IBM Maximo Asset Management
Version 7 Release 6

# *Installation Guide*

*(Oracle WebLogic Server)*

IBM

# Contents

# Chapter 1. Preparing for installation

These topics provide information about product media, preinstallation considerations, overview of the installation procedure, and instructions on using the Maximo® Asset Management launchpad.

Use the planning information to familiarize yourself with the overall process of a Maximo Asset Management deployment before you use this information to conduct the installation.

## Software installation images

You access the IBM® Maximo Asset Management product software from IBM Passport Advantage.

The installation images that you download from Passport Advantage can comprise multiple downloadable files. Download all files in the package to a single directory and extract the files for execution.

For instructions and a full list of installation images, see the IBM Maximo Asset Management 7.6 Download Document (http://www.ibm.com/support/docview.wss?uid=swg24038431).

## Before you begin

You must prepare your environment before you install Maximo Asset Management. In some cases, to perform steps for preparation, you must be logged in as a user with administrator privileges on Windows or as root on UNIX. Some processes, such as anti-virus programs, can negatively affect Maximo Asset Management installation on the system. You must temporarily shut down any noncritical processes before running the Maximo Asset Management installation program.

Make a copy of the image of the system, database, and application server on which you are planning to install the product.

Ensure that you have adequate disk space for the future on the systems being used for the Maximo Asset Management deployment. Filling up the disk space on a Maximo Asset Management deployment system can cause problems with Maximo Asset Management operations.

Fully qualified host names provided to the installation programs must resolve between systems involved in the product deployment. Ensure all IP addresses configured for systems targeted for the product deployment are reachable using the ping command from the administrative workstation.

### Programmatically verifying prerequisites from the command-line.

You can use the prerequisite verification utility from the command-line to verify that installation program prerequisites are present on a system. Use this utility before you start the product installation programs.

## About this task

When started from the command line, the prerequisite verification utility accepts various parameters. The prerequisite verification utility must be run on the system that hosts the prerequisite to check. You cannot use this utility to check prerequisites on a remote system.

*Table 1. System verification parameters*

| Parameters | Details |
|---|---|
| `-component` | Specifies the type of component to verify. At least one component must be used with the prerequisite verification utility.<br><br>**dbserver**<br>    Use the **dbserver** parameter to have the prerequisite verification utility check for DB2® prerequisites.<br><br>**pip**<br>    Use the **pip** parameter to have the prerequisite verification utility check for prerequisites that are required by Maximo Asset Management.  Run the prerequisite verification utility with the **pip** parameter on the administrative workstation.<br><br>**asset_mgt_upg**<br>    Use the **asset_mgt_upg** parameter to have the prerequisite verification utility check for prerequisites that are required to connect remotely during configuration. .<br><br>**rxa**<br>    Use the **rxa** parameter to have the prerequisite verification utility check for prerequisites that are required by Maximo Asset Management.<br><br>**performance**<br>    Use the **performance** parameter to have the prerequisite verification utility check for prerequisites performance settings for Maximo Asset Management.<br><br>Syntax example:<br><br>`tpae_req_check.bat -component pip`<br><br>If you do not use the **-component** parameter, you are prompted to specify components when you use the utility.<br><br>Multiple components can be specified as comma-separated values. |

*Table 1. System verification parameters  (continued)*

| Parameters | Details |
|---|---|
| **-input** | The prerequisite verification utility can accept a property file as input. The utility verifies property values that are found in the file.<br><br>For example, the default DB2 server port parameter is **DB2_SERVER_PORT=50000**. The prerequisite verification utility verifies that this port is available on the DB2 server host.<br><br>The default input property file is tpae.properties, and is found on the middleware installation media in the SystemRequirements directory. Several sample property files are found in the SystemRequirements\ sample_property_files directory. These sample property files contain custom values that are defined for particular operating systems. You can copy these property files to the system and modify them, or create your own, before you run the prerequisite verification utility.<br><br>Syntax example:<br><br>`tpae_req_check.bat -component dbserver -input path to the property file`<br><br>If you do not use the **input** parameter when you run the prerequisite verification utility, the utility is run in interactive mode. You are prompted for individual values to be verified. |
| **-mode** | The prerequisite verification utility can be run in silent or interactive mode.<br><br>**interactive**<br>    By default, the prerequisite verification utility is run in interactive mode. If you do not specify the **-mode** parameter, the utility defaults to running in interactive mode.<br><br>**silent**<br>    If you use the **silent** qualifier, you can also use the **input** parameter and supply a property file. If an input file not provided, default property values are used. The output must also be directed to a file to view the results.<br><br>Syntax example:<br><br>`tpae_req_check.bat`<br>`-component dbserver`<br>`-mode silent`<br>`-input path to the property file > prereqresults.log` |
| **-trace** | Parameter that is used to specify trace output statements while the utility is running.<br><br>**None**<br>    Selecting this qualifier results in no trace information that is generated while the utility is running.<br><br>**Verbose**<br>    Selecting this qualifier results in detailed trace information that is generated while the utility is running.<br><br>**Normal**<br>    Selecting this qualifier results in default trace information that is generated while the utility is running.<br><br>Syntax example:<br><br>`tpae_req_check.bat`<br>`-component dbserver`<br>`-trace None` |

## Procedure

1. Log on to the system you are checking for prerequisites with a user ID that has permission to run scripts on the system. Ensure that the middleware installation media is mounted or otherwise available to the system. This example procedure uses values that are found in the default `tpae.properties` file to verify that a system is suitable to host DB2 for a Maximo Asset Management deployment.

2. Open a command-line window and change directory to the `SystemRequirements` directory of the middleware installation media.

3. Run the prerequisite verification utility. Specify the component to check and the property file to use.

   ```
   tpae_req_check.bat
   -component dbserver
   -input d:\SystemRequirements\tpae.properties
   ```

   In this example, the `tpae.properties` file is in the same directory as the `tpae_req_check.bat` script.

## Results

After the prerequisite verification utility successfully completes the verification task, results are printed to the screen.

```
CTGIN8117I : The Tivoli Pre-requisite Scanner has been launched.
CTGIN8118I : The Tivoli Pre-requisite Scanner exited with the return code
IBM Prerequisite Scanner
    Version  : 1.0.34
    Build    : 20101109
    OS Name  : Microsoft Windows Server 2003, Enterprise Edition Service Pack 2
    User Name: Administrator

Machine Info
    Machine name : MYMACHINE
    Serial Number: KKKKKK0
    OS Serial    : 66666-666-6666666-66666

PAE [not defined] [version 07500000]:
Property                                 Result  Found                                    Exp...
========                                 ======  =====                                    ===...
os.totalPhysicalMemory                   PASS    2.00GB                                   1.90GB
network.hasFQDN                          FAIL    False                                    True
Disk#1 (C:\ibm\tivoli\mwi\workspace)     PASS    27.99GB                                  300MB
Disk#2 (C:\Temp\1)                       PASS    27.99GB                                  1000MB
Disk#3 (C:\Temp\1)                       PASS    27.99GB                                  250MB
network.availablePorts.db2               PASS    135,445,1025,2967,3389,5800,5900,139     50000
network.availablePorts.ctginst           PASS    135,445,1025,2967,3389,5800,5900,139     50005
Disk#4 (C:\Program Files\IBM\SQLLIB)     PASS    27.99GB                                  1.40GB

ALL COMPONENTS :
Property            Result    Found    Exp...
========            ======    =====    ===...
C:                  PASS      27.99GB  2.91GB

Prereq Scanner Overall Result: FAIL
prereq_checker.bat 1
```

These values can also be redirected to a file when you start the command from the command line.

If any of the verification steps report a failure, resolve the issue and rerun the verification utility before you install Maximo Asset Management components.

The Table 2 table contains a list of the properties checked. Use this information to analyze prerequisite verification utility results.

*Table 2. Prerequisite verification utility properties*

| Property | Description |
|---|---|
| browser.Version | Checks browser version. |
| CpuArchitecture | Verifies that the machine architecture is supported. |
| network.availablePorts.* | Verifies that required ports are available.<br><br>The following example checks to ensure a port that is required by DB2 is available.<br><br>`network.availablePorts.db2inst=50000` |
| network.dns | Verifies that there is a DNS entry for the system on the DNS server. |
| network.fqdn | Verifies that the system host name is fully qualified. |
| os.architecture | Verifies that the operating system architecture is supported. |
| os.dir.tmp/home | Verifies required permissions for the system directories |
| os.FreePagingSpace | Verifies that adequate free paging space is available on the system. |
| os.iodevicestatus | Verifies the I/O device status of the system. |
| os.kernelversion | Checks kernel version. |
| os.lib.* | Verifies that a prerequisite library is available on the system. |
| os.mountcheck | Checks for the existence of `nosuid` on the file system. |
| os.MozillaVersion | Checks browser version. |
| os.package.atk | Verifies a prerequisite library that is required for GTK on AIX® is available on the system. |
| os.package.cairo | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.expat | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.fontconfig | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.freetype2 | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.gettext | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.glib2 | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.gtk2 | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.libjpeg | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.libpng | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.libtiff | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.pango | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.pixman | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |

*Table 2. Prerequisite verification utility properties  (continued)*

| Property | Description |
|---|---|
| os.package.rpm | Verifies that prerequisite packages are available on the system. |
| os.package.xcursor | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.xft | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.xrender | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.package.zlib | Verifies a prerequisite library that is required for GTK on AIX is available on the system. |
| os.RAMSize | Reports system RAM available. |
| os.SeaMonkeyVersion | Checks browser version. |
| os.SELinux | Determines whether `selinux` is enabled. |
| os.ServicePack | Verifies that required service packs are installed on the system. |
| os.shell.default | Determines the default shell for users. |
| os.space.* | Determines disk space availability for various file systems.<br><br>The following example shows the check that is made for the middleware installation program workspace.<br><br>`os.space.var=[dir:root=/var,unit:MB]512` |
| os.totalPhysicalMemory | Reports physical RAM available to the operating system. |
| os.ulimit | Identifies user limits that are configured on the system. |
| OS.Version | Reports the operating system type and version. |

# Checking port availability

Before you use the product installation programs, you must ensure that certain ports in your environment are available for use with the middleware that you are installing. You can either use the prerequisite verification utility to check for port availability or you can check manually.

## About this task

Default port values are assigned for the middleware that you deploy.

For example, port 50000 or an alternative port must be made available for DB2.

Port 7001 or an alternative port must be made available for Oracle WebLogic Server.

For a full list of port requirements, see the System requirements information on the Maximo Asset Management wiki.

**Procedure**

1. Check the available ports on the host system:

| Option | Description |
|---|---|
| Automatic | Run the prerequisite verification utility. The `network.availablePorts.*` parameter indicates the availability of the required ports. |
| Manual | Run the following command to see a list of active connections: `netstat -ao` |

2. If you find that a port is already assigned, specify the value of an open port when you are prompted to by the installation program.

# AIX tar command requirements

Both the native UNIX **tar** command and the GNU version of the **tar** command are required by the installation and configuration programs. Because the native utility does not support long file names, ensure that GNU **tar** version 1.14 or higher is installed. GNU **tar** version 1.14 ensures that installation files can be extracted.

Verify that the system path variable contains both native UNIX **tar** and GNU **tar** paths. The GNU **tar** path must be defined before the native UNIX tar path. For example, the native **tar** utility is installed in `/usr/bin` and the GNU tar utility is installed in `/opt/freeware/bin/tar`.

If you have set a symbolic link to overwrite the native UNIX **tar** command with the GNU **tar** command an error occurs.

http://www.ibm.com/systems/p/os/aix/linux/toolbox/download.html

# AIX font requirements

AIX requires specific fonts to produce reports.

## About this task

When you produce reports from AIX systems, TrueType fonts must be available on the system.

## Procedure

1. Install TrueType fonts on the AIX system.
2. Ensure the fonts-path environment variable refers to the location of the fonts.

# Enabling asynchronous I/O on AIX

IBM Tivoli® Directory Server requires asynchronous I/O be enabled on AIX systems.

## About this task

Asynchronous I/O must be enabled on AIX systems if the system hosts IBM Tivoli Directory Server.

Without asynchronous I/O, DB2 database instances cannot be started successfully.

Without asynchronous I/O, Oracle database instances cannot be started successfully.

To enable asynchronous I/O, follow these steps:

### Procedure

1. Log in to the system as root.
2. Open a terminal and run the following command:

   `smit chgaio`
3. From the System Management Interface tool (SMIT) dialog box, change STATE to be configured at system restart from **defined** to **available**.
4. Click **OK**.
5. Exit SMIT.
6. Run the following command from the command line:

   `smit aio`
7. In the System Management Interface tool (SMIT) dialog box, select **Configure Defined Asynchronous I/O**, and then click **Enter**.
8. Restart the system to enable the changes.

## Checking for required libraries on Linux

The Maximo Asset Management deployment requires certain Linux system libraries.

### Procedure

1. Locate the `libstdc++.so.5`, `libstdc++.so.6`, and `libaio.so.1` libraries in the `/usr/lib64/` directory. If these libraries are not installed, the Maximo Asset Management installation program produces an error when run in graphical mode. If you cannot locate this library on your system, locate the Red Hat Package Manager package for your system that contains these libraries and install the packages.
2. Ensure that you have the `libstdc++33-32bit-3.3.3-11.9.x86_64.rpm` package that is installed before you run the product installation program. SUSE Linux Enterprise Server 10 64-bit systems that host manually configured middleware must host this package.

## Configuring the JRE in Linux

In some cases, the Maximo Asset Management installation or configuration program fails on Red Hat Enterprise Linux systems, or other systems with Security-Enhanced Linux (SELinux) enabled.

### About this task

In one scenario, the Maximo Asset Management installation or configuration program fails with an error that states that the Java™ Runtime Environment (JRE) cannot be found. In another scenario, the Maximo Asset Management installation or configuration program fails stating that it cannot find the VM.

### Procedure

1. Temporarily disable SELinux by using the **setenforce 0** command.
2. Run the Maximo Asset Management installation or configuration program.
3. Re-enable SELinux by using the **setenforce 1** command.

4. Manually issue the **chcon** command.

   `chcon -R -t textrel_shlib_t` *install_home*`/jvm/jre`

### Results

The Maximo Asset Management installation or configuration program is now able to locate the JRE. Alternatively, you can edit the `/etc/selinux/config` file and set **SELINUX** to either `permissive` or `disabled` for a more permanent fix. This solution, however, affects the level of security for the entire system.

# Setting the ulimit

Set the ulimit in Linux. The ulimit setting is used to define user system and process resource limits.

### About this task

For Linux systems, you must set the ulimit for the system before you use the Maximo Asset Management installation program.

If you set the ulimit in `.profile` for root, the ulimit setting applies to all processes.

### Procedure

1. From a command line, type **ulimit -f unlimited**
2. From a command line, type **ulimit -n 8192**

# Setting the swap size

Maximo Asset Management can be a resource-intensive application. Configure and tune your system for maximum performance. Set the size of the swap space that is used in Linux systems.

### About this task

Typically, the swap size for Linux is set to twice the amount of physical RAM in the server. See the product documentation for your Linux distribution for more information.

More swap space can be made available to the system.

### Procedure

- Increase the size of the existing swap partition.
- Create a new, extra swap partition.
- Create a swap file.

# Setting shared memory

Set a minimum shared memory value in Linux.

### About this task

For Linux systems, you must set a minimum shared memory value for the system before you use the Maximo Asset Management installation program.

### Procedure

1. From a command line, type `sysctl kernel.shmmax` and determine whether the value is less than 268435456 bytes (256 Mb).
2. If you must increase the value, from a command line, type `sysctl -w kernel.shmmax=268435456`.
3. Update the value in `/etc/sysctl.conf`.

## Remote configuration enablement

The Maximo Asset Management configuration program can automatically configure middleware. You must enable a remote access protocol for each system on which you intend to configure the middleware.

Use SSH for accessing remote Linux and UNIX systems. Use Windows SMB accessing remote Windows systems. Windows SMB is a Windows protocol. The IBM JRE on the Maximo Asset Management administrative workstation includes SSH.

Before you start the Maximo Asset Management configuration program, ensure that you can log on to any remote servers with the protocols that you intend to use. Use the credentials that you plan to supply to the Maximo Asset Management configuration program.

For remote Windows systems, ensure that the following requirements are met before you configure the software.

- The user name that you provide to the Maximo Asset Management configuration program must exist as a local account on the remote system. This user must be a member of the Windows Administrators group.
- The following Windows services must be started on the remote system before you begin a remote configuration:
  - `winmgmt` (Windows Management Instrumentation)
  - `RemoteRegistry` (Remote Registry)
  - `lanmanserver` (Service)
- The SMB protocol must be enabled and configured to send NetBIOS over TCP/IP, by choosing to use port 139. Alternatively, you can configure SMB to use TCP/IP as the transport protocol, without NetBIOS, by configuring it to use port 445.
- Ensure that any ports that you use for remote protocols are not blocked by firewall software or security policies, including ports 137 and 139. Port 139 is used if SMB is configured to run on NetBIOS over TCP/IP. Port 445 is used if SMB is run directly on TCP/IP, without NetBIOS.
- To disable simple file sharing, start Windows Explorer. Click **Tools** > **Folder Options**, and clear the **Use Simple File Sharing** check box.
- The Windows administrative share (C$) and the interprocess communications (IPC$) folder must be shared.
- For Microsoft Windows Server 2008 systems that support password-protected sharing, disable password-protection. Shares must be shared for the Guest or Everyone accounts.
- For Windows systems that have User Account Control (UAC) enabled, it must be disabled before software can be remotely installed and configured.
- If Cygwin is installed on the remote Windows system the SSH daemon (sshd) must be uninstalled or disabled.

For remote Linux or UNIX systems, ensure that the following requirements are met before you configure the software.

- For AIX systems, set the following SSH parameters in the `/etc/ssh/sshd_config` file.
    - ClientAliveInterval 900
    - ClientAliveCountMax 10

    Stop the SSH daemon with the `stopsrc -s sshd` command and then restart it using the `startsrc -s sshd` command.
- For AIX systems, set the *TMOUT* and *TIMEOUT* variables in the user profile script to 0. This setting prevents the user from idling out and being logged off the remote system during the configuration.
- The user name that you provide to the Maximo Asset Management configuration program must exist as a privileged account (for example, root) on the remote systems.
- Ensure that a current version of OpenSSH is installed and running. Do not use OpenSSH 4.7.0.5302.
- For Oracle Solaris systems, the remote access protocols require the use of internal shell scripts that must be run within the Korn (ksh) shell. The methods need ksh, even if the user ID that you use to log on to the remote system is configured to use a different shell. Oracle Solaris systems must have the ksh environment installed and properly configured.
- If you plan to remotely configure software on remote Linux or UNIX computers, ensure that SSH is installed.

Remote configuration does not support accessing network drives on the local or remote system.

## Microsoft SQL Server preparation

Microsoft SQL Server must be configured using specific options before you can use it with Maximo Asset Management.

Ensure you have the following options set for the Microsoft SQL Server database you intend to use with Maximo Asset Management. These settings must be configured before you use the Maximo Asset Management configuration program to configure Maximo Asset Management.

- Use the SQL Server Configuration Manager to disable TCP/IP dynamic ports usage on the server.
- Verify that you enabled the Full-text Search setting during the installation of Microsoft SQL Server.
- Set Microsoft SQL Server Database Collation settings to the following options:
    - Dictionary order
    - Case-insensitive
    - For use with 1252 Character set

## System password policy settings

Familiarize yourself with the password policies of systems you are using as part of a Maximo Asset Management deployment.

Your organization might have password policy regulations that are enforced on systems on your network. Before deploying Maximo Asset Management, be sure that you are familiar with the password policies of systems that are used in the deployment.

For example, Microsoft Windows Server 2008 systems have a stricter set of password requirements than previous versions configured by default. If you are not familiar with these stronger password requirements, you might experience an error during the installation of Maximo Asset Management. This error occurs when you create users on a Microsoft Windows Server 2008 system.

Password values that you provide during the Maximo Asset Management installation and configuration must be compliant with the password policies set for the target system. Passwords are not validated against the password policies of target systems at the time you enter them on the installation and configuration program panels. If your passwords do not conform to the password policies, errors occur when the installation or configuration program attempts to create these passwords.

# Starting the launchpad

The IBM Maximo Asset Management launchpad serves as a centralized interface for starting the product installation and configuration programs. The launchpad also contains links to product information.

## Before you begin

Before you start the launchpad, ensure that you meet the middleware prerequisites. Ensure that you have the correct service pack levels for your environments.

The launchpad program uses the system default browser to run. If the default browser on AIX is Firefox, it is likely that the launchpad program does not run properly due to the ksh shell interface. If you must use the launchpad with the Firefox browser, follow these steps to modify it.

1. Download and extract the launchpad images which is described in the download document.
2. Modify th e/user_dir/launchpad/Firefox.sh file and remove the following lines

   ```
   typeset +r LOGNAME 2>/dev/null
   LOGNAME=lp_user_$$; export LOGNAME
   ```
3. Run the launchpad from /user_dir.

## About this task

The launchpad generates messages that are captured in a hidden log frame while the launchpad is running. To show the log frame on the launchpad panels, hold the Ctrl key and simultaneously click the banner frame of the launchpad. Messages that are generated while the launchpad is running are not automatically saved on the hard disk drive. You can save the messages from a session by clicking **Save** at the bottom of the log frame.

Always use fully qualified domain names when you enter values for the installation and configuration programs.

**Procedure**

1. Log on to an account with system administration privileges on the system where you want to install Maximo Asset Management.

2. Start the launchpad from the root directory of the installation image.

| Option | Description |
|---|---|
| **Windows** | Start the launchpad by using the `launchpad64.exe` program. |
| **AIX and Linux** | Start the launchpad from the root directory by using the `launchpad.sh` program, for example, `./launchpad.sh` |

**Related information**:

Software Product Compatibility Reports

# Installation program overview

You use the Maximo Asset Management installation program to install Maximo Asset Management, IBM WebSphere® Application Server, and IBM DB2.

Select packages to install from the Figure 1 user interface.
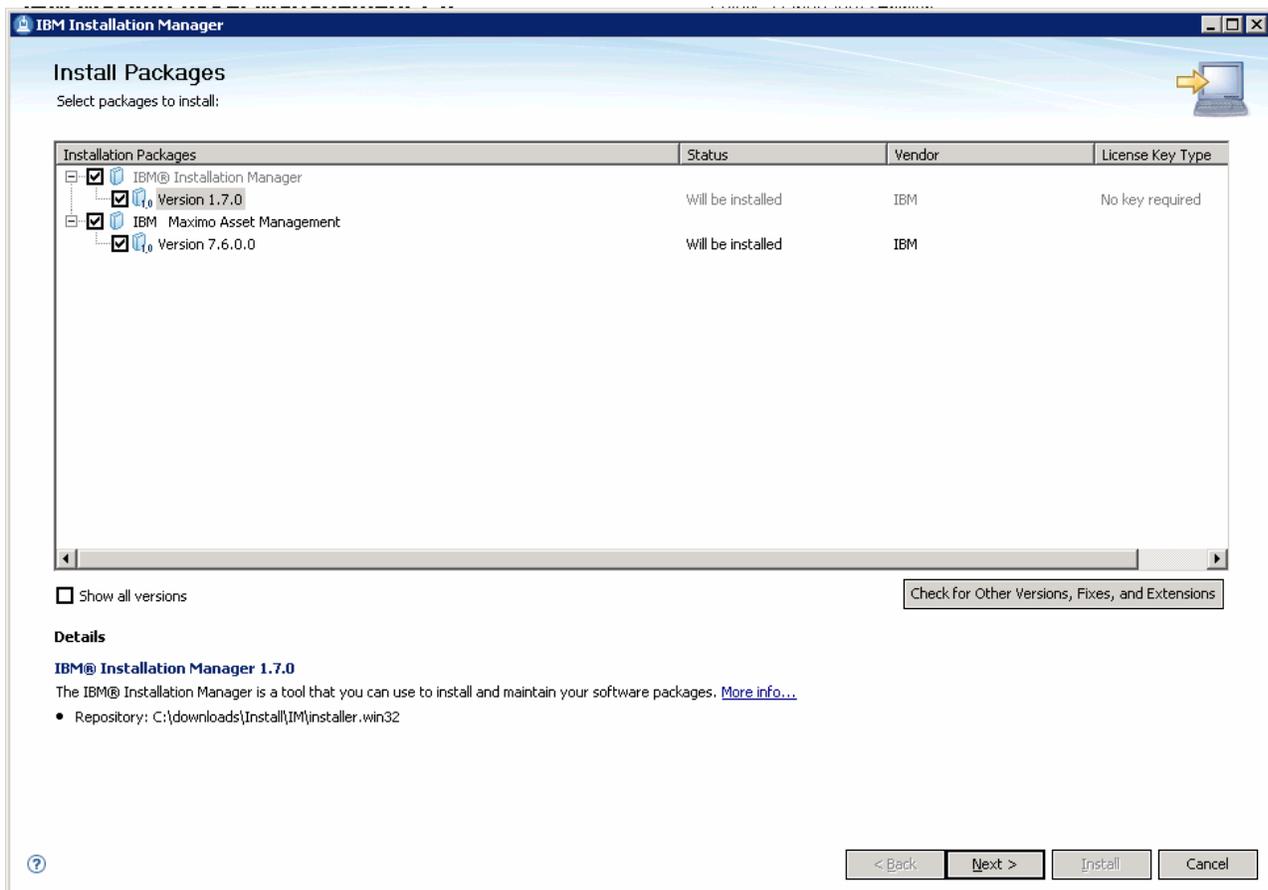


Figure 1. Maximo Asset Management installation program

You must run the installation program on the target system to install the component. For example, to install IBM WebSphere Application Server, you start the installation program locally on the system that is designated as the IBM WebSphere Application Server server.

Use the **Check for Other Versions, Fixes, and Extensions** button to search for fixes, updates, and extensions for the packages listed. The installation program downloads the latest version of the packages listed from IBM Fix Central.

# Configuration program overview

You use the Maximo Asset Management configuration program to configure Maximo Asset Management and associated middleware.

