
Version 6.1.3

MessageWay AS2 Installation Guide



Document History

Part Number	Product Name	Date
MW410-575	MessageWay AS2 Installation Guide	06/2018
MW410-580	MessageWay AS2 Installation Guide	01/2020
MW410-585	MessageWay AS2 Installation Guide	03/01/22

Copyright

© 2022 Progress Software Corporation and/or one of its subsidiaries or affiliates. All rights reserved.

These materials and all Progress® software products are copyrighted and all rights are reserved by Progress Software Corporation. The information in these materials is subject to change without notice, and Progress Software Corporation assumes no responsibility for any errors that may appear therein. The references in these materials to specific platforms supported are subject to change.

Chef, Chef (and design), Chef Infra, Code Can (and design), Compliance at Velocity, Corticon, DataDirect (and design), DataDirect Cloud, DataDirect Connect, DataDirect Connect64, DataDirect XML Converters, DataDirect XQuery, DataRPM, Defrag This, Deliver More Than Expected, DevReach (and design), Icenium, Inspec, Ipswitch, iMacros, Kendo UI, Kinvey, MessageWay, MOVEit, NativeChat, NativeScript, OpenEdge, Powered by Chef, Powered by Progress, Progress, Progress Software Developers Network, SequeLink, Sitefinity (and Design), Sitefinity, Sitefinity (and design), SpeedScript, Stylus Studio, Stylized Design (Arrow/3D Box logo), Styleized Design (C Chef logo), Stylized Design of Samurai, TeamPulse, Telerik, Telerik (and design), Test Studio, WebSpeed, WhatsConfigured, WhatsConnected, WhatsUp, and WS_FTP are registered trademarks of Progress Software Corporation or one of its affiliates or subsidiaries in the U.S. and/or other countries.

Analytics360, AppServer, BusinessEdge, Chef Automate, Chef Compliance, Chef Desktop, Chef Habitat, Chef WorkStation, Corticon.js, Corticon Rules, Data Access, DataDirect Autonomous REST Connector, DataDirect Spy, DevCraft, Fiddler, Fiddler Everywhere, FiddlerCap, FiddlerCore, FiddlerScript, Hybrid Data Pipeline, iMail, JustAssembly, JustDecompile, JustMock, KendoReact, NativeScript Sidekick, OpenAccess, PASOE, Pro2, ProDataSet, Progress Results, Progress Software, ProVision, PSE Pro, Push Jobs, SafeSpaceVR, Sitefinity Cloud, Sitefinity CMS, Sitefinity Digital Experience Cloud, Sitefinity Feather, Sitefinity Insight, Sitefinity Thunder, SmartBrowser, SmartComponent, SmartDataBrowser, SmartDataObjects, SmartDataView, SmartDialog, SmartFolder, SmartFrame, SmartObjects, SmartPanel, SmartQuery, SmartViewer, SmartWindow, Supermarket, SupportLink, Unite UX, and WebClient are trademarks or service marks of Progress Software Corporation and/or its subsidiaries or affiliates in the U.S. and other countries. Java is a registered trademark of Oracle and/or its affiliates. Any other marks contained herein may be trademarks of their respective owners.

This document was published on Friday, February 18, 2022 at 13:24.

MessageWay AS2 Installation Guide

Contents

Introduction	3
Overview of AS2 Interface	3
Purpose and Scope	3
Audience	3
Critical Dependencies	3
MessageWay AS2 Default Ports.....	4
MessageWay AS2 Interface Default Locations	4
Technical Support.....	7
Installing the AS2 Interface	8
Licensing Requirements for the AS2 Interface	8
Prerequisites for the MessageWay AS2 Interface	9
Install the AS2 Servlets.....	9
Enable SSL (optional)	12
Install the AS2 Adapter	12
Start and Stop the AS2 Interface	12
Post-Install Best Practices	16
Upgrading the AS2 Interface	17
Uninstalling the MessageWay AS2 Interface	17
Uninstalling the MessageWay AS2 Interface from Windows	17
Uninstalling the MessageWay AS2 Interface from UNIX/Linux.....	18

This page intentionally blank

Introduction

Overview of AS2 Interface

Applicability Statement 2 (AS2) is a specification to transfer structured business data between computers using the Internet Web page protocol, the Hypertext Transfer Protocol (HTTP). Although any structured data format may be used, the prevalent formats are XML or Electronic Data Interchange (EDI), such as ANSI X12 or UN/EDIFACT.

