FICO® Blaze Advisor®

decision rules management system

Version 7.8

INSTALLATION AND SETUP

For Java

FICO®

FICO® Blaze Advisor® 7.8
Last Revised October 31, 2021
Version 7.8
Template LG9.0

Contents

CHAPTER 1

Installation and Setup

This section explains how to install the FICO® Blaze Advisor®
decision rules management system on the supported platforms. It contains these topics:

In the Blaze Advisor documentation, you will see references to `<ADVISOR_HOME>`. This is an abbreviated way of referring to the full path to your Blaze Advisor installation. For example: `C:\Blaze\Advisor78`.

Prerequisites for Installing Blaze Advisor

The requirements for installation include:

- A minimum disk space of approximately 2 gigabytes.
- The minimum memory is 4 gigabytes.
- An installed Java Virtual Machine (JVM).
- A valid license for Blaze Advisor and for any separately licensed features that you want to use.

License Options

Blaze Advisor is licensed as two products: Development and Deployment.

- A Development license allows you to use all of the design components to create rule maintenance applications and to run non-production or test deployments. The kit contains all of the files, documentation, examples, and tutorials that are included with Blaze Advisor.
- A Deployment license allows you to deploy rule services in a production environment. The kit contains only those files needed to run Blaze Advisor rule services.

A separate license is required for the Decision Simulator and PMML Executor features.

- A Decision Simulator license allows you to run Blaze Advisor projects using historical data to assess the potential business impact of any proposed changes.
- A PMML Executor license allows you to import PMML models that conform to the PMML 3.0 to 3.2 or 4.0 to 4.4 specification.

System Requirements for Developing Blaze Advisor Applications

Development machines must have the following software installed:

- One of the supported operating systems and a compatible Java Development Kit (JDK) or Java Runtime Environment (JRE).
- One of the supported versions of Eclipse.
- A supported web browser for viewing reports and rule maintenance applications. See **"System Requirements for Rule Maintenance Application Deployment" on page 9**.
- Blaze Advisor Development license.

Blaze Advisor supports the following platforms and JDKs and JREs:

Platforms (64-bit)	Supported JDKs and JREs(64-bit)
Microsoft Windows 10	■ Oracle Java SE Development Kit (JDK) 8 ■ Oracle Java SE Runtime Environment (JRE) 8 ■ Oracle Java Development Kit (JDK) Java 11 ■ Amazon Corretto 8 and 11 ■ IBM SDK 8

The following links offer free downloads for some of the supported JDKs and JREs:

- **https://www.oracle.com/technetwork/java/javase/downloads/index.html**
- **https://www.ibm.com/developerworks/java/jdk/**
- **https://docs.aws.amazon.com/corretto/latest/corretto-11-ug/downloads-list.html**
- **https://docs.aws.amazon.com/corretto/latest/corretto-8-ug/downloads-list.html**

Blaze Advisor supports the following versions of Eclipse:

- Eclipse 2021-06
  64-bit Eclipse with a 64-bit version of Oracle JDK 11 or Amazon Corretto 11
- Eclipse 2020-06
  64-bit Eclipse with a 64-bit version of Oracle JDK 8 or Amazon Corretto 8

Other versions of Eclipse are not supported.

Note: FICO recommends installing Eclipse IDE for Java EE Developers.You can use other Eclipse packages but, if you do, you will be prompted to install additional plug-ins while installing the Blaze Advisor Eclipse plug-in.

In the case of Eclipse 2020-06, a recommendation to download node.js is noted on the download site. If you do not install node.js, you may see a warning message pop-up in the IDE.

To install the Blaze Advisor Deployment Feature that allows you to generate an Eclipse Web Project and WAR deployment from the Rule Maintenance Application Generator, you must install Eclipse IDE for Java EE Developers for Eclipse 2021-06 or Eclipse 2020-06.

System Requirements for Deploying Blaze Advisor Applications

Deployment machines must have the following software installed:

- One of the supported operating systems listed and a compatible JDK or JRE.
- Blaze Advisor Deployment license.

This section contains these topics:

Operating Systems and JDKs and JREs for Deployment

Blaze Advisor supports the following operating systems and JDKs for rule service deployments:

Platforms (64-bit)	Supported JDKs (64-bit)
■ Microsoft Windows 10 ■ Microsoft Windows Server 2019	■ Oracle Java SE Development Kit (JDK) 8 ■ Oracle Java SE Runtime Environment (JRE) 8 ■ Oracle Java Development Kit (JDK) Java 11 ■ Amazon Corretto 8 and 11 ■ IBM SDK 8
IBM AIX 7.2 and later	IBM SDK 8
IBM z/OS 2.1 and later, zLinux SuSE	IBM SDK 8
Linux RHEL 8	■ Oracle Java SE Development Kit (JDK) 8 ■ Oracle Java SE Runtime Environment (JRE) 8 ■ Oracle Java Development Kit (JDK) Java 11 ■ Amazon Corretto 8 and 11 ■ IBM SDK 8

The following links offer free downloads for some of the supported JDKs and JREs listed previously:

- **https://www.oracle.com/technetwork/java/javase/downloads/index.html**
- **https://www.ibm.com/developerworks/java/jdk/**

- **https://docs.aws.amazon.com/corretto/latest/corretto-11-ug/downloads-list.html**
- **https://docs.aws.amazon.com/corretto/latest/corretto-8-ug/downloads-list.html**

Server Frameworks for Rule Service Deployment

Server support includes both a runtime library that provides a generic integration mechanism with server environments and a set of deployment-time facilities (Deployment Definition entities, Quick Deployer template files, and a Generate Deployment Wizard in the IDE) to assist with the deployment of decision services.

This table outlines the servers that are supported for rule service deployments:

Server	Supported Deployments	Oracle JDK	Other JDK Support
Apache Tomcat 9	XML Document Web Services and Java POJO Web Services	Oracle JDK 8 Oracle JDK 11	Amazon Corretto 8 Amazon Corretto 11
Apache Tomcat 10	XML Document Web Services and Java POJO Web Services	Oracle JDK 8 Oracle JDK 11	Amazon Corretto 8 Amazon Corretto 11
JBoss Application Server (7 EAP)	All	Oracle JDK 11	Amazon Corretto 11
Spring 5.3.7	Java only	Oracle JDK 11	Amazon Corretto 11
Oracle WebLogic 12c (R2 12.2.1)	All	Oracle JDK 11	Amazon Corretto 11
Oracle WebLogic 14.1.1	All	Oracle JDK 11	Amazon Corretto 11
IBM WebSphere 9.0	All	N/A	IBM SDK 8
IBM WebSphere 9.0.5	All	N/A	IBM SDK 8
IBM WebSphere Liberty 19.0.0.3 and later	All	N/A	IBM SDK 8
IBM WebSphere Liberty 21	All	N/A	IBM SDK 8

Note: In this release, FICO supports two versions of Apache Tomcat (10 and 9). Note that the Vendor drop-down list in the System Definition wizard lists only the option for the higher version. To use an Apache Tomcat 9 server for your rule service deployment, select **Tomcat 10 and later** in the Vendor drop-down list to generate your deployment.