The Maximo Asset Management configuration program is used to perform the following configuration tasks.
- Prepare IBM WebSphere Application Server for configuration of your product
- Configure a new Maximo Asset Management deployment
- Update database access credentials
- Change the Maximo Asset Management security model
- Change the database server host
- Update IBM WebSphere Application Server access credentials
- Change IBM WebSphere Application Server host
- Update database and build and deploy application EAR files
- Install product help information in another language
- Remove IBM WebSphere Application Server configuration
- Remove Maximo Asset Management configuration

The Maximo Asset Management configuration program records configuration choices that you make about your Maximo Asset Management deployment and then deploys Maximo Asset Management based on the information that you entered.
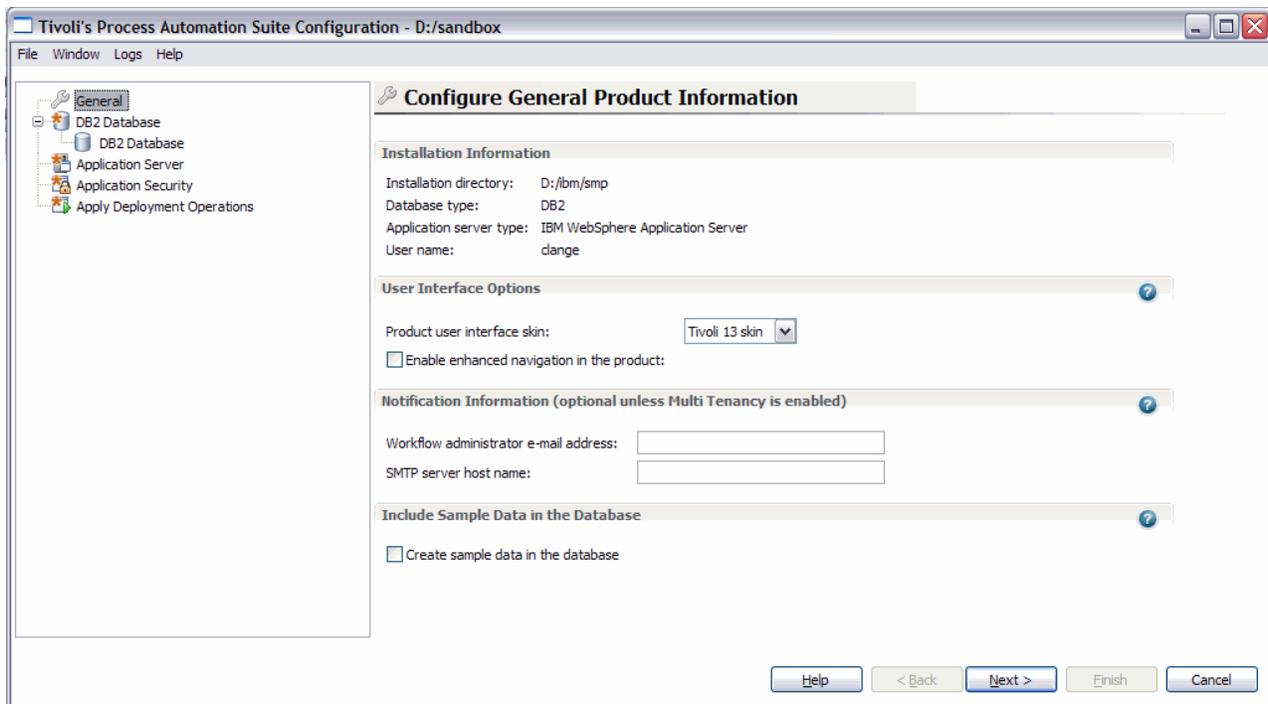
*Figure 2. Maximo Asset Management configuration program*

Information that you enter for the Maximo Asset Management configuration program is stored in the `maximo.properties` and `install.properties` files. Some of this information is also written to the Maximo database. These values are displayed in the panel fields of the Maximo Asset Management configuration program during an upgrade or fix pack application.

The Maximo Asset Management configuration program validates the values that you enter in configuration item fields. Some configuration item fields are validated as a pair, for example, user name and password fields.

In most cases, you can run the configuration program locally or remotely. You are required to run the configuration program locally on the IBM WebSphere Application Server server when you are preparing it for Maximo Asset Management configuration.

The Maximo Asset Management configuration program cannot automate the configuration of Oracle WebLogic Server.

If you choose not to have the Maximo Asset Management configuration program automatically configure middleware, you must configure that piece of middleware manually before the configuration of Maximo Asset Management.

**Note:** If you are using a directory server to secure Maximo Asset Management, be aware of the product-specific syntax rules for using special characters in LDAP strings that you enter. In most cases, special characters must be preceded by an escape character to make them readable by the directory server.

Many directory server products consider a blank space to be a special character in an LDAP string. If you enter an LDAP string with an unescaped blank character at the end of a field value, you encounter Maximo Asset Management errors.

For more information about special character usage in LDAP strings, see the product documentation for your directory server.

# Chapter 2. Automatically configuring middleware during deployment

Use the Maximo Asset Management installation and configuration programs to install and automatically configure a Maximo Asset Management deployment within your enterprise.

## About this task

This information provides a high-level overview or road map of tasks you need to complete in order to deploy Maximo Asset Management with automatic middleware configuration.

In this scenario, you use the Maximo Asset Management installation and configuration programs to install and automatically configure new instances of the following components:

- DB2
- Maximo Asset Management

You use the Maximo Asset Management installation program to install Maximo Asset Management and the middleware you want to use in your deployment. You then use the Maximo Asset Management configuration program to configure both middleware and Maximo Asset Management.

Oracle WebLogic Server must always be configured manually.

You can use the Maximo Asset Management installation program to install DB2. You then use the Maximo Asset Management configuration program to automatically configure it.

## Installing DB2 v10.5 using the Maximo Asset Management version 7.6 installation program

Use the Maximo Asset Management version 7.6 installation program to install DB2 v10.5.

### Procedure

1. Log in to the target system as a user with administrative authority. If you are running the IBM Maximo Asset Management version 7.6 installation program from a Linux or UNIX terminal window, you must be logged in as the root user.
2. Start the IBM Maximo Asset Management version 7.6 installation program from the launchpad.
   a. Start the launchpad.

      **Windows**
      From the downloaded installation image, browse to the root directory and run the following command: `launchpad64.exe`.

      **Linux and UNIX**
      From the downloaded installation image, browse to the root directory and run the following command: `launchpad.sh`.

b. Select a language for the installation and click **OK**.

c. In the launchpad navigation pane, click **Install Product**.

d. From the Install Product panel, select **IBM DB2 v10.5** and then click **Install IBM Maximo Asset Management components**.

3. In the package selection panel, click **Next**.

4. In the package prerequisite validation panel, review the results of the prerequisite check and then click **Next**. If any errors are reported on this panel, resolve the issue and then click **Recheck Status** before continuing.

5. In the license agreement panel, review the license information for each package being installed, select **I accept the terms in the license agreements** if you agree with the terms, and then click **Next**.

6. In the Installation Manager installation location panel, specify path information for the shared resources directory and the Installation Manager installation directory, and then click **Next**. Installation Manager is the installation framework that is used to install Maximo Asset Management version 7.6 components. The shared resources directory is a common workspace directory used by Installation Manager when installing packages.

7. In the package installation location panel, select a package group, specify the path information for its installation directory, and then click **Next**. Repeat this process for each package group listed.

8. In the package translations panel, specify language support for DB2, and then click **Next**.

9. In the package features panel, leave all default options checked, and then click **Next**.

10. In the package configuration panel, specify configuration information for DB2 v10.5, and then click **Next**.

11. In the package summary panel, review the information for the planned installation, and then click **Install**.

# Running the Maximo Asset Management 7.6 installation program

Use the Maximo Asset Management version 7.6 installation program to install Maximo Asset Management version 7.6.

## About this task

In order to install Maximo Asset Management version 7.6, run the Maximo Asset Management version 7.6 installation program on the Maximo Asset Management administrative system.

## Procedure

1. Log in to the Maximo Asset Management administrative workstation. If you run the IBM Maximo Asset Management version 7.6 installation program from a Linux or UNIX terminal window, you must be logged in as the root user.

2. Launch the IBM Maximo Asset Management version 7.6 installation program from the launchpad.

a. Start the launchpad.

**Windows**

From the downloaded installation image, browse to the root directory and run the following command: **launchpad64.exe**.

**Linux and UNIX**

From the downloaded installation image, browse to the root directory and run the following command: `launchpad.sh`.

    b. Select a language for the installation session and click **OK**.

    c. In the launchpad navigation pane, click **Install Product**.

    d. From the **Install Product** panel, select **IBM Maximo Asset Management v7.6** and then click **Install IBM Maximo Asset Management components**.

3. In the package selection panel, click **Next**.

4. In the package prerequisite validation panel, review the results of the prerequisite check, and then click **Next**. If any errors are reported on this panel, resolve the issue and then click **Recheck Status** before continuing.

5. In the license agreement panel, review the license information for each package being installed, select **I accept the terms in the license agreements** if you agree with the terms, and then click **Next**.

6. In the Installation Manager installation location panel, specify path information for the shared resources directory and the Installation Manager installation directory, and then click **Next**. Installation Manager is the installation framework that is used to install Maximo Asset Management version 7.6 components. The shared resources directory is a common workspace directory used by Installation Manager when installing packages.

7. In the package installation location panel, select the IBM Tivoli's process automation suite package group, specify the path information for the Maximo Asset Management version 7.6 installation directory, and then click **Next**.

8. In the package features panel, leave all default options checked, and then click **Next**.

9. In the package summary panel, review the information for the planned installation, and then click **Install**. If you install in a non-English environment, you might notice the environment summary is listed as English. You configure supported languages for Maximo Asset Management later with the Maximo Asset Management configuration program.

10. After the installation is complete, select the option to start the Maximo Asset Management version 7.6 configuration program, and then click **Finish**. The Maximo Asset Management version 7.6 installation program exits and the Maximo Asset Management version 7.6 configuration program is started automatically.

### What to do next

Use the Maximo Asset Management version 7.6 configuration program to configure Maximo Asset Management version 7.6.

## Configuring Maximo Asset Management version 7.6 using the Maximo Asset Management version 7.6 configuration program

Use the Maximo Asset Management version 7.6 configuration program to configure Maximo Asset Management version 7.6.

### Procedure

1. If the Maximo Asset Management version 7.6 configuration program is not open, start it from the Maximo Asset Management launchpad. In the launchpad

navigation pane, click **Configure Product**, specify Maximo Asset Management installation location, and then click **Launch the Tivoli's Process Automation Suite configuration program**.

2. In the IBM Maximo Asset Management configuration operations page, click **Configure a New Deployment**.

3. In the Define Deployment Environment panel, specify information about the your database and application server, in this example, DB2 and WebLogic Server servers that you installed and prepared. Select **Create and configure the database** and **Complete configuration of WebSphere for your product** to automatically configure DB2.  After you define your deployment environment, click **Finish**.

4. In the Configure General Product Information panel, review summary details about the product components that you are installing. Specify the appearance and navigation features for your product, add an e-mail address to receive workflow messages, and choose whether or not to deploy sample data.

5. In the Configure the DB2 Instance panel, specify information about the DB2 instance to create for Maximo Asset Management.

6. In the Configure the DB2 Database panel, specify information about the DB2 database to create for Maximo Asset Management.

7. In the Configure Application Security panel, Enter the user names and passwords for users that must be created for Maximo Asset Management. Do not use the user name as a password value.

8. Choose the base language and any additional languages you want to install.

9. In the Apply Deployment Operations panel, select all available deployment operations, and then click **Finish**.

## Results

Maximo Asset Management version 7.6 is installed and configured to use WebSphere Application Server Network DeploymentWebLogic Server and DB2.

# Chapter 3. Automatically configuring existing middleware during deployment

Use the Maximo Asset Management configuration program to automatically configure existing middleware within your enterprise for use with Maximo Asset Management

## About this task

This information provides a high-level overview or roadmap of tasks you must complete to deploy Maximo Asset Management automatically. You use middleware that is already established in your enterprise.

In this scenario, you use the Maximo Asset Management configuration program automatically configure existing middleware resources.

Oracle WebLogic Server must always be configured manually.

The Maximo Asset Management configuration program is used to automatically configure an existing DB2 server in your environment.

## Running the Maximo Asset Management 7.6 installation program

Use the Maximo Asset Management version 7.6 installation program to install Maximo Asset Management version 7.6.

### About this task

In order to install Maximo Asset Management version 7.6, run the Maximo Asset Management version 7.6 installation program on the Maximo Asset Management administrative system.

### Procedure

1. Log in to the Maximo Asset Management administrative workstation. If you run the IBM Maximo Asset Management version 7.6 installation program from a Linux or UNIX terminal window, you must be logged in as the root user.

2. Launch the IBM Maximo Asset Management version 7.6 installation program from the launchpad.

   a. Start the launchpad.

      **Windows**
      From the downloaded installation image, browse to the root directory and run the following command: `launchpad64.exe`.

      **Linux and UNIX**
      From the downloaded installation image, browse to the root directory and run the following command: `launchpad.sh`.

   b. Select a language for the installation session and click **OK**.

   c. In the launchpad navigation pane, click **Install Product**.

   d. From the **Install Product** panel, select **IBM Maximo Asset Management v7.6** and then click **Install IBM Maximo Asset Management components**.

3. In the package selection panel, click **Next**.

4. In the package prerequisite validation panel, review the results of the prerequisite check, and then click **Next**. If any errors are reported on this panel, resolve the issue and then click **Recheck Status** before continuing.

5. In the license agreement panel, review the license information for each package being installed, select **I accept the terms in the license agreements** if you agree with the terms, and then click **Next**.

6. In the Installation Manager installation location panel, specify path information for the shared resources directory and the Installation Manager installation directory, and then click **Next**. Installation Manager is the installation framework that is used to install Maximo Asset Management version 7.6 components. The shared resources directory is a common workspace directory used by Installation Manager when installing packages.

7. In the package installation location panel, select the IBM Tivoli's process automation suite package group, specify the path information for the Maximo Asset Management version 7.6 installation directory, and then click **Next**.

8. In the package features panel, leave all default options checked, and then click **Next**.

9. In the package summary panel, review the information for the planned installation, and then click **Install**. If you install in a non-English environment, you might notice the environment summary is listed as English. You configure supported languages for Maximo Asset Management later with the Maximo Asset Management configuration program.

10. After the installation is complete, select the option to start the Maximo Asset Management version 7.6 configuration program, and then click **Finish**. The Maximo Asset Management version 7.6 installation program exits and the Maximo Asset Management version 7.6 configuration program is started automatically.

### What to do next

Use the Maximo Asset Management version 7.6 configuration program to configure Maximo Asset Management version 7.6.

## Configuring Maximo Asset Management version 7.6 using the Maximo Asset Management version 7.6 configuration program

Use the Maximo Asset Management version 7.6 configuration program to configure Maximo Asset Management version 7.6.

### Procedure

1. If the Maximo Asset Management version 7.6 configuration program is not open, start it from the Maximo Asset Management launchpad. In the launchpad navigation pane, click **Configure Product**, specify Maximo Asset Management installation location, and then click **Launch the Tivoli's Process Automation Suite configuration program**.

2. In the IBM Maximo Asset Management configuration operations page, click **Configure a New Deployment**.

3. In the Define Deployment Environment panel, specify information about the your database and application server, in this example, DB2 and WebLogic Server servers that you installed and prepared. Select **Create and configure the database** and **Complete configuration of WebSphere for your product** to automatically configure DB2. After you define your deployment environment, click **Finish**.

4. In the Configure General Product Information panel, review summary details about the product components that you are installing. Specify the appearance and navigation features for your product, add an e-mail address to receive workflow messages, and choose whether or not to deploy sample data.

5. In the Configure the DB2 Instance panel, specify information about the DB2 instance to create for Maximo Asset Management.

6. In the Configure the DB2 Database panel, specify information about the DB2 database to create for Maximo Asset Management.

7. In the Configure Application Security panel, Enter the user names and passwords for users that must be created for Maximo Asset Management. Do not use the user name as a password value.

8. Choose the base language and any additional languages you want to install.

9. In the Apply Deployment Operations panel, select all available deployment operations, and then click **Finish**.

## Results

Maximo Asset Management version 7.6 is installed and configured to use WebSphere Application Server Network DeploymentWebLogic Server and DB2.

# Chapter 4. Manually configuring WebLogic Server

WebLogic Server can be used with Maximo Asset Management if it is manually configured.

## About this task

You can use a WebLogic Server as your J2EE server. The Maximo Asset Management installation process prepares EAR files on the Maximo Asset Management administrative workstation. These files are then manually deployed to your WebLogic Server. The Maximo Asset Management installation program does not automatically configure a WebLogic Server environment It does not install or otherwise deploy Maximo Asset Management applications into a WebLogic Server server.

Manually configuring WebLogic Server consists of the following tasks:

## Creating the MAXIMOSERVER domain

This section details how to create the MAXIMOSERVER domain within WebLogic Server.

## About this task

To create the MAXIMOSERVER domain within WebLogic Server, complete the following steps:

## Procedure

1. Launch the Configuration wizard tool.
   - For Windows systems, select **Start** > **Programs** > **BEA Products** > **Tools** > **Configuration Wizard**.
   - For UNIX systems, launch the Configuration wizard using the `config.sh` command located in `<WebLogic_Home>/weblogic10/common/bin`.
2. From the welcome panel, select **Create a new WebLogic Server domain**, and then click **Next**.
3. From the Select a Domain Source panel, select **Generate a domain configured automatically to support the following products**, and then click **Next**.
4. From the Specify Domain Name and Location panel, name the domain MAXIMOSERVER, and then click **Next**
5. From the Configure Administrator Username and Password panel, enter an administrator user ID and password, and then click **Next**.
6. From the Configure Server Start Mode and JDK panel, enter the following information, and then click **Next**.

   **WebLogic Server Domain Startup Mode**
   > Select **Production**.
   >
   > Selecting this option requires that you provide a user ID and password in order to deploy applications.

   **JDK Selection**
   > Select a JDK with a version number 1.6.0 or greater from the supplied JDKs.

7. Proceed through the next few panels until you reach the Customize Environment and Services Settings panel. From the Customize Environment and Services Settings panel, select the default option **No**, and then click **Next**.

8. From the Create WebLogic Server Domain panel, change the Domain Name value to `MaximoServer`, and then click **Create** to create the domain.

9. Once the domain has been created, ensure that the **Start Admin Server** option is cleared, and then click **Done** to close the Configuration wizard. On UNIX systems, this option is not displayed.

# Configuring JMS options for integration framework

This procedure provides details on steps to configure JMS queues.

## Before you begin

You need to be familiar with the Oracle WebLogic Server Administration Console to complete the following procedure. See your Oracle WebLogic Server documentation on how to navigate through the Oracle WebLogic Server Administration Console.

## Procedure

1. Launch Internet Explorer and open the `MAXIMOSERVER` Administration Console by typing the following URL:

   `http://host_name:7001/console`

2. Sign in by entering your user ID and password.

3. Create three persistent file stores with the following configuration parameters:

| Option | Description |
|---|---|
| **File Store Name** | sqinstore, sqoutstore, cqinstore |
| **File Store Target** | MAXIMOSERVER |
| **File Store Directory** | C:\WebLogic, or another directory of your choice |

   File stores have better performance in general. However, you can choose to create database-based stores. To choose between file stores and database-based stores, consult your Oracle WebLogic Server documentation.

4. Create three JMS servers with the following configuration parameters:

| Option | Description |
|---|---|
| **Properties Name** | sqinserver, sqoutserver, cqinserver |
| **Server Target** | MAXIMOSERVER |
| **Bytes Maximum** | 20% or below of the allotted JVM heap size |

   It is recommended that you configure the Bytes Maximum for the JMS servers to a value of 20% or below of the allotted JVM heap size to prevent the Oracle WebLogic Server from running out of memory.

5. Create a JMS system module named `intjmsmodule` with MAXIMOSERVER as the target server.

6. Add a connection factory resource to the JMS module:

| Option | Description |
|---|---|
| **Name** | intjmsconfact |
| **JNDI Name** | jms/maximo/int/cf/intcf |
| **Target** | MAXIMOSERVER |

7. Add three queue resources to the JMS Module:
   a. Create the queue resource type.
   b. Create a new subdeployment with the same name as the queue.
   c. Assign the *subdeployment_name*server server target to the subdeployment.

| Option | Description |
| --- | --- |
| Queue 1 | Name: sqin JNDI Name: jms/maximo/int/queues/sqin |
| Queue 2 | Name: sqout JNDI Name: jms/maximo/int/queues/sqout |
| Queue 3 | Name: cqin JNDI Name: jms/maximo/int/queues/cqin |
| Template | None |

8. Enable the JMS connection factory for integration framework:
   a. Select the **intjmsconfact** link.
   b. From the **Transactions** tab, check the **XA Connection Factory Enabled** check box.
   c. Set the **Maximum Messages per Session** to -1.
9. Save and activate the changes.

### What to do next

You need to stop and restart the MAXIMOSERVER application server for the update to take effect.

## Configuring JMS for the email listener in WebLogic Server

You use the email listeners application to receive and process requests through email in the form of email messages.

### About this task

To configure JMS queues for the WebLogic Server, complete the following steps:

### Procedure

1. Log in to the WebLogic Server administration console.
2. Create a persistent store.
   a. In the Domain Structure panel, expand **Services** and then click **Persistent Stores**.
   b. Click **New** and select the option to create a file store.
   c. Enter lsnrstore for the name of the file store and c:\wls10 as the directory, and then click **Finish**.
3. Create a JMS Server
   a. In the Domain Structure panel, under **Services** > **Messaging** , click **JMS Servers**, enter the following information, and then click **Next**.

      **Name** Enter a name for a new JMS server, for example, lsnrserver.

      **Persistent Store**
             Select **lsnrstore**.
   b. For the **Target**, select **AdminServer**, and then click **Finish**.
4. Create a JMS System Module

a. In the Domain Structure panel, under **Services** > **Messaging** , click **JMS Modules**, and then click **New**.

b. Enter a value for the **Name** field, such as lsnrjmsmodule, and then click **Next**.

c. Select the option for **AdminServer**, and then click **Next**.

d. Select the option for **Would you like to add resources to the JMS system module?**, and then click **Finish**.

5. Add a connection factory to the JMS module.

a. From the Summary of Resources table of the lsnrjmsmodule module, click **New** to add a resource.

b. Select the **Connection Factory** option, and then click **Next**.

c. From the Connection Factory Properties panel, enter the following information, and then click **Next**.

**Name**    Enter a name for the new connection factory. For example, lsnrconnfact.

**JNDI Name**
Enter **jms/mro/lsnr/lsnrcf**.

d. Ensure AdminServer is selected, and then click **Finish**

6. Add a queue to the JMS module.

a. From the Summary of Resources table of the lsnrjmsmodule module, click **New** to add a resource.

b. Select the **Queue** option, and then click **Next**.

c. From the JMS Destination Properties panel, enter the following information, and then click **Next**.

**Name**    Enter a name for the new queue. For example, lsnrqueue.

**JNDI Name**
Enter **jms/mro/lsnr/qin**.

**Template**
No template is necessary.

d. Click **Create a new Subdeployment**, select lsnrserver, and then click **Finish**.

7. Set the XA option for the connection factory.

a. Select lsnrconnfact, and then select the Transactions tab.

b. Ensure that the XA Connection Factory Enabled option is selected, and then click **Save**.

8. Click **Activate Changes** to save and activate all changes.

9. To enable an email listener to use JMS queues, a Message Driven Bean is configured through a deployment descriptor file that is part of the installation. You configure the Message Driven Bean by removing comment lines from sections within the deployment descriptor files. Edit the ejb-jar.xml and weblogic-ejb-jar.xml files.

a. Open the *install_dir*\applications\maximo\mboejb\ejbmodule\META-INF\ ejb-jar.xml file in an editor, uncomment the following sections, and then save the file:

```
Email Listener JMS is not deployed by default
<message-driven id="MessageDriven_LSNRMessageBean">
<ejb-name>LSNRMessageBean</ejb-name>
ejb-class>psdi.common.emailstner.LSNRMessageBean</ejb-class>
```

```
transaction-type>Container</transaction-type>
<message-destination-type>javax.jms.Queue</messagedestination-
type>
</message-driven>
```

```
Email Listener JMS is not deployed by default
<container-transaction>
<method>
<ejb-name>LSNRMessageBean</ejb-name>
<method-name>*</method-name>
</method>
<trans-attribute>Required</trans-attribute>
</container-transaction>
```

b. Open the `c:\ibm\smp\maximo\applications\maximo\mboejb\ejbmodule\META-INF\weblogic-ejb-jar.xml` file in an editor, uncomment the following section, and then save the file:

```
Email Listener JMS is not deployed by default
<weblogic-enterprise-bean>
<ejb-name>LSNRMessageBean</ejb-name>
<message-driven-descriptor>
<destination-jndi-name>jms/mro/lsnr/qin</destination-jndiname>
<connection-factory-jndi-name>jms/mro/lsnr/lsnrcf</
connection-factory-jndi-name>
</message-driven-descriptor>
<transaction-descriptor>
<trans-timeout-seconds>600</trans-timeout-seconds>
</transaction-descriptor>
<jndi-name>LSNRMessageBean</jndi-name>
</weblogic-enterprise-bean>
```

# Post installation for Oracle WebLogic Server deployments

There is a post installation step required for Oracle WebLogic Server deployments.

## About this task

When using Oracle WebLogic Server as the application server for your Maximo Asset Management deployment, the installation program produces an EAR file. You must then manually deploy that file to Oracle WebLogic Server.

When using Oracle WebLogic Server, you must modify the `weblogic.xml` file and rebuild the `maximo.ear` file before deploying it.

## Procedure

1. Install the updated JDK 1.6 program.
   a. Obtain the latest JDK program with a version number 1.6.0 SR7 or greater, from http://www.oracle.com/technetwork/java/index.html
   b. Start the installation program.
   c. Accept the license agreement.
   d. Change the installation destination path to your Oracle home folder, and click **OK** , then **Next**.
   e. Accept any additional prompts to install the JDK program so that the installation process can complete successfully.
2. If you are using an Oracle database, copy the `oraclethin.jar` file.
   a. Copy the `oraclethin.jar` file from the administrative workstation to the server where you installed Oracle WebLogic Server. Use any file transfer protocol that is available.

b. The `oraclethin.jar` file is in the *install_home*\maximo\applications\
maximo\lib directory. The default *install_home* directory is
C:\IBM\SMP\Maximo\Applications\Maximo\lib. Copy the `oraclethin.jar` file
to the J2EE server and into the wls11g/server/lib directory. For example,
C:/Oracle/wls11g/server/lib.

3. Edit the domain configuration file for appropriate Java settings.

   a. Navigate to *Install_dir*\user_projects\domains\mydomain\bin.

   b. Right-click and edit (do not double-click) `setDomainEnv.cmd` (Windows) or
   `setDomainEnv.sh` (UNIX).

   c. Using your text editor, search for and replace `jdk` entries with entries that
   configure Oracle WebLogic Server to use the updated JDK program. Two
   instances must be replaced.

   d. (Oracle only) Search for `PRE_CLASSPATH`. Set the parameter as follows:

   ```
   set PRE_CLASSPATH=%WL_HOME%\server\lib\oraclethin.jar (Windows)
   PRE_CLASSPATH={WL_HOME}/server/lib/oraclethin.jar (UNIX)
   ```

   **Note:** This step is required for connecting to Oracle databases from Oracle
   WebLogic Server.

   e. Increase the Java Heap parameters by searching for `MEM_ARGS` and setting
   the parameter as follows:

   ```
   set MEM_ARGS=-Xms512m -Xmx1024m -XX:MaxPermSize=512m (Windows)
   MEM_ARGS=-Xms512m -Xmx1024m -XX:MaxPermSize=512m (UNIX)
   ```

   These parameters are case-sensitive. Ensure to include the minus (-) sign
   before each parameter and to include the `m` after each value.

   f. Search for `set JAVA_OPTIONS=%JAVA_OPTIONS% %JAVA_PROPERTIES%`. Insert an
   option directly after this string, as follows:

   ```
   -Dcom.sun.xml.namespace.QName.useCompatibleSerialVersionUID=1.0
   ```

   The entry looks like this example when complete:

   ```
   set JAVA_OPTIONS=%JAVA_OPTIONS% %JAVA_PROPERTIES%
   -Dcom.sun.xml.namespace.QName.useCompatibleSerialVersionUID=1.0
   -Dwlw.iterativeDev=%iterativeDevFlag% -Dwlw.testConsole=%testConsoleFlag%
   -Dwlw.logErrorsToConsole=%logErrorsToConsoleFlag%
   ```

   g. Save and close the `setDomainEnv.cmd` file.