The AS2 standard uses both Multi-Purpose Internet Mail Extensions (MIME) and Secure Multi-Purpose Internet Mail Extensions (S/MIME) to package data. It uses HTTP or a more secure version, HTTPS, to transmit data over the Internet. It supports data integrity and authenticity through the optional use of X.509 (**RFC2459** (<http://www.ietf.org/rfc/rfc2459.txt>)) certificates, an international standard (Recommendation) for public key infrastructure (PKI) certificates.

The MessageWay AS2 interface provides the AS2 protocol interface between MessageWay and a remote host.

The AS2 Interface is based on the following reference:

Applicability Statement 2 (AS2) **RFC 4130** (<http://www.ietf.org/rfc/rfc4130.txt>)

Purpose and Scope

This document covers the installation procedures for MessageWay AS2 Interface on perimeter servers.

For installation instructions for the MessageWay AS2 Adapter, refer to topic ***Installing or Upgrading Additional Adapters or Services*** in the ***MessageWay Installation Guide***.

Audience

This document is intended for users who want to install MessageWay AS2 Interface on their server.

Whoever installs MessageWay AS2 Servlets should be familiar with their operating system.

Critical Dependencies

Critical dependencies include versions of software that users must install prior to installing MessageWay AS2 Interface, or it will not install or operate successfully. This software is not packaged with MessageWay AS2 installations.

MessageWay AS2 version 6.1.3 has the following critical dependencies:

MessageWay Component	Dependency
AS2 Interface	<ul style="list-style-type: none">▪ Java version 8.x▪ Apache Tomcat version 7.x, 8.x or 9.x▪ MessageWay version 6.1 or later

MessageWay AS2 Default Ports

The following tables describe the default ports used by AS2:

Server	Port #	Description
MessageWay AS2 Interface	8080	HTTP Listener
	8443	HTTPS Listener (optional)

MessageWay AS2 Interface Default Locations

The following are the default locations where MessageWay AS2 Interface files are installed. The locations vary depending on the system, Windows or UNIX/Linux.

MessageWay Files and Locations for Windows

The following table shows the default locations where MessageWay AS2 Interface installs files that support its servers on a Windows system.

Description	Locations and Files
AS2 Interface	<pre> <i>webserversdirectory</i> \mwas2 \WEB-INF \classes GetMWayAS2Ver.sh GetMWayAS2Ver.class GetVer_inedi.class log4j2.xml MWayAS2In\$1.class MWayAS2In\$About.class MWayAS2In\$SessionContent.class MWayAS2In.class MWayAS2Out\$1.class MWayAS2Out\$About.class MWayAS2Out\$SessionContent.class MWayAS2Out.class \lib inedi.jar log4j-api-2.17.0.jar log4j-core-2.17.0.jar mwayservice.jar mwas2.conf web.xml \Windows\System32\NTEventLogAppender.dll </pre>

MessageWay Files and Locations for UNIX/Linux

The following table shows the default locations where MessageWay installs files that support its servers on a Linux or UNIX system.

