



# Overview

---

This chapter provides an overview of what happens as a result of the installation process, such as which packages are installed and how environment variables are set. This guide explains how to install the servers used by both the web applications and the WAE platform.

- [Collector Server Upgrades](#)—Describes options for migrating the Collector server files and upgrading the Collector server database.
- [Before You Start](#)—Requirements, best practices, and pre-installation instructions.
- [Online Installation](#)—Procedure for installing the software when you have access to the Internet.
- [Offline Installation](#)—Procedure for installing the software when you do not have access to the Internet.
- [Post Installation](#)—Optional procedures to follow after having installed the product, including pointers on where to go for configuration information and how to log in to the web UI.
- [License Installation](#)—Procedure for installing licenses for all products.
- [WAE Design Floating License Server](#)—Procedure for installing and setting up the FlexNet Publisher license server. This is applicable only if you have WAE Design and only if you need to administer floating licenses for these end users.

## Installer

An installer is an executable that runs a script to install the WAE software in the proper locations. Although WAE features are installed, they are active only if you have a license for them.

The installer automates the installation process by asking you a series of questions and performing the following tasks.

- Verifies the following.
  - Whether the device has the appropriate system requirements, including proper operating system, disk space, total memory, and required software packages.

If the installer sees that you do not have the appropriate system requirements, it either exits, gives a warning, or prompts you whether to continue. Since the installer might be checking for more than what your specific requirements are, you can press “y” to attempt to continue.
  - Whether there are existing package installations; if there are, they are preserved
- Stops all services that are running.
- Installs all RPM packages, creates the software directory structure, and creates symbolic links to the most recently installed packages.

- Sets the maximum file descriptors and processors as follows.
  - /etc/sysctl.conf file  
Sets fs.file-max=512000 if it is less than 65535.
  - /etc/security/limits.conf  
Sets nofile to 65535 for the WAE username.
  - /etc/security/limits.d/90-nproc.conf  
Sets nproc to 8192 for the WAE username.
- Sets the file capabilities for all binaries in `$CARIDEN_HOME/lib/ext/pmacct/sbin`, which enables you to collect flow data using `flow_manage` and `flow_get` without having to change the file capabilities for the flow collection server. This is the equivalent of executing the following command.
 

```
setcap 'cap_net_bind_service=+ep' $CARIDEN_HOME/lib/ext/pmacct/sbin/*
```
- Automatically starts the web server (`wae-web-server`), WAE Network Interface (NI) server (`wae-ni`), and all System services (`wae-svcs-*`). For detailed information on services, including how to start, stop, restart, and check their status, see the *WAE System Administration Guide*.
- Upon logging out and back in, the WAE username and associated permissions are set, and the environment variables are set.
  - For executables, the WAE user has read, write, and execute permissions. Users listed in `/etc/group` have read and execute permissions. All others have read-only permissions.
  - For non-executables, only the WAE user has read and write permissions. All other users have read-only permissions.

## RPM Packages

The `jdk-7u60` package is installed in `/usr/java/latest`. All other RPM packages are installed in `/opt/cariden/software`. To list these packages, enter this command.

```
rpm -qa | grep wae-
```

Package	Description
<code>jdk-7u60</code>	Oracle Java version 7 (1.7)
<code>wae-appenginecore</code>	Service API that manages and routes requests to load the network model to the appropriate <code>wae-designapiserver</code> .
<code>wae-core</code>	WAE Core server that enables the use of WAE Core REST and Thrift APIs.
<code>wae-db</code>	WAE Core database.
<code>wae-demo</code>	Demonstration UI for applications developed for WAE Core.
<code>wae-designapiserver</code>	Service that enables <code>wae-appenginecore</code> to use the Design APIs. This wraps the Design APIs and controls Design API instances.
<code>wae-dlc</code>	WAE Design, WAE Live, and WAE Collector executables and binaries.
<code>wae-messaging</code>	WAE messaging system that uses JMS (Java Message Service).
<code>wae-ni</code>	WAE NI server that enables the northbound WAE NI APIs. WAE NI is used for continuous polling and continuous PCEP LSP collection.
<code>wae-osc</code>	Open SDN Controller.

Package	Description
wae-platsvcs	Package that distributes the Automation platform services in a high-availability environment.
wae-svcs-client	Component that provides the data to the aggregation server for displaying statistics, such as logs, diagnostics, and processes.
wae-svcs-server	Aggregation server for gathering and collating the web statistics sent to it by the wae-svcs-client.

## Current Links

The current symbolic link for the DLC package is `/opt/cariden/software/mate/current`.

**Example RPM name:** `wae-dlc-1.0.2-123456789`

**Corresponding current link name:** `/opt/cariden/software/mate/current`

Other than the DLC package, all other packages have a current symbolic link using the package name as the link name.

**Example RPM name:** `wae-core-1.0.2-123456789`

**Example corresponding current link name:** `wae-core`

## Versions

To determine the release version, enter the following command.

```
license_check -version
```

To determine the current RPM versions, enter the following command.

```
rpm -qa | grep wae
```

## Environment Variables

Upon logging out and back in as the WAE user after an installation, the environment variables are automatically set using `.sh` scripts in `/etc/profile.d` directory

Environment Variable	Default
<code>\$WAE_ROOT</code>	<code>/opt/cariden</code>
<code>\$WAE_HOME</code>	<code>/opt/cariden/software</code>
<code>\$CARIDEN_ROOT</code>	<code>/opt/cariden</code>
<code>\$CARIDEN_HOME</code>	<code>/opt/cariden/software/mate/current</code>
<code>\$CASSANDRA_HOME</code>	<code>/opt/cariden/software/wae-db</code>
<code>\$COLLECTOR_HOME</code>	<code>/opt/cariden/software/wae-ni</code>

Environment Variable	Default
<code>\$JAVA_HOME</code>	<code>/usr/java/latest</code>
<code>\$PYTHONPATH</code>	<code>/opt/cariden/software/wae-demo/lib/python2.7/site-packages</code> For use by the <code>/etc/profile.d/wae-demo.sh</code> script.