The lack of an out-of-the-box Quick Deployer (QD) template for a specific version of a JEE container does not necessarily prevent a Blaze Advisor rule service from being deployed to the container. It simply requires you to provide the integration code that normally would be generated by the Quick Deployer. If you need this type of solution, FICO advises you to modify, as needed, the code generated by the Quick Deployer for an earlier version of the target container to satisfy any norms/constraints imposed by the new container version.

This recommendation assumes that the implementation of the target JEE container has not introduced any significant architectural shifts/API changes that are incompatible with the rule service model supported by this release of Blaze Advisor.

Platforms for COBOL Deployment

The COBOL code that Blaze Advisor generates is ANSI X3.23 1985 / ISO 1989:1985 compliant.

The following platforms are supported:

- IBM OS/390 V2 R8
  Blaze Advisor supports COBOL compiler version 2, release 1 or higher.
- IBM z/OS 2.1 and later

System Requirements for Rule Maintenance Application Deployment

Blaze Advisor supports the following web browsers:

- Google Chrome
- Mozilla Firefox
- Microsoft Edge Chromium

To specify the browser that will be used to view an RMA generated from the Blaze Advisor IDE, see “Blaze Advisor General Preferences” in *DevelopingRuleProjects.pdf*.

The following platforms are supported for Blaze Advisor RMAs:

Platforms (64-bit)	Supported JDKs (64-bit)
■ Microsoft Windows 10 ■ Microsoft Windows Server 2019 ■ Linux RHEL 8	■ Oracle Java SE Development Kit (JDK) 8 ■ Oracle Java SE Runtime Environment (JRE) 8 ■ Oracle Java Development Kit (JDK) Java 11 ■ Amazon Corretto 8 and 11

This table outlines the supported servers for RMA deployments:

Server	Oracle JDK	Other JDK Support
Apache Tomcat 9	■ Oracle JDK 8 ■ Oracle JDK 11	■ Amazon Corretto 8 ■ Amazon Corretto 11
Apache Tomcat 10	■ Oracle JDK 8 ■ Oracle JDK 11	■ Amazon Corretto 8 ■ Amazon Corretto 11
JBoss Application Server (7 EAP)	Oracle JDK 11	Amazon Corretto 11
Spring 5.3.7	Oracle JDK 11	Amazon Corretto 11
Oracle WebLogic 12c (R2 12.2.1)	Oracle JDK 8	Amazon Corretto 8
Oracle WebLogic 14.1.1	Oracle JDK 11	Amazon Corretto 11
IBM WebSphere 9.0	N/A	IBM SDK 8
IBM WebSphere 9.0.5	N/A	IBM SDK 8

Server	Oracle JDK	Other JDK Support
IBM WebSphere Liberty 19.0.0.3 and later	N/A	IBM SDK 8
IBM WebSphere Liberty 21	N/A	IBM SDK 8

Note: FICO RMAs are not tested with third-party add-ons. If the browser you will be using for RMAs has third-party add-ons, be aware that this may result in some unpredictable behavior.

Supported Databases

These databases are supported for Decision Simulator, database business object models (BOMs), database repositories, and other Blaze Advisor features that permit database access:

Decision Simulator

- Oracle
- IBM DB2
- MS SQL Server
- HSQLDB

Database business object models (BOMs)

- Oracle
- IBM DB2
- MS SQL Server

LCM Access Control

- Oracle
- IBM DB2
- MS SQL Server
- HSQLDB
- MySQL

Database Repositories

- Oracle
- IBM DB2
- MS SQL Server
- MongoDB

  Blaze Advisor supports MongoDB 4.4.6 and later for 64-bit Microsoft Windows machines and Linux machines with RHEL 8.

Supported Source Control System

Blaze Advisor supports the following third-party versioning service with Blaze Advisor repositories and workspaces:

- Subversion® (SVN) open source version control system
  SVNKit 1.10 (for the client) with SVN server 1.10.x

Installing a Development or Deployment Kit on Microsoft Windows

This section explains how to install Blaze Advisor on Microsoft Windows. It contains these topics:

To use the Blaze Advisor Development kit, you must install the Blaze Advisor Eclipse plug-in after you install Blaze Advisor. See **"Installing the Blaze Advisor Plug-in on Eclipse" on page 13**.

Invoking the Installer on Microsoft Windows

For best results, FICO recommends the following:

- Install Blaze Advisor to a clean directory. Do not install to a directory where you have previously installed or uninstalled another version of Blaze Advisor.
- Use the same version of the JDK that you will use when installing the Blaze Advisor Eclipse plug-in. Using the same version will minimize the possibility of errors caused by the Java Platform Module System. For information about how to specify the JDK version, see **"Modifying the Eclipse Configuration File" on page 16**.
- Install Blaze Advisor to a directory name without spaces.

To use the installer

1. Download the Blaze Advisor release.
2. Double-click `blaze-advisor_7.8.v(`*`number`*`)_Windows.exe`.
3. When the installer opens, read the text and click **Next**.
4. Retain **Development** to install the Development kit or select **Deployment** for the Deployment kit, and click **Next.**
5. Select a Java Development Kit (JDK) or a Java Runtime Engine (JRE) from the list to use for the `<JAVA_HOME>` variable in the Blaze Advisor scripts. If the JDK or JRE you want to use is not listed, click **Choose Java Executable** to select it from your machine. If you do not select one before you click **Next,** the Java executable at the top of the list is used by default.
6. Read the license, click **I accept the terms of the License Agreement** and click **Next**.
7. Do one of the following to specify where you want Blaze Advisor installed.
   - Enter the full path to the directory and click **Next**.
   - Click **Choose** and choose the directory and click **Next**.
   - Click **Next** to use the default installation directory.

8 Enter the license keys now or wait until after installation is complete.

For example, if the development license key is NdDevelopment=11-123-5-6abc-ABCDEFGHI1234-1234-12345678, enter only the numbers, letters, and the hyphens. A development license begins with `11`, a deployment license begins with `12`, a Decision Simulator license begins with `8`, and a PMML Executor license begins with `16`.

To add the license after the installation is complete, click **Next** then follow the instructions in **"Adding or Updating Licenses After Installation on Microsoft Windows" on page 12**.

9 Review the details of the installation and do one of the following:

- Click **Install** to install Blaze Advisor.
- Click **Previous** to return to the page where you want to make a change. Make your change, click **Next** until you return to the last wizard page, and then click **Install.**

10 After the installer finishes, click **Done**.

Adding or Updating Licenses After Installation on Microsoft Windows

If the license keys were not entered during installation, enter them using the license utility. You can also use this utility to enter any corrections to the license keys.