Description	Locations and Files
AS2 Interface	<pre> <i>webserversdirectory</i> /mwas2 /WEB-INF /classes GetMWayAS2Ver.class GetMWayAS2Ver.sh GetVer_inedi.class log4j2.xml MWayAS2In\$1.class MWayAS2In\$About.class MWayAS2In\$SessionContent.class MWayAS2In.class MWayAS2Out\$1.class MWayAS2Out\$About.class MWayAS2Out\$SessionContent.class MWaAS2Out.class /lib inedi.jar log4j-api-2.17.0.jar log4j-core-2.17.0.jar mwayservice.jar mwas2.conf web.xml </pre>

Technical Support

The MessageWay Technical Support hub is an information and diagnostic center available for customers to:

- Obtain advice on proper product installation, configuration, and operation
- Report any product problems and receive timely resolutions
- Request a software enhancement
- Request software updates
- Inquire about software release contents and status
- View publications
- See how to contact Technical Support
- See hours of availability for Technical Support

To visit the MessageWay Technical Support hub, please follow the below link:

<https://www.progress.com/support/messageway>

The Technical Support Web site is available 24/7, portions of which require a valid Progress ID. If you have not already done so, you can follow the instructions in the following URL to obtain a valid Progress ID:

<https://knowledgebase.progress.com/articles/Article/how-to-create-a-progress-id>

Installing the AS2 Interface

This is a check-list of the tasks you will perform to install the AS2 Interface and then configure it for initial testing. The following instructions install the interface. To configure the AS2 Interface, refer to topic *Configuring the AS2 Interface* in the *MessageWay User's Guide and Reference*.

The installation process installs the components of the AS2 Interface. The installation process also requires a Java Runtime Environment and a Web container, such as Apache Tomcat. For specific MessageWay and third-party requirements, refer to the topic, *Prerequisites for the MessageWay AS2 Interface* (on page 9).

These tasks assume that you have already installed MessageWay, which includes the following components of interest here:

- MessageWay Messaging Server, which processes messaging requests
- MessageWay User Server, which controls access to MessageWay
- MessageWay Service Interface, which provides access to MessageWay from MessageWay servers and the Internet
- MessageWay Manager, which provides the user interface to configure MessageWay

These are the basic tasks to install the MessageWay AS2 Interface:

- Install the MessageWay AS2 inbound and outbound servlets on any system within a Web container in the LAN or WAN
- Install the MessageWay AS2 adapter on the system where the MessageWay Server runs

After you have installed the AS2 Interface, refer to topic *Configuring the AS2 Interface* in the *MessageWay User's Guide and Reference* to perform the following tasks to configure and test the system:

- Modify the configuration file for the Service Interface on the MessageWay system
- Modify the configuration files for the AS2 Interface
- Configure MessageWay users and locations
- Create and populate Java Key Store (.jks) file to manage X.509 certificates
- Start the Service Interface
- Start the Web container, such as Apache Tomcat, which starts the AS2 servlets
- Test the connection from a browser to the AS2 Interface
- Test inbound AS2 transmissions
- Start the AS2 adapter
- Test outbound AS2 transmissions

Licensing Requirements for the AS2 Interface

Licenses are required for the MessageWay AS2 servlets and the AS2 adapter. For more information, contact **MessageWay Technical Support**.

IMPORTANT: You must obtain a license before starting the AS2 adapter. The license for the AS2 servlets is already embedded in the software.

Prerequisites for the MessageWay AS2 Interface

The MessageWay AS2 Interface has been developed with and requires at a minimum specific versions of MessageWay software as well as third-party software. These are the prerequisites:

Software Component	Minimum Requirement
<i>Java Runtime Environment (JRE)</i> (http://www.java.com)	Java version 8.x
Web container that supports Java Servlet specification 2.4, such as <i>Apache Tomcat</i> (http://tomcat.apache.org/)	Apache Tomcat version 7.x, 8.x or 9.x
MessageWay	Version 6.1 or later

IMPORTANT: Tomcat must be installed as a service. Not all versions of Tomcat do this automatically. Refer to the installation instructions provided with Tomcat, since the instructions may vary between versions.

