServlet</servlet-class>
  <init-param>
    <description>
```

```

    This is the targetURI</description>
    <param-name>targetURI</param-name>
    <param-value>/services/exampleXML</param-value>
  </init-param>
  <init-param>
    <description>
      This is the target context i.e. /wsstack</description>
    <param-name>targetContext</param-name>
    <param-value>/wsstack</param-value>
  </init-param>
</servlet>
<servlet-mapping>
  <servlet-name>forward</servlet-name>
  <url-pattern>/xmlrt</url-pattern>
</servlet-mapping>

```

The old endpoint `http://hostname:port/entirex/xmlrt` is built from the name of the Web application ("entirex.war") and the URL pattern of the servlet `entirex-forward ("/xmlrt")`. The new endpoint is built from the `targetContext (/wsstack)` and the `targetURI ("/services/exampleXML")`.

7. Restart Tomcat.
8. Open the Software AG Designer and change to the EntireX perspective.
9. Choose **Windows > Show View > XML Tester** to open the XML Tester.
10. Browse for `<installation home>/EntireX/examples/RPC/basic/example/XMLClient/example.xmm`.
11. Create a sample request document.
12. Start Java RPC server with script `jrpcserver.bat` or `jrpcserver.bsh` (depending on platform) in directory `bin` of the EntireX installation that is used as a back-end.
13. Select **HTTP Test (XML/SOAP listener)**.
14. Set **HTTP URI** to `"http://localhost:10010/wsstack/services/exampleXML"`.
15. Send request to see the service works.
16. Set **HTTP URI** to `"http://localhost:10010/entirex/xmlrt"`.
17. Send request again (the request is sent to "old" address and will be processed in the XML/SOAP Listener embedded in Web Services Stack).