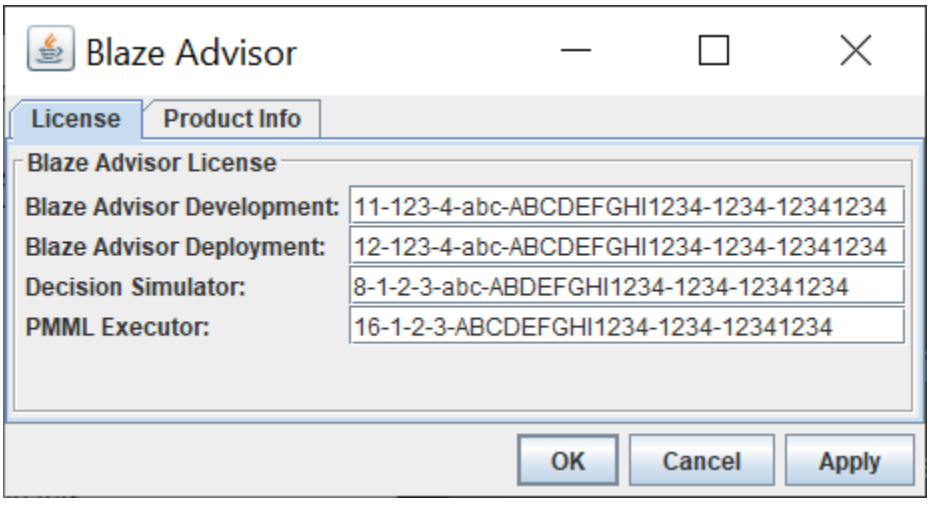
If you will be installing the Blaze Advisor Eclipse plug-in, you can add the license using the Preferences window in Eclipse. See "Adding or Updating the Blaze Advisor License in Eclipse" in *DevelopingRuleProjects.pdf*.

To add or update licenses on Microsoft Windows

1 Open a Command Prompt window and locate the `<ADVISOR_HOME>\bin` directory and enter this string:

```
installLicense.bat
```

2 In the utility, enter the applicable licenses in the fields provided, as shown in this figure:



3 Click **Apply**, then click **OK.**

Verifying a Blaze Advisor Installation on Microsoft Windows

Blaze Advisor includes a script you can use to verify the installation. After the script runs, the following information is printed in the window:

- Java Virtual Machine (JVM) information, version number, vendor number, and OS
- Classpath
- FICO products that are installed
- Product version, build number, and build date for each product or module
- License information

To verify the Blaze Advisor installation on Windows

1. Open a Command Prompt window and change to the `<ADVISOR_HOME>\bin` directory.
2. Run the installation verification script:

   ```
   verifyInstall.bat
   ```

   If the installation was successful, this message appears when the script finishes:

   ```
   Blaze Advisor has successfully been installed.
   ```

Installing the Blaze Advisor Plug-in on Eclipse

The Blaze Advisor Development environment is an Eclipse plug-in. This section describes how to install the Blaze Advisor Eclipse plug-in.

It contains these topics:

You can use the Eclipse Extension Plug-in mechanism if you want to integrate a third-party application or a database or source control management system with Blaze Advisor. See **“Integrating Third-party Tools with Blaze Advisor” on page 29**.

Important Tips for Installing the Blaze Advisor Eclipse Plug-in

- FICO recommends creating a new instance of Eclipse and using it to install a new version of the Blaze Advisor Eclipse plug-in.

- FICO recommends using the same JDK version that was selected during the Blaze Advisor installation. Using the same version will minimize the possibility of errors caused by the Java Platform Module System. For information about adding a path to the JDK in the `eclipse.ini` file, see **"Modifying the Eclipse Configuration File" on page 16**.
- To reduce the installation time, confirm that a valid Blaze Advisor development license and licenses for any additional features that you want to use are in your Blaze Advisor installation. Alternatively, you can add the licenses after the installation. See **"Adding or Updating Licenses After Installation on Microsoft Windows" on page 12**.
- During installation, if a login dialog or wpad error message appears, you may need to configure the Eclipse proxy settings or consult with your network engineers to allow Eclipse to contact update sites directly.

Installation Procedure for the Blaze Advisor Eclipse Plug-in

After Blaze Advisor and one of the supported Eclipse platforms is installed, use the procedure in this section to install the Blaze Advisor Eclipse plug-in.

There are optional features you can install with the Blaze Advisor plug-in or install later after the Blaze Advisor plug-in is installed.

- Blaze Advisor Deployment Feature: If this feature is installed, you can generate an Eclipse Web Project and WAR deployment from the Rule Maintenance Application Generator. See "Using the Rule Maintenance Application Generator" in *DevelopingRuleMaintenanceApplications.pdf*.
- Decision Simulator: If this feature is installed, you can create, manage and generate simulations. These simulations enable business users to run Blaze Advisor projects using historical data to assess the potential business impact prior to deployment into a production environment.A valid license must be included in your Blaze Advisor installation.
- PMML Executor: If this feature is installed, you can import PMML models that conform to the PMML 3.0 to 3.2 or 4.0 to 4.4 specification. A valid license must be included in your Blaze Advisor installation. See "Importing Other PMML Models" in the RMA User Guide accessible from an RMA.

To install the Blaze Advisor plug-in

1. Launch Eclipse.
2. Do one of the following:
   - Select a folder to use as the Eclipse workspace**.**
   - Enter the fully qualified path to the Eclipse workspace**.**
3. (Optional) Select the check box next to **Use this as the default and do not ask again.**
4. Click **Launch**.
5. In Eclipse, select **Help > Install New Software.**
   a. Click **Add** to add a site.
   b. In the Add Repository dialog box, enter a name for Blaze Advisor.

c Click **Local** to locate the Blaze Advisor site folder at `<ADVISOR_HOME>\EclipsePluginUpdateSite`.

By default, `ADVISOR_HOME` is `C:\Blaze\Advisor<version number>`

d Click **Select Folder**.

e Back in the Add Repository dialog box, click **Add**.

6 Select the check box next to **Blaze Advisor** in the **Name** column.

7 (Optional) Select the check box next to any additional features that you want to add.

If you install the Blaze Advisor Deployment Feature, ensure that the check box next to **Contact all update sites during install to find required software** is selected. Otherwise, you can clear this check box.

This option checks for any requirements and dependencies related to the selected plug-ins and automatically installs them for you. For example, if you selected the Blaze Advisor Deployment Feature and you are using a non-JEE version of Eclipse, the Eclipse Java EE Developer Tools are automatically installed for you. As a result, the installation may take longer depending on your network.

8 Click **Next**. After a few moments, the Install Details page appears. Click **Next**.

9 Read the license and select **I accept the terms of the license agreement**, and click **Finish**.

10 Click **Restart Now** to close and restart Eclipse.

11 If you did not select the **Use this as the default and do not ask again** check box in Step 3, when Eclipse restarts, retain the workspace showing in the **Workspace** field or select another one and click **OK**.

12 The remaining steps are only applicable if a valid Blaze Advisor license was not available when the installation process was started.

a Ensure that the Blaze Advisor installation includes a valid development license.

b In the **Blaze Advisor License** window, do one of the following:

- Browse to locate the `keys.properties`, select it, and click **Open**.
- Enter each license key. If a license key is valid, the expiration date of the license appears in the License Details section. If the license is invalid, a message appears instead.

c Click **Finish**. You will be prompted to restart Eclipse.

13 Click **Yes** to close and restart Eclipse. If Decision Simulator or PMML Executor is installed and a valid license is not in your Blaze Advisor installation, you will be prompted to enter the license key.

14 Enter the license key and click **Finish**.

15 Click **Yes** to restart Eclipse.

To complete your Eclipse configuration, see **"Modifying the Eclipse Configuration File" on page 16**. To make changes to the licenses, see **"Adding or Updating a Blaze Advisor License" on page 18**.

Resuming the Installation

Note: If you selected Cancel during the installation, you can continue with the installation process at any time.