Install the AS2 Servlets

IMPORTANT: Older versions of the AS2 servlets must be uninstalled before the new AS2 servlets can be installed. For information on how to uninstall AS2 servlets, refer to the topic, *Uninstalling the MessageWay AS2 Interface* (on page 17).

The installation process for the AS2 servlets varies depending on the operating system where you install the components, UNIX/Linux or Windows.

To Install the AS2 Servlets on UNIX/Linux

Make sure you have already installed the Web container, Apache Tomcat. For more information, refer to the topic, *Prerequisites for the MessageWay AS2 Interface* (on page 9).

- 1 Log onto the perimeter server as user **root**.
- 2 Copy the MessageWay AS2 Interface zip file, **mwas2servlet-6.1.3.zip** to the **mwayinstall** directory on the perimeter server (create if does not exist).
- 3 Go to the **mwayinstall** directory, and unzip the MessageWay AS2 Interface zip file, **mwas2servlet-6.1.3.zip**. Type the following command:

```
unzip mwas2servlet-6.1.3.zip -d mwas2servlet-6.1.3
```

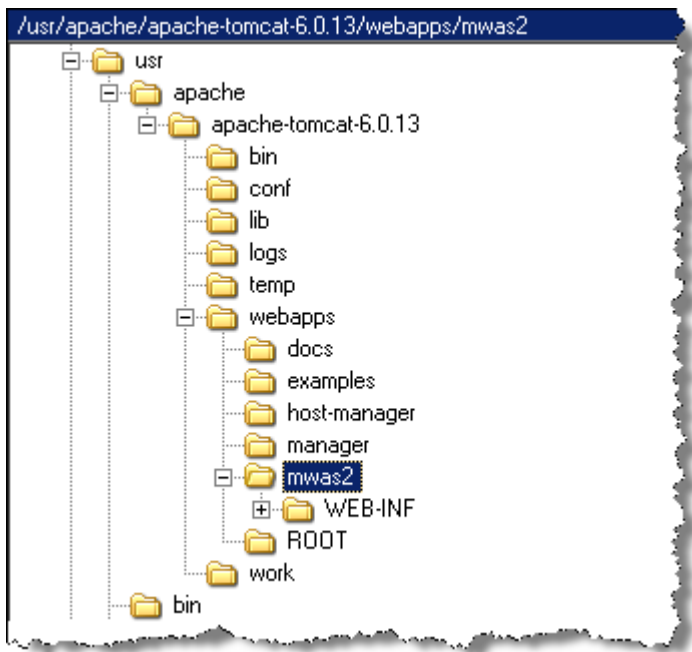
This creates a subdirectory of the name of the install file.

- 4 Go into the directory structure you just created, and copy the directory, **mwas2**, and all of its contents to the **/webapps** subdirectory of the Apache Tomcat install directory (**{tomcat-root}**). Type the following command:

```
cp -rp ./mwas2 {tomcat-root}/webapps/
```

- 5 Optionally, delete the file `NTEventLogAppender.dll` and `NTEventLogAppender.amd64.dll` from the `mwas2` directory, which is for the Windows installations only.

Your directory structure should look something like the following:



To Install the AS2 Servlets on Windows

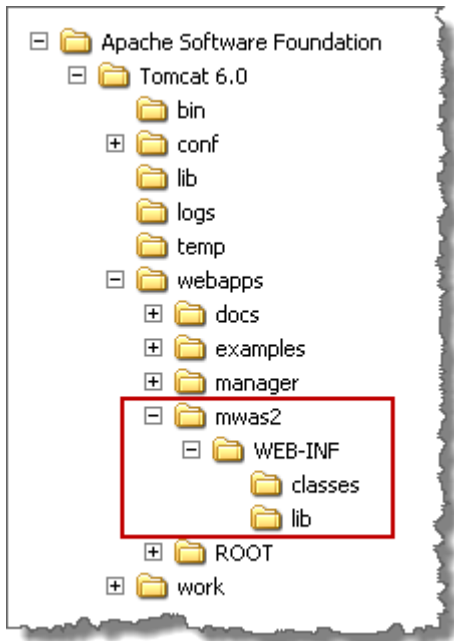
To install the AS2 inbound and outbound servlets, you unzip the installation package and copy a directory structure to the web container. The web container, Apache Tomcat, *must* be installed as a pre-requisite and *must* be installed as a service. Not all versions of Tomcat automatically install as a service. Refer to the installation instructions provided with Tomcat, since the instructions may vary between versions.