4. Edit the \IBM\SMP\maximo\applications\maximo\maximouiweb\webmodule\WEB-
INF\weblogic.xml file and add the following entries:

   ```
   <session-descriptor>
   <cookie-http-only>false</cookie-http-only>
   </session-descriptor>
   ```

5. Start the server without being prompted for user name and password. If you
are working in Production mode and do not want to be prompted for the
Oracle WebLogic Server user name and password, create a `boot.properties`
file. This procedure stores the user name and password in an encrypted format.
Place the following two lines in a text file:

   ```
   username=<administrator username>
   password=<administrator password>
   ```

   The user name and password values must match an existing user account in
   the Authentication provider for the default security realm. If you save the file
   as `boot.properties` and locate it in the `mydomain` folder (for example,
   C:\BEA\User_projects\domains\mydomain), the server automatically uses this file
   during its subsequent startup cycles. The first time you use this file to start a

server, the server reads the file and then overwrites it with an encrypted version of the user name and password.

6. Verify settings by starting Oracle WebLogic Server. To ensure that the edits that you made to the previous Oracle WebLogic Server files were accurate, start the Oracle WebLogic Server.

7. Rebuild the `maximo.ear` file.

   `\IBM\SMP\maximo\deployment\buildmaximoear`

8. Deploy the `maximo.ear` file to Oracle WebLogic Server.

## Deploying ear files

You are now ready to deploy the EAR files from the Oracle WebLogic Server Administration Console. Deploying the EAR files starts the Maximo Application Server.

### Before you begin

`Windows` **Windows**
>    If `MAXIMOSERVER` is not running, start it from a command prompt.

`UNIX` **UNIX**
>    Ensure that the Oracle WebLogic Server daemon is running.

### About this task

In order to view the Oracle WebLogic Server Administration Console, you need to also install a Java Virtual Machine (JVM). Newer versions of Windows might not ship with JVM.

### Procedure

1. Change to the *my_domain* directory:

   `Windows` **Windows**
   >    From a command prompt, change directory path to:
   >
   >    `drive_name:\weblogic_install_dir\user_projects\domain\my_domain`

   `UNIX` **UNIX**
   >    Open a Terminal window (or a command prompt, if you are telneting from a Windows computer) and change directory as follows:
   >
   >    `cd /home/mxadmin/weblogic_install_dir/user_projects/domains/my_domain`

2. Type

   `Windows` **Windows**
   >    **startWebLogic.cmd**

   `UNIX` **UNIX**
   >    **./startWebLogic.sh**

   and press Enter. Once the phrase: `Server started in RUNNING mode` displays, the Oracle WebLogic Server is running.

3. Start the `MAXIMOSERVER` Oracle WebLogic Server Administration Console by typing the following URL in the address bar of your browser:

   `http://host_name:7001/console`

4. Log in to the administrative console with the administrative user name and password.

5. Click the **Lock&Edit** button in the **Change Center** section to set the server to edit mode.

6. Click the **Deployments** link in the **Domain Structure** section.

7. In the Summary of Deployments section, click the **Install** button. The Install Application Assistant opens.

8. Click the **upload your file(s)** link.

9. Click the **Browse** button next to the **Deployment Archive** field. Browse to where you have `maximo.ear` file installed (`c:/ibm/smp/maximo/deployment/default/` of the administrative workstation), select the file and click **Open**.

10. Click **Next** to upload the file to the Oracle WebLogic Server server.

11. Click the radio button next to the `maximo.ear` file, and **Next** to continue the deployment.

12. Accept the default value to install the deployment as an application and click **Next**.

13. Accept all other default values and click **Finish** to start the deployment process.

14. When the deployment process is complete, click the **Activate Changes** button in the Change Center to apply the changes. The following message displays: `All changes have been activated. No restarts are necessary.`

15. Repeat Steps 8 through 14 to deploy `maximoiehs.ear`.

16. Click the checkbox next to the **maximo** application.

17. Click the dropdown arrow of the **Start** button and select **Servicing all requests**. The Start Application Assistant displays.

18. Click **Yes** to start the system.

## Setting the Oracle WebLogic Server host name

Setting the Oracle WebLogic Server host name manually to access application help information.

### About this task

By default, the property used to store the host name for the Maximo Asset Management help system is set to `localhost`. The fully qualified host name of the Oracle WebLogic Server must be configured in the help system property to view Maximo Asset Management application help topics. Complete this step after you have deployed Maximo Asset Management EAR files to the Oracle WebLogic Server.

### Procedure

1. Log in to the Maximo Asset Management console as maxadmin.

2. Browse to **Go To** > **System Configuration** > **Platform Configuration** > **System Properties**

3. Using the Filter feature, search for the **mxe.help.host** Property Name.

4. Expand the **mxe.help.host** property and set the attribute to the fully qualified host name of the Oracle WebLogic Server.

5. Select the **mxe.help.host** record check box.

6. Click the Live Refresh icon in the toolbar.

7. From the Live Refresh dialog, click **OK**.

**Results**

You are now able to access application help topics in the Maximo Asset Management user interface.

# Chapter 5. Manually configuring existing middleware during deployment

You can deploy Maximo Asset Management by reusing existing middleware servers and manually configuring them to work with Maximo Asset Management.

This information provides a high-level overview or road map of tasks you need to complete in order to deploy Maximo Asset Management automatically, using middleware already established in your enterprise.

In this scenario, you manually configure existing middleware resources before running the Maximo Asset Management configuration program.

## Maximo Asset Management installation with manual middleware configuration

Manually configured installations involve configuring middleware components, the database server, the directory server, and the J2EE server, to work with Maximo Asset Management before you use the Maximo Asset Management installation and configuration programs.

You can automatically configure one or more Maximo Asset Management middleware components with the Maximo Asset Management configuration program. Alternatively, you can manually configure middleware servers to work with Maximo Asset Management before you run the Maximo Asset Management installation and configuration programs..

Before you begin, ensure that the following prerequisite conditions are met:
- You designate a Windows or UNIX server to start the Maximo Asset Management configuration program.

You must complete the manual configuration of each middleware server before you use the Maximo Asset Management configuration program to configure Maximo Asset Management to work with the servers.

Ensure that the middleware you intend to use with Maximo Asset Management is supported.

### Manual configuration of the database

You can manually configure the database server that is used with Maximo Asset Management. You must manually configure the database server before you use the Maximo Asset Management installation and configuration programs.

For DB2 on UNIX and Linux, a minimum of 8 gigabytes (binary) free of space is required for DB2 table spaces. This space must be available to the database instance home directory (/home/ctginst1).

For DB2 on Windows, ensure that you have a minimum of 8 gigabytes of free space in the DB2 installation directory.

#### Manually configuring DB2 v10.5
Manually configuring DB2 v10.5 servers for use by Maximo Asset Management.

## About this task

To configure an existing DB2 v10.5 server for use with Maximo Asset Management, complete the following steps before you start the Maximo Asset Management installation and configuration programs:

## Procedure

1.  Log in to the system as a user that has administrative permissions on the system.
2.  Create system users if they do not exist on the system.
    *   Windows
        - db2admin
        - maximo
    *   Linux or UNIX
        - db2admin
        - maximo
        - ctgfenc1
        - ctginst1

        The ctginst1 user ID must be assigned to db2iadm1 as its primary group.
3.  Open a console window and set up the DB2 environment:
    *   Windows

        db2cmd

    *   Linux or UNIX

        Ensure that the `/opt/ibm/db2/V10.5/bin`, `/opt/ibm/db2/v10.5/instance`, and `/opt/ibm/db2/V10.5/adm` directories are added to your PATH.
4.  Create the DB2 instance:
    *   Windows

        ```
        db2icrt -s ese -u db2admin,myPassword -r 50005,50005 ctginst1
        set db2instance=ctginst1
        db2start
        db2 update dbm config using SVCENAME 50005 DEFERRED
        db2stop
        db2set DB2COMM=tcpip
        db2start
        ```

    *   Linux orUNIX

        ```
        db2icrt -s ese -u ctgfenc1 -p 50005 ctginst1
        . /home/ctginst1/sqllib/db2profile
        db2start
        db2 update dbm config using SVCENAME 50005 DEFERRED
        db2stop
        db2set DB2COMM=tcpip
        db2start
        ```

5.  Create the database.

    ```
    db2 create db 'maxdb76' ALIAS 'maxdb76' using codeset UTF-8 territory US pagesize 32 K
    db2 connect to 'maxdb76'
    db2 GRANT DBADM ON DATABASE TO USER db2admin (windows only)
    db2 GRANT SECADM ON DATABASE TO USER db2admin (windows only)
    db2 connect reset
    ```

6.  Configure the database.

    ```
    db2 update db cfg for maxdb76 using SELF_TUNING_MEM ON
    db2 update db cfg for maxdb76 using APPGROUP_MEM_SZ 16384 DEFERRED
    db2 update db cfg for maxdb76 using APPLHEAPSZ 2048 AUTOMATIC DEFERRED
    db2 update db cfg for maxdb76 using AUTO_MAINT ON DEFERRED
    db2 update db cfg for maxdb76 using AUTO_TBL_MAINT ON DEFERRED
    db2 update db cfg for maxdb76 using AUTO_RUNSTATS ON DEFERRED
    db2 update db cfg for maxdb76 using AUTO_REORG ON DEFERRED
    db2 update db cfg for maxdb76 using AUTO_DB_BACKUP ON DEFERRED
    db2 update db cfg for maxdb76 using CATALOGCACHE_SZ 800 DEFERRED
    db2 update db cfg for maxdb76 using CHNGPGS_THRESH 40 DEFERRED
    db2 update db cfg for maxdb76 using DBHEAP AUTOMATIC
    db2 update db cfg for maxdb76 using LOCKLIST AUTOMATIC DEFERRED
    db2 update db cfg for maxdb76 using LOGBUFSZ 1024 DEFERRED
    db2 update db cfg for maxdb76 using LOCKTIMEOUT 300 DEFERRED
    db2 update db cfg for maxdb76 using LOGPRIMARY 20 DEFERRED
    db2 update db cfg for maxdb76 using LOGSECOND 100 DEFERRED
    db2 update db cfg for maxdb76 using LOGFILSIZ 8192 DEFERRED
    db2 update db cfg for maxdb76 using SOFTMAX 1000 DEFERRED
    db2 update db cfg for maxdb76 using MAXFILOP 32768 DEFERRED #32-bit Windows
    db2 update db cfg for maxdb76 using MAXFILOP 65335 DEFERRED #64-bit Windows
    db2 update db cfg for maxdb76 using MAXFILOP 30720 DEFERRED #32-bit UNIX
    ```

```
db2 update db cfg for maxdb76 using MAXFILOP 61440 DEFERRED #64-bit UNIX
db2 update db cfg for maxdb76 using PCKCACHESZ AUTOMATIC DEFERRED
db2 update db cfg for maxdb76 using STAT_HEAP_SZ AUTOMATIC DEFERRED
db2 update db cfg for maxdb76 using STMTHEAP AUTOMATIC DEFERRED
db2 update db cfg for maxdb76 using UTIL_HEAP_SZ 10000 DEFERRED
db2 update db cfg for maxdb76 using DATABASE_MEMORY AUTOMATIC DEFERRED
db2 update db cfg for maxdb76 using AUTO_STMT_STATS OFF DEFERRED
db2 update db cfg for maxdb76 using STMT_CONC LITERALS DEFERRED
db2 update alert cfg for database on maxdb76 using db.db_backup_req SET THRESHOLDSCHECKED YES
db2 update alert cfg for database on maxdb76 using db.tb_reorg_req SET THRESHOLDSCHECKED YES
db2 update alert cfg for database on maxdb76 using db.tb_runstats_req SET THRESHOLDSCHECKED YES
db2 update dbm cfg using PRIV_MEM_THRESH 32767 DEFERRED
db2 update dbm cfg using KEEPFENCED NO DEFERRED
db2 update dbm cfg using NUMDB 2 DEFERRED
db2 update dbm cfg using RQRIOBLK 65535 DEFERRED
db2 update dbm cfg using HEALTH_MON OFF DEFERRED
#
# Windows: AGENT_STACK_SZ 1000
# Unix   : AGENT_STACK_SZ 1000
#
db2 update dbm cfg using AGENT_STACK_SZ 1000 DEFERRED
db2 update dbm cfg using MON_HEAP_SZ AUTOMATIC DEFERRED
db2set DB2_SKIPINSERTED=ON
db2set DB2_INLIST_TO_NLJN=YES
db2set DB2_MINIMIZE_LISTPREFETCH=Y
db2set DB2_EVALUNCOMMITTED=YES
db2set DB2_FMP_COMM_HEAPSZ=65536
db2set DB2_SKIPDELETED=ON
db2set DB2_USE_ALTERNATE_PAGE_CLEANING=ON
```

7. For Linux orUNIX, log into the system as the ctginst1 user, and then restart the DB2 command-line environment.

   ```
   su - ctginst1
   db2
   ```

8. Restart DB2.

   ```
   db2stop force
   db2start
   ```

9. Reconnect to the database.

   ```
   db2 connect to 'maxdb76'
   ```

10. Create a buffer pool.

    ```
    db2 CREATE BUFFERPOOL MAXBUFPOOL IMMEDIATE SIZE 4096 AUTOMATIC PAGESIZE 32 K
    ```

11. Create table spaces.

    ```
    db2 CREATE REGULAR TABLESPACE MAXDATA PAGESIZE 32 K MANAGED BY AUTOMATIC STORAGE INITIALSIZE 5000 M BUFFERPOOL MAXBUFPOOL
    db2 CREATE TEMPORARY TABLESPACE MAXTEMP PAGESIZE 32 K MANAGED BY AUTOMATIC STORAGE BUFFERPOOL MAXBUFPOOL
    db2 CREATE REGULAR TABLESPACE MAXINDEX PAGESIZE 32 K MANAGED BY AUTOMATIC STORAGE INITIALSIZE 5000 M BUFFERPOOL MAXBUFPOOL
    db2 GRANT USE OF TABLESPACE MAXDATA TO USER MAXIMO
    ```

12. Create Schema

    ```
    db2 create schema maximo authorization maximo
    ```

13. Grant authority to maximo.

    ```
    db2 GRANT DBADM,CREATETAB,BINDADD,CONNECT,CREATE_NOT_FENCED_ROUTINE,IMPLICIT_SCHEMA,
    LOAD,CREATE_EXTERNAL_ROUTINE,QUIESCE_CONNECT,SECADM ON DATABASE TO USER MAXIMO
    ```

    ```
    db2 GRANT USE OF TABLESPACE MAXDATA TO USER MAXIMO
    ```

    ```
    db2 GRANT  CREATEIN,DROPIN,ALTERIN ON SCHEMA MAXIMO TO USER MAXIMO
    ```

14. Break the database connection.

    ```
    db2 connect reset
    ```

15. Exit the DB2 command-line environment

16. Install the appropriate fix pack that is indicated in the prerequisites topic in the chapter that describes how to prepare for the installation. Ensure that you review and complete all the installation and postinstallation tasks that are contained within the fix pack readme file. Failure to do so can potentially cause the Maximo Asset Management installation to fail.

## Manually configuring Oracle 12c

Use the following instructions to manually configure Oracle 12c for use with Maximo Asset Management.

### Before you begin

Oracle instance names must be limited to 8 characters in length. For Oracle databases, the schema owner must use the same user ID as the database user.

**About this task**

To configure an existing Oracle 12c server for use with Maximo Asset
Management, complete the following steps before you start the Maximo Asset
Management installation program:

**Procedure**

1. Log in as the Oracle software user. Typically this user is named `oracle`.
2. Create the database listener if it does not exist. The listener manages requests to
   connect to the database.
   a. Open the Oracle Network Configuration Assistant application.
   b. From the Welcome panel, select **Listener configuration**, and then click **Next**.
   c. From the action panel, select **Add**, and then click **Next**.
   d. Enter a name for the listener or accept the default value, and then click
      **Next**.
   e. Accept the default Selected Protocols that are listed by clicking **Next**.
   f. From the port panel, select **Use the standard port of 1521**, and then click
      **Next**.
   g. Select **No** to indicate that you are finished configuring listeners, and then
      click **Next**.
   h. From the Listener Configuration Done panel, click **Next**.
   i. Click **Finish**.
3. Create a database for use by Maximo Asset Management.
   a. Open the Oracle Database Configuration Assistant.
   b. Select **Create Database**, and then click **Next**.
   c. Select **Advanced Mode**, and then click **Next**.
   d. Select **General Purpose or Transaction Processing**, and then click **Next**.
   e. Enter `ctginst1` for both the **Global Database Name** value and the **SID**
      value, and then click **Next**.
   f. Leave the defaults selected, and click **Next**.
   g. Ensure **Use the Same Administrative Password for All Accounts** is
      selected, enter a password for Oracle users, and then click **Next**.
   h. Leave the defaults selected, and click **Next**.
   i. Ensure **File System** is selected as the storage type to use for the database.
      Ensure **Use Database File Locations from Template** is selected as the value
      to use for database file location, leave defaults selected for the database
      recovery options, and then click **Next**.
   j. From the Sample Schemas panel, click **Next**.
   k. From the Initialization Parameters panel, on the Memory tab, select **Custom
      Settings** and provide the following values (measured in Mb):

      **Memory Management**
              Set this value to **Manual Shared Memory Management**.

      **Shared Pool**
              Set this value to 152.

      **Buffer Cache**
              Set this value to 36.

      **Java Pool**
              Set this value to 32.

**Large Pool**
Set this value to 8.

**PGA Size**
Set this value to 36.

l. From the Character Sets tab, select **Use Unicode (AL32UTF8)**.

m. Click **All Initialization Parameters...**.

n. Click **Show Advanced Parameters**.

o. Locate the following parameters, change them to the values indicated, and then click **Close**.

**nls_length_semantics**
Change this value to **CHAR**.

**open_cursors**
Change this value to 1000.

**cursor_sharing**
Set this value to FORCE.

p. From the Initialization Parameters panel, click **Next**.

q. From the Creation Options panel, click **Next**.

r. From the Summary panel, click **Finish**.

s. Click **Exit** to exit the Database Configuration Assistant. The database is successfully created.

**Note:** The Oracle Database Configuration Assistant starts the `ORACLE_HOME/ctx/admin/defaults/drdefus.sql` script as part of the configuration of the CTXSYS user. You must start this script manually if you do not use the Oracle Database Configuration Assistant.

4. Create a table space with the following command in SQL*Plus:

```
Create tablespace maxdata datafile
'C:\oracle\product\12.1.0.1\db_1\dbs\maxdata.dbf'
size 1000M autoextend on;
```

Change the directory that is specified in the example to the database location. If the directory does not exist, this command fails.

5. Create a temporary table space with the following command in SQL*Plus

```
create temporary tablespace maxtemp tempfile
'C:\oracle\product\12.1.0.1\db_1\dbs\maxtemp.dbf'
size 1000M autoextend on maxsize unlimited;
```

Change the directory that is specified in the example to the database location. If the directory does not exist, this command fails.

6. Create the maximo user and grant permissions with the following command in SQL*Plus:

```
create user maximo identified by maximo default tablespace maxdata temporary
tablespace maxtemp;
grant connect to maximo;
grant create job to maximo;
grant create trigger to maximo;
grant create session to maximo;
grant create sequence to maximo;
grant create synonym to maximo;
grant create table to maximo;
grant create view to maximo;
grant create procedure to maximo;
grant alter session to maximo;
grant execute on ctxsys.ctx_ddl to maximo;
alter user maximo quota unlimited on maxdata;
```

If you created a separate table space for indexing, you must also grant access to that index table space to the maximo user. Use the following command if you created a separate table space for indexing called TSI_MAM_OWN.":

```
alter user maximo quota unlimited on TSI_MAM_OWN
```

**What to do next**

If you modified default role sets assigned to the user ID used to connect to the database, or restricted the default privileges that are granted to user IDs, you must explicitly grant them to the maximo user. For example, if you do not grant a role such as the select_catalog_role role, you must explicitly grant that role to the maximo user. This assignment can be done by running this SQL*Plus command:

```
grant select_catalog_role to maximo
```

## Manually configuring Microsoft SQL Server 2012

You can manually configure Microsoft SQL Server for use with Maximo Asset Management.

**Before you begin**

Because Microsoft SQL Server does not support UTF-8, Maximo Asset Management does not have multilingual support when deployed with Microsoft SQL Server.

Microsoft SQL Server Database Collation settings must be set to the following options:
- Dictionary order
- Case-insensitive
- For use with 1252 Character set

**About this task**

To configure an existing Microsoft SQL Server for use with Maximo Asset Management, complete the following steps before you start the Maximo Asset Management installation and configuration programs:

**Procedure**

1. Configure the listener port.

   If The default instance of the Microsoft SQL Server Database Engine listens on TCP port 1433. Named instances of the Microsoft SQL Server Database Engine and Microsoft SQL Server Compact Edition are configured for dynamic ports, which means they select any available port when the service starts. When you connect to a named instance across a firewall, configure the Database Engine to listen on a specific port, to open this port in the firewall.

   a. Open **Programs** > **Microsoft SQL Server 2012** > **Configuration Tools** > **SQL Server Configuration Manager**

   b. From the Microsoft SQL Server Configuration Manager navigation pane, expand **SQL Server Network Configuration** > **Protocols for <*instance name*>,** and then double-click **TCP/IP**.

   c. In the TCP/IP Properties dialog box, click the **IP Addresses** tab.

   d. For each IP address listed, ensure that the **TCP Dynamic Ports** field is blank. If the **TCP Dynamic Ports** field contains a value of 0, that IP address is using dynamic ports. Since Maximo Asset Management requires Microsoft SQL Server to listen on a static port, this field must be blank.

e. For each IP address listed, enter 1433 for the **TCP Port** field.

f. Click **OK**.

g. From the SQL Server Configuration Manager navigation pane, click **SQL Server Services**.

h. Right-click **SQL Server (*<instance name>*)** and then click **Restart**, to stop, and restart Microsoft SQL Server.

i. Verify the Microsoft SQL Server instance is running, and then close SQL Server Configuration Manager.

2. Verify that you enabled the Full-text Search setting during the installation of Microsoft SQL Server 2012. To determine if Full-text Search is installed on your existing Microsoft SQL Server database, complete the following steps:

a. Open SQL Server Management Studio, select **SQL Server Authentication** as the authentication type, and enter the credentials for the **sa** user ID, and then click **Connect**.

b. Click **New Query**.

c. Type the following command and then click **Execute**.:

```
select FULLTEXTSERVICEPROPERTY ( 'IsFulltextInstalled' )
```

If you did not install Full-text Search (the resulting value is zero), you must do so now. To install Full-text Search, insert the Microsoft SQL Server 2012 product media into the system and start the installation wizard. Navigate through the wizard and enable the Full-Text Search option that is on the Custom portion of the Setup Type dialog box. To finish the installation process, restart the server.

3. Create a Microsoft SQL Server database.

a. Open SQL Server Management Studio (**Start** > **Programs** > **Microsoft SQL Server 2012** > **SQL Server Management Studio**)

b. Right-click the databases folder from the navigation tree, and select **New Database**.

c. In the database name entry field, specify a unique database name. For example, enter maxdb76

d. For the maxdb76 Logical Name, change the Initial Size (MB) attribute to 500 (MB), and also set the value of the **Autogrowth / Maxsize** field to **By 1 MB, Unlimited**.

e. If you prefer, modify the log settings to accommodate your production environment.

f. Click **OK**.

4. Create the Maximo User for Microsoft SQL Server.

a. Open Microsoft SQL Server Management Studio (**Start** > **Programs** > **Microsoft SQL Server 2012** > **SQL Server Management Studio**)

b. From the SQL Server Configuration Manager navigation pane, click **Databases**.

c. Right-click the **maxdb76** database and select **New Query**.

d. Enter the following script to create the Maximo database user MAXIMO with a password that adheres to the password policy of the system.

```
sp_addlogin MAXIMO,password
go
```

This value is case-sensitive.

e. Click **Execute**.

f. Enter the following script to change the database owner to maximo.

```
sp_changedbowner MAXIMO
go
```

g.  Click **Execute**.

**What to do next**

If you add logical names to the database and set their file group to a value other than PRIMARY, complete these steps after you set up the database and create the Maximo user:

1.  Run the Maximo Asset Management installation program and choose the **Do not run the configuration step now** option.
2.  Add the following property to the *install_home*\etc\install.properties file:

    Database.SQL.DataFilegroupName=*your_logical_name*

3.  Run the configuration tool to complete the deployment.

Complete these additional steps only if you added logical names to the database and set their file group to a value other than PRIMARY.

# Manual configuration of the directory server

You can manually configure a directory server for Maximo Asset Management to manage authentication and authorization.

There are several methods to secure Maximo Asset Management.If want to use a directory server to secure Maximo Asset Management, you must complete the manual configuration of the directory server before you use the Maximo Asset Management installation and configuration programs.

**Note:** The base dn, bind user, and other various node values that are listed are default values. These values are replaced with values applicable to existing LDAP hierarchies within your organization.

**Important:** When you enter LDAP values for Maximo Asset Management configuration panel fields, entries in LDIF files, or values you enter directly into a directory instance with directory server tools, be aware of the product-specific syntax rules for using special characters in an LDAP string. In most cases, special characters must be preceded by an escape character to make it readable by the directory server. Failing to escape special characters that are contained in an LDAP string that is used with Maximo Asset Management results in Maximo Asset Management errors.

Many directory server products consider a blank space as a special character that is part of the LDAP string. If you mistakenly enter an LDAP string that contains a blank, at the end of a field value, for example, and you do not precede the blank character with an escape character, you encounter Maximo Asset Management errors that are difficult to troubleshoot.

For more information about special characters in LDAP strings, see the product documentation for your directory server.

## Manually configuring IBM Tivoli Directory Server
Manually configure IBM Tivoli Directory Server for use with Maximo Asset Management.

## About this task

To configure IBM Tivoli Directory Server before you start the Maximo Asset Management installation and configuration programs, you must create an instance of IBM Tivoli Directory Server.

**Note:** While you can technically share a DB2 instance between Maximo Asset Management and the one needed by IBM Tivoli Directory Server, it might lead to problems. During the installation, the database instance is restarted, which might disrupt the availability of IBM Tivoli Directory Server to your enterprise. If you are using the automated installation programs, separate instances are created for use by Maximo Asset Management and IBM Tivoli Directory Server.

## Procedure

1. Using your preferred method, create a user on the system and assign it to the appropriate group.

   **Windows**
   > Create the user `db2admin` and make it a member of the following groups:
   > - Windows Administrators
   > - DB2ADMNS
   > - DB2USERS

   **UNIX**    Create the user `idsccmdb` and make it a member of the following groups:
   > - dasadmn1
   > - idsldap
   > - dbsysadm

   > The root user must also be a member of the dasadm1, idsldap, and dbsysadm groups.