To resume the installation later

1. If necessary, launch Eclipse.
2. Select a folder to use as the workspace, and click **OK.**
3. In Eclipse, select **Help > Install New Software.**
4. From the **Work with** drop-down menu, select the Blaze Advisor site.
5. Start at Step 5 in **Installation Procedure for the Blaze Advisor Eclipse Plug-in** and complete the remaining steps to finish the installation process.

Modifying the Eclipse Configuration File

During the Blaze Advisor Eclipse plug-in installation, an `eclipse.ini` file is installed under `<ECLIPSE_HOME>`. This file includes Eclipse startup options. To complete the Eclipse installation, you must make some modifications to the `eclipse.ini` file.

Important: Using the Java Virtual Machine (JVM) that is packaged with Eclipse can result in errors caused by the Java Platform Module System. To minimize the possibility of errors, FICO recommends specifying the same JDK version that was selected during the Blaze Advisor installation in the `eclipse.ini` file. In addition, you can include the `--illegal-access=warn` argument to the `-vmargs` section of the `eclipse.ini` file. For information about the Java Platform Module System, see **"Stricter Access Rules in the Java Platform Module System" on page 34**.

You must also set a property that improves the performance when using the Blaze Advisor Eclipse plug-in. If you do not add this property, a message displays suggesting that you add this property the first time you launch Eclipse with the Blaze Advisor Eclipse plug-in installed

For general information about the `eclipse.ini` file, see **https://wiki.eclipse.org/Eclipse.ini.**

To set the configurations

1. Open the file in a text editor.
2. Under the `-vm` option, replace the default path with a path to a specific JDK executable.

   You can determine this path by searching for `JAVA_HOME` in the `setenv.bat` file under `<ADVISOR_HOME>\bin`. If the path displayed in this file is not the complete path to a JDK executable, for example, `C:\Program Files\Java\jdk-11.0.12`, append `\bin` to the path and specify the JDK executable in the `\bin` directory. For example:

   ```
   C:\Program Files\Java\jdk-11.0.12\bin\javaw.exe
   ```
3. To enhance the performance when using the Blaze Advisor Eclipse plug-in, enter this property at the end of the `-vmargs` properties:

   ```
   -Dosgi.bundlefile.limit=1000
   ```

Important: If you are working with Decision Simulator, simulation executions will pass only if you set this Eclipse property with the value specified.

This figure shows the updates to the `eclipse.ini` file.

Verifying the Java Versions for Blaze Advisor and Eclipse

For best results, ensure that both Blaze Advisor and Eclipse are using the same version of a supported JDK or JRE. Using the same version will minimize the possibility of errors caused by the Java Platform Module System. See **"System Requirements for Developing Blaze Advisor Applications."**

To verify the version you are using with Blaze Advisor

- To verify the Java version you are using with Blaze Advisor, do one of the following:
  - Open the `installer_debug.txt` file under `<Blaze>` in a text editor and search for the `JAVA_HOME` value.
  - Open the `setenv.bat` file under `<ADVISOR_HOME>\bin` in a text editor and search for the `JAVA_HOME` value.
  - Open a command prompt and change the directory to `<ADVISOR_HOME>\bin` and run the `verifyInstall.bat`.

To verify the version you are using with Eclipse

1. Select **Help > About Eclipse IDE.**
2. Click **Installation Details** and select the **Configuration** tab.
3. In the System properties section, scroll down to find the `-vm` flag for the JDK version that is being used.

Installing Additional Features After Installing the Blaze Advisor Plug-in

If the Blaze Advisor plug-in is already installed in Eclipse, you can install any or all of the following:

- Blaze Advisor Deployment Feature
- Decision Simulator
- PMML Executor

For a description of the features, see **"Installation Procedure for the Blaze Advisor Eclipse Plug-in" on page 14**.

To install a feature after Blaze Advisor is installed

1. Launch Eclipse.
2. In Eclipse, select **Help > Install New Software.**
3. In the **Work with** field, select the Blaze Advisor site folder at `<ADVISOR_HOME>\EclipsePluginUpdateSite`. By default, `ADVISOR_HOME` is `C:\Blaze\Advisor<version number>`
4. Select the check box next to **Blaze Advisor** in the **Name** column.
5. To install the Blaze Advisor Deployment Feature, expand the **Blaze Advisor Optional** folder and select **Blaze Advisor Deployment Feature**. Ensure that the check box next to **Contact all update sites during install to find required software** is selected. Otherwise, you can clear this check box.

   This option checks for any requirements and dependencies related to the selected plug-ins and automatically installs them for you. For example, if you selected the Blaze Advisor Deployment Feature and you are using a non-JEE version of Eclipse, the Eclipse Java EE Developer Tools are automatically installed for you.
6. (Optional) To install Decision Simulator, select **Decision Simulator**.
7. (Optional) To install PMML Executor, expand the **Blaze Advisor Analytics Integration** folder and select **PMML Executor**.
8. Click **Next**. After a few moments, the Install Details page appears. Click **Next**.
9. Read the license and select **I accept the terms of the license agreement**, and click **Finish**.
10. Click **Restart Now** to close and restart Eclipse.
11. If you did not select the **Use this as the default and do not ask again** check box in Step 3, when Eclipse restarts, retain the workspace showing in the **Workspace** field or select another one and click **OK**.

Adding or Updating a Blaze Advisor License

You can update the Blaze Advisor development and deployment licenses in Eclipse at any time.

To add or update a license

1. Select **Window > Preferences > Blaze Advisor.**
2. Click **Install new license**, and do one of the following:
   - Enter or modify the license key(s).
   - Click **Browse** to select a `keys.properties` file, and click **Open**.
3. Click **Finish**.
4. When prompted to restart Eclipse now, click **Yes**.

Updating the Decision Simulator License in Eclipse

You can update the license in Eclipse at any time.

To update the license

1. Select **Window > Preferences > Blaze Advisor > Decision Simulator.**
2. Click **Install new license.**
3. In the **Decision Simulator** field, enter the license key.
4. Click **Finish**.
5. When prompted to restart Eclipse now, click **Yes**.

Updating the PMML Executor License in Eclipse

You can update the license in Eclipse at any time.

To update the license

1. Select **Window > Preferences > Blaze Advisor > PMML Executor.**
2. Click **Install new license.**
3. In the **PMML Executor** field, enter the license key.
4. Click **Finish**.
5. Click **Restart Now** to close and restart Eclipse.

Changing the License Expiration Notification Period

Sixty days before your Blaze Advisor licenses are set to expire, you will start to see a license expiration notification in the Eclipse Console.

You can change the length of the notification period by setting the value for a flag in the `eclipse.ini` configuration file. For example, to change the notification period from 60 days to 45 days, open the `eclipse.ini` file and add this line:

```
-Dnd.licenseCheckWarningDuration=45
```

You can also turn off the notification by setting another flag in the `eclipse.ini` file. To turn off the notification, open the `eclipse.ini` file and add this line:

```
-Dnd.disableLicenseCheckWarning=true
```

To turn the notification on again, change the value to `false` or remove the flag from the `eclipse.ini` file.

Testing the Blaze Advisor Eclipse Plug-in

You can test the Eclipse plug-in installation by importing a project from the Examples Repository. You can find additional information about the examples in the Help system.