- 1 Log onto the perimeter server as the administrative user that will own the AS2 servlets.
- 2 Copy the MessageWay AS2 Interface zip file, **mwas2servlet-6.1.3.zip** to the **mwayinstall** directory on the perimeter server (create if does not exist).
- 3 Go to the **mwayinstall** directory and unzip the MessageWay AS2 Interface zip file, **mwas2servlet-6.1.3.zip**, using the folder path names.

This creates a subdirectory of the name of the install file.

- 4 Go into the directory structure you just created, and copy the directory, **mwas2**, and all of its contents to the **\webapps** subdirectory of the Apache Tomcat install directory (**{tomcat-root}**).

This creates a subdirectory within the webapps directory. Your structure should look something like the following:



5 For 32-bit systems:

- a) Move the file NTEventLogAppender.dll (32-bit) from the mwas2 directory to the \Windows\System32 directory.
- b) Optionally, delete the file NTEventLogAppender.dll from the mwas2 directory, which is for the 64-bit Windows installation.

- or -

For 64-bit systems:

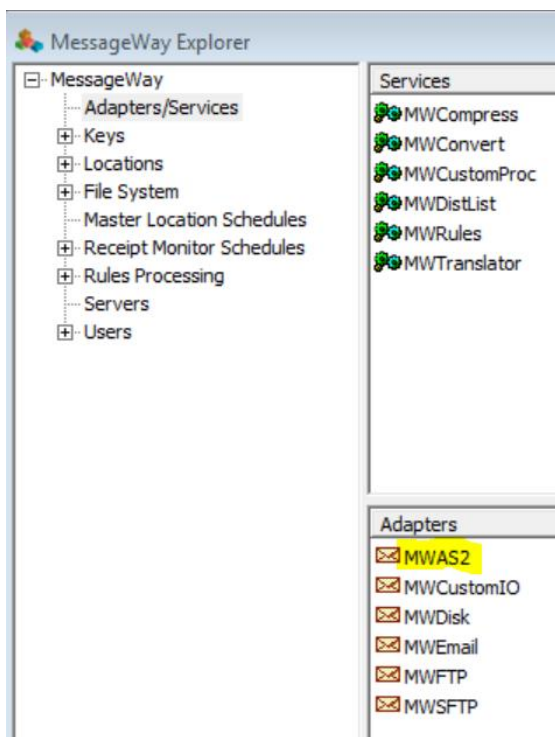
- a) Move the file NTEventLogAppender.amd64.dll (64-bit) to the \Windows\System32 directory.
- b) Move the file NTEventLogAppender.dll (32-bit) from the mwas2 directory to the \Windows\SysWOW64 directory.

Enable SSL (optional)

If you want to enable SSL for enhanced security on the front-end, follow the instructions in **enablingSSL.txt** included in the installation zip file.

Install the AS2 Adapter

During the MessageWay install, if you selected AS2 Adapter on the **Select optional components to install**: page, then AS2 Adapter has already been installed. To verify, go into the Manager and select Adapters/Services in the left-hand window pane:



and confirm that MWAS2 resides under the list of Adapters. If not, refer to topic *Installing or Upgrading Additional Adapters or Services* in the *MessageWay Installation Guide*.

Start and Stop the AS2 Interface

You start and stop the AS2 Interface differently, depending on the operating system where the server resides, UNIX/Linux or Windows.