2. If the Instance Administration tool is not already started, ensure that you are logged in as an administrator on the system, and then start the tool

   **Windows**
   > Select **Programs** > **IBM Tivoli Directory Server 6.3** > **Instance Administration Tool**.

   **UNIX**    Type **/opt/IBM/ldap/V6.3/sbin/idsxinst** at the command line.

3. In the Instance Administration tool, click **Create an instance**.

4. In the Create a new instance window, click **Create a new directory server instance**, and then click **Next**.

5. From the Instance details window, enter values for the following fields, and then click **Next**.

   **User name**
   > Select `idsccmdb` as the system user ID of the user who owns the instance. This name is also the name of the instance.

   **Install location**
   > Enter the location where the instance files are stored.

   **Encryption seed string**
   > Type a string of characters that are used as an encryption seed. This value must be a minimum of 12 characters.

**Instance description**

Enter a brief description of the instance.

6. In the DB2 instance details panel, enter `idsccmdb` as the value for the DB2 instance name field, and then click **Next**.

7. In the TCP/IP settings for multihomed hosts panel, select **Listen on all configured IP addresses**, and then click **Next**.

8. In the TCP/IP port settings panel, complete the following fields, and then click **Next**.

**Server port number**

Enter 389 as the contact port for the server.

**Server secure port number**

Enter 636 as the secure port for the server.

**Admin daemon port number**

Enter 3538 as the administration daemon port.

**Admin daemon secure port number**

Enter 3539 as the administration daemon secure port.

9. In the Option steps panel, leave the following options selected, and then click **Next**.

**Configure admin DN and password**

You want to configure the administrator DN and password for the instance now.

**Configure database**

You want to configure the database for the directory server now.

10. In the Configure administrator DN and password panel, complete the following fields, and then click **Next**.

**Administrator DN**

Enter `cn=root` for the administrator distinguished name.

**Administrator Password**

Enter a password for the Administrator DN.

11. From the Configure database panel, complete the following fields, and then click **Next**.

**Database user name**

Enter `idsccmdb` as the database user.

**Password**

Enter the password for the idsccmdb user.

**Database name**

Enter `idsccmdb` as the database to be used with this directory instance.

12. In the Database options panel, complete the following fields, and then click **Next**.

**Database install location**

Type the location for the database.

> **Windows**
>
> For Windows platforms, this value must be a drive letter.

> **UNIX** For non-Windows systems, the location must be a directory name, such as `/home/ldapdb`.

> Ensure that you have at least 80 MB of free hard disk space in the
> location you specify. More disk space must be available to
> accommodate growth as new entries are added to the directory.

**Character-set option**
> Leave the **Create a universal DB2 database (UTF-8/UCS-2)** option
> selected.

13. In the Verify settings panel, review the instance creation details that are
    provided, and then click **Finish** to create the idsccmdb instance.

14. Click **Close** to close the window and return to the main window of the
    Instance Administration tool.

15. Click **Close** to exit the Instance Administration tool.

16. Start the IBM Tivoli Directory Server Configuration tool:

    **Windows**
    > Select**Programs** > **IBM Tivoli Directory Server 6.3** > **Instance
    > Administration Tool**.

    **UNIX**  Type **./opt/IBM/ldap/V6.3/sbin/idsxcfg** at the command line.

17. Select **Manage suffixes**.

18. In the Manage suffixes panel, type the following suffix, and then click **Add**.
    ```
    o=IBM,c=US
    ```

19. Click **OK**.

20. Create and save an LDIF file.

    Add the DN information, for example:

    - ou=SWG,o=IBM,c=US
    - ou=users

    **Note:** ou=SWG,o=IBM,c=US in this example is an organization unit called
    SWG. SWG houses the OU=Users organization units to place the users that
    are created for Maximo Asset Management. DC=IBM and DC=COM would
    indicate a domain forest of ibm.com®. You can replace the example with the
    directory structure of your own organization.
    Define the following users and their positions within the ou=users DN's you
    created. These users are defined in order for Virtual Member Manager to be
    used to secure Maximo Asset Management.

    **Important:** Before you begin this procedure, create the following users in the
    root of your LDAP repository:

*Table 3. Base Maximo Asset Management users*

| User |
| --- |
| maxadmin |
| mxintadm |
| maxreg |

Here is an example of the default base LDIF data:
```
dn: o=ibm,c=us
objectClass: top
objectClass: organization
o: IBM

dn: ou=SWG, o=ibm,c=us
ou: SWG
```

```
objectClass: top
objectClass: organizationalUnit

dn: ou=users,ou=SWG, o=ibm,c=us
ou: users
objectClass: top
objectClass: organizationalUnit

dn: uid=maxadmin,ou=users,ou=SWG, o=ibm,c=us
userPassword: maxadmin
uid: maxadmin
objectClass: inetorgperson
objectClass: top
objectClass: person
objectClass: organizationalPerson
sn: maxadmin
cn: maxadmin

dn: uid=mxintadm,ou=users,ou=SWG, o=ibm,c=us
userPassword: mxintadm
uid: mxintadm
objectClass: inetorgperson
objectClass: top
objectClass: person
objectClass: organizationalPerson
sn: mxintadm
cn: mxintadm

dn: uid=maxreg,ou=users,ou=SWG, o=ibm,c=us
userPassword: maxreg
uid: maxreg
objectClass: inetorgperson
objectClass: top
objectClass: person
objectClass: organizationalPerson
sn: maxreg
cn: maxreg
```

**Note:** If you create the LDIF file on Windows, ensure that you remove the ^M characters from the file before you use it.

**Note:** Before you can import an LDIF file on UNIX systems, you must run the dos2unix command to format the file.

21. In the IBM Tivoli Directory Server Configuration tool, click **Import LDIF data**.
22. Click **Browse** to locate the LDIF file.
23. Click **Import**.
24. Close the IBM Tivoli Directory Server Configuration tool and restart the server.

## Manually configuring Microsoft Active Directory version 2012

You can manually configure Microsoft Active Directory for use with Maximo Asset Management.

### Before you begin

Manually configure Microsoft Active Directory for authorization and authentication within Maximo Asset Management.

**Important:** The users and passwords you create in the following steps must match this exact configuration for the deployment to succeed. It is sometimes necessary to temporarily alter theMicrosoft Active Directory security policy to create these

accounts in the required format. After installation, you can change these default passwords to conform to a stricter security policy.

## About this task

Before you start theMaximo Asset Management installation and configuration programs, configure Microsoft Active Directory. Complete the following steps

## Procedure

1. Add the Active Directory Domain Services role.
    a. Start the Server Manager.
    b. Click **Add roles and features**.
    c. From the Before You Begin page, click **Next**.
    d. For Installation Type, select **Role-based or feature-based installation**, and then click **Next**.
    e. Select the **Select a server from the server pool** option, and then choose your local server from the Server Pool list. Click **Next**
    f. In the Roles list, select **Active Directory Domain Services**.
    g. Click **Add Features**.
    h. Click **Next**.
    i. From the Select Features page , accept the defaults and click **Next**
    j. On the confirmation page, click **Install**
    k. Click **Close** when the installation is complete and then restart the server.
2. Promote the server to a Domain Controller.
    a. Start the Server Manager.
    b. Click **AD DS** in the dashboard.
    c. Click the **Configuration required for Active Directory Domain Services** warning indicator.
    d. From the All Servers Task Details and Notifications page, click the **Promote this server to a domain controller** action.
    e. From the Deployment Configuration page, select **Add a new forest**. Specify the root domain name with a unique fully qualified domain name, for example, `qawin01ldap.swg.usma.ibm.com`, and then click **Next**
    f. From the Domain Controller Options page, clear the **Domain Name System (DNS) server** option, and provide a password for Directory Services Restore Mode (DSRM). Click **Next**.
    g. Accept the default NetBIOS name and click **Next**.
    h. Accept the default paths for **AD DS** database, log files, and SYSVOL. Click **Next**.
    i. Review the summary, click **Next**, and then click **Install**.
    j. Restart the server

    When the system restarts, you must include the domain that you specified with your login credentials. In this example, a QAWIN01LDAP domain was created. Specify `QAWIN01LDAP\Administrator` as the user when you log in to the system.
3. Create containers, users, and groups for Maximo Asset Management.
    a. Select **Control Panel** > **Administrative Tools** > **Active Directory Users and Computers**.
    b. Select your server, right-click, and choose **New** > **Organizational Unit**.

c. Specify a name for the organizational unit, for example, SWG, and then click **OK**.

d. Select the new organizational unit that is located under your server, in this example, SWG, right-click, and choose **New** > **Organizational Unit**.

e. Specify a name for the organizational unit, for example, users, and then click **OK**.

f. Create another organizational unit under SWG and name it groups.

g. Right-click the **users** organizational unit, and click **New** > **User**.

h. Create a user who is called maxadmin. Specify maxadmin as the **First name** and **User logon name**. Click **Next**.

i. Provide a password for maxadmin. Click **Next** and then **Finish**.

j. Create two more users who are named mxintadm and maxreg.

k. Right-click the **groups** organizational unit and select and choose **New** > **Group**

l. Name the group maximousers and click **OK**.

m. Right-click the maximousers group and select **Properties**.

n. On the Members tab, click **Add**.

o. Type maxadmin in the **Enter the object names to select** field, and then click **OK**. The maxadmin user is now a member of the maximousers Members list.

p. Add the mxintadm and maxreg users to the maximousers group.

q. Click **Apply**, then **OK**.

### What to do next

Microsoft Active Directory configuration is finished and you are now ready to configure the J2EE server to use Microsoft Active Directory.

## Manual configuration of the J2EE server

Manually configuring an existing J2EE server for use by Maximo Asset Management.

Manual configuration of the J2EE server is required if you choose to deploy Maximo Asset Management with Oracle WebLogic Server. You must complete the manual configuration before you use the Maximo Asset Management installation and configuration programs.

Maximo Asset Management requires Java 7. Java 7 must be installed and configured on the J2EE server you want tu use with Maximo Asset Management.

### Manually configuring WebLogic Server
WebLogic Server can be used with Maximo Asset Management if it is manually configured.

### About this task

You can use a WebLogic Server as your J2EE server. The Maximo Asset Management installation process prepares EAR files on the Maximo Asset Management administrative workstation. These files are then manually deployed to your WebLogic Server. The Maximo Asset Management installation program

does not automatically configure a WebLogic Server environment It does not install or otherwise deploy Maximo Asset Management applications into a WebLogic Server server.

Manually configuring WebLogic Server consists of the following tasks:

**Creating the MAXIMOSERVER domain:**

This section details how to create the MAXIMOSERVER domain within WebLogic Server.

**About this task**

To create the MAXIMOSERVER domain within WebLogic Server, complete the following steps:

**Procedure**
1. Launch the Configuration wizard tool.
   - For Windows systems, select **Start** > **Programs** > **BEA Products** > **Tools** > **Configuration Wizard**.
   - For UNIX systems, launch the Configuration wizard using the `config.sh` command located in `<WebLogic_Home>/weblogic10/common/bin`.
2. From the welcome panel, select **Create a new WebLogic Server domain**, and then click **Next**.
3. From the Select a Domain Source panel, select **Generate a domain configured automatically to support the following products**, and then click **Next**.
4. From the Specify Domain Name and Location panel, name the domain MAXIMOSERVER, and then click **Next**
5. From the Configure Administrator Username and Password panel, enter an administrator user ID and password, and then click **Next**.
6. From the Configure Server Start Mode and JDK panel, enter the following information, and then click **Next**.

   **WebLogic Server Domain Startup Mode**
   > Select **Production**.
   >
   > Selecting this option requires that you provide a user ID and password in order to deploy applications.

   **JDK Selection**
   > Select a JDK with a version number 1.6.0 or greater from the supplied JDKs.
7. Proceed through the next few panels until you reach the Customize Environment and Services Settings panel. From the Customize Environment and Services Settings panel, select the default option **No**, and then click **Next**.
8. From the Create WebLogic Server Domain panel, change the Domain Name value to MaximoServer, and then click **Create** to create the domain.
9. Once the domain has been created, ensure that the **Start Admin Server** option is cleared, and then click **Done** to close the Configuration wizard. On UNIX systems, this option is not displayed.

**Configuring JMS options for integration framework:**

This procedure provides details on steps to configure JMS queues.

**Before you begin**

You need to be familiar with the Oracle WebLogic Server Administration Console to complete the following procedure. See your Oracle WebLogic Server documentation on how to navigate through the Oracle WebLogic Server Administration Console.

**Procedure**

1. Launch Internet Explorer and open the `MAXIMOSERVER` Administration Console by typing the following URL:

   `http://host_name:7001/console`

2. Sign in by entering your user ID and password.

3. Create three persistent file stores with the following configuration parameters:

| Option | Description |
|---|---|
| File Store Name | `sqinstore, sqoutstore, cqinstore` |
| File Store Target | `MAXIMOSERVER` |
| File Store Directory | `C:\WebLogic`, or another directory of your choice |

   File stores have better performance in general. However, you can choose to create database-based stores. To choose between file stores and database-based stores, consult your Oracle WebLogic Server documentation.

4. Create three JMS servers with the following configuration parameters:

| Option | Description |
|---|---|
| Properties Name | `sqinserver, sqoutserver, cqinserver` |
| Server Target | `MAXIMOSERVER` |
| Bytes Maximum | 20% or below of the allotted JVM heap size |

   It is recommended that you configure the Bytes Maximum for the JMS servers to a value of 20% or below of the allotted JVM heap size to prevent the Oracle WebLogic Server from running out of memory.

5. Create a JMS system module named `intjmsmodule` with MAXIMOSERVER as the target server.

6. Add a connection factory resource to the JMS module:

| Option | Description |
|---|---|
| Name | `intjmsconfact` |
| JNDI Name | jms/maximo/int/cf/intcf |
| Target | `MAXIMOSERVER` |

7. Add three queue resources to the JMS Module:
   a. Create the queue resource type.
   b. Create a new subdeployment with the same name as the queue.
   c. Assign the *subdeployment_name*server server target to the subdeployment.

| Option | Description |
|---|---|
| Queue 1 | Name: `sqin` JNDI Name: jms/maximo/int/queues/sqin |
| Queue 2 | Name: `sqout` JNDI Name: jms/maximo/int/queues/sqout |
| Queue 3 | Name: `cqin` JNDI Name: jms/maximo/int/queues/cqin |
| Template | None |

8. Enable the JMS connection factory for integration framework:

   a. Select the **intjmsconfact** link.

   b. From the **Transactions** tab, check the **XA Connection Factory Enabled** check box.

   c. Set the **Maximum Messages per Session** to **-1**.

9. Save and activate the changes.

**What to do next**

You need to stop and restart the MAXIMOSERVER application server for the update to take effect.

**Configuring JMS for the email listener in WebLogic Server:**

You use the email listeners application to receive and process requests through email in the form of email messages.

**About this task**

To configure JMS queues for the WebLogic Server, complete the following steps:

**Procedure**

1. Log in to the WebLogic Server administration console.

2. Create a persistent store.

   a. In the Domain Structure panel, expand **Services** and then click **Persistent Stores**.

   b. Click **New** and select the option to create a file store.

   c. Enter lsnrstore for the name of the file store and c:\wls10 as the directory, and then click **Finish**.

3. Create a JMS Server

   a. In the Domain Structure panel, under **Services** > **Messaging** , click **JMS Servers**, enter the following information, and then click **Next**.

      **Name**   Enter a name for a new JMS server, for example, lsnrserver.

      **Persistent Store**
           Select **lsnrstore**.

   b. For the **Target**, select **AdminServer**, and then click **Finish**.

4. Create a JMS System Module

   a. In the Domain Structure panel, under **Services** > **Messaging** , click **JMS Modules**, and then click **New**.

   b. Enter a value for the **Name** field, such as lsnrjmsmodule, and then click **Next**.

   c. Select the option for **AdminServer**, and then click **Next**.

   d. Select the option for **Would you like to add resources to the JMS system module?**, and then click **Finish**.

5. Add a connection factory to the JMS module.

   a. From the Summary of Resources table of the lsnrjmsmodule module, click **New** to add a resource.

   b. Select the **Connection Factory** option, and then click **Next**.

c. From the Connection Factory Properties panel, enter the following
      information, and then click **Next**.

   **Name**   Enter a name for the new connection factory. For example,
   `lsnrconnfact`.

   **JNDI Name**
   > Enter **jms/mro/lsnr/lsnrcf**.

   d. Ensure AdminServer is selected, and then click **Finish**

6. Add a queue to the JMS module.

   a. From the Summary of Resources table of the lsnrjmsmodule module, click
      **New** to add a resource.

   b. Select the **Queue** option, and then click **Next**.

   c. From the JMS Destination Properties panel, enter the following information,
      and then click **Next**.

   **Name**   Enter a name for the new queue. For example, `lsnrqueue`.

   **JNDI Name**
   > Enter **jms/mro/lsnr/qin**.

   **Template**
   > No template is necessary.

   d. Click **Create a new Subdeployment**, select `lsnrserver`, and then click
      **Finish**.

7. Set the XA option for the connection factory.

   a. Select `lsnrconnfact`, and then select the Transactions tab.

   b. Ensure that the XA Connection Factory Enabled option is selected, and then
      click **Save**.

8. Click **Activate Changes** to save and activate all changes.

9. To enable an email listener to use JMS queues, a Message Driven Bean is
   configured through a deployment descriptor file that is part of the installation.
   You configure the Message Driven Bean by removing comment lines from
   sections within the deployment descriptor files. Edit the `ejb-jar.xml` and
   `weblogic-ejb-jar.xml` files.

   a. Open the *install_dir*`\applications\maximo\mboejb\ejbmodule\META-INF\`
      `ejb-jar.xml` file in an editor, uncomment the following sections, and then
      save the file:

      ```
      Email Listener JMS is not deployed by default
      <message-driven id="MessageDriven_LSNRMessageBean">
      <ejb-name>LSNRMessageBean</ejb-name>
      ejb-class>psdi.common.emailstner.LSNRMessageBean</ejb-class>
      transaction-type>Container</transaction-type>
      <message-destination-type>javax.jms.Queue</messagedestination-
      type>
      </message-driven>
      ```

      ```
      Email Listener JMS is not deployed by default
      <container-transaction>
      <method>
      <ejb-name>LSNRMessageBean</ejb-name>
      <method-name>*</method-name>
      </method>
      <trans-attribute>Required</trans-attribute>
      </container-transaction>
      ```

   b. Open the `c:\ibm\smp\maximo\applications\maximo\mboejb\ejbmodule\META-`
      `INF\weblogic-ejb-jar.xml` file in an editor, uncomment the following
      section, and then save the file:

```
Email Listener JMS is not deployed by default
<weblogic-enterprise-bean>
<ejb-name>LSNRMessageBean</ejb-name>
<message-driven-descriptor>
<destination-jndi-name>jms/mro/lsnr/qin</destination-jndiname>
<connection-factory-jndi-name>jms/mro/lsnr/lsnrcf</
connection-factory-jndi-name>
</message-driven-descriptor>
<transaction-descriptor>
<trans-timeout-seconds>600</trans-timeout-seconds>
</transaction-descriptor>
<jndi-name>LSNRMessageBean</jndi-name>
</weblogic-enterprise-bean>
```

# Running the Maximo Asset Management 7.6 installation program

Use the Maximo Asset Management version 7.6 installation program to install Maximo Asset Management version 7.6.

## About this task

In order to install Maximo Asset Management version 7.6, run the Maximo Asset Management version 7.6 installation program on the Maximo Asset Management administrative system.

## Procedure

1. Log in to the Maximo Asset Management administrative workstation. If you run the IBM Maximo Asset Management version 7.6 installation program from a Linux or UNIX terminal window, you must be logged in as the root user.

2. Launch the IBM Maximo Asset Management version 7.6 installation program from the launchpad.

   a. Start the launchpad.

      **Windows**
      From the downloaded installation image, browse to the root directory and run the following command: `launchpad64.exe`.

      **Linux and UNIX**
      From the downloaded installation image, browse to the root directory and run the following command: `launchpad.sh`.

   b. Select a language for the installation session and click **OK**.

   c. In the launchpad navigation pane, click **Install Product**.

   d. From the **Install Product** panel, select **IBM Maximo Asset Management v7.6** and then click **Install IBM Maximo Asset Management components**.

3. In the package selection panel, click **Next**.

4. In the package prerequisite validation panel, review the results of the prerequisite check, and then click **Next**. If any errors are reported on this panel, resolve the issue and then click **Recheck Status** before continuing.

5. In the license agreement panel, review the license information for each package being installed, select **I accept the terms in the license agreements** if you agree with the terms, and then click **Next**.

6. In the Installation Manager installation location panel, specify path information for the shared resources directory and the Installation Manager installation directory, and then click **Next**. Installation Manager is the installation framework that is used to install Maximo Asset Management

version 7.6 components. The shared resources directory is a common workspace directory used by Installation Manager when installing packages.

7. In the package installation location panel, select the IBM Tivoli's process automation suite package group, specify the path information for the Maximo Asset Management version 7.6 installation directory, and then click **Next**.

8. In the package features panel, leave all default options checked, and then click **Next**.

9. In the package summary panel, review the information for the planned installation, and then click **Install**. If you install in a non-English environment, you might notice the environment summary is listed as English. You configure supported languages for Maximo Asset Management later with the Maximo Asset Management configuration program.

10. After the installation is complete, select the option to start the Maximo Asset Management version 7.6 configuration program, and then click **Finish**. The Maximo Asset Management version 7.6 installation program exits and the Maximo Asset Management version 7.6 configuration program is started automatically.

### What to do next

Use the Maximo Asset Management version 7.6 configuration program to configure Maximo Asset Management version 7.6.

## Configuring Maximo Asset Management version 7.6 with manually configured middleware

This procedure explains how to use the Maximo Asset Management version 7.6 configuration program to configure Maximo Asset Management version 7.6.

### Procedure

1. If the Maximo Asset Management version 7.6 configuration program is not open, start it from the Maximo Asset Management launchpad. In the launchpad navigation pane, click **Configure Product**, specify Maximo Asset Management installation location, and then click **Launch the Tivoli's Process Automation Suite configuration program**.

2. In the IBM Maximo Asset Management configuration operations page, click **Configure a New Deployment**.

3. In the Define Deployment Environment panel, specify information about DB2 and WebLogic Server After you define your deployment environment, click **Finish**.

4. In the Configure General Product Information panel, review summary details about the product components you are installing. Specify the appearance and navigation features for your product, add an e-mail address to receive workflow messages, and choose whether or not to deploy sample data.

5. In the Configure the DB2 Instance panel, specify information about the DB2 instance for Maximo Asset Management.

6. In the Configure the DB2 Database panel, specify information about the DB2 database for Maximo Asset Management.

7. In the Configure Application Security panel, Enter the user names and passwords for users that must be created for Maximo Asset Management.

8. Choose the base language and any additional languages you want to install.

9. In the Apply Deployment Operations panel, specify user interface preferences, select all available deployment operations, and then click **Finish**.

**Results**

Maximo Asset Management version 7.6 is installed and configured to use
WebSphere Application Server Network DeploymentWebLogic Server and DB2.

# Post installation for Oracle WebLogic Server deployments

There is a post installation step required for Oracle WebLogic Server deployments.

## About this task

When using Oracle WebLogic Server as the application server for your Maximo
Asset Management deployment, the installation program produces an EAR file.
You must then manually deploy that file to Oracle WebLogic Server.

When using Oracle WebLogic Server, you must modify the `weblogic.xml` file and
rebuild the `maximo.ear` file before deploying it.

## Procedure

1. Install the updated JDK 1.6 program.
   a. Obtain the latest JDK program with a version number 1.6.0 SR7 or greater,
      from http://www.oracle.com/technetwork/java/index.html
   b. Start the installation program.
   c. Accept the license agreement.
   d. Change the installation destination path to your Oracle home folder, and
      click **OK** , then **Next**.
   e. Accept any additional prompts to install the JDK program so that the
      installation process can complete successfully.
2. If you are using an Oracle database, copy the `oraclethin.jar` file.
   a. Copy the `oraclethin.jar` file from the administrative workstation to the
      server where you installed Oracle WebLogic Server. Use any file transfer
      protocol that is available.
   b. The `oraclethin.jar` file is in the *install_home*\maximo\applications\
      maximo\lib directory. The default *install_home* directory is
      C:\IBM\SMP\Maximo\Applications\Maximo\lib. Copy the `oraclethin.jar` file
      to the J2EE server and into the `wls11g/server/lib` directory. For example,
      `C:/Oracle/wls11g/server/lib`.
3. Edit the domain configuration file for appropriate Java settings.
   a. Navigate to *Install_dir*\user_projects\domains\mydomain\bin.
   b. Right-click and edit (do not double-click) `setDomainEnv.cmd` (Windows) or
      `setDomainEnv.sh` (UNIX).
   c. Using your text editor, search for and replace `jdk` entries with entries that
      configure Oracle WebLogic Server to use the updated JDK program. Two
      instances must be replaced.
   d. (Oracle only) Search for `PRE_CLASSPATH`. Set the parameter as follows:
      ```
      set PRE_CLASSPATH=%WL_HOME%\server\lib\oraclethin.jar (Windows)
      PRE_CLASSPATH={WL_HOME}/server/lib/oraclethin.jar (UNIX)
      ```

      **Note:** This step is required for connecting to Oracle databases from Oracle
      WebLogic Server.
   e. Increase the Java Heap parameters by searching for `MEM_ARGS` and setting
      the parameter as follows:

```
set MEM_ARGS=-Xms512m -Xmx1024m -XX:MaxPermSize=512m (Windows)
MEM_ARGS=-Xms512m -Xmx1024m -XX:MaxPermSize=512m (UNIX)
```

These parameters are case-sensitive. Ensure to include the minus (-) sign before each parameter and to include the m after each value.

f.  Search for `set JAVA_OPTIONS=%JAVA_OPTIONS% %JAVA_PROPERTIES%`. Insert an option directly after this string, as follows:

```
-Dcom.sun.xml.namespace.QName.useCompatibleSerialVersionUID=1.0
```

The entry looks like this example when complete:

```
set JAVA_OPTIONS=%JAVA_OPTIONS% %JAVA_PROPERTIES%
-Dcom.sun.xml.namespace.QName.useCompatibleSerialVersionUID=1.0
-Dwlw.iterativeDev=%iterativeDevFlag% -Dwlw.testConsole=%testConsoleFlag%
-Dwlw.logErrorsToConsole=%logErrorsToConsoleFlag%
```

g.  Save and close the `setDomainEnv.cmd` file.

4. Edit the `\IBM\SMP\maximo\applications\maximo\maximouiweb\webmodule\WEB-INF\weblogic.xml` file and add the following entries:

```
<session-descriptor>
<cookie-http-only>false</cookie-http-only>
</session-descriptor>
```

5. Start the server without being prompted for user name and password. If you are working in Production mode and do not want to be prompted for the Oracle WebLogic Server user name and password, create a `boot.properties` file. This procedure stores the user name and password in an encrypted format. Place the following two lines in a text file:

```
username=<administrator username>
password=<administrator password>
```

The user name and password values must match an existing user account in the Authentication provider for the default security realm. If you save the file as `boot.properties` and locate it in the `mydomain` folder (for example, `C:\BEA\User_projects\domains\mydomain`), the server automatically uses this file during its subsequent startup cycles. The first time you use this file to start a server, the server reads the file and then overwrites it with an encrypted version of the user name and password.