To import a project from the Examples Repository

1. In Eclipse, select **Window > Perspective > Open Perspective > Other**.
2. In the **Open Perspective** dialog, select **Blaze Advisor**.
3. Click **Open**.
4. Select **File > Import**.
5. Expand the **Blaze Advisor** folder and select **Rule Project**.
6. Click **Next**.
7. Select **Examples Repository** and click **Next**.
8. Expand the folder and locate the project.
9. Select the project and click **Finish**.

Installing a Deployment Kit on UNIX or IBM Platforms

This section describes how to install a Blaze Advisor Deployment kit on other supported platforms. It contains these topics:

Installing on Linux and UNIX Systems

This section explains how to install a Blaze Advisor Deployment kit on Linux and UNIX systems. Note that the Development kit is not supported for installation on Linux or UNIX systems.

This section contains these topics:

Invoking the Installer

For UNIX installations, the Blaze Advisor installer creates a temporary directory called ismp??? in the $HOME directory that it uses for temporary files. If you log in as root, the

$HOME may be "/" and the temporary directory is created at the "/" mount point. If there is not adequate disk space at the "/" directory, the installation will not proceed.

To invoke the installer

1. Download your Blaze Advisor release.
2. Locate `blaze-advisor_7.8.v(number)_Windows.bin`.
3. Open a shell window and run the `blaze-advisor_7.8.v(number)_Windows.bin`. For example:
   ```
   sh blaze-advisor_7.8.v(number)_Windows.bin
   ```
4. When the installer opens, read the text and click **Next**.
5. Select **Deployment** and click **Next.**
6. Select a Java Development Kit (JDK) or a Java Runtime Engine (JRE) to use as the `<JAVA_HOME>` variable in the Blaze Advisor scripts. Select one from the list or click **Choose Java Executable** to select one from your machine. If you do not select one before you click **Next,** the Java executable at the top of the list is used by default.
7. Read the license and click **I accept the terms of the License Agreement**.
8. Do one of the following to specify where you want Blaze Advisor installed.
   - Enter the full path to the directory, and click **Next.**
   - Click **Choose** and choose the directory, and click **Next.**

     To use the default location, change the installation directory from `c:\Blaze` to `/usr/local/Blaze`.
9. Enter the license keys now or wait until after installation is complete.

   For example, if the development license key is NdDeployment=12-123-5-6abc-ABCDEFGHI1234-1234-12345678, enter only the numbers, letters, and the hyphens.

   To add the license after the installation is complete, see **"Adding or Updating Licenses After Installation on Microsoft Windows" on page 12**.
10. Review the details of the installation and do one of the following:
    - Click **Install** to install Blaze Advisor.
    - Click **Previous** to return to the page where you want to make a change. Make your change, click **Next** until you return to the last wizard page, and then click **Install.**
11. After the installer finishes, click **Done**.

Installing Blaze Advisor Without UI Support

When you install Blaze Advisor using without the UI support, you may see warning messages during the installation. These warnings are benign and can be ignored. Also, for the license text, you may need to press **Enter** several times in order to read through the license.

To install Blaze Advisor without the UI support

- Open a shell window and enter the following:
  ```
  blaze-advisor_7.8.v(number)_Windows.bin -i silent
  -DUSER_INSTALL_DIR="$BLAZE_LOC"
  ```
  `-DUSER_INSTALL_DIR` is used to specify the installation directory.

Adding or Updating the License After Installation on Linux and UNIX

If the license key was not entered during installation, enter it using the license utility. You can also use this utility to enter any corrections to the license key.

To add or update licenses on Linux and UNIX

1. Open a shell window.
2. Locate the `<ADVISOR_HOME>\bin` directory.
3. Add executable permissions to the `.sh` files by entering:
   ```
   chmod +x *.sh
   ```
4. Enter:
   ```
   sh installLicense.sh
   ```
5. In the utility, enter the Deployment license.
6. Click **Apply**, and then click **OK.**

Verifying a Blaze Advisor Installation on Linux and UNIX

Blaze Advisor includes a script you can use to verify the installation. After the script runs, the following information is printed in the window:

- Java Virtual Machine (JVM) information, version number, vendor number, and OS
- Classpath
- FICO products that are installed.
- Product version, build number, and build date for each product or module
- License information

To verify the Blaze Advisor installation on Linux and UNIX

1. Locate the `<ADVISOR_HOME>\bin` directory.
2. Run the installation verification script:
   ```
   sh verifyInstall.sh
   ```
   If the installation was successful, this message appears when the script finishes:
   ```
   Blaze Advisor has successfully been installed.
   ```

Installing on IBM z/OS

This section explains how to install a Blaze Advisor Deployment kit on the IBM z/OS system. It contains these topics:

Transferring the Installation Files to the z/OS system

FICO recommends that you install a Blaze Advisor Deployment kit on a Microsoft Windows machine using a deployment license. After the installation is complete, transfer the Blaze Advisor installation files to the IBM z/OS machine.

To transfer the installation files

1 Install Blaze Advisor on a Microsoft Windows machine. See **"Installing a Development or Deployment Kit on Microsoft Windows" on page 11**. When you use the Blaze Advisor installer, install a Deployment kit.

Enter the license during installation or enter it after the installation by using the license utility. See **"Adding or Updating Licenses After Installation on IBM z/OS" on page 26**. Be sure to enter the license before transferring the license file to the z/OS machine.

2 Create the following directories on the z/OS machine.

- `<ADVISOR_HOME>/bin`
- `<ADVISOR_HOME>/lib`

3 Locate these Blaze Advisor shell files from the installation on your local machine:

Files	Comments
bin/ndrimiregistry.sh	Required for remote debugging purposes.
bin/setenv.sh	Always required.
bin/verifyInstall.sh	Always required.

You may need to edit the shell scripts to correspond with the location of your Blaze Advisor installation.

4 Transfer these files to the z/OS machine in ASCII format only. This example shows one way to transfer the files:

a Enter your user name and password.
```
ftp> open <hostname>
```

b Change the directory to the remote install location.
```
ftp> cd <your dir>
```

c Change to the local install location.
```
ftp> lcd <local installdir>
```

d Switch to ASCII mode.
```
ftp> ascii
```

e Turn off the same question for each file.
```
ftp> prompt
```

f Execute the command to transfer the files.
```
ftp> mput *
```

5 Locate these files from the Blaze Advisor installation on your local machine to transfer to the OS/390 or z/OS machine:

Java Archive and Zip File	Comments
lib/AdvisorSvr.jar	Always required.
lib/Advisor.jar	Always required.

Java Archive and Zip File	Comments
lib/AdvCommon.jar	Always required.
lib/AdvTest.jar	Only required if you want to use the brUnit Module.
lib/AdvRmaServer.jar	Only required if you want to deploy rule maintenance applications.
lib/InnovatorRT.jar	Required for access to repositories.
lib/jaxen-1.1.4.jar and saxpath.jar or lib/xalan-2.7.jar	The default XPath engine for the XML BOM is Jaxen. If you have not changed the XPath engine, jaxen-1.1.4.jar and saxpath.jar are required. If these files are not transferred over, Blaze Advisor's wrapper around the Java XPath engine is used. If you are using the Xalan XPath engine, the xalan-2.7.jar is required. If you are using Blaze Advisor's wrapper around the Java XPath engine, no files need to be transferred.
lib/mongo-java-driver-3.12.10.jar	Required if a MongoDB repository is being used.
lib/xerces2.jar	xerces2.jar is only required if your environment does not include an XML processor.
lib/util.jar	Required for installation verification procedure.
lib/wstx.jar	SAX parser that is used for cut/copy/paste functionality.