To Start or Stop the AS2 Interface on UNIX/Linux

Before you start the AS2 servlets, you must configure the file that identifies the location of the servlet configuration file. Then you start the Web container, Apache Tomcat, which starts the AS2 servlets.

IMPORTANT: These instructions allow you to start the servlets. To complete the configurations, refer to topic, *Configuring the AS2 Interface* in the *MessageWay User's Guide and Reference* or in the Manager online help.

- 1 Edit the file, web.xml, within the {tomcat-root}/webapps/mwas2/WEB-INF directory, to specify the current location of the servlet configuration file for both the inbound and outbound parameters, and save your changes. Typically, you would only replace the value {tomcat-root} with your install location, but be sure to check the entire path.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE web-app
  PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
  "http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app>

  <display-name>MessageWay AS2 Interface</display-name>

  <servlet>
    <display-name>MessageWay AS2 Interface Inbound Servlet</display-name>
    <servlet-name>MWayAS2In</servlet-name>
    <servlet-class>MWayAS2In</servlet-class>
    <load-on-startup>1</load-on-startup>
    <init-param>
      <param-name>mwas2-conf</param-name>
      <param-value>{tomcat-root}/webapps/mwayas2/WEB-INF/mwas2.conf</param-value>
    </init-param>
    <init-param>
      <param-name>eventlevel</param-name>
      <param-value>INFO</param-value>
    </init-param>
  </servlet>

  <servlet>
    <display-name>MessageWay AS2 Interface Outbound Servlet</display-name>
    <servlet-name>MWayAS2Out</servlet-name>
    <servlet-class>MWayAS2Out</servlet-class>
    <load-on-startup>1</load-on-startup>
    <init-param>
      <param-name>mwas2-conf</param-name>
      <param-value>{tomcat-root}/webapps/mwayas2/WEB-INF/mwas2.conf</param-value>
    </init-param>
    <init-param>
      <param-name>eventlevel</param-name>
      <param-value>INFO</param-value>
    </init-param>
  </servlet>
</web-app>
```

- 2 To start the Apache Tomcat Web container, which in turn starts the servlets, from the /bin directory of Apache Tomcat, type:
./startup.sh.
- 3 To stop the Apache Tomcat Web container, which in turn stops the servlets, from the /bin directory of Apache Tomcat, type:
./shutdown.sh.

NOTE: For Red Hat 7.x, Apache Tomcat supports the `systemctl` utility, including automatically starting Apache Tomcat Web container when the perimeter server is rebooted. To manually start the Apache Tomcat Web container, type `systemctl start tomcat` logged on as **root**. To manually stop the Apache Tomcat Web container, type `systemctl stop tomcat` logged on as **root**. To check for status, type `systemctl status tomcat`.

- 4 Test to make sure the servlets are running and you can access them:
 - a) To test access to the inbound servlet, from your Web browser, type:
`http://localhost:8080/mwas2/in`
 - b) To test access to the outbound servlet, from your Web browser, type:
`http://localhost:8080/mwas2/out`
-

NOTE: If you do not receive messages, "You have reached the MessageWay AS2 ... Interface", test to see if you can access the Web container. Type, `http://localhost:8080`. If the Web browser is on a different machine, replace **localhost** with the IP Address of the machine hosting the servlets.

To Start or Stop the AS2 Interface on Windows

Before you start the AS2 servlets, you must configure the file that identifies the location of the servlet configuration file. Then you start the Web container, Apache Tomcat, which starts the AS2 servlets.

IMPORTANT: These instructions allow you to start the servlets. To complete the configurations, refer to topic *Configuring the AS2 Interface* in the *MessageWay User's Guide and Reference* or in the manager online help.

- 1 Edit the file, `web.xml`, within the `{tomcat-root}\webapps\mwas2\WEB-INF` directory, to specify the current location of the servlet configuration file for both the inbound and outbound parameters, and save your changes. Typically, you would only replace the value `{tomcat-root}` with your install location, but be sure to check the entire path.


