

Using Blaze Rules with BPM and CAF

Version 9.6

April 2014

This document applies to webMethods Product Suite Version 9.6 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2009-2014 Software AG, Darmstadt, Germany and/or Software AG USA Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://documentation.softwareag.com/legal/>.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at <http://documentation.softwareag.com/legal/> and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices and license terms, please refer to "License Texts, Copyright Notices and Disclaimers of Third Party Products". This document is part of the product documentation, located at <http://documentation.softwareag.com/legal/> and/or in the root installation directory of the licensed product(s).

Table of Contents

About this Guide.....	5
Document Conventions.....	5
Documentation Installation.....	6
Online Information.....	6
Getting Started.....	9
Special Considerations Regarding Version Compatibility.....	10
Using Blaze Rules Development for Software AG Designer.....	11
About Blaze Advisor Rules.....	11
Deploying Blaze Rule Projects to webMethods Servers.....	12
Deploying RMAs to My webMethods Server.....	13
Types of Users Who Interact with Blaze Rules.....	16
Storing Blaze Rules and Controlling Versions.....	16
Before You Can Use Blaze Rules in a webMethods Process.....	17
Installing Blaze Rules Development for Software AG Designer.....	18
Defining the Blaze Advisor Installation Directory.....	18
Adding JDBC Files.....	19
Changing the Logging Level of the Blaze Eclipse Plug-in.....	19
Upgrading to Blaze Advisor 7.1.....	19
Upgrading from Version 8.2 SP2 and Earlier.....	20
Upgrading from Version 9.0 SP1.....	21
Setting Up the Blaze Repository.....	23
Supported Blaze Repository Types.....	24
Creating a Blaze Database Repository.....	24
Creating the Database Schema and Repository Separately.....	25
Creating the Database Schema and Repository at the Same Time.....	25
Preparing to View Rules for the First Time.....	28
Using Blaze Rules in Business Processes.....	31
Overview.....	32
Creating Blaze Projects.....	32
Defining Server Connection Properties for Blaze Projects.....	33
Using IS Document Types in Blaze Rules.....	34
Supported Data Types in Blaze Advisor.....	35
Creating Blaze Custom Business Object Models.....	36
Returning the Results of a Decision Table or Tree as an IS Document Type (webMethods Version 8.0 and Later).....	37
Guidelines for Blaze Rules Used in webMethods Processes.....	38
Rule Inputs and Outputs.....	38
Deploying Rules to the Integration Server.....	39
Adding Rule Steps to a Process Model.....	40

Before You Can Manage Blaze Rules.....	41
Requirements for Managing Blaze Rules in My webMethods.....	42
Granting Users Permission to Manage Blaze Rules.....	42
Configuring the My webMethods Server Connection for a Blaze Project.....	44
Updating the Integration Server and Repository Credentials for an RMA.....	44
Updating the Integration Server.....	45
Updating the Database Repository.....	45
Checking the Project into the Blaze Versioning System.....	46
Deploying RMAs to the My webMethods Server.....	46
Managing Blaze Rules in My webMethods.....	49
Overview.....	50
Viewing Rules in My webMethods.....	50
Viewing the Version History of a Rule Instance.....	51
Updating a Rule Instance.....	51
Creating a New Rule Instance.....	53
Deleting a Rule Instance.....	54
Deleting an RMA.....	54
Blaze Rule KPIs.....	55
Viewing Blaze Rule KPIs.....	56
Index.....	59

About this Guide

This guide describes how to deploy Blaze rules created in Blaze Advisor to a webMethods Integration Server so that the rules can be used in a webMethods BPM process. The guide also describes how to manage rules dynamically in My webMethods and to view key performance indicators that illustrate how changes to rules affect business processes.

To use this guide effectively, you should understand structured rule language (SRL) and know how to use Blaze Advisor to develop rule projects.

Detailed information about FICO Blaze Advisor is beyond the scope of this guide. You can find FICO Blaze Advisor documentation in the following locations:

- *Software AG_directory*\blaze\Advisor71\doc
- Software AG Documentation website

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Narrowfont	Identifies storage locations for services on webMethods Integration Server, using the convention <i>folder.subfolder:service</i> .
UPPERCASE	Identifies keyboard keys. Keys you must press simultaneously are joined with a plus sign (+).
<i>Italic</i>	Identifies variables for which you must supply values specific to your own situation or environment. Identifies new terms the first time they occur in the text.
Monospace font	Identifies text you must type or messages displayed by the system.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.