6. Transfer these files to the z/OS machine in binary mode. This example shows one way to transfer the files:
   a. Change to the local install location (if you are not already in that location).
      ```
      ftp> lcd <local installdir>
      ```
   b. Switch to binary mode.
      ```
      ftp> binary
      ```
   c. Turn off the same question for each file.
      ```
      ftp> prompt
      ```
   d. Execute the command to transfer the files.
      ```
      ftp> mput *
      ```
7. Locate the license key file:
   ```
   cd license/com/blazesoft/licenseKeys
   ```
8. Transfer the license key file to the z/OS machine platform in ASCII format only.
9. Edit the `setenv.sh` file to set the CLASSPATH and other variables correctly for this environment.

   Make sure that the license folder in your Blaze Advisor home directory is part of the CLASSPATH. This is required to locate the license key file.

Transferring Rule Project and Rule Server Files to the IBM z/OS

This section provides information on how to transfer rule projects, rule server (.server) and deployment manager (.dmanager) configuration files to an IBM z/OS system.

To transfer rule projects, you must first export the projects as Blaze Advisor projects on your Microsoft Windows machine. A Blaze Advisor project is composed of project resource files such as (`.adv`, `.jcp`, `.ccp`, `.dbcp`, `.xcp`, `.rb`, `.flow`).

If you use FTP in binary mode to transfer a project resource file, a rule server, or a deployment manager configuration file, it is readable by Blaze Advisor, but it cannot be edited in an z/OS text editor. If you transfer the configuration file in ASCII mode, the FTP program automatically converts the file to EBCDIC. The configuration file becomes readable by an z/OS editor. However, you must specify the correct encoding in the configuration file so that the XML parser can read it as well. You can do this by adding the encoding attribute to the configuration file. For example:

```
<?xml version="1.0" encoding="ebcdic-cp-us" ?>
```

Your FTP program can be configured on the z/OS side to use a certain codepage for converting ASCII into EBCDIC. Make sure that the configuration file encoding is set to the same codepage.

Blaze Advisor uses Xerces as the default parser. Xerces supports the "ebcdic-cp-us" encoding, which is equivalent to the CP037 encoding on the z/OS, as well as several other EBCDIC encodings. However, if your z/OS is configured to use Latin 1 EBCDIC encoding (CP1047) or some other format, then Xerces is not able to read it unless you convert it to CP037.

When you transfer the Blaze Advisor project files, the .adv file should always be transferred in ASCII mode. How you transfer `.jcp`, `.ccp`, `.dbcp`, `.xcp`, `.rb`, and `.flow` files depends on whether or not the resources are compressed. If you selected the **Compress resources** option for the project in the Blaze Advisor IDE, the project resources are compressed, and should be transferred in binary mode. (To set the option in the Blaze Advisor IDE, select Project > Properties > Engine/Project Optimization.) Otherwise, they should be transferred in ASCII mode. Note that `.class` files should always be transferred in binary mode.

If you create a compressed resource on a UTF8 platform, and want to read it on an EBCDIC platform, you need to specify the encoding in the exported Blaze Advisor project (`.adv`) file. You need to do this even though the `.jcp` file is binary because the content originates from ASCII/UTF8.

You can add the encoding for each resource in the exported project (`.adv`) file. To do this, open the `.adv` file in a text editor, and insert this line for each resource:

```
charset: UTF8
```

Example

Blaze Advisor automatically creates the first, second, and fourth lines in the `.adv` file when a resource is compressed. You just need to insert `charset: UTF8` as the third line, as shown below:

```
resource: StringTokenizer.jcp
class: com.blazesoft.engines.rules.java.NdFileJavaClassProviderResource
charset: UTF8
compressionType: Zip
```

Repeat this step for each resource in the `.adv` file.

Adding or Updating Licenses After Installation on IBM z/OS

If you did not provide your deployment license during installation, you can enter it in a utility after installation. You can also use this utility to enter any corrections to the license.

To add or update your license

1. Return to your Microsoft Windows machine.
2. Open a Command Prompt window, locate the `<ADVISOR_HOME>\bin` directory, and enter:
   `installLicense.bat`
3. In the utility, enter your Deployment license in the **Blaze Advisor Deployment** field.
4. Click **Apply**, then click **OK.**
5. Transfer the license file from the Microsoft Windows machine in ASCII mode to the z/OS machine. See **"Installing on IBM z/OS" on page 22**.

Verifying a Blaze Advisor Installation on IBM z/OS

Blaze Advisor includes a script you can use to verify the installation. After the script runs, the following information is printed in the window:

- Java Virtual Machine (JVM) information, version number, vendor number, and OS
- Classpath
- FICO products that are installed
- Product version, build number, and build date for each product or module
- License information

To verify the Blaze Advisor installation

1. Locate the `<ADVISOR_HOME>\bin` directory.
2. Run the `verifyInstall.sh` script.
   If the installation was successful, this message appears when the script finishes:
   `Blaze Advisor has successfully been installed.`

Using the French Version of the RMA

Blaze Advisor includes localized strings for the RMA in French. To view the RMA strings in French, change the language setting in Microsoft Chromium Edge, Mozilla Firefox or Google Chrome.

To change the language setting

1. Close all RMA instances.
2. Open the browser.
3. Locate the window or panel (associated with your browser) where you can change the language preference and select `French (France) [fr_FR]`.

4 Relaunch one more instances of the RMA.

Setting Workspace Preferences and Project Properties

Blaze Advisor provides workspace preferences that you can set for your Eclipse workspace and properties that can be set for individual projects. For example, if you have projects that use the same Java classes, you can set the classpath in each project using the Properties window or you can set the classpath globally using the Preferences window.

Review the options available for the workspace and for projects before importing existing projects or creating new ones. See these topics:

- "Setting Workspace Preferences for Blaze Advisor" in *DevelopingRuleProjects.pdf*.
- "Setting Properties for Blaze Advisor Projects" in *DevelopingRuleProjects.pdf*.

Using Rule Maintenance Applications Created in Prior Releases

Rule maintenance applications created prior to Blaze Advisor 7.8 must be regenerated in order to use the new RMA features available in this release.

Uninstalling Blaze Advisor

This section explains how to uninstall Blaze Advisor on various platforms. Blaze Advisor includes a utility to uninstall the product on Microsoft Windows and on UNIX.

This section contains these topics:

Note: After you uninstall, do not install a new version of Blaze Advisor to the directory where you have previously uninstalled the product. Install a new version of Blaze Advisor to an empty directory.

Uninstalling on Microsoft Windows

Use the Uninstaller to uninstall Blaze Advisor.

To uninstall using the Uninstaller utility

1 Double-click the `Uninstaller` utility under `<Blaze>\Uninstall` directory.

2 When the Uninstall wizard appears, click **Uninstall**.

3 After the operation has completed, click **Done.**

Uninstalling the Blaze Advisor Plug-ins from Eclipse

Use the Eclipse SDK Installation Details window to uninstall Blaze Advisor and any of the optional Blaze Advisor plug-ins.