```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE web-app
  PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
  "http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app>

  <display-name>MessageWay AS2 Interface</display-name>

  <servlet>
    <display-name>MessageWay AS2 Interface Inbound Servlet</display-name>
    <servlet-name>MWayAS2In</servlet-name>
    <servlet-class>MWayAS2In</servlet-class>
    <load-on-startup>1</load-on-startup>
    <init-param>
      <param-name>mwas2-conf</param-name>
      <param-value>{tomcat-root}/webapps/mwayas2/WEB-INF/mwas2.conf</param-value>
    </init-param>
    <init-param>
      <param-name>eventlevel</param-name>
      <param-value>INFO</param-value>
    </init-param>
  </servlet>

  <servlet>
    <display-name>MessageWay AS2 Interface Outbound Servlet</display-name>
    <servlet-name>MWayAS2Out</servlet-name>
    <servlet-class>MWayAS2Out</servlet-class>
    <load-on-startup>1</load-on-startup>
    <init-param>
      <param-name>mwas2-conf</param-name>
      <param-value>{tomcat-root}/webapps/mwayas2/WEB-INF/mwas2.conf</param-value>
    </init-param>
    <init-param>
      <param-name>eventlevel</param-name>
      <param-value>INFO</param-value>
    </init-param>
  </servlet>
</web-app>
```

- 2 Start the Apache Tomcat Web container, which also starts the servlets. The task varies depending on how you installed the software:
 - From the Microsoft Management Console (MMC) Services window, start it as a Windows service.
- or -
 - Right-click the icon in the system tray, and click **Start service**.
- 3 Stop the Apache Tomcat Web container, which also stops the servlets. The task varies depending on how you installed the software:
 - From the Microsoft Management Console (MMC) Services window, stop it as a Windows service.
- or -
 - Right-click the icon in the system tray, and click **Stop service**.
- 4 Test to make sure the servlets are running and you can access them:
 - a) To test access to the inbound servlet, from your Web browser, type:
http://localhost:8080/mwas2/in
 - b) To test access to the outbound servlet, from your Web browser, type:
http://localhost:8080/mwas2/out

NOTE: If you do not receive messages, "You have reached the MessageWay AS2 ... Interface", test to see if you can access the Web container. Type, <http://localhost:8080>. If the Web browser is on a different machine, replace **localhost** with the IP Address of the machine hosting the servlets.

Post-Install Best Practices

The following sections outline changes to your AS2 environment that will improve performance.

Managing Memory and Garbage Can Collection on UNIX/Linux

To improve overall performance and thru-put for AS2 Servlets, we recommend that you add the following command to the {tomcat-root}/bin/catalina.sh file:

```
JAVA_OPTS="$JAVA_OPTS -server -Xms4096m -Xmx4096m -XX:+UseConcMarkSweepGC  
-XX:MaxPermSize=512m -XX:ParallelGCThreads=2 -XX:+UseParNewGC -XX:+DisableExplicitGC"
```

If further performance tuning or trouble shooting is required, the above command can be modified to allow tracing of JAVA_OPTS settings:

```
JAVA_OPTS="$JAVA_OPTS -server -Xms4096m -Xmx4096m -XX:+UseConcMarkSweepGC  
-XX:MaxPermSize=512m -XX:ParallelGCThreads=2 -XX:+UseParNewGC -XX:+DisableExplicitGC  
-XX:+PrintGCTimeStamps -XX:+PrintGCDetails -Xloggc:/var/log/tomcat5/printgc.log"
```

NOTE: MaxPermSize has been replaced with MaxMetaspaceSize in Java version 8 or newer.

NOTE: catalina.sh has been replaced with tomcat.conf in newer versions of Apache Tomcat.

Managing Memory and Garbage Can Collection on Windows

To improve overall performance and thru-put for AS2 Servlets, we recommend that you add the following command to the {tomcat-root}\bin\catalina.bat file:

```
JAVA_OPTS="$JAVA_OPTS -server -Xms4096m -Xmx4096m -XX:+UseConcMarkSweepGC  
-XX:MaxPermSize=512m -XX:ParallelGCThreads=2 -XX:+UseParNewGC -XX:+DisableExplicitGC"
```

If further performance tuning or trouble shooting is required, the above command can be modified to allow tracing of JAVA_OPTS settings:

```
JAVA_OPTS="$JAVA_OPTS -server -Xms4096m -Xmx4096m -XX:+UseConcMarkSweepGC  
-XX:MaxPermSize=512m -XX:ParallelGCThreads=2 -XX:+UseParNewGC -XX:+DisableExplicitGC  
-XX:+PrintGCTimeStamps -XX:+PrintGCDetails -Xloggc:/var/log/tomcat5/printgc.log"
```

NOTE: MaxPermSize has been replaced with MaxMetaspaceSize in Java version 8 or newer.

Upgrading the AS2 Interface

There is no automatic upgrade process for the MessageWay AS2 Interface. Periodically Progress will send out Maintenance Releases for the MessageWay AS2 Interface, and these MR's will contain detailed instructions on how to install the MR. Refer to the release notes for each MR for further upgrade instructions.

Uninstalling the MessageWay AS2 Interface

There is no automatic uninstall process for the MessageWay AS2 Interface. You must manually delete the necessary files and folders.

How you uninstall the MessageWay AS2 Interface depends on the whether you are running the AS2 Interface on Windows or UNIX/Linux.

Before you uninstall the AS2 Interface, if you want to save information for a future reinstallation, save these files:

Server Location	Default Location
Windows	<ul style="list-style-type: none"> ▪ {tomcat-root}\webapps\mwas2\WEB-INF\mwas2.conf ▪ {tomcat-root}\webapps\mwas2\WEB-INF\web.xml
UNIX/Linux	<ul style="list-style-type: none"> ▪ {tomcat-root}/webapps/mwas2/WEB-INF/mwas2.conf ▪ {tomcat-root}/webapps/mwas2/WEB-INF/web.xml

NOTE: depending on where you installed your java key store (JKS), you may want to back it up as well if it is installed within the folders being deleted.

Uninstalling the MessageWay AS2 Interface from Windows

When you uninstall MessageWay AS2 Interface from a Windows system, you must manually delete the necessary files and folders.

To perform an uninstall, proceed as follows:

- 1 Log on as user who has authority to delete AS2 Interface files and folders.
- 2 Stop the AS2 Interface. The task varies depending on how you installed the Apache Tomcat Web container:
 - From the Microsoft Management Console (MMC) Services window, stop the Windows service.
 - or -
 - Right-click the icon in the system tray, and click **Stop service**.
- 3 Optionally back up any config and license files as noted above for a future reinstall.

- 4 Delete all files and folders in the `{tomcat-root}\webapps\mwas2` folder, including the `mwas2` folder itself.

Uninstalling the MessageWay AS2 Interface from UNIX/Linux

When you uninstall MessageWay AS2 Interface from a UNIX/Linux system, you must manually delete the necessary files and folders.

To perform an uninstall, proceed as follows:

- 1 Log onto the perimeter server as user `root`.
- 2 Stop the AS2 Interface. From the `/bin` directory of Apache Tomcat, type:
`./shutdown.sh`.

NOTE: For Red Hat 7.x, Apache Tomcat supports the `systemctl` utility. To stop the Apache Tomcat Web container, type `systemctl stop tomcat` logged on as `root`. To check for status, type `systemctl status tomcat`.

- 3 Optionally back up any config and license files as noted above for a future reinstall.
- 4 Delete all files and folders in the `{tomcat-root}/webapps/mwas2` folder, including the `mwas2` folder itself.