6. Verify settings by starting Oracle WebLogic Server. To ensure that the edits that you made to the previous Oracle WebLogic Server files were accurate, start the Oracle WebLogic Server.

7. Rebuild the `maximo.ear` file.

   `\IBM\SMP\maximo\deployment\buildmaximoear`

8. Deploy the `maximo.ear` file to Oracle WebLogic Server.

# Deploying ear files

You are now ready to deploy the EAR files from the Oracle WebLogic Server Administration Console. Deploying the EAR files starts the Maximo Application Server.

## Before you begin

`Windows` **Windows**

> If `MAXIMOSERVER` is not running, start it from a command prompt.

`UNIX` **UNIX**

> Ensure that the Oracle WebLogic Server daemon is running.

## About this task

In order to view the Oracle WebLogic Server Administration Console, you need to also install a Java Virtual Machine (JVM). Newer versions of Windows might not ship with JVM.

## Procedure

1. Change to the *my_domain* directory:

   **Windows** **Windows**
   
   From a command prompt, change directory path to:
   
   *drive_name*:\\*weblogic_install_dir*\\*user_projects*\\domain\\*my_domain*

   **UNIX** **UNIX**
   
   Open a Terminal window (or a command prompt, if you are telneting from a Windows computer) and change directory as follows:
   
   cd /home/mxadmin/*weblogic_install_dir*/*user_projects*/domains/*my_domain*

2. Type

   **Windows** **Windows**
   `startWebLogic.cmd`

   **UNIX** **UNIX**
   `./startWebLogic.sh`

   and press Enter. Once the phrase: `Server started in RUNNING mode` displays, the Oracle WebLogic Server is running.

3. Start the `MAXIMOSERVER` Oracle WebLogic Server Administration Console by typing the following URL in the address bar of your browser:
   
   `http://`*host_name*`:7001/console`

4. Log in to the administrative console with the administrative user name and password.

5. Click the **Lock&Edit** button in the **Change Center** section to set the server to edit mode.

6. Click the **Deployments** link in the **Domain Structure** section.

7. In the Summary of Deployments section, click the **Install** button. The Install Application Assistant opens.

8. Click the **upload your file(s)** link.

9. Click the **Browse** button next to the **Deployment Archive** field. Browse to where you have `maximo.ear` file installed (`c:/ibm/smp/maximo/deployment/default/` of the administrative workstation), select the file and click **Open**.

10. Click **Next** to upload the file to the Oracle WebLogic Server server.

11. Click the radio button next to the `maximo.ear` file, and **Next** to continue the deployment.

12. Accept the default value to install the deployment as an application and click **Next**.

13. Accept all other default values and click **Finish** to start the deployment process.

14. When the deployment process is complete, click the **Activate Changes** button in the Change Center to apply the changes. The following message displays: `All changes have been activated. No restarts are necessary.`

15. Repeat Steps 8 on page 32 through 14 on page 32 to deploy `maximoiehs.ear`.

16. Click the checkbox next to the `maximo` application.
17. Click the dropdown arrow of the **Start** button and select **Servicing all requests**. The Start Application Assistant displays.
18. Click **Yes** to start the system.

# Setting the Oracle WebLogic Server host name

Setting the Oracle WebLogic Server host name manually to access application help information.

## About this task

By default, the property used to store the host name for the Maximo Asset Management help system is set to `localhost`. The fully qualified host name of the Oracle WebLogic Server must be configured in the help system property to view Maximo Asset Management application help topics. Complete this step after you have deployed Maximo Asset Management EAR files to the Oracle WebLogic Server.

## Procedure

1. Log in to the Maximo Asset Management console as maxadmin.
2. Browse to **Go To** > **System Configuration** > **Platform Configuration** > **System Properties**
3. Using the Filter feature, search for the `mxe.help.host` Property Name.
4. Expand the `mxe.help.host` property and set the attribute to the fully qualified host name of the Oracle WebLogic Server.
5. Select the `mxe.help.host` record check box.
6. Click the Live Refresh icon in the toolbar.
7. From the Live Refresh dialog, click **OK**.

## Results

You are now able to access application help topics in the Maximo Asset Management user interface.

# Chapter 6. Silent installation

The Maximo Asset Management silent installation option interacts with the Maximo Asset Management installation program by using a command prompt and a response file.

You must install the Installation Manager software before you can silently install the Maximo Asset Management or related middleware packages

## Creating silent response files for installation

A response file records selections that you make and values that you provide when using the Maximo Asset Management installation program. The Maximo Asset Management installation program can then be run silently using the response file for input.

### Before you begin

You can create a silent installation response file by setting the **record** environment variable, and then starting the installation program through the Maximo Asset Management launchpad. The response file is generated in the home directory of the user that started the program. You can determine the home directory of the current user the command appropriate for your operating system by using the following commands.

**Windows**
        set HOMEPATH

**UNIX**
        echo $HOME

The name of the file is tpaeInstallerRsp*date_and_time_stamp*.xml, for example, tpaeInstallerRsp20130716105258.xml

### Procedure

1. Open a command prompt and set the **record** environment variable.

| Option | Description |
| --- | --- |
| Windows | set record=true |
| Linux or UNIX | export record=true |

2. Start the Maximo Asset Management launchpad.

| Option | Description |
| --- | --- |
| Windows | launchpad64.exe |
| Linux or UNIX | ./launchpad.sh |

3. Click **Install Product**.
4. Select the components that you want to install, and then click **Install** to start the Maximo Asset Management installation program.

5. Select the packages you want to install silently, and specifying the shared resources and installation directories. When you are satisfied with your installation choices, click **Install** to install the packages.

6. Click **Finish** to exit the Maximo Asset Management installation program.

# Sample silent installation response files

Instead of using the tpaeInstallerRsp*date_and_time_stamp*.xml file, you can use a sample response file and modify it according to your needs.

## Sample files

Sample response files are provided in the directory where you extracted the IBM Maximo Asset Management 7.6 launchpad. If you update the values that are used in the sample response files, limit your updates to parameters such as installation locations, host names, port numbers, and user names. If you need to update non-environmental values, such as the main product name or the Installation Manager repository, create your own silent response file with the Installation Manager software.

*Table 4. Samples of silent response files*

| File | Description |
|---|---|
| *launchpad_directory*\SilentResponseFiles\ Installer\Unix\ ResponseFile_MAM_Install_Unix.xml | Install Maximo Asset Management on Linux and UNIX. |
| *launchpad_directory*\SilentResponseFiles\ Installer\Unix\ ResponseFile_MAM_Uninstall_Unix.xml | Uninstall Maximo Asset Management on Linux and UNIX. |
| *launchpad_directory*\SilentResponseFiles\ Installer\Windows\ ResponseFile_MAM_Install_Windows.xml | Install Maximo Asset Management on Windows. |
| *launchpad_directory*\SilentResponseFiles\ Installer\Windows\ ResponseFile_MAM_Uninstall_Windows.xml | Uninstall Maximo Asset Management on Windows. |
| *launchpad_directory*\SilentResponseFiles\ Installer\Unix\ ResponseFile_DB2_Install_Unix.xml | Install IBM DB2 on Linux and UNIX. |
| *launchpad_directory*\SilentResponseFiles\ Installer\Windows\ ResponseFile_DB2_Install_Windows.xml | Install IBM DB2 on Windows. |
| *launchpad_directory*\SilentResponseFiles\ Installer\Unix\ ResponseFile_DB2_Uninstall_Unix.xml | Uninstall IBM DB2 on Linux and UNIX. |
| *launchpad_directory*\SilentResponseFiles\ Installer\Windows\ ResponseFile_DB2_Uninstall_Windows.xml | Uninstall IBM DB2 on Windows. |
| *launchpad_directory*\SilentResponseFiles\ Installer\Unix\ ResponseFile_TPAE_WAS_Install_Unix.xml | Install base components of WebSphere Application Server on Linux and UNIX. |
| *launchpad_directory*\SilentResponseFiles\ Installer\Windows\ ResponseFile_TPAE_WAS_Install_Windows.xml | Install base components of WebSphere Application Server on Windows. |

*Table 4. Samples of silent response files  (continued)*

| File | Description |
|------|-------------|
| *launchpad_directory*\SilentResponseFiles\ Installer\Unix\ ResponseFile_TPAE_WAS_Uninstall_Unix.xml | Uninstall base components of WebSphere Application Server on Linux and UNIX. |
| *launchpad_directory*\SilentResponseFiles\ Installer\Windows\ ResponseFile_TPAE_WAS_Uninstall_Windows.xml | Uninstall base components of WebSphere Application Server on Windows. |

# Updating password values in silent response files

The Maximo Asset Management installation program requires the encryption of password values that are in the response files. If you need to update encrypted password values in a response file, use the Installation Manager string encryption utility.

## About this task

The Installation Manager string encryption utility accepts a plain text password value and produces an encrypted version of that password. You must then copy and paste the encrypted value into the password field of the response file.

## Procedure

1. Start the Installation Manager string encryption utility from the command line.

| Option | Description |
|--------|-------------|
| **Windows** | c:\Program Files\IBM\ InstallationManager\eclipse\tools\ imutilsc *newpasswordvalue* |
| **AIX or Linux** | /opt/IBM/InstallationManager/eclipse/ tools/imutilsc encryptString *newpasswordvalue* |

   Replace *newpasswordvalue* with your new plain text password value.
2. Copy the string in the **Encrypted version of the string** field.
3. Open the response file in a text editor, replace the password value with the copied encrypted password, and then save the file.

# Silently installing the Installation Manager software

In order to install Maximo Asset Management and related middleware silently, the Installation Manager software must first be installed.

## Procedure

1. Open a command prompt on the system and change to the directory where you unzipped the launchpad files.
2. Change directory to the folder that contains the installation manager program for your operating system, for example, \Install\IM\installer.win64.
3. Optional: Open the silent-install.ini file in a text editor and update default values.
4. Start the Installation Manager silent installation process.

| Option | Description |
|---|---|
| **Windows** | `installc —acceptLicense` |
| **Other operating systems** | `./installc —acceptLicense` |

### Results

The Installation Manager software is successfully installed on the system. You can now proceed with silently installing Maximo Asset Management.

## Silently installing Maximo Asset Management and related middleware

You can deploy Maximo Asset Management and associated middleware silently to perform an unattended installation based on predefined settings.

### Before you begin

You must complete a successful installation to produce a response file that can be used to silently install Maximo Asset Management and related middleware. Alternatively, you can use one of the sample response files that are provided on the product media and modify the sample file.

If the response file was created on a different system, ensure that all values that are recorded in the response file are valid for the target system. Ensure that the path values that were defined when the response file was created are still valid for the target system. For example, you might need to update the **repository location** parameter values to indicate the correct path for product and middleware installation images.

If you are installing the product in a directory different from the path used during creation of the response file, update the **installLocation** parameter value. If you update the value for the **installLocation** parameter, you must also provide a new value for the **profile id** parameter.

### Procedure

1. Copy the response file that you generated earlier and the Maximo Asset Management files to the target system.
2. Open a command prompt and change directory to the location of the Installation Manager program that is appropriate for your operating system, for example, `cd /opt/IBM/InstallationManager/eclipse/tools/`.
3. Silently start the Installation Manager program.

   ```
   imcl -input responsefilename
   -log logfilename-acceptLicense
   ```

   The **-input** parameter determines the response file to use, for example, `/SI/tpaeInstallerRsp20130716105258.xml`. The **-log** parameter determines where log files are written, for example, `/tmp/linuxDB2WASMAM_Log.xml`. You must include the full path when you specify the response and log files.

### Results

The installation proceeds silently with the values that you provided in the response file. Using different response files as input, you can silently install middleware on separate systems. You can then use another response file to silently install Maximo

Asset Management on the system that is designated as the administrative workstation.

# Silently uninstalling Maximo Asset Management and related middleware

You can silently uninstall Maximo Asset Management and associated middleware based on settings that were used during the installation.

## Procedure

1. Open a command prompt and change to the *launchpad_home*\ `SilentResponseFiles\Installer` directory, which contains the sample uninstallation response file.
2. Update the sample uninstallation response file with values that are used in your environment.
3. Silently start the Maximo Asset Management uninstallation program.

   ```
   imcl -input responsefilename
   -log logfilename-acceptLicense
   ```

   The default home directory for the **imcl** command is `c:\Program Files\IBM\Installation Manager\eclipse\tools\` on Windows systems and `/opt/IBM/InstallationManager/eclipse/` on Linux and UNIX systems.

   The **-input** parameter determines the response file to use, for example, `C:\tmp\install_response.xml`.

   The **-log** parameter determines where log files are written, for example, `C:\tmp\silent.log`.

   You must include the full path when you specify the response and log files. The **-acceptLicense** parameter is used to accept the license automatically.
4. You can remove the Installation Manager software from the system using the uninstall program.

   ```
   uninstallc
   ```

*Table 5. Location of* `uninstallc` *command*

| Windows | | Other operating systems |
|---|---|---|
| Admin | | Admin |
| **64-bit** | `C:\ProgramData\IBM\Installation Manager\uninstall` | `var/ibm/InstallationManager/uninstall` |
| User | | User |
| **64-bit** | `C:\Users\`*User ID*`\AppData\ Roaming\IBM\Installation Manager\uninstall` | *User Home*`/var/ibm/InstallationManager/ uninstall` |

# Chapter 7. Silent configuration

The silent configuration option allows you to interact with the Maximo Asset Management configuration program by using a command prompt and a response file.

The following tasks are required for silent configuration:

1. "Creating silent response files for middleware configuration"
2. "Creating a response file for silent Maximo Asset Management configuration" on page 66
3. "Silently configuring middleware" on page 67
4. "Silently configuring Maximo Asset Management" on page 67

## Creating silent response files for middleware configuration

You create a response file by modifying one of the sample response files that is provided. The configuration program can then be run silently using the response file for input.

### Procedure

1. Create a copy of the sample file, for example, `my-cfg-silent-win.properties`. Sample response files are in *launchpad_home*`\SilentResponseFiles\ConfigTool`.
2. Open the response file in a text editor, update values as required, and save the file.

   Ensure that all values listed in the response file are valid for the target system, particularly path values. For example, you might need to update the `WAS.InstallLocation` parameter value to indicate the correct path for WebSphere Application Server Network Deployment relative to the target system.

   If there are any Maximo Asset Management components you do not want to automatically configure, set the appropriate property value to `false`. For example, if you do not want to configure WebSphere Application Server Network Deployment on the system, set the `WAS.ND.AutomateConfig` value to `false`.

### What to do next

After you have completed silent middleware configuration, you can proceed with silently configuring Maximo Asset Management.

**Related tasks**:

"Silently configuring middleware" on page 67
Middleware used with Maximo Asset Management can be configured silently by using a configuration response file.

## Sample silent middleware configuration response files

Several sample response files are provided on the IBM Maximo Asset Management7.6 product image. You can use a sample response file and modify it according to your needs.

### Sample files

Silent response files are in the *launchpad_home*\SilentResponseFiles\ConfigTool directory of the product image.

*Table 6. Samples of silent response files for middleware*

| File | Description |
|------|-------------|
| cfg-silent-aix.properties | Configure Maximo Asset Management middleware on AIX. |
| cfg-silent-linux.properties | Configure Maximo Asset Management middleware on Linux. |

# Creating a response file for silent Maximo Asset Management configuration

You create a response file by modifying one of the sample response files provided according to your needs. The configuration program can then be run silently using the response file for input.

### Before you begin

There are several sample response files provided on the IBM Maximo Asset Management product image. The files are in the directory where you uncompressed the product image under the \SilentResponseFile\ConfigTool directory.

### About this task

When you run the configuration program in silent mode, it uses the data in the response file to configure the product.

### Procedure

1. Create a copy of the sample file and name it something appropriate, for example, my-maximocfg-silent-win.properties.
2. Open the response file in a text editor, update values as required, and save the file.

   Ensure that all values listed in the response file are valid for the target system, particularly path values. For example, you might need to update the **WAS.InstallLocation** parameter value to indicate the correct path for WebSphere Application Server Network Deployment relative to the target system.

   If there are any Maximo Asset Management components you do not want to automatically configure, set the appropriate property value to false. For example, if you do not want to configure WebSphere Application Server Network Deployment virtual member manager on the system, set the **WAS.VmmFullyAutomatedConfig** value to false.

**Related tasks**:

"Silently configuring Maximo Asset Management" on page 67
Maximo Asset Management can be configured silently by using an configuration response file.

# Silently configuring middleware

Middleware used with Maximo Asset Management can be configured silently by using a configuration response file.

### Procedure

1. Copy the response file you created earlier to the target system.
2. Open a command prompt and start the configuration program silently.

```
reconfigurePae
-action deployMiddlewareConfiguration
-inputFile responsefilename
```

The *responsefilename* parameter determines the response file to use, for example, `C:\tmp\my-cfg-silent.properties`. You must include the full path when you specify the response and log files.

### Results

The configuration proceeds silently with the values that you provided in the response file.

### What to do next

You can modify the silent configuration response file, copy it to another server, and perform further configuration. For example, you can use the configuration program to silently configure WebSphere Application Server Network Deployment on one server, and then modify the response file to configure IBM Tivoli Directory Server on another server.

# Silently configuring Maximo Asset Management

Maximo Asset Management can be configured silently by using an configuration response file.

### Procedure

1. Copy the response file you created earlier to the target system.
2. Open a command prompt and start the Maximo Asset Management configuration program silently.

```
reconfigurePae
-action deployConfiguration
-inputFile responsefilename middlewarecomponents
```

The *responsefilename* parameter determines the response file to use, for example, `C:\tmp\my-cfg-silent.properties`. The *middlewarecomponents* parameter indicates which middleware component to configure, for example, `-dbvendor DB2` or `-j2eevendor WebSphere`. In most cases, you configure middleware components together. You must include the full path when you specify the response files.

### Results

The configuration proceeds silently with the values that you provided in the response file.

### What to do next

You can modify the silent configuration response file, copy it to another server, and perform further configuration. For example, you can use the configuration program to silently configure WebSphere Application Server Network Deployment on one server, and then modify the response file to configure IBM Tivoli Directory Server on another server.

## Silently remove middleware configuration

You can use the same file you used to silently configure middleware to remove the configuration information.

### About this task

Maximo Asset Management middleware configuration can be removed silently using the same configuration response file that was used to configure the product previously. You must edit the configuration response file that was used to configure the product previously before it can be used to remove configuration information from the product.

### Procedure

1. Change to the directory containing the response file used to silently configure previously.
2. Make a copy of the response file and rename it. For example, rename the file to `my-uncfg-silent.properties`.
3. Open the file in a text editor and change the **MW.Operation** property value to **unconfigure**, and then save the file.
4. Open a command prompt and start the Maximo Asset Management unconfiguration program silently.

   ```
   reconfigurePae
   -action deployMiddlewareConfiguration
   -inputFile responsefilename
   ```

   The *responsefilename* parameter determines the response file to use, for example, `C:\tmp\my-uncfg-silent.properties`. You must include the full path when you specify the response file.

## Remove product configuration silently

You can use the **removeConfiguration** action of the **reconfigurePae** command to remove Maximo Asset Management configuration information.

### About this task

The **removeConfiguration** action is used to remove Maximo Asset Management configuration from the database and application server that was performed by the configuration tool. This action is equivalent to running the **removeJ2EEConfiguration** and **removeDatabaseConfiguration** actions together.

### Procedure

1. Log on to the Maximo Asset Management administrative workstation.
2. Open a command prompt and change directory to *Install_Home*\ConfigTool\ scripts.

3. Start the Maximo Asset Management configuration program silently.

```
reconfigurePae -action removeConfiguration
-dbuser userid
-dbpwd password
-wasuser userid
-waspwd password
```

The **-dbuser** and **-dbpwd** parameters are the user ID and password used to configure the database. The **-wasuser** and **-waspwd** parameters are the user ID and password used to configure IBM WebSphere Application Server Network Deployment.

# Chapter 8. Programmatically verifying that the installation was successful

After you have exited the Maximo Asset Management installation program without encountering any errors, you can verify that the installation completed successfully. The installation is programmatically verified through the use of post installation validation utilities.

## Before you begin

The *JAVA_HOME* environment variable must be set on the system before you can use the verification utilities.

## About this task

During installation, the Maximo Asset Management installation program performs a simple health check. This health check consists of logging in to the application to verify availability. This health check might not be sufficient to verify a successful installation in all deployment scenarios. For example, if, during installation, you select the option to deploy the application EAR file at a later date, this health check cannot verify the installation. This simple health check is also insufficient for verifying an upgrade.

Post installation validation utilities are available after the product has been deployed.

The installValidation utility is used to validate the product installation. These command-line utilities are used to verify the installation and configuration of the product in a more complete manner. These utilities can also be used to verify an existing deployment after changes in the environment, such as changes to host names, user IDs, and passwords.

Results of the installValidation utility are logged in `installValidationTrace00.log`. This log is found in the `\smp\configtools\logs` directory on the administrative workstation.

## Procedure

1. Log in to the server using the user ID that was used to run the installation program. If you intend to use the installValidation utility, log in to the administrative workstation.
2. Ensure the *JAVA_HOME* environment variable is set properly to the location of a JRE 1.7 installation.
3. To verify product installation, from the administrative workstation, change directory to `ibm\smp\configtool\scripts` and run the `installValidation.[sh|bat]` command. For either command, use the parameters in Table 1.

*Table 7. Verification utilities parameters*

| Program | Syntax parameters | Input |
|---|---|---|
| installValidation.bat | **-action** | Use **-action** validate to start the validation of the product installation.<br><br>The **-action** parameter is the only mandatory parameter when using the installValidation utility. |
|  | **-trace** | Verification progress information is written to the screen during the verification process. You can modify the output using the **-tracing** parameter.<br>• minimal<br>  Progress information is limited to error information.<br>• normal<br>  Information includes individual test progress, individual test results, and overall progress of the entire verification process.<br>  This is the default mode of tracing.<br>• verbose<br>  In addition to providing normal progress information, verbose tracing includes the test class name, test method name, and corrective action information. |
|  | **-confirm** | You are prompted before each test is started. You must confirm each test to be performed. There is no input for this parameter. |
|  | **-component** | You can provide a comma-delimited list of directories containing the test objects to limit testing to a specific set of tests.<br><br>These test objects must be located in the \ibm\smp\HealthValidation\ directory. For the product installation program, only the CTGIN folder is provided. |
|  | **-dbuser** | Provide the user ID used to access the database. |
|  | **-dbpwd** | Provide the password of the user ID used to access the database. |
|  | **-maxuser** | Provide the Maximo Asset Management administrator user ID. For example, maxadmin. |
|  | **-maxpwd** | Provide the password for the Maximo Asset Management administrator user ID. |

For example,

```
installValidation.bat -action validate
```

**Results**

The `installValidation.bat` validation utility results are logged in `ctginstallvalidationtrace00.log`. This log is found in the `\smp\configtools\logs` directory on the administrative workstation.

# Chapter 9. Verifying the installation manually

After you exit the Maximo Asset Management installation program without errors, you can further verify that the installation completed successfully. The installation is manually verified by locating key artifacts and performing a few simple tasks.

## Before you begin

The product installation program performs installation verification, or a health check, by logging on to the product before exiting the installation. The product installation program returns a success message if all installation steps, including the product log in task, are successful. If the health check task is unsuccessful, a `HealthCheckFailed` error is thrown for the `getUrlContentString` function. This error is in the *install_home*\configtools\logs\CTGConfiguration*XX*.log file.

## About this task

To further verify that the Maximo Asset Management installation was completed successfully, perform the following verification procedures:

## Procedure

1. Examine the CTGConfiguration*XX*.log files for success messages.

   The following **success** messages are found in the CTGConfigurationTrace*XX*.log file and can be used to verify which phases of the installation were successful:

   - CTGIN2114I: The database was created successfully
   - CTGIN2135I: Tablespace maxdata created successfully
   - CTGIN2135I: Tablespace maxtemp created successfully
   - CTGIN2079I: process automation engine database configuration completed successfully (This message indicates that maxinst finished successfully.)
   - CTGIN2253I: buildmaximoear.cmd completed successfully
   - CTGIN2253I: buildhelpear.cmd completed successfully
   - CTGIN2208I: runConfigurationStep completed successfully
   - CTGIN2370I: The installation finished successfully

   Search the CTGConfigurationTrace*XX*.log file for the following message, which indicates that the maxinst task completed successfully:

   `CTGIN2079I, maxinst.*completed`

2. Compare the packages chosen for installation with the packages that were installed. The Package Summary panel of the Maximo Asset Management installation program provides a summary of all the packages to be installed. You can compare the contents of that panel with a listing of what was installed on the system by using the configuration tool and selecting the update database task. The resulting list matches the packages listed in the Package Summary panel.

3. Log in to the product and verify that applications are present. Being able to log in to the product manually is a good indicator that the installation was successful. Once you have logged in, navigate through the **GoTo** menu to verify that it has been populated with the applications you expect. If you experience access problems with the product, clear the cache of your browser and try again.

# Chapter 10. Installation and configuration programs log information

Log files generated by the Maximo Asset Management and configuration programs can provide details on completed and configuration tasks and any errors that might occur.

## Maximo Asset Management installation and configuration programs log locations

Maximo Asset Management and associated middleware is installed through Installation Manager. The log files of Installation Manager are common to any installation, regardless of which components you choose to install. Log files for the Maximo Asset Management installation program are found on the system where it is run.

*Table 8. Log file directories*

| Log file directory |
| --- |
| `C:\ProgramData\IBM\Installation Manager\logs` |
| `/var/ibm/InstallationManager/logs` |
| `C:\IBM\SMP\ConfigTool\logs` |
| `opt/IBM/SMP/ConfigTool/logs` |

When engaging IBM product support services, be prepared to provide these log files in an archive file. The LogZipper utility provided in the *install_home\*`configtool\scripts` directory can be used for this task. If you use the LogZipper utility, all relevant log files are archived in *install_home*`/configtool/debug/`*YYYYMMDD_hhmmss*`.z`ip.

# Chapter 11. Post installation tasks

There are several post installation tasks required in order to complete the Maximo Asset Management deployment.

## Accessing product documentation

Accessing the Maximo Asset Management Knowledge Center documentation and product online help.

If you manually configured the application server after you ran the product installation program, you must set the property for the product documentation server. For more information, see https://www.ibm.com/support/docview.wss?uid=swg21508594.

## Installing help information in a new language

Help information is available in multiple languages. If you need to support multiple languages for Maximo Asset Management, you optionally install the help information for other languages. Before you can use multi-language help information in Maximo Asset Management, you must first install it using the configuration program.

### About this task

Multi-language help information for Maximo Asset Management is available on Fix Central as a fix pack that includes multiple language plug-ins. The fix pack must be downloaded from Fix Central and then applied with the Maximo Asset Management configuration program.