To uninstall the Blaze Advisor plug-ins

1. Select **Help > About Eclipse SDK** or select **About Eclipse IDE.** (The command name depends on the Eclipse version you are using.)
2. Click **Installation Details**.
3. Select **Blaze Advisor IDE** and **Blaze Advisor libraries** and any other Blaze Advisor features that are currently installed. This enables the Uninstall button.
4. Click **Uninstall**.
5. Click **Finish**. A dialog box appears asking you whether or not you want to restart Eclipse.
6. Click **Restart Now** to close and restart Eclipse.

Note: FICO recommends creating a new instance of Eclipse to install another version of the Blaze Advisor Eclipse plug-in.

Uninstalling on UNIX

Use the uninstaller.jar file to uninstall Blaze Advisor.

To uninstall Blaze Advisor on UNIX

1. Open a shell window and change to the `uninst` directory:
2. Run the `uninstaller.jar` file:
   ```
   java -cp uninstaller.jar uninstall
   ```
3. Press **Enter.**
4. Follow the instructions on the screen.

CHAPTER 2

Integrating Third-party Tools with Blaze Advisor

Use the Eclipse Extension plug-in mechanism to integrate a third-party application, a database, or a source control management system with Blaze Advisor. The Blaze Advisor Eclipse extension plug-in allows you to load classes with dependencies on third-party JAR files in the Blaze Advisor IDE in Eclipse. You can also use this mechanism if you are using classes where there are dependencies on classes bundled with the Blaze Advisor plug-ins.

This section contains these topics:

To implement a Subversion repository, see "Implementing a Subversion Repository" in *ManagingRepositories.pdf* in the Blaze Advisor Help system or in `ManagingRepositories.pdf`.

Integrating Third-party Tools Using the Eclipse Extensions Plug-in

If you attempt to import the classes for a third-party application where there are dependencies using the Blaze Advisor Java Business Object Model Wizard, you may see an NullPointerException (NPE). This is due to the way that Eclipse loads external classes with dependencies. Eclipse uses the OSGi class loader that has special rules to load classes with bundles. The OSGi class loader does not rely on the classpath to load the classes, but loads them from different plug-ins (bundles), which specify the dependencies among themselves. When Blaze Advisor attempts to load the classes, this in turn loads other classes. All these classes have to be found by a single OSGi class loader and if some of these classes are unable to load, an NPE is thrown.

If you want to integrate third-party applications with Blaze Advisor, use the `com.fairisaac.eclipse.extensions plugin` to make sure that all classes and their dependencies can be found by the OSGi class loader so that these classes can be used in the Blaze Advisor IDE in Eclipse. This plug-in is part of the Blaze Advisor installation.

If the classes you want to load using the Blaze Advisor Eclipse extensions plug-in has dependencies on Blaze Advisor classes, see **"Using Classes Bundled with Blaze Advisor" on page 30**.

To integrate third-party tools with Blaze Advisor IDE

1. Add the files to the <ADVISOR_HOME>/EclipseDropins/com.fairisaac.eclipse.extensions folder.

   For example, to add an Oracle database driver to the Blaze Advisor IDE, add the ZIP file to this directory.

2. In the MANIFEST.MF file located in <ADVISOR_HOME>/EclipseDropins/com.fairisaac.eclipse.extensions/META-INF, enter the ZIP or JAR file name to the Bundle-Classpath list. This is a comma-separated list. Do not include extra white space between the items in this list.

   Example:

   ```
   Manifest-Version: 1.0
   Export-Package: oracle.sql,oracle.jdbc.driver
   Bundle-Vendor: FICO
   Bundle-ActivationPolicy: lazy
   Bundle-ClassPath: .,oracle(number).zip
   Bundle-Version: 7.8.v(number)
   Bundle-Name: %Bundle-Name
   Bundle-ManifestVersion: 2
   Bundle-SymbolicName: com.fairisaac.eclipse.extensions
   Bundle-RequiredExecutionEnvironment: JavaSE-1.8
   ```

3. Add any packages you need to use in the Blaze Advisor IDE to the Export-Package list. This is also a comma-separated list.

4. Copy the entire com.fairisaac.eclipse.extensions folder to this directory:

   <ECLIPSE_HOME>/dropins directory.

   The <ECLIPSE_HOME> directory is the location of your Eclipse installation, for example C:/eclipse.

5. Restart Eclipse.

Adding a JavaWrapper Class

If you have a JavaWrapper class, ensure that the class is in a named package and put it in a JAR file. The OSGi bundles do not allow referencing classes from the default package. After you have created the JAR file, you can add it to the Blaze Advisor Eclipse Extensions plug-in, add it to the Export-Package list in the MANIFEST.MF file and copy the folder to the <ECLIPSE_HOME>/dropins directory. If you need to use the class in the Blaze Advisor project, import the class using the Java Business Object Model Wizard.

Using Classes Bundled with Blaze Advisor

If the classes you want to load using the Blaze Advisor Eclipse extensions plug-in have dependencies on classes bundled with the Blaze Advisor plug-ins, add the following line to the MANIFEST.MF file located in <ADVISOR_HOME>/EclipseDropins/com.fairisaac.eclipse.extensions/META-INF:

Example:

```
Manifest-Version: 1.0
Export-Package: oracle.sql,oracle.jdbc.driver
Bundle-Vendor: FICO
Bundle-ActivationPolicy: lazy
Eclipse-BuddyPolicy:global
Bundle-ClassPath: ,oracle(number).zip
```

```
Bundle-Version: 7.8.v(number)
Bundle-Name: %Bundle-Name
Bundle-ManifestVersion: 2
Bundle-SymbolicName: com.fairisaac.eclipse.extensions
Bundle-RequiredExecutionEnvironment: JavaSE-1.8
```

The `Eclipse-BuddyPolicy:global` tells Eclipse to look for classes in the calling plug-in which is `com.fairisaac.blazesoft.advisor`.

CHAPTER 3

Issues with Java that Affect Blaze Advisor

This section describes known issues with the Oracle Java Development Kit (JDK) that affect the FICO® Blaze Advisor® decision rules management system.

Before troubleshooting issues, FICO recommends that you confirm that both Blaze Advisor and Eclipse are using a supported version of a Java Virtual Machine. See **"Verifying the Java Versions for Blaze Advisor and Eclipse"**.

Problems Resulting from an Incompatible Java Virtual Machine

If you installed the Blaze Advisor Eclipse plug-in successfully and then encountered a "Trying to execute the disabled command" error or have problems opening a project in the IDE, this problem may be due to the use of an incompatible Java Virtual Machine (JVM).

You must specify a JDK version on which Eclipse runs by adding a path to the JDK in the `eclipse.ini` file. FICO recommends that Blaze Advisor and Eclipse use the same JDK version.

The `eclipse.ini` file is located in the directory where you installed Eclipse, also known as `<ECLIPSE_HOME>`.

To edit the eclipse.ini file

1. Open the `eclipse.ini` file in a text editor.

   Locate the `-vm` option.

2. Under the `-vm` option, replace the default path with a path to a specific JDK executable.

   You can determine this path by searching for `JAVA_HOME` in the `setenv.bat` file under `<ADVISOR_HOME>\bin`. If the path displayed in this file is not the complete path to a JDK executable, for example, `C:\Program Files\Java\jdk-11.0.12`, append `\bin` to the path and specify the JDK executable in the `\bin` directory. For example:

   ```
   C:\Program Files\Java\jdk-11.0.12\bin\javaw.exe
   ```

Changes in Locale Load Order

In Java 11, the Unicode Common Locale Data Repository (CLDR) provides a mechanism to support languages around the world with the largest and most extensive standard repository of locale data available. The CLDR was first added to Oracle JDK 8 and is now the default provider for Java 11. This change in locale provider load ordering may affect display formats and parsing behavior for dates, times, timestamps, as well as numeric data types.

Existing Blaze Advisor users who are impacted by this change can revert to the provider used in prior releases by specifying a load ordering through the `java.locale.providers` Java system property. The following setting will restore the provider load order to what it was prior to the use of CLDR as the default provider:

```
java.locale.providers=COMPAT, CLDR, SPI
```

When and Where Blaze Advisor Users Should Override Locale Provider Load Order

Generally speaking, this Java system property should be specified wherever a Java Virtual Machine (JVM) is started. Below is a list of some common locations:

- Generated RMA “startserver” startup script.
- Eclipse IDE `eclipse.ini` file.
  - Affects Blaze Advisor IDE users.
  - RMA users for RMAs launched from within the IDE.
- Generated Blaze Advisor deployment “Run” startup scripts.

For more information about these internationalization enhancements, see:

https://docs.oracle.com/javase/9/intl/internationalization-enhancements-jdk-9.htm#JSINT-GUID-974CF488-23E8-4963-A322-82006A7A14C7

Java Garbage Collection Issue with Java 11

For rule services executed on Java 11 or later, the default garbage collector algorithm is G1GC (Garbage-First Garbage Collector), which is designed to run concurrently with other threads and aimed at providing quick release of garbage objects and minimum GC pauses.

Although this GC keeps memory usage more constant and mostly eliminates GC pauses, this comes at a cost of a moderate loss of throughput (about 20% in average in our tests), so may not be the optimal GC option for services that need to maximize throughput. For those, FICO recommends that you explicitly select the older "Parallel" GC, which was the default GC until Java 8, using the JVM option `-XX:+UseParallelGC`.

For more information, see:

Java 8: **https://docs.oracle.com/javase/8/docs/technotes/guides/vm/gctuning/collectors.html#sthref28**

Java 11: **https://docs.oracle.com/en/java/javase/11/gctuning/available-collectors.html#GUID-F215A508-9E58-40B4-90A5-74E29BF3BD3C**

Stricter Access Rules in the Java Platform Module System

Starting with Java 9, stricter access rules are implemented in the Java Platform Module System. These restrictions can affect any Blaze Advisor entity, including the Blaze Advisor IDE, the RMA, Blaze Advisor server deployments, and so on.

If you encounter illegal reflective access errors while using Blaze Advisor, you can work around these issues by configuring module access through arguments passed to the Java Virtual Machine (JVM) at the time it is invoked.

FICO recommends that you report all illegal reflective access errors as follows:

- For errors that prevent you from accessing a Blaze Advisor class, contact **http://www.fico.com/support**.
- For errors that prevent you from accessing a third-party class, contact the third-party developer.

For information about the Java Platform Module System and its access rules, go to: **https://cr.openjdk.java.net/~mr/jigsaw/spec/**.

CHAPTER 4

Issues with Eclipse that Affect Blaze Advisor

This section describes known issues with Eclipse that affects the Blaze Advisor IDE in the FICO® Blaze Advisor® decision rules management system. This section contains these topics:

Underlying Eclipse Name of Blaze Advisor Project Items Appears in Some Eclipse Editors

In some of the native Eclipse editors and screens, Blaze Advisor projects are referred to by their underlying Eclipse name instead of the display name in the Project Explorer. For example, if a project is run and there is an error in a function called `main`, it appears as `main.advprom` in the **Resource** column in the Console. This is a display issue only and does not affect the Blaze Advisor project.

Errors Seen in Eclipse Error Log When Search is Used in the Help

It is a known defect that when you use the Help > Search functionality in Eclipse, the following errors appear in the Error Log:

```
Help documentation could not be indexed completely.
```

```
Unable to consume Lucene index from bundle 'version number'. The
index should be rebuilt with Lucene 6.1.
```

This issue has been reported to the Eclipse foundation and is yet to be fixed. The bug can be found here: **https://bugs.eclipse.org/bugs/show_bug.cgi?id=466237**

Errors Seen in Eclipse Error Log When Opening an HTML Example Document

Most of the Blaze Advisor examples include an HTML document that explains how the example works. When using Eclipse, errors may appear if you open one of the HTML documents in Eclipse. This is the first error in a sequence of related error messages:

```
No line style provider for org.eclipse.wst.html.SCRIPT.EVENTHANDLER
```

This issue has been reported to the Eclipse foundation and is yet to be fixed. The bug can be found here: **https://bugs.eclipse.org/bugs/show_bug.cgi?id=566686**

Illegal Reflective Access Errors

The Java Virtual Machine (JVM) that is packaged with Eclipse is known to cause illegal reflective access errors when used with the Blaze Advisor Eclipse plug-in. To work around this issue, you can configure module access rules for the virtual machine in the `eclipse.ini` file. For more information, see **“Modifying the Eclipse Configuration File” on page 16**.

CHAPTER 5

Issues with Microsoft Windows that Affect Blaze Advisor

This section describes known issues with Microsoft Windows that affects the FICO® Blaze Advisor® decision rules management system. It contains these topics:

Full Path Exceeds Windows Maximum Length

While performing some editing or versioning operations in a Blaze Advisor file repository, you may encounter a problem on Windows if the full path of the item in the native file system exceeds the Windows maximum length of 260 characters (defined as the constant MAX_PATH in the Windows APIs). Symptoms of the problem include: error messages indicating that a file path is too long or error messages indicating that a template or instance cannot be found when you know the file should be present in the current project. If such symptoms occur on a deeply nested item in the repository, verify whether the full path exceeds that limit.

To workaround this problem, you can map the folder containing the repository or workspace to a Windows network drive and refer to the repository/workspace using this network drive. That will shorten the path used for the Blaze Advisor operation. Another recommended workaround for rule maintenance applications (RMAs) generated from repositories with BVS private workspaces is to specify an explicit, absolute location for the 'workspaces' directory (the directory containing user workspaces) located near the root of the file system.

Keyboard Shortcut May Bring Up Other Languages

If you have additional languages installed on your machine, using Ctrl + Space in fields created by the Free-form Code editor may appear in another language other than the one selected in the Region and Language dialog in the Control Panel.

While using Ctrl + Space is the standard keyboard binding for code completion for most IDEs, Microsoft chose to use the same keyboard binding for switching languages. This problem occurs only when the Language Settings toolbar is enabled and there is more than one language installed on your machine.

To work around this issue, select **Programs > Control Panel > Region and Language > Keyboards and Languages** and customize the keyboard binding for the language switch by selecting a different key combination.

Index

I

J

K

L

M

O

P

S

U

V