### Procedure

1. Download the fix pack that contains the multi-language Maximo Asset Management documentation from the Fix Central support site to a temporary location on the administrative workstation.
2. Start the Maximo Asset Management launchpad.
3. Click **Configure Product**.
4. Select the location of the Maximo Asset Management installation you want to configure, and then click **Launch the Tivoli's Process Automation Suite configuration program**.
5. From the Configuration Actions page, click **Add Additional Help Plug-ins**.
6. Click **Browse** to locate the fix pack file that you downloaded.
7. Optional: Select **Deploy the product help EAR to the application server** to install the updated help information to Maximo Asset Management now. If you do not select this option, the Maximo Asset Management help information EAR file is rebuilt, but not deployed to the application server.
8. Click **Finish** to rebuild the Maximo Asset Management help information EAR file. If you selected the **Deploy the product help EAR to the application server** option, the help information EAR file is deployed to the application server.

# Configuring initial data

After you successfully install and configure Maximo Asset Management components, you must complete several data configuration tasks before you use Maximo Asset Management.

## Before you begin

- Create security groups and add users to the security groups.
- If you are using a directory server as part of your deployment, ensure that the user names are created in your LDAP repository.
- To ensure that you have the accounting information that you need, consult the Financial department of your organization.

## Procedure

1. Log in to Maximo Asset Management by entering the user name `maxadmin` and the password that you specified during the installation.

2. If you didn't set the SMTP server by using the Configuration tool, configure the SMTP server to send users email notifications of system events. To learn about the range of system properties in Maximo Asset Management, see the System properties information.

   a. From the **Go To** menu, select **System Configuration** > **Platform Configuration** > **System Properties**.

   b. In the Global Properties table window, use the filter feature to search for and expand the `mail.smtp.host` property.

   c. In the **Global Value** field, specify the SMTP host name.

   d. Select the `mail.smtp.host` check box.

   e. From the **Common Actions** menu, click **Live Refresh**.

   f. In the Live Refresh window, click **OK**.

   g. In the Global Properties table window, use the filter feature to search for the `mxe.adminEmail` property.

   h. Expand the `mxe.adminEmail` property and in the **Global Value** field, specify your email address.

   i. Select the `mxe.adminEmail` record check box.

   j. From the **Common Actions** menu, click **Live Refresh**.

   k. In the Live Refresh window, click **OK**.

3. Define a currency code.

   a. From the **Go To** menu, select **Financial** > **Currency Codes**

   b. Click **New Row** and specify a currency code and a description. For example, enter `USD` for United States of America Dollar.

   c. Click **Save Currency**.

4. Define item sets and company sets.

   a. From the **Go To** menu, select **Administration** > **Sets**.

   b. Click **New Row**.

   c. Specify an item set name. For example, enter `IT Items`.

   d. In the **Type** field, specify `ITEM`.

   e. Click **New Row**.

   f. Specify a company set name. For example, enter `IT Comps`.

   g. In the **Type** field, specify `COMPANY`.

   h. Click **Save Sets**.

5. Create an organization.
   a. From the **Go To** menu, select **Administration** > **Organizations**.
   b. From the toolbar, click **New Organization**.
   c. Specify an organization name. For example, enter EAGLENA, and enter a long description.
   d. Specify the base currency that you defined in step 3.
   e. Specify the item set and the company set that you defined in step 4.
   f. In the **Default Item Status** field, set the status to PENDING.
   g. Click the **Sites** tab and select **New Row** under the Sites table window.
   h. Specify a site name. For example, enter Factory01 and enter a long description.
   i. Click **Save Organization**.
6. Create a general ledger account component.
   a. From the **Go To** menu, select **System Configuration** > **Platform Configuration** > **Database Configuration**.
   b. From the **More Actions** menu, select **GL Account Configuration**.
   c. Click **New Row**.
   d. Specify a component name. For example, enter MYCOMPONENT.
   e. Specify a length for the component. For example, enter 5.
   f. Specify a type for the component. For example, select Alphanumeric.
   g. Click **OK**.
7. Apply changes to the database.
   a. From the **More Actions** menu, select **Manage Admin Mode**.
   b. Select **Turn Admin Mode ON** and click **OK**. This task takes several minutes to complete. You can click **Refresh Status** to view the progress.
   c. From the **More Actions** menu, select **Apply Configuration Changes**. Ensure that the status To Be Changed does not appear in the status column of the listed objects.
   d. From the **More Actions** menu, select **Manage Admin Mode**.
   e. Select **Turn Admin Mode OFF** and click **OK**. If you do not turn off Admin Mode, cron tasks fail.
8. Create a general ledger account.
   a. From the **Go To** menu, select **Financial** > **Chart of Accounts**.
   b. In the Organizations table window, select your organization.
   c. From the **More Actions** menu, select **GL Component Maintenance**.
   d. In the Components table window, select the component that you entered in step 6 and click **New Row**.
   e. Specify a GL Component value and a description, and then click **OK**.
   f. In the GL Accounts table window, click **New Row**.
   g. Specify a General Ledger Account and click **Save GL Account**.
   h. From the **Go To** menu, select **Administration** > **Organizations**.
   i. Find your organization and select the record.
   j. In the **Clearing Account** field, specify the general ledger account that you created.
   k. Select **Active**.
   l. Click **Save Organization**.
9. Authorize a security group to modify a general ledger component type.

a. From the **Go To** menu, select **Security** > **Security Groups**.

b. Select the group that provides authorization. For example, select **FINANCE**.

c. Click the **GL Components** tab.

d. For each GL Component that is listed, select the **Authorized** check box. As a shortcut, you can select **Authorize Group to Change All GL Component Types**.

e. Click **Save Group**.

10. Update the company-related accounts.

a. From the **Go To** menu, select **Financials** > **Chart of Accounts**.

b. Select your organization and from the **More Actions** menu, select **Company-Related Accounts**.

c. Click **New Row** and specify the company type `Courier`.

d. Enter an account number in the **RBNI Account**, **AP Suspense Account**, and **AP Control Account** fields. You can specify the same account number in each field.

e. Click **OK**.

f. From the **More Actions** menu, select **Update Database** and click **OK**.

11. Create a default insert site.

a. From the **Go To** menu, select **Security** > **Users**.

b. Find **maxadmin** and select the record.

c. In the **Default Insert Site** field, specify the site name that you created in step 5.

d. In the **Storeroom Site for Self-Service Requisitions** field, specify the same site name.

e. Click **Save User**.

12. Define work types. Work types denote the importance of the work task.

a. From the **Go To** menu, select **Administration** > **Organizations**.

b. Find your organization and select the record.

c. From the **More Actions** menu, select **Work Order Options** > **Work Type**.

d. Click **New Row**.

e. In the **Work Order Class** field, select `WORKORDER`.

f. Specify a **Work Type**. For example, enter `MAJOR`.

g. Set the **Start Status** to **In Progress**.

h. Set the **Complete Status** to **Completed**.

i. Click **New Row** and repeat steps f-i to create another work order class with a different work type. For example, enter `MINOR`.

j. Click **New Row** and repeat steps f-i to create a `CHANGE` work order class with a different work type. For example, enter `SIG` to represent a significant change.

k. Click **OK** and click **Save Organization**.

**Related information**:

Integrating data with external applications

System properties

# Synchronizing users and groups

When you select application server security, the scheduled synchronization of users and groups that occurs between LDAP repositories and Maximo Asset Management is governed by the federated repositories.

## Before you begin

View the cron task configuration information in the Administering section of the Knowledge Center.

## About this task

LDAP repositories are synchronized with Oracle WebLogic Server using the LDAP cron task.

To configure the synchronization schedule between LDAP repositories and Maximo Asset Management, complete the following steps:

## Procedure

1. Open a web browser and point to http://*host name:port*/maximo.
2. Log in to Maximo Asset Management using the maxadmin user ID.
3. From the Maximo Asset Management interface, navigate to **Go To** > **System Configuration** > **Platform Configuration** > **Cron Task Setup**.
4. Search for the appropriate cron task in the **Cron Task** field and configure it.
5. Set the task to **active**.

## What to do next

By default, the cron task performs its task every 5 minutes. Change the **Schedule** field of the cron task if you want to change the interval.

# Adding sample data to the Maximo Asset Management database post configuration

Using the `maxinst` command, you can add sample data to the Maximo Asset Management database after the initial installation and configuration of Maximo Asset Management.

## Before you begin

The `maxinst` command does not provide default values for table space parameters. You must specify the data table space name and the index table space name that you specified during the installation. The `maxinst` command replaces the current database schema and data. Do not add sample data to the Maximo Asset Management database if you want to preserve records currently stored in the database.

## About this task

If you created a database either automatically or manually during the installation, you can use the `maxinst` command to create a sample data in that database instance.

## Procedure

1. Back up the existing database.
2. Unzip the *install_home*\IBM\SMP\Maximo\tools\maximo\en\maxdemo.*dbtype*.zip file. Replace the *dbtype* variable with the extension that is appropriate for the type of database you used for the Maximo Asset Management deployment. Available options are ora, sqs, or db2.
3. Change the *install_home*\IBM\SMP\Maximo\tools\maximo\en\maxdemo.*dbtype* file to the name of the Maximo Asset Management database you configured during the initial deployment. For example, rename the file to max76db.db2
4. Open a command prompt and change to the *install_home*\IBM\SMP\Maximo\ tools\maximo directory.
5. Populate the database with sample data. You must specify the table space name parameters when you run the maxinst command.

    maxinst -s*tablespacename* -t*tablespacename* -imax76db For example, for Oracle Database or for DB2, enter maxinst -sMAXDATA -tMAXDATA -imax76db. For SQL Server, enter maxinst -tPRIMARY -sPRIMARY -imax76db. The system reads the maximo.properties file for database connectivity information. The maximo.properties file is in the *install_home*\IBM\SMP\Maximo\Applications\ Maximo\Properties directory.
6. Populate the database by running commands with specific parameter values. Use the following **maxinst** database parameters:

| Parameter | Description |
|---|---|
| **-a** | Database alias. If not specified, the alias mxe.db.url.property is used. |
| **-d** | Log file directory. If you are using the -l parameter, the log file is sent to the specified directory. Otherwise, the log file is sent to the log directory, for example *install_home*\IBM\SMP\Maximo\tools\ maximo\logs. |
| **-e** | Runs the SQL. This parameter is required and already present in the maxinst.bat file. |
| **-f** | File name for the properties file. If not specified, maximo.properties is used. |
| **-i** | File name of the input file (without path or extension). If not specified, the default file name Unlcvt is used. |
| **-k** | Directory of the properties file. |
| **-l** | Creates a detailed log file. This parameter is already present in the maxinst.bat file. |
| **-m2** | Optional value for SQL Server: Multiplies the column width for strings by 2. This value is required for double-byte character sets. |
| **-o** | If you are using the -l parameter, the -o parameter specifies the file name for the log file. |
| **-p** | Password for the database connection. If not specified, the mxe.db.password property or MAXIMO is used. If MAXIMO is used, it must be entered in uppercase letters. |

| Parameter | Description |
|-----------|-------------|
| -s | Required value: Table space for index storage. |
| -t | Required value: Table space for table storage. |
| -u | User name for database connection. If not specified, the mxe.db.user property or MAXIMO is used. If MAXIMO is used, it must be entered in uppercase letters. |
| -x | Required value for UNIX: Fixes the doclink file separators in UNIX environments. Note: If a UNIX environment is deployed without using this parameter, the attached documents do not function properly. |
| -y | Multiplies the column width for strings by 2. This value is required for double-byte character sets. |

7. From the *install_home*\IBM\SMP\ETC directory, open the install.properties
   file and specify the installation-related properties. You can add these properties
   to the database from the System Properties application.

## Results

The Maximo Asset Management database is populated with sample data.

If an error occurs when you run the **maxinst** command, check the log file in the
*install_home*\IBM\SMP\Maximo\tools\maximo\logs directory. If the **maxinst**
command fails, you must correct the error and rerun the **maxinst** command to
complete the Maximo database schema deployment before you can start the
application.

.

# Chapter 12. Installing Version 7.5 industry solutions

Version 7.5 of industry solutions can be installed on Maximo Asset Management 7.6.

Version 7.5 of industry solutions can be installed on Maximo Asset Management 7.6 using several methods. The industry solution can be installed from the launchpad if one is included with the industry solution package. The industry solution can also be installed from the command-line. Silent installation is also supported.

Ensure that you download the latest version and fixes available for the industry solution or add-on before you install it. Refer to the Maximo Upgrade Resources support page for the latest information about industry solution and add-on requirements.

➦ http://www.ibm.com/support/docview.wss?uid=swg21266217

## Installing version 7.5 industry solutions from the launchpad

If a launchpad is included with an industry solution package, you can use it to install the industry solution on Maximo Asset Management 7.6.

### Procedure

1. Log on to the administrative workstation.
2. Create a backup image of your Maximo Asset Management 7.6 administrative workstation. When you install version 7.5 industry solutions in Maximo Asset Management 7.6, you do not have an automated uninstallation method that returns the system to its previous state.
3. Open the archive for your industry solution package, and start the launchpad.
4. Click **Install** from the menu of options.
5. Follow the steps in the installation program.
6. When the installation is complete, click **Done**. After the industry solution installation program exits, the Maximo Asset Management 7.6 configuration tool starts.
7. From the Deployment Operations panel, select the options for applying changes to the database, and building and deploying application EAR files to complete the deployment.

## Installing version 7.5 industry solutions from the command line

Version 7.5 industry solutions can be installed in Maximo Asset Management 7.6 from the command line.

### Procedure

1. Log on to the administrative workstation.
2. Create a backup image of your Maximo Asset Management 7.6 administrative workstation. When you install version 7.5 industry solutions in Maximo Asset Management 7.6, you do not have an automated uninstallation method that returns the system to its previous state.

3. Start the process solution installation program by changing to the *install_home*\bin directory of your Maximo Asset Management 7.6 installation and use the **solutionInstaller** command.

   ```
   solutioninstaller.bat -pkgpath path_to_industry_solution_package
   -license accept
   ```

   The -pkgpath option specifies the path to the industry solution package. Replace *path_to_industry_solution_package* with the path to the industry solution package file you are installing. The -license option is used to accept license terms. For example,

   ```
   solutioninstaller.bat -pkgpath
   C:\TEMP\Spatial75_launchpad\Install\SPATIAL\Spatial75.zip
    -license accept
   ```

4. Follow the steps in the installation program.

5. When the installation is complete, click **Done**. After the industry solution installation program exits, the Maximo Asset Management 7.6 configuration tool starts.

6. From the Deployment Operations panel, select the options for applying changes to the database, and building and deploying application EAR files to complete the deployment.

# Installing version 7.5 industry solutions silently

Version 7.5 industry solutions can be installed in Maximo Asset Management 7.6 silently from the command line.

## Procedure

1. Log on to the administrative workstation.

2. Create a backup image of your Maximo Asset Management 7.6 administrative workstation. When you install version 7.5 industry solutions in Maximo Asset Management 7.6, you do not have an automated uninstallation method that returns the system to its previous state.

3. Start the process solution installation program by changing to the *install_home*\bin directory of your Maximo Asset Management 7.6 installation and use the **solutionInstallerGUI** command.

   ```
   solutioninstallerGUI.bat -pkgpath path_to_industry_solution_package -nl locale
   -S -license accept
   ```

   The -pkgpath option specifies the path to the industry solution package. Replace *path_to_industry_solution_package* with the path to the industry solution package file you are installing. The -nl option specifies the language that is used during the installation. Replace *locale* with the abbreviation for the language to use. The -s option indicates that the installation is processed silently. The -license option is used to accept license terms. For example,

   ```
   solutioninstallerGUI.bat -pkgpath
   C:\TEMP\Spatial75_launchpad\Install\SPATIAL\Spatial75.zip
   -nl es -S -license accept
   ```

   .

4. Follow the steps in the installation program.

5. When the installation is complete, click **Done**. After the industry solution installation program exits, configure it with the Maximo Asset Management 7.6 configuration tool.

6. From the Deployment Operations panel, select the options for applying changes to the database, and building and deploying application EAR files to complete the deployment.

# Chapter 13. Configuring and managing Oracle WebLogic Server

This section describes common tasks on configuring and administering the Oracle WebLogic Server.

For more comprehensive information on running and administering Oracle WebLogic Server visit the Oracle Web site.

## Starting Oracle WebLogic Server

Starting the Oracle WebLogic Server.

### About this task

To start the `MAXIMOSERVER` application, complete the following steps:

### Procedure

1. Open a command prompt window and change the directory to:

   <kbd>Windows</kbd> **Windows:**
   `C:\bea\user_projects\domains\mydomain`

   <kbd>UNIX</kbd> **UNIX:**
   `weblogic_install_dir/user_projects/domains/mydomain`

2. Launch:

   <kbd>Windows</kbd> **Windows:**
   `startWebLogic.cmd`

   <kbd>UNIX</kbd> **UNIX:**
   `./startWebLogic.sh`

3. If prompted, type the administrator user name and password. Once you see the phrase: `server started in RUNNING Mode` or `Started weblogic Admin Server MAXIMOSERVER for domain` *mydomain* `running in production mode` (depending on your operating system), the Oracle WebLogic Server is running.

## Stopping the Oracle WebLogic Server

You can use the Administrative Console to stop the Oracle WebLogic Server.

### About this task

To stop the MAXIMOSERVER from the Administrative Console, complete the following steps:

### Procedure

1. Open a browser window and enter the URL: http://*localhost*:7001/console This URL is the administration URL for Oracle WebLogic Server domain you created.
2. Log in to the Administration Console with the administrator user name and password.

3. In the Domain Structure section, click **Environment** > **Servers**.
4. In the main section of the console, click the **AdminServer** link.
5. Click the **Control** tab.
6. Click **Shutdown** and then **Force Shutdown Now**. The MAXIMOSERVER is stopped.

# Chapter 14. Maintaining EAR files

This section contains information about maintaining Maximo Asset Management EAR files.

The following instructions are used to manually build or uninstall the Maximo Asset Management `maximo.ear` and `maximoiehs.ear` files on Oracle WebLogic Server.

- If you modify any database connection parameters in the `maximo.properties` file after the initial installation, you must rebuild of the `maximo.ear` file, and then redeploy it in Oracle WebLogic Server. In this scenario, you would likely only rebuild and redeploy the `maximo.ear` file. You would not be required to rebuild and redeploy the `maximoiehs.ear` file.

- If you have installed Maximo Asset Management into a development environment, you might at some point like to migrate the deployment into a test or production environment. In this scenario, you must deploy both the maximo and maximo help applications into the new environment.

## Building EAR files

You can manually build Maximo Asset Management EAR files if, for example, you modify a database connection parameter in the `maximo.properties` file after the initial installation.

### About this task

To manually build Maximo Asset Management EAR files, complete the following steps:

### Procedure

1. Build the maximo EAR file:

   **Windows**
   
   > *install_home*\maximo\deployment\buildmaximoear.cmd

   **Linux and UNIX**
   
   > *install_home*/maximo/deployment/buildmaximoear.sh

2. Build the maximo help system EAR file:

   **Windows**
   
   > *install_home*\maximo\deployment\buildmxiehsear.cmd

   **Linux and UNIX**
   
   > *install_home*/maximo/deployment/buildmxiehsear.sh

## Manually uninstalling applications from Oracle WebLogic Server

This section contains information about uninstalling Maximo Asset Management applications from Oracle WebLogic Server.

## Manually uninstalling applications from Oracle WebLogic Server

### Procedure

1. Log into the Administration Console. For example, http://<server>:<port>/console
2. Click **Lock & Edit**.
3. Click **Deployments**.
4. Select the application for the EAR to uninstall.
5. Click Delete.

# Chapter 15. Command line configuration

Maximo Asset Management includes a configuration program that is accessed from the command line. This command line configuration program performs the same duties as the Maximo Asset Management configuration program user interface. Command line configuration is used to quickly update configuration values when you choose to not use the configuration tool user interface.

When you install a fix pack or upgrade Maximo Asset Management, the configuration program uses values that are recorded from the previous deployment. These values are stored on the administrative system in the `install.properties` and `maximo.properties` files that are located under the *install_home*\etc and *install_home*\maximo\applications\maximo\properties directories.

If you make any environmental changes to any of the systems that are used for the original deployment, they must be recorded in the `install.properties` and `maximo.properties` files. Typically, these types of changes would include changing a host name or updating a password, for example.

You can update these values manually, or you can use the configuration command line interface tool to update configuration values for your existing deployment when they change. Properties that are specified for the configuration command line interface tool, either as parameters, or in an input properties file, replace existing properties in the `maximo.properties` and `install.properties` files. Properties are encrypted when they are saved.

You can complete the following tasks with the configuration command line interface tool:

- Change the security model for the original deployment. For example, you can migrate from Maximo Asset Management-based security to WebSphere Application Server security. This option is not available for Oracle WebLogic Server.
- Confirm that updated property values conform to required input domains. Validate credentials, host names, and available ports.
- Update database or application server configuration settings without the server available. These values cannot be validated in this scenario.
- Clone an existing environment and use the configuration command line interface tool to reconfigure the administrative workstation to use the new servers. The cloned environment must use the same directory structure as the original.
- Create an environment (non-cloned) and use the configuration command line interface tool to create all required artifacts database, table spaces, and application server objects, for example.
- Remove configuration from a previously configured database.

## Configuration parameter validation

Use the Maximo Asset Management command line configuration program to validate input in the same way the Maximo Asset Management configuration program validates input in the user interface.

You can validate the following items with the Maximo Asset Management command line configuration program.

- Host names or IP addresses are correctly formatted and are reachable.
- User IDs and passwords meet length and character set criteria.
- Specified credentials are used to authenticate to WebSphere Application Server.
- Specified port values are listening on the corresponding host. Port values contain numbers only and are within a valid range.
- Middleware installation directories that are supplied exist.
- Preconfigured middleware can authenticate to the product database.
- Required objects such as JMS queues, SIB destinations, databases, and table spaces exist.
- Required WebSphere Application Server virtual member manager users exist.
- Sufficient authority is available to create WebSphere Application Server virtual member manager users.
- Remote login credentials are valid.
- There is enough space in the supplied directories to create database objects.

## Input properties files

You can use property files as input for the Maximo Asset Management command-line configuration program.

The following sample input property files are provided with the Maximo Asset Management command-line configuration program.

- DB2_Sample_input.properties
- Oracle_Sample_input.properties
- SQLServer_Sample_input.properties
- WebSphere_App_Server_Sample_input.properties

These sample files are in the ibm\smp\configtools\samples directory. Each of these sample files contains properties that are associated with a specific database or application server type. They contain comments and properties you can update with values from your environment. You can then use the updated file as input for the configuration command line interface tool with the **–input** parameter.

All input properties, introduced from either the command line or in the input properties file, are validated before they are used. After validation is successful, the maximo.properties and install.properties files are updated with the values you provided.

# Command line interface parameters

The following parameters can be used with the configuration command line program to configure your environment.

*Table 9. Maximo Asset Management command line configuration program parameters*

| Parameter name | Description |
|---|---|
| **-action** | Configuration action.<br><br>• **addLanguages** – Adds additional languages to a current deployment.<br><br>• **configureProducts** - Configures more products after an initial installation and configuration. This action can configure multiple products after they are installed and can also be used in an upgrade scenario.<br><br>• **deployConfiguration** - Configures initial installation. This action can configure multiple products after they are installed. This action is used for new installations only.<br><br>• **deployDatabaseConfiguration** - Validates command input and defines configuration property values for a new database for the current product.<br><br>• **deployJ2eeConfiguration** - Validates command input and defines configuration property values for a new application server for the current product.<br><br>• **deployMiddlewareConfiguration** – Configures initial middleware installation.<br><br>• **disableAppSecurity** - Disables application security for the application. Security is handled entirely through Maximo Asset Management. This action sets the **mxe.useAppServerSecurity** property to a value of 0. This value is written to the database when updated.<br><br>• **disableDB2TextSearch** - Enable the full text search feature on a DB2 database.<br><br>• **enableAppSecurity** – Enables application security for the application. This action sets the **mxe.useAppServerSecurity** property to a value of 1, and updates the **mxe.ldapUserManagement** flag. This value is written to the database when the database is updated.<br><br>• **enableDB2TextSearch** – Enables the full text search feature on a DB2 database.<br><br>• **reapplyAppServerTuning** – Reapplies tuning configuration to an application server.<br><br>• **removeDatabaseConfiguration** – Removes product configuration information from a previously configured database.<br><br>• **removeJ2EEConfiguration** – Removes application server configuration that was added by the configuration tool.<br><br>• **updateApplication** – Updates the Maximo Asset Management application by updating the database and redeploying application EAR files. |

*Table 9. Maximo Asset Management command line configuration program parameters  (continued)*

| Parameter name | Description |
|---|---|
| | • **updateDatabaseConfiguration** – Updates existing database configuration values. |
| | • **updateJ2eeConfiguration** - Updates existing application server configuration values. |
| | • **validateAndUpdateDatabaseConfiguration** – Validates reconfiguration command line interface tool input and then updates existing database configuration property values. |
| | • **validateAndUpdateJ2eeConfiguration** - Validates reconfiguration command line interface tool input and then updates existing database configuration property values. |
| | • **validateDatabaseConfiguration** - Validates database configuration values specified as input for the reconfiguration command line interface tool. |
| | • **validateJ2eeConfiguration** - Validates application server configuration values that are specified as input for the reconfiguration command line interface tool. |
| | • **removeConfiguration** - Removes database and application configuration performed by the configuration tool. |
| | The actions that modify the security setting for Maximo Asset Management also updates Maximo Asset Managementweb.xml files. |
| **-additionalLangs** | Adds one or more additional languages to the deployment. |
| | This parameter requires one or more locale abbreviations for input. For example, ES for Spanish. |
| **-allowUpdateDB** | The database update task is run to add the languages that are specified to the database. |
| **-applicationServerName** | Name of the application server. |
| **-applicationServerNode** | Name of the application server node. |
| **-automatedbconfig** | Automates the configuration of the database. Use this parameter to create the database instance, database users, the database, table spaces, and creates database schema. |
| **-automatej2eeconfig** | Automates the configuration of WebSphere Application Server. Using this parameter creates, for example, JMS queues and profiles. |
| **-buildAndDeployEAR** | Rebuilds and deploys application EAR files. Application EAR files must be rebuilt and redeployed for configuration changes to take effect in the application. |
| **-buildears** | Rebuilds application EAR files. |

| Parameter name | Description |
|---|---|
| **-bypassJ2eeValidation** | Skips the validation and configuration of the application server.<br><br>If you use this parameter, then you must manually configure the application server before you can deploy Maximo Asset Management.<br><br>Validation of a manually configured application server requires logging in to the application server with application server administrative credentials. If you do not want to provide those credentials to the configuration program, you can skip the validation task.<br><br>If you use the **bypassJ2eeValidation** parameter, you cannot also use the following parameters:<br>• **applicationServerName**<br>• **applicationServerNode**<br>• **automatej2eeconfig**<br>• **buildAndDeployEAR**<br>• **createResourcesIfMissing**<br>• **deployhelpear**<br>• **deploymaximoear**<br>• **enableappsecurity**<br>• **enableEnhancedNavigation**<br>• **enableSkin**<br>• **inputfile**<br>• **j2eeserverport**<br>• **usermanagement**<br>• **wasuser**<br>• **waspwd** |
| **-bypassUpdateDB** | Specifies that the database update task is not run and the installation program does not update the database.<br><br>Languages files for the locales that are specified are built into the application EAR file.<br><br>Use this parameter if you have run the database update task and included the language files in the database through another method. |
| **-createResourcesIfMissing** | Creates any resources that are not found during configuration. |
| **-db2_english_only** | Specifies that alphanumeric data is stored as a `varchar` data type. Using this parameter limits you to using English in the user interface.<br><br>If you do not use this parameter, alphanumeric data is stored as a `vargraphic` data type. |
| **-db2textsearchport** | Port that is used by the DB2 server for full text search. |
| **-dbname** | Name of the database. |
| **-dbpwd** | Password for the user ID that accesses the database. |
| **-dbrxapwd** | Password for the user ID used to access a remote middleware server. |
| **-dbrxauser** | User ID used to access a remote middleware server. |

*Table 9. Maximo Asset Management command line configuration program parameters  (continued)*

| Parameter name | Description |
|---|---|
| **-dbserverhost** | Host name of the database server. |
| **-dbserverport** | Port name of the database server. |
| **-dbuser** | User ID that accesses the database. |
| **-dbvendor** | Database software name.<br>• DB2<br>• Oracle<br>• SQLServer |
| **-deleteInstanceUsersAndGroups** | Removes the instance owner and database user IDs and associated groups from the DB2 server. This parameter has cannot be used with and Microsoft SQL Server.<br><br>The **deleteInstanceUsersAndGroups** parameter requires the **dbrxauser** and **dbrxapwd** and the **dbuser** and **dbpwd** parameters. |
| **-deleteWASprofiles** | Deletes the WebSphere Application Server profiles that are created for the application. |
| **-deployDemoData** | Includes sample data. |
| **-deployhelpear** | Deploys Maximo help application EAR file. |
| **-deploymaximoear** | Deploys Maximo application EAR file. |
| **-enableappsecurity** | Enables application security for the application. This parameter works the same as the enableAppSecurity action.<br><br>This action sets the **mxe.useAppServerSecurity** property to a value of 1, and updates the **mxe.ldapUserManagement** flag. This value is written to the database when updated. |
| **-enableEnhancedNavigation** | Enables enhanced navigational elements in the user interface, including enhancements for switching between applications and viewing record lists.<br><br>The **enableEnhancedNavigation** parameter must be used with the **enableSkin** parameter.<br><br>If you enable enhanced navigation, and then you enable a different user interface skin with the –**enableSkin** parameter, you must re-enable enhanced navigation. |
| **-enableMultiTenancy** | Enables the multitenancy feature.<br><br>Multitenancy refers to software or services that can be configured to serve multiple groups of clients referred to as tenants. Each group of tenants is served by an instance of the software that uses data that only they can access. |

*Table 9. Maximo Asset Management command line configuration program parameters  (continued)*

| Parameter name | Description |
|---|---|
| **-enableRestrictivePasswordPolicy** | Enables a more restrictive password policy for Maximo Asset Management users.<br>• Login attempts<br>  Standard policy allows 10 login attempts before the account is locked.<br>  Restrictive policy allows 3 login attempts before the account is locked.<br>• Number of times that the forgotten password link can be used<br>  Standard policy allows 5 uses of the forgotten password link before locking the account.<br>  Restrictive policy allows 3 uses of the forgotten password link before locking the account.<br>• Password duration (days)<br>  Standard policy passwords never expire.<br>  Restrictive policy passwords expire after 90 days.<br>• Password expiration warning (days)<br>  Standard policy does not warn the user that the account password is about to expire.<br>  Restrictive policy warns the user that the account password is about to expire seven days before it expires.<br>• Password threshold<br>  Standard policy does not enforce a password threshold.<br>  Restrictive policy enforces a password threshold of 270 characters. |
|  | • Password minimum character length<br>  Standard policy passwords must be at least six characters.<br>  Restrictive policy passwords must be at least eight characters.<br>• Limit of allowed consecutive characters<br>  Standard policy has no restrictions to the amount of times a character can be repeated consecutively in a password.<br>  Restrictive policy allows a character to be repeated two times consecutively in a password value.<br>• Password can be same value as user name<br>  Standard policy allows you to use the same value for the user name and password.<br>  Restrictive policy does not allow you to use the same value for the user name and password.<br>• Number of numeric characters required in the password<br>  Standard policy does not require numeric characters in a password value.<br>  Restrictive policy requires one or more numeric characters in a password value.<br>• Number of alphabetic characters required in the password<br>  Standard policy does not require alphabetic characters in a password value.<br>  Restrictive policy requires one or more alphabetic characters in a password value. |

*Table 9. Maximo Asset Management command line configuration program parameters  (continued)*

| Parameter name | Description |
|---|---|
| **-enableSkin** | Change the appearance of the user interface.<br><br>• Classic<br>  Removes settings for the following properties:<br>  – **mxe.webclient.homeButtonHeaders**<br>  – **mxe.webclient.systemNavBar**<br>  – **mxe.webclient.tabBreadCrumbs**<br>  – **mxe.webclient.verticalLabels**<br>  – **mxe.webclient.skin**<br>• tivoli09<br>  Removes settings for the following properties:<br>  – **mxe.webclient.homeButtonHeaders**<br>  – **mxe.webclient.systemNavBar**<br>  – **mxe.webclient.tabBreadCrumbs**<br>  – **mxe.webclient.verticalLabels**<br>  Sets the following property:<br>  – **mxe.webclient.skin=tivoli09**<br>• tivoli13<br>  Sets the following properties to a value of 1:<br>  – **mxe.webclient.homeButtonHeaders**<br>  – **mxe.webclient.systemNavBar**<br>  – **mxe.webclient.tabBreadCrumbs**<br>  – **mxe.webclient.verticalLabels**<br>  Sets the following property:<br>  – **mxe.webclient.skin=tivoli13** |
| **-force** | Indicates that an input property file updates the properties file and skips the validation of parameters input to the command line tool through an input properties file.<br><br>When you use this parameter, you are not prompted for confirmation of the property update task. |
| **-inputfile** | Fully qualified path to the properties file that is used to set properties in an environment.<br><br>Values that are specified as command line parameters for the configuration command line interface tool supersede values from the input file. |
| **-j2eeserverhost** | Host name of the domain manager of the application server.<br><br>If deployment manager host name or server port is specified the **thinwsadmin** scripts are updated accordingly. |
| **-j2eeserverport** | Port name of the application server domain manager server.<br><br>If deployment manager host name or server port is specified the **thinwsadmin** scripts are updated accordingly. |
| **-j2eevendor** | Application server software.<br>• WebSphere<br>• WebLogic |
| **-mtadminpwd** | Password for the user ID of the multitenancy configuration manager. |

*Table 9. Maximo Asset Management command line configuration program parameters (continued)*

| Parameter name | Description |
|---|---|
| **-mtadminuser** | User ID of the multitenancy configuration manager. |
| **-mtdbpwd** | Password for the user ID of the multitenancy configuration manager that is used to connect to the database. |
| **-mtdbuser** | User ID of the multitenancy configuration manager that is used to connect to the database. |
| **-removeCreatedArtifacts** | Removes any artifacts that are created by the configuration tool. Examples include directories that are created by the configuration program, JMS objects, and the application server. |
| **-removeInstance** | Removes the database instance for DB2 and Oracle. This parameter cannot be used with Microsoft SQL Server.<br><br>The **removeInstance** parameter requires the **dbrxauser** and **dbrxapwd**, and the **dbuser** and **dbpwd** parameters. |
| **-setjdbcurl** | Sets the JDBC connection URL string. |
| **-stopAppServer** | The application server must be stopped when you add languages. If you do not use the **stopAppServer** parameter with the **addLanguages** action, you must stop the application server manually before you use the **addLanguages** action.<br><br>This parameter is used for WebSphere Application Server only. |
| **-unsetjdbcurl** | Removes the currently used JDBC connection URL string. |
| **-updatedb** | Runs the **updatedb** configuration task. |
| **-usermanagement** | Changes the security configuration of the existing Maximo deployment.<br>• j2ee<br>   Change the security setting in Maximo so that both Maximo users and Maximo groups are managed through application server security mechanisms.<br>• mixed<br>   Manage users through application server security mechanisms, and manage groups using Maximo. |
| **-validateForNewDeploy** | Validate middleware configuration.<br><br>When this parameter is specified, validation is performed as if the tool was configuring the middleware. Additional configuration details are checked, including disk space availability and remote access credentials.<br><br>If this parameter is not specified, validation is performed in the context of a component that is already established in the environment. |
| **-validateUsers** | Used with the **enableAppSecurity** action to validate users exist in the LDAP repository.<br><br>If the users do not exist, the existing security model is not modified. |
| **-waspwd** | WebSphere Application Server administrator user ID password. |
| **-wasrxapwd** | Password for user ID to access remote WebSphere Application Server server host. |

*Table 9. Maximo Asset Management command line configuration program parameters (continued)*

| Parameter name | Description |
|---|---|
| -wasrxauser | Operating system user ID used to access remote WebSphere Application Server host. Typically this user ID is defined either as Administrator or root. |
| -wasuser | WebSphere Application Server administrator user ID.<br><br>This is the user ID that is used to log in to the WebSphere Application Server administrative client application. Typically, this user ID is defined as wasadmin. |

# Command line configuration program actions

Use the command line configuration program to configure new or existing deployments of Maximo Asset Management.

## action updateDatabaseConfiguration

The **updateDatabaseConfiguration** action is used to update existing database configuration values.

```
-action updateDatabaseConfiguration [-force] [-buildAndDeployEAR]
[-dbserverhost hostname] [-dbserverport port number]
[-dbname database name] [-dbuser userid]
[-dbpwd password] [-dbrxauser userid]
[-dbrxapwd password] [-setjdbcurl jdbcurl]
[-unsetjdbcurl] [-inputfile path to input properties file]
```

## action validateDatabaseConfiguration

Use the **validateDatabaseConfiguration** action to validate current database configuration properties. This action can be used to verify database properties before they are updated, or following an update action.

```
-action validateDatabaseConfiguration [-validateForNewDeploy]
[-dbserverhost hostname]
[-dbserverport port number]
[-dbname database name]
[-dbuser userid] [-dbpwd password]
[-dbrxauser userid]
[-dbrxapwd password] [-setjdbcurl jdbcurl]
[-unsetjdbcurl] [-enableMultiTenancy]
[-mtadminuser userid] [-mtadminpwd password]
[-mtdbuser userid] [-mtdbpwd password]
[-inputfile path to input properties file]
```

The following properties are required if the **-validateForNewDeploy** parameter is used.

- **Database.DB2.InstanceAdminGroup**
- **Database.DB2.InstanceAdminUserName** (Windows only)
- **Database.DB2.InstanceAdminPassword** (Windows only)
- **Database.DB2.FencedUser**
- **Database.DB2.FencedUserPassword**
- **Database.DB2.FencedGroupName**
- **Database.DB2.InstanceUserPassword**

These properties must be added manually to the install.properties file.

## action validateAndUpdateDatabaseConfiguration

The **validateAndUpdateDatabaseConfiguration** action is used to both validate reconfiguration command-line interface tool input and then update existing database configuration property values.

```
-action validateAndUpdateDatabaseConfiguration [-validateForNewDeploy] [-force]
[-buildAndDeployEAR] [-dbserverhost hostname]
[-dbserverport port number] [-dbname database name]
[-dbuser userid] [-dbpwd password]
[-dbrxauser userid] [-dbrxapwd password]
[-setjdbcurl jdbcurl] [-unsetjdbcurl] [-enableMultiTenancy]
[-mtadminuser userid] [-mtadminpwd password] [-mtdbuser userid]
[-mtdbpwd password] [-inputfile path to input properties file]
```

The following properties are required if the **-validateForNewDeploy** parameter is used.

- **Database.DB2.InstanceAdminGroup**
- **Database.DB2.InstanceAdminUserName** (Windows only)
- **Database.DB2.InstanceAdminPassword** (Windows only)
- **Database.DB2.FencedUser**
- **Database.DB2.FencedUserPassword**
- **Database.DB2.FencedGroupName**
- **Database.DB2.InstanceUserPassword**

These properties that must be added manually to the install.properties file.

The **-enableMultiTenancy** parameter must be used with the **-validateForNewDeploy** parameter.

## action deployDatabaseConfiguration

The **deployDatabaseConfiguration** action is used to validate command-line configuration program input and define configuration property values for a new database for the currently deployed product. This action can be run after a successful **deployConfiguration** action.

```
-action deployDatabaseConfiguration [-buildAndDeployEAR] [-createResourcesIfMissing]
[-dbserverhost hostname] [-dbserverport port number]
\[-dbname database name] [-dbuser userid]
[-dbpwd password] [-dbrxauser userid] [-dbrxapwd password]
[-setjdbcurl jdbcurl] [-unsetjdbcurl] [-enableMultiTenancy]
[-mtadminuser userid] [-mtadminpwd password] [-mtdbuser userid]
[-mtdbpwd password] [-deployDemoData]
[-inputfile path to input properties file] [-enableRestrictivePasswordPolicy]
```

The following properties are required if the **-createResourcesIfMissing** parameter is used.

- **Database.DB2.InstanceAdminGroup**
- **Database.DB2.InstanceAdminUserName** (Windows only)
- **Database.DB2.InstanceAdminPassword** (Windows only)
- **Database.DB2.FencedUser**
- **Database.DB2.FencedUserPassword**
- **Database.DB2.FencedGroupName**
- **Database.DB2.InstanceUserPassword**

These properties that must be added manually to the install.properties file.

## action removeDatabaseConfiguration

The **removeDatabaseConfiguration** action is used to remove configuration settings from a previously configured database.

```
-action removeDatabaseConfiguration [-removeInstance] [-dbrxauser userid]
[-dbrxapwd password]  [-dbuser userid]
[-dbpwd password]  [-deleteInstanceUsersAndGroups]
```

## action enableDB2TextSearch

The **enableDB2TextSearch** action is used to enable the full text search feature for a DB2 database.

```
-action enableDB2TextSearch [-db2textsearchport port number]
```

## action updateJ2eeConfiguration

The **updateJ2eeConfiguration** action is used to update existing application server configuration values.

```
-action updateJ2eeConfiguration [-force] [-buildAndDeployEAR]
[-j2eeserverhost hostname] [-j2eeserverport port number]
[-wasuser userid] [-waspwd password]
[-applicationServerNode nodeName] [-applicationServerName appServerName]
[-inputfile path to input properties file]
```

## action validateJ2eeConfiguration

The **validateJ2eeConfiguration** action is used to validate application server configuration values specified as input for the reconfiguration command-line interface tool.

```
-action validateJ2eeConfiguration [-validateForNewDeploy] [-j2eeserverhost hostname]
[-j2eeserverport port number] [-wasuser userid]
[-waspwd password] [-applicationServerNode nodeName]
[-applicationServerName appServerName] [-inputfile path to input properties file]
```

The following properties are required if the **–validateForNewDeploy** parameter is used and the **WAS.SibPersistMessages** property is set to true.

- **Database.DB2.FencedUser**
- **Database.DB2.FencedUserPassword**
- **Database.DB2.InstanceUserPassword**

These properties that must be added manually to the install.properties file.

## action validateAndUpdateJ2eeConfiguration

The **validateAndUpdateJ2eeConfiguration** action is used to both validate reconfiguration command-line interface tool input and then update existing application server configuration property values.

```
-action validateAndUpdateJ2eeConfiguration [-force] [-validateForNewDeploy]
[-buildAndDeployEAR] [-j2eeserverhost hostname]
[-j2eeserverport port number] [-wasuser userid]
[-waspwd password] [-applicationServerNode nodeName]
[-applicationServerName appServerName]
[-inputfile path to input properties file]
```

The following properties are required if the **–validateForNewDeploy** parameter is used and the **WAS.SibPersistMessages** property is set to true.

- **Database.DB2.FencedUser**
- **Database.DB2.FencedUserPassword**

- `Database.DB2.InstanceUserPassword`

These properties must be added manually to the `install.properties` file.

## action deployJ2eeConfiguration

The **deployJ2eeConfiguration** action is used to validate command line configuration program input and define configuration property values for a new application server for the currently deployed product. This action can be run after a successful **deployConfiguration** action.

```
-action deployJ2eeConfiguration [-bypassJ2eeValidation] [-buildAndDeployEAR]
[-createResourcesIfMissing] [-j2eeserverhost hostname]
[-j2eeserverport port number] [-wasuser userid]
[-waspwd password] [-applicationServerNode nodeName]
[-applicationServerName appServerName]
[-inputfile path to input properties file]
[—enableSkin Classic|tivoli09|tivoli13] [-enableEnhancedNavigation]
```

The following properties are required if the **-validateForNewDeploy** parameter is used and the **WAS.SibPersistMessages** property is set to `true`.

- `Database.DB2.FencedUser`
- `Database.DB2.FencedUserPassword`
- `Database.DB2.InstanceUserPassword`

These properties that must be added manually to the `install.properties` file.

## action removeJ2EEConfiguration

The **removeJ2EEConfiguration** action is used to remove configuration settings from a previously configured application server. Items removed include domain manager and application server profiles, and JMS objects.

```
-action removeJ2EEConfiguration [-wasuser userid] [waspwd password]
[-wasrxauser userid] [-wasrxapwd password]
[-removeCreatedArtifacts] [-deleteWasProfiles]
```

## action deployConfiguration

Use the **deployConfiguration** action to configure the initial installation. This action is used for new Maximo Asset Management installation scenarios only. This action can configure multiple middleware products after they are installed. The **deployConfiguration** action requires the **inputfile** parameter and an input properties file populated with the configuration values you want to use.

```
-action deployConfiguration [-inputfile path to input properties file]
[-dbvendor Oracle|DB2|SQLServer] [-j2eevendor WebSphere|WebLogic]
[-bypassJ2eeValidation][-automatedbconfig] [-automatej2eeconfig]
[-usermanagement j2ee|mixed] [-buildears] [-deploymaximoear] [-deployhelpear]
[-enableappsecurity] [-deployDemoData] [-enableMultiTenancy] [-db2_english_only]
[—enableSkin Classic|tivoli09|tivoli13] [-enableEnhancedNavigation]
[-enableRestrictivePasswordPolicy]
```

## action removeConfiguration

The **removeConfiguration** action is used to removes database and application configuration that is performed by the configuration tool. This action is equivalent to running the **removeJ2EEConfiguration** and **removeDatabaseConfiguration** actions together

```
-action removeConfiguration -dbuser userid
-dbpwd password -wasuser userid
-waspwd password [-wasrxauser userid]
```

```
[-wasrxapwd password] [-dbrxauser userid]
[-dbrxapwd password] [-removeInstance] [-deleteInstanceUsersAndGroups]
[-removeCreatedArtifacts] [-deleteWasProfiles]
[-inputfile path to input properties file]
```

## action reapplyAppServerTuning

The action is used to reapply application server tuning.

```
-action reapplyAppServerTuning [-wasuser userid]
[-waspwd password] [-applicationServerNode nodeName]
[-applicationServerName appServerName]
```

## action enableAppSecurity

The **enableAppSecurity** action is used to enable application security for the application. This action sets the **mxe.useAppServerSecurity** property to a value of 1, and updates the **mxe.ldapUserManagement** property according to the setting of the **–usermanagement** parameter. This value is written to the database when updated. The maximouiweb, maxrestweb, meaweb, and mboweb web.xml files are updated during this action.

```
-action enableAppSecurity -usermanagement {j2ee,mixed}
[-buildAndDeployEAR] [-validateUsers] [-force]
```

## action disableAppSecurity

The **disableAppSecurity** action is used to disable application security for the application. This action sets the **mxe.useAppServerSecurity** property to a value of 0. This value is written to the database when updated. Using this action reverts the security implementation to Maximo security for users and groups. The maximouiweb, maxrestweb, meaweb, and mboweb web.xml files are updated during this action.

```
-action disableAppSecurity [-buildAndDeployEar] [-force]
```

## action updateApplication

The **updateApplication** action is used to update the application by updating the database and redeploying application EAR files.

```
-action updateApplication [-updatedb] [-deploymaximoear] [-deployhelpear]
[-wasuser userid] [-waspwd password]
[–enableSkin Classic|tivoli09|tivoli13] [-enableEnhancedNavigation]
[-enableRestrictivePasswordPolicy]
```

## action addLanguages

Use the **addLanguages** action to configure additional languages for a deployment.

```
-action addLanguages [-additionalLangs locale1,locale2,locale3]
[-buildAndDeployEAR] [-allowUpdateDB | -bypassUpdateDB] [-stopAppServer]
[-inputfile path to input properties file]
```

The application server must be stopped when you add languages. If you do not use the **stopAppServer** parameter with the **addLanguages** action, you must stop the application server manually before you use the **addLanguages** action.

The following locale values are supported.

*Table 10. Locales supported by the configuration program*

| Language | Locale |
|---|---|
| Arabic | ar |
| Brazilian Portuguese | pt_BR |

*Table 10. Locales supported by the configuration program  (continued)*

| Language | Locale |
|---|---|
| Croatian | hr |
| Czech | cs |
| Danish | da |
| Dutch | nl |
| Finnish | fi |
| French | fr |
| German | de |
| Hebrew | he |
| Hungarian | hu |
| Italian | it |
| Japanese | ja |
| Korean | ko |
| Norwegian | no |
| Polish | pl |
| Russian | ru |
| Simplified Chinese | zh_CN |
| Slovak | sk |
| Slovenian | sl |
| Spanish | es |
| Swedish | sv |
| Traditional Chinese | zh_TW |
| Turkish | tr |

# action configureProducts

The **configureProducts** action is used to configure more products that are installed after an initial installation and configuration. This action verifies a full installation and configuration of an existing product was completed successfully before it continues. This action can configure multiple products after they are installed. This action can also be used in an upgrade scenario.

```
-action configureProducts [-updatedb] [-buildears] [-deploymaximoear]
[-deployhelpear] [-wasuser userid] [-waspwd password]
```

# action deployMiddlewareConfiguration

The **deployMiddlewareConfiguration** action is used to configures middleware after it is installed.

```
-action deployMiddlewareConfiguration [-inputfile path to input properties file]
```

The following properties can be included in the input properties file used with the **deployMiddlewareConfiguration** action.

*Table 11. Properties that can be used in the deployMiddlewareConfiguration action input file*

| Category | Property |
|---|---|
| HTTP Server properties | IHS.HTTPPort |

*Table 11. Properties that can be used in the deployMiddlewareConfiguration action input file  (continued)*

| Category | Property |
|---|---|
| | IHS.InstallLocation |
| | IHS.WebserverName |
| LDAP server properties | LDAP.AdminDN |
| | LDAP.AdminPassword |
| | LDAP.BaseEntry |
| | LDAP.GroupSuffix |
| | LDAP.OrgContainerSuffix |
| | LDAP.ServerHostName |
| | LDAP.ServerPort |
| | LDAP.UserSuffix |
| | LDAP.Vendor |
| General middleware configuration properties | MW.Operation |
| WebSphere properties | PLG.InstallLocation |
| | WAS.AdminPassword |
| | WAS.AdminUserName |
| | WAS.ApplicationServerName |
| | WAS.CellName |
| | WAS.DeploymentManagerNodeName |
| \ | WAS.DeploymentManagerProfileName |
| | WAS.InstallLocation |
| | WAS.LDAPAutomatedConfig |
| | WAS.ND.AutomateConfig |
| | WAS.NodeName |
| | WAS.ServerProfileName |
| | WAS.SOAPConnectorPort |
| | WCT.InstallLocation |

# Configuration program properties

Maximo Asset Management properties can be modified with the command line configuration program by using an `input.properties` file. When you set a property value to a Windows path, you must use two consecutive backslashes, for example, `C:\\IBM\\WebSphere\\AppServer`. When you set property values to a Linux or UNIX path, use single forward slashes. For example, `/opt/IBM/db2/V10.5`.

*Table 12. Maximo Asset Management properties*

| Category | Property | Definition |
|---|---|---|
| Shared database properties | mxe.db.schemaowner | Owner of the database schema. For example, `maximo`. This value is written to the database when updated. |

*Table 12. Maximo Asset Management properties (continued)*

| Category | Property | Definition |
|---|---|---|
| | mxe.db.user | User ID that accesses the database. |
| | mxe.db.password | Password for the user ID that accesses the database. |
| | Database.RemoteAccessUserName | Database server system user ID that is used for configure the database remotely.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.RemoteAccessPassword | Password for user ID named in **Database.RemoteAccessUserName**.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| DB2 Properties | Database.DB2.ServerHostName | Fully qualified host name of the DB2 server.<br><br>For example, mymachine.mydomain.com.<br><br>This value is written to the database when updated. |
| | Database.DB2.ServerPort | Database server port.<br><br>For example, 50005.<br><br>This value is written to the database when updated. |
| | Database.DB2.InstanceName | Name of the database instance.<br><br>For example, ctginst1.<br><br>This value is written to the database when updated.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.InstanceUserPassword | Password for the database instance owner<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | Database.DB2.InstanceAdminGroup | Group for the instance administrator.<br><br>For example, `ctgiadm1`.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.DatabaseName | Name of the database.<br><br>For example, `maxdb76`.<br><br>This value is written to the database when updated. |
| | Database.DB2.InstallLocation | Install location of the database.<br><br>For example, `/opt/IBM/db2/V10.5`.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.LogFileSize | Set the size for transaction logs.<br><br>For example, 8192.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.AppCtlHeapSize | Application control heap size.<br><br>For example, 1024.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.ApplHeapSize | Application heap size.<br><br>For example, 1024.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |

*Table 12. Maximo Asset Management properties (continued)*

| Category | Property | Definition |
|---|---|---|
| | Database.DB2.LockListSize | Size that is allocated to the lock list.<br><br>For example, AUTOMATIC.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.LogSecond | Number of secondary log files allowed.<br><br>For example, 100.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.ServiceUser | User ID used to autostart.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified.<br><br>This property is only required if DB2 is installed on a Windows system. |
| | Database.DB2.ServicePassword | Password for **Database.DB2.ServiceUser**.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified.<br><br>This property is only required if DB2 is installed on a Windows system. |
| | Database.DB2.ExtentSize | Number of pages per extent (group of pages).<br><br>For example, 32.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | Database.DB2.FencedUser | Fenced user ID for DB2 on Linux or UNIX systems.<br><br>For example, **db2fenc1**.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.FencedUserPassword | Password for the fenced user ID for DB2 on Linux or UNIX systems.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.FencedGroupName | Default group for database fenced user.<br><br>For example, ctgfgrp1.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified.<br><br>This property is only required if DB2 is installed on a Linux or UNIX system. |
| | Database.DB2.AuthType | Method DB2 uses to authenticate users.<br><br>For example, server.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.DataTablespaceName | DB2 table space name for the product database.<br><br>For example, maxdata. |
| | Database.DB2.BufferPoolName | DB2 buffer pool name.<br><br>For example, MAXBUFPOOL.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
|  | Database.DB2.BufferPoolSize | Size of the buffer pool.<br><br>For example, 4096.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
|  | Database.DB2.DataTablespaceLocation | Location of DB2 database table space data files.<br><br>For example, CTGDAT.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
|  | Database.DB2.DataTablespaceSize | Table space size, which is measured in Mb.<br><br>For example, 5000.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
|  | Database.DB2.TempTablespaceName | Temporary table space name.<br><br>For example, maxtemp.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
|  | Database.DB2.TempTablespaceLocation | Location of temporary table space.<br><br>For example, CTGTMP.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
|  | Database.DB2.TempTablespaceSize | Temporary table space size, which is measured in Mb.<br><br>For example, 1000.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | Database.DB2.IndexTablespaceName | Index table space name.<br><br>For example, maxdata. |
| | Database.DB2.IndexTablespaceLocation | Location of index table space.<br><br>For example, CTGDAT.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.IndexTablespaceSize | Index table space size, which is measured in Mb.<br><br>For example, 5000.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DB2.InstanceAdminUserName | Administrative user or the database instance.<br><br>For example, db2admin.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified.<br><br>For Unix and Linux platforms, this value must be the same as the instance owner. |
| | Database.DB2.InstanceAdminPassword | Password for the user ID specified for **Database.DB2.InstanceAdminUserName**.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified.createResourcesIfMissing |
| Oracle | Database.Oracle.SchemaPassword | Password for the schema owner. |
| | Database.Oracle.InstanceName | Oracle instance name.<br><br>This value is written to the database when updated. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | Database.Oracle.SoftwareOwner | Owner of the software installation.<br><br>For example, `oracle`.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.Oracle.SoftwareOwnerPassword | Password for the user ID listed in **Database.Oracle.SoftwareOwner**.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.Oracle.InstallLocation | Oracle installation location. For example, `/opt/app/oracle/product/10.2.0/db_1`.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.Oracle.DataTablespaceName | Oracle table space name for the product database.<br><br>For example, `maxdata`. |
| | Database.Oracle.InstanceLocation | Oracle instance location.<br><br>For example, `/opt/app/oracle/product/10.2.0/db_1`.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.Oracle.DataTablespaceSize | Table space size, which is measured in Mb.<br><br>For example, `5000`.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | Database.Oracle.TempTablespaceName | Temporary table space name.<br><br>For example, `maxtemp`.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.Oracle.TempTablespaceSize | Temporary table space size, which is measured in Mb.<br><br>For example, `1000`.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.Oracle.IndexTablespaceName | Index table space name.<br><br>For example, `maxdata`. |
| | Database.Oracle.IndexTablespaceSize | Index table space size, which is measured in Mb.<br><br>For example, `1000`.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.Oracle.ServerHostName | Fully qualified host name of the Oracle server. |
| | Database.Oracle.ServerPort | Port number that is used by Oracle.<br><br>For example, `1521`.<br><br>This value is written to the database when updated. |
| | Database.DBAUserName | Oracle DBA user name.<br><br>For example, `sys`.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.DBAPassword | Password for user ID listed for **Database.DBAUserName**.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| Microsoft SQL Server | Database.SQL.DatabaseName | Name of the database.<br><br>For example, maxdb76.<br><br>This value is written to the database when updated. |
| | Database.SQL.DataFileName | A way to specify the name of the data file used for the database.<br><br>For example, maxdb76_dat.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.SQL.DataFileMaxSize | Maximum size for data file for the database. |
| | Database.SQL.DataFileSize | Initial size for data file for the database in kb.<br><br>For example, 5000.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.SQL.LogFileName | A way to specify the name for the database transaction log file.<br><br>For example, maxdb76_log.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.SQL.LogFileSize | Microsoft SQL Server transaction log file size.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.SQL.DataFilegroupName | Database logical name file group.<br><br>For example, PRIMARY.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | Database.SQL.ServerHostName | Host name of the database server.<br><br>For example, `myhost.mydomain.com`. |
| | Database.SQL.ServerPort | Database server port.<br><br>For example, `1433`.<br><br>This value is written to the database when updated. |
| | Database.SQL.InstanceAdminUserName | Administrative user for the Microsoft SQL Server instance. Used during installation for creating and modifying the database and database user.<br><br>For example, `sa`.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| | Database.SQL.InstanceAdminPassword | Administrative user password.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatedbconfig** parameters are specified. |
| WebSphere Application Server Network Deployment | WAS.InstallLocation | Installation location for WebSphere Application Server Network Deployment.<br><br>For example, `C:\\IBM\\WebSphere\\ AppServer`.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.DeploymentManagerHostName | Host name of the WebSphere Application Server Network Deployment deployment manager.<br><br>This value is written to the database when updated. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
|  | WAS.DeploymentManagerProfileName | WebSphere Application Server Network Deployment profile name.<br><br>For example, `ctgDmgr01`.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
|  | WAS.DeploymentManagerProfileRoot | Location of the WebSphere Application Server Network Deployment profile.<br><br>For example, `C:\\ IBM\\WebSphere\\AppServer\\ profiles\\ctgDmgr01`<br><br>This property is only required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
|  | WAS.NodeName | WebSphere Application Server Network Deployment node name.<br><br>For example, `ctgNode01` |
|  | WAS.ApplicationServerName | WebSphere Application Server Network Deployment application server name.<br><br>For example, `MXServer`.<br><br>This value is written to the database when updated. |
|  | WAS.CellName | WebSphere Application Server Network Deployment cell name.<br><br>For example, `ctgCell01`. |
|  | WAS.AdminUserName | WebSphere Application Server Network Deployment administrator name.<br><br>For example, `wasadmin`. |

*Table 12. Maximo Asset Management properties (continued)*

| Category | Property | Definition |
|---|---|---|
| | WAS.AdminPassword | WebSphere Application Server Network Deployment administrator password. |
| | WAS.RemoteAccessUserName | WebSphere Application Server Network Deployment deployment manager system user ID used for tasks such as copying Integrated Solutions Console web archive files and fetching the keystore.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.RemoteAccessPassword | WebSphere Application Server Network Deployment deployment manager system user password.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.SOAPConnectorPort | SOAP port for WebSphere Application Server Network Deployment deployment manager.<br><br>For example, 8879. |
| | WAS.VirtualHost | Name of the WebSphere Application Server Network Deployment virtual host.<br><br>For example, maximo_host. |
| | WAS.WebServerHostName | Host name where the HTTP server is located. |
| | WAS.SibName | Name of the service integration bus.<br><br>For example, intjmsbus. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | WAS.SibHiMsg | Service integration bus high message count.<br><br>For example, 500000.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatej2eeconfig** parameters are specified.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.WebServerName | Name of the WebSphere Application Server Network Deployment web server. Used to manage HTTP server from within WebSphere Application Server Network Deployment.<br><br>For example, webserver1. |
| | WAS.SibPersistMessages | Binary value that indicates if service integration bus messages are persisted in either the product database or a local derby database.<br><br>A value of true indicates that the messages are persisted.<br><br>This property is only required if the **createResourcesIfMissing**, **validatefornewdeploy**, or **automatej2eeconfig** parameters are specified.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | WAS.SibDSName | Service integration bus data source name that is created to access the service integration bus persistence store.<br><br>For example, `intjmsds`.<br><br>This property is only required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.SibDBName | Name of the service integration bus messages database.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.SibDBInstance | Instance name of the service integration bus database.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.SibDBInstancePassword | IBM DB2 instance users password for the SIB IBM DB2 data store.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to `true`.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | WAS.SibDBServerName | Server name of the system that is hosting the service integration bus message database.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.SibDBServerPort | Database server port for the database that contains the service integration bus messages.<br><br>For example, 50005.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.SibDBUserName | User ID used to access the persistence data store database for service integration bus messages.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | WAS.SibDBUserPass | Password for user ID named in **WAS.SibDBUserName**.<br><br>This property is only required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.SibDBInstallDir | Where the service integration bus database is installed.<br><br>For example, c:\\program files\\ibm\\sqllib.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.SibDbFencedUser | Fenced user ID for the service integration bus database. This property is only used for databases that are hosted on Linux and UNIX systems.<br><br>For example, db2fenc1.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required if the database is hosted on a Linux or UNIX system.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |

*Table 12. Maximo Asset Management properties (continued)*

| Category | Property | Definition |
|---|---|---|
| | WAS.SibDbFencedPassword | Password for the fenced user ID for the service integration bus database.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required if the database is hosted on a Linux or UNIX system.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.SibDbInstanceAdminUser | Instance owner for the service integration bus database.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required if the database is hosted on a Windows system.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.SibDbInstanceAdminPassword | Password for the instance owner of the service integration bus database.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required if the database is hosted on a Windows system.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | WAS.SibDbRemoteAccessUser | Database server system user that is used to configure the service integration bus remotely.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.SibDbRemoteAccessPassword | Password for user ID named in **WAS.SibDbRemoteAccessUser**.<br><br>This property is required if the **WAS.SibPersistMessages** property is set to true.<br><br>This property is required in the properties file that is specified in one of the following properties.<br>• **WAS.MIFSIBPropertiesFile**<br>• **WAS.CronSIBPropertiesFile**<br>• **WAS.UISIBPropertiesFile** |
| | WAS.VmmGroupRDN | Relative Distinguished Name of the location of virtual member manager groups.<br><br>For example, ou=groups,ou=SWG,o=IBM,c=US. |
| | WAS.VmmUserRDN | Relative Distinguished Name of the location of virtual member manager users.<br><br>For example, ou=users,ou=SWG,o=IBM,c=US. |
| | WAS.UseClustersForDeploy | Deploy the application EAR to a cluster.<br><br>For example, true. |
| | WAS.UIClusterName | Name of the cluster that manages UI tasks.<br><br>Set a value for this parameter if **WAS.UseClustersForDeploy** is set to true. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | WAS.WAS.UIEnableJMS | Enable or disable JMS for Maximo Enterprise Adapter for the UI cluster.<br><br>For example, `true`.<br><br>Set a value for this parameter if **WAS.UseClustersForDeploy** is set to `true`. |
| | WAS.UISIBPropertiesFile | File that contains SIB persistence properties if you are persisting messages in the UI clusters bus.<br><br>For example, `c:\\sibdb_ui.properties`.<br><br>Set a value for this parameter if **WAS.UseClustersForDeploy** is set to `true`. |
| | WAS.CronClusterName | Name of the cluster that manages cron tasks.<br><br>For example, `maximocron`.<br><br>Set a value for this parameter if **WAS.UseClustersForDeploy** is set to `true`. |
| | WAS.CronEnableJMS | Enable JMS for Maximo Enterprise Adapter for the cron task cluster.<br><br>For example, `true`.<br><br>Set a value for this parameter if **WAS.UseClustersForDeploy** is set to `true`. |
| | WAS.CronSIBPropertiesFile | File that contains SIB persistence properties if you are persisting messages in the cron clusters bus.<br><br>For example, `c:\\sibdb_cr.properties`.<br><br>Set a value for this parameter if **WAS.UseClustersForDeploy** is set to `true`. |
| | WAS.ReportingClusterName | Name of the cluster that manages reporting tasks.<br><br>For example, `maximorpt`.<br><br>Set a value for this parameter if **WAS.UseClustersForDeploy** is set to `true`. |

*Table 12. Maximo Asset Management properties (continued)*

| Category | Property | Definition |
|---|---|---|
| | WAS.MIFClusterName | Name of the cluster that manages Maximo Integration Framework tasks.<br><br>For example, `maximomea`.<br><br>Set a value for this parameter if **WAS.UseClustersForDeploy** is set to `true`. |
| | WAS.MIFEnableJMS | Enable JMS for Maximo Enterprise Adapter for the Maximo Integration Framework task cluster.<br><br>For example, `true`.<br><br>Set a value for this parameter if **WAS.UseClustersForDeploy** is set to `true`. |
| | WAS.MIFSIBPropertiesFile | File that contains SIB persistence properties if you are persisting messages in the Maximo Integration Framework clusters bus.<br><br>For example, `c:\\`<br>`sibdb_mif.properties`.<br><br>Set a value for this parameter if **WAS.UseClustersForDeploy** is set to `true`. |
| Multi-tenancy | mt.sysprovider.tenantcode | Tenant code identifier for the multi-tenancy system provider.<br><br>For example, `MTM`.<br><br>This property is required to enable multi-tenancy. |
| | mt.sysprovider.desc | Description of the multi-tenancy manager. |
| | mt.configmanager.adminuserid | User ID of the multi-tenancy configuration manager.<br><br>For example, `maxadmin`.<br><br>This property is required to enable multi-tenancy.<br><br>This value can be the same value used for the `mtadminuser` user. |
| | mt.configmanager.adminpassword | Password for the User ID of the multi-tenancy configuration manager.<br><br>This property is required to enable multi-tenancy.<br><br>This value can be the same value used for the `mtadminpwd` value. |

*Table 12. Maximo Asset Management properties  (continued)*

| Category | Property | Definition |
|---|---|---|
| | mt.configmanager.tenantcode | Tenant code identifier for the multi-tenancy configuration manager.<br><br>For example, GA.<br><br>This is property is required to enable multi-tenancy. |
| | mt.configmanager.desc | Description of the multi-tenancy global administrator. |
| | mt.configmanager.dbuser | User ID of the multi-tenancy configuration manager used to connect to the database.<br><br>For example, ADMIN.<br><br>This property is required to enable multi-tenancy.<br><br>This value can be the same value used for the mtdbuser user. |
| | mt.configmanager.dbpassword | Password for the user ID of the multi-tenancy configuration manager that is used to connect to the database.<br><br>This property is required to enable multi-tenancy.<br><br>This value can be the same value used for the mtdbpwd value. |
| | mxe.mt.enabled | Value that indicates whether multi-tenancy is enabled.<br><br>For example, a value of 1 indicates that multi-tenancy is enabled. A value of 0 indicates that multi-tenancy is not enabled for the product. |
| | mxe.mt.db.adminuser | This value is the same as the value set for the mtdbuser parameter.<br><br>This property is set in the maximo.properties file during the **maxinst** operation. |
| | mxe.mt.demo.extratenants | Number of sample tenants to create when you enable multi-tenancy.<br><br>For example, 10.<br><br>This property is set in the maximo.properties file during the **maxinst** operation. |

# Chapter 16. Configuring your product with performance analysis data

Configure Maximo Asset Management with optimized settings.

## About this task

The IBM Performance Analysis Suite checks Maximo Asset Management for compliance with best practices and optimized performance settings. The IBM Performance Analysis Suite collects configuration data about the database, host operating system settings, WebSphere Application Server Network Deployment and IBM HTTP Server, applies a set of performance metrics to the data, and then generates a report.

The system tuner utility takes IBM Performance Analysis Suite data and automatically configures your environment and middleware.

## Procedure

1. Click the link on the Maximo Asset Management launchpad to download the IBM Performance Analysis Suite from the IBM developerWorks community and install it on the Maximo Asset Management administrative workstation.
2. Analyze your Maximo Asset Management deployment with the IBM Performance Analysis Suite and export the IBM Performance Analysis Suite workspace. Refer to the IBM Performance Analysis Suite developerWorks community for information about using IBM Performance Analysis Suite.
3. Log on to the Maximo Asset Management administrative workstation, open a command prompt, and change to the *install_home*\scripts directory.
4. Run the **tuneSystem** command. All parameters listed are required.

   ```
   tuneSystem -d install_dir -e env_name -w workspace_zip_file
   -r [generate|update] -c [warning|critical] -o output_dir -p plugin_name
   ```

   **--debug**
   Use this parameter to turn on debug logging.

   **-d** *install_dir*
   Use this parameter to specify where the IBM Performance Analysis Suite is installed on the system.

   **-e** *env_name*
   Use this parameter to enter the name of the environment you defined when you used the IBM Performance Analysis Suite to collect data about your environment.

   **-w** *workspace_zip_file*
   Use this parameter to enter the name of the zipped file containing the data that was collected when you used the IBM Performance Analysis Suite.

   **-r [generate|update]**
   Use this parameter to generate scripts to update Maximo Asset Management configuration.

   If you use **generate**, the scripts are generated in the output directory you specify. You must manually run these scripts to re-configure Maximo Asset Management. A file with an extension of .sql is generated to update the Maximo Asset Management database. A file with an extension of .py is

generated to update the Maximo Asset Management application server. A file with an extension of `.sh` or `.bat` is generated to update the Maximo Asset Management administrative workstation.

If you use **update**, the scripts are generated and then run on the system to update Maximo Asset Management configuration.

**-c [warning|critical]**
Use this parameter to specify which conditions identified by the IBM Performance Analysis Suite to update in Maximo Asset Management. If you use **critical**, configuration values that were categorized as critical by the IBM Performance Analysis Suite are updated by the Maximo Asset Management system tuner utility. If you use **warning**, configuration values that were categorized as either critical or warnings by the IBM Performance Analysis Suite are updated by the Maximo Asset Management system tuner utility.

**-o** *output_dir*
Use this parameter to specify where to output the scripts generated by the Maximo Asset Management system tuner utility.

**-p** *plugin_name*
Use this parameter to enter the name of the plugin you specified when you used the IBM Performance Analysis Suite.

**-h** Use this parameter to display help for the **tuneSystem** command.

```
tuneSystem -d install_dir -e env_name -w workspace_zip_file
-r [generate|update] -c [warning|critical] -o output_dir -p plugin_name
```

## What to do next

You should use the IBM Performance Analysis Suite and the **tuneSystem** command to perform Maximo Asset Management configuration updates as part of your regularly scheduled maintenance program.

**Related information**:

⮕ https://www.ibm.com/developerworks/community/blogs/28cb6d68-ab67-4203-96f9-5538e654a5ff/entry/june_16_2013_11_07_pm

⮕ https://www.ibm.com/developerworks/community/wikis/home?lang=en-us#!/wiki/W16f314e69702_46b3_9252_04023c060b9a

# Chapter 17. Backup and restoration

Like all important business data, it is a good idea to establish a process and schedule for backing up Maximo Asset Management data.

Back up and restore middleware application data using the methods described in the documentation for that product is important. Also, establish a process for backing up data contained on the Maximo Asset Management administrative workstation.

The default installation directory on the Maximo Asset Management administrative workstation is `C:\ibm`. This directory contains the critical files for your Maximo Asset Management deployment, which includes all class files and customizations that have been performed in your environment, the current Enterprise Archive (EAR) file, and the properties files and installation tools used for your environment. Plan to back up Maximo Asset Management administrative workstation data just after initial deployment and schedule periodic backups on an ongoing basis.

## Backing up the administrative workstation

It is recommended that you back up all Maximo Asset Management middleware applications and Maximo Asset Management administrative workstation on a regular basis.

### About this task

The default installation directory on the administrative workstation is `\ibm`. This directory contains the critical files for your Maximo Asset Management deployment.

Specifically, the administrative workstation contains the following items:
- Class files and customizations performed in your environment.
- The current Enterprise Archive (EAR) file.
- The properties files and installation tools used for your environment.

It is important to make a back up of the database at the same time that you back up the administrative workstation. During restoration, you restore the database back up at the same time you restore the administrative workstation back up it was paired with.

To back up critical Maximo Asset Management information, complete the following steps:

### Procedure
1. Back up the Maximo Asset Management database, J2EE server, and authentication server information using the instructions provided by your middleware vendors.
2. Create a backup of the installation directory. By default, this directory is `\IBM\SMP`. Ensure that all file permissions are preserved.

# Restoring the administrative workstation

This section details how to restore previously backed up Maximo Asset Management administrative workstation information. This information can be used to return an existing Maximo Asset Management administrative workstation to a previous state.

## Before you begin

It is important to restore the back up of the database that was made when you backed up the administrative workstation. A database back up should be restored with the administrative workstation back up it was paired with.

## About this task

To restore Maximo Asset Management information to an administrative workstation, complete the following steps:

## Procedure

1. Restore the database back up that was paired with the administrative workstation back up you are restoring.
2. Log on to the target administrative system with the same user ID that was used to install the product on the existing administrative workstation.
3. Copy the Maximo Asset Management installation files and directories to the file system of the target administrative system. You must maintain the directory structure of the original installation. For example, if the Maximo Asset Management installation directory on the existing administrative system is \IBM\SMP, you cannot copy those files to a \NewAdminWS\IBM\SMP directory on the target administrative workstation.

# Chapter 18. Uninstalling the product

Uninstalling Maximo Asset Management version 7.6 includes removing configuration values from Maximo Asset Management and associated middleware, and then uninstalling Maximo Asset Management.

Maximo Asset Management uninstallation is a comprehensive procedure and does not support partial removal of individual components or process managers, including process managers that are deployed by other products. Process managers of previously deployed products are also removed when you uninstall Maximo Asset Management.

Maximo Asset Management can only be uninstalled using the Maximo Asset Management installation and configuration programs as directed. Do not use other methods to attempt to uninstall Maximo Asset Management, such as using the Add/Remove Programs panel.

After the Maximo Asset Management uninstallation process is complete, you can reinstall Maximo Asset Management by using the Maximo Asset Management installation and configuration programs.

## Remove Maximo Asset Management configuration

The first step to uninstall Maximo Asset Management is to remove configuration values using the Maximo Asset Management configuration program.

### About this task

If you choose the option to delete the database instance, table spaces, the database, and the database instance are removed. The instance user, database users, and system users that were created with the configuration program can also be removed. For WebSphere Application Server Network Deployment, specify credentials for the WebSphere administrative user to uninstall the Maximo Asset Management application. If you choose the option to delete WebSphere artifacts, the application server and JMS queue information are removed.

### Procedure

1. Log on to the WebSphere Application Server Network Deployment server with the WebSphere Application Server admin console and stop all of the Maximo Asset Management applications.
2. If the Maximo Asset Management configuration program is not open, start it from the Maximo Asset Management launchpad. In the launchpad navigation pane, click **Configure Product**, specify Maximo Asset Management installation location, and then click **Launch the Tivoli's Process Automation Suite configuration program**.
3. In the IBM Maximo Asset Management configuration operations page, click **Remove Product Configuration Information**.
4. In the Remove Product Configuration Information panel, specify credentials to access the database and WebSphere Application Server Network Deployment servers you previously configured for Maximo Asset Management.
5. Click **Finish**, and then **OK** to confirm.

# Removing WebSphere Application Server Network Deployment configuration

Removing WebSphere Application Server Network Deployment configuration with the Maximo Asset Management configuration program deletes all configuration data.

## Procedure

1. If the Maximo Asset Management configuration program is not open, start it from the Maximo Asset Management launchpad. In the launchpad navigation pane, click **Configure Product**, specify Maximo Asset Management installation location, and then click **Launch the Tivoli's Process Automation Suite configuration program**.
2. In the IBM Maximo Asset Management configuration operations page, click **Unconfigure WebSphere Application Server**.
3. In the Unconfigure WebSphere panel, specify credentials for the WebSphere administrative user to remove WebSphere Application Server Network Deployment profile information, and then click **Next**.
4. In the Unconfigure Middleware panel, select the **Unconfigure WebSphere Application Server Network Deployment** option, click **Finish**, and then **OK** to confirm.

# Uninstalling Maximo Asset Management and middleware

Use the Installation Manager software to uninstall Maximo Asset Management and any middleware that was installed with the Maximo Asset Management installation program.

## About this task

After you use the Maximo Asset Management configuration program to unconfigure Maximo Asset Management and associated middleware, use the Installation Manager software to uninstall previously installed components. You must run Installation Manager locally on the system that hosts the component you want to uninstall.

## Procedure

1. Stop all DB2 and WebSphere Application Server Network Deployment processes on the system.
2. Open a command prompt and start Installation Manager.

| Windows | UNIX |
|---|---|
| `C:\Program Files\IBM\Installation Manager\eclipse\IBMIM.exe` | IBMIM.sh |

3. From the IBM Installation Manager interface, select **Uninstall**.
4. Select the packages that you want to remove and then click **Next**.
5. Review the package summary information and click **Uninstall** to remove the selected packages.

# Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

## Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

# Index

## A

action addLanguages 108
action configureProducts 109
action deployConfiguration 107
action deployDatabaseConfiguration 105
action deployJ2eeConfiguration 107
action
  deployMiddlewareConfiguration 109
action disableAppSecurity 108
action enableAppSecurity 108
action enableDB2TextSearch 106
action reapplyAppServerTuning 108
action removeConfiguration 107
action
  removeDatabaseConfiguration 106
action removeJ2EEConfiguration 107
action updateApplication 108
action updateDatabaseConfiguration 104
action updateJ2eeConfiguration 106
action validate And Update Database
  Configuration 105
action validate And Update J2ee
  Configuration 106
action
  validateDatabaseConfiguration 104
action validateJ2eeConfiguration 106
administrative workstation
  backing up 135
  restoring 136
AIX
  asynchronous I/O, enabling 7
  fonts 7
  tar command 7
asynchronous I/O
  enabling on AIX 7

## B

backup 135

## C

command line configuration program
  input properties files 96
  parameters 97
Command line configuration program
  actions 104
configuration 80
configuration actions
  addLanguages 108
  configureProducts 109
  deployConfiguration 107
  deployDatabaseConfiguration 105
  deployJ2eeConfiguration 107
  deployMiddlewareConfiguration 109
  disableAppSecurity 108
  enableAppSecurity 108
  enableDB2TextSearch 106
  reapplyAppServerTuning 108
  removeConfiguration 107

configuration actions *(continued)*
  removeDatabaseConfiguration 106
  removeJ2EEConfiguration 107
  updateApplication 108
  updateDatabaseConfiguration 104
  updateJ2eeConfiguration 106
  validate And Update Database
    Configuration 105
  validate And Update J2ee
    Configuration 106
  validateDatabaseConfiguration 104
  validateJ2eeConfiguration 106
configure Maximo Asset Management
  silently 67
configure middleware silently 67
configuring
  database server 35
  silent 65
configuring Maximo Asset Management
  command line configuration
    program 95
  validate configuration parameters
    from the command line 96

## D

database server 35
DB2
  manual configuration, v10.5 36
DB2 v10.5
  installing 17
deploying ear files 31, 56
directory server
  manually configuring 42

## E

e-mail listener
  configuring in WebLogic Server 27,
    51
EAR files
  building manually 93
  maintaining 93

## F

fonts
  AIX 7

## H

help 79
help information
  installing in multiple languages 79

## I

IBM Tivoli Directory Server
  manually configuring 43

industry solutions 87
  installing from command-line 87
  installing silently 88
  installing with launchpad 87
install
  automatic middleware
    configuration 17
install silently 62
installation
  automatically configuring existing
    middleware 21
  deploying using manual middleware
    configuration 35
  deploying with manually configured
    middleware 35
  preparing 1
  program logs 77
Installation
  post-installation tasks 80
Installation Manager
  installing silently 61
installation program
  logs 77
installing
  silent 59
integration framework
  JMS options
    configuring 26, 50

## J

J2EE server
  manually configuring 48
JRE
  configuring in Linux 8

## L

launchpad
  starting 12
libraries
  Linux 8
    required libraries 8
Linux
  JRE, configuring 8
  shared memory, setting 9
  swap size, setting 9
  ulimit, setting 9

## M

Maximo Application Server
  starting 31, 56
Maximo Asset Management
  configuring with the Maximo Asset
    Management configuration
    program 19, 22, 54
  installing version 7.6 18, 21, 53
  remove configuration 137
  uninstalling 138

**IBM** ®

Printed in USA