Convention	Description
	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the symbol.
[]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Documentation Installation

You can download the product documentation using the Software AG Installer. The documentation is downloaded to a central directory named `_documentation` in the main installation directory (SoftwareAG by default).

Online Information

You can find additional information about Software AG products at the locations listed below.

If you want to...	Go to...
Access the latest version of product documentation.	Software AG Documentation website http://documentation.softwareag.com
Find information about product releases and tools that you can use to resolve problems. See the Knowledge Center to:	Empower Product Support website https://empower.softwareag.com
<ul style="list-style-type: none"> ■ Read technical articles and papers. ■ Download fixes and service packs (9.0 SP1 and earlier). ■ Learn about critical alerts. See the Products area to:	<ul style="list-style-type: none"> ■ Download products.

If you want to...	Go to...
<ul style="list-style-type: none"> ■ Download certified samples. ■ Get information about product availability. ■ Access older versions of product documentation. ■ Submit feature/enhancement requests. 	
<ul style="list-style-type: none"> ■ Access additional articles, demos, and tutorials. ■ Obtain technical information, useful resources, and online discussion forums, moderated by Software AG professionals, to help you do more with Software AG technology. ■ Use the online discussion forums to exchange best practices and chat with other experts. ■ Expand your knowledge about product documentation, code samples, articles, online seminars, and tutorials. ■ Link to external websites that discuss open standards and many web technology topics. ■ See how other customers are streamlining their operations with technology from Software AG. 	<p>Software AG Developer Community for webMethods</p> <p>http://communities.softwareag.com/</p>

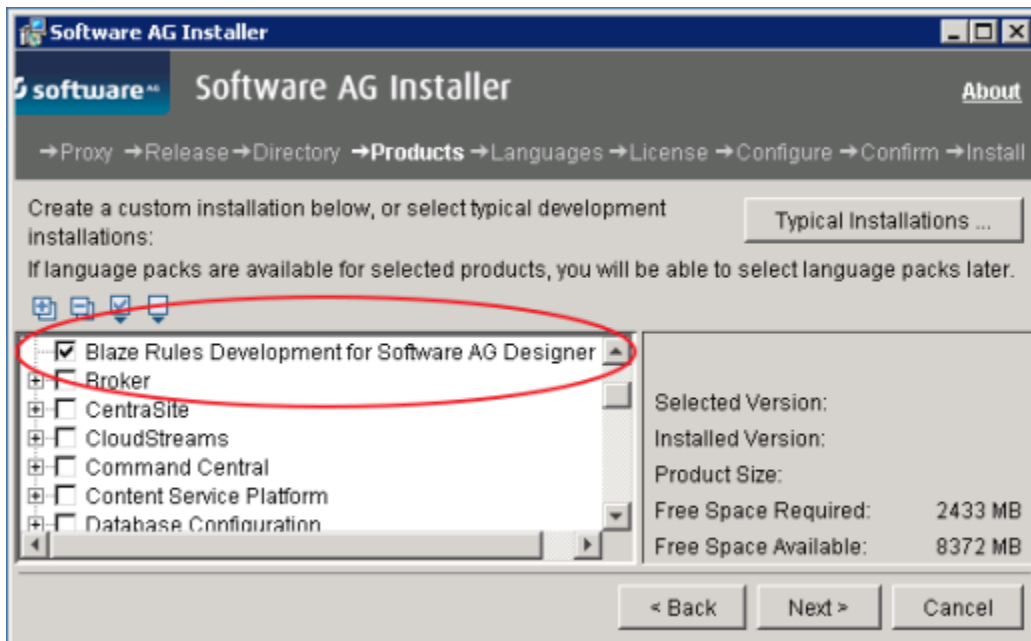
1 Getting Started

■ Special Considerations Regarding Version Compatibility	10
■ About Blaze Advisor Rules	11
■ Deploying Blaze Rule Projects to webMethods Servers	12
■ Types of Users Who Interact with Blaze Rules	16
■ Storing Blaze Rules and Controlling Versions	16
■ Before You Can Use Blaze Rules in a webMethods Process	17
■ Upgrading to Blaze Advisor 7.1	19

Special Considerations Regarding Version Compatibility

Software AG Designer version 9.5.1 and later is built on Eclipse v.4.3. However, the FICO Blaze Advisor 7.1 Eclipse plug-in is not compatible with Eclipse 4.3, thus preventing users from being able to use FICO Blaze Advisor in Software AG Designer 9.5.1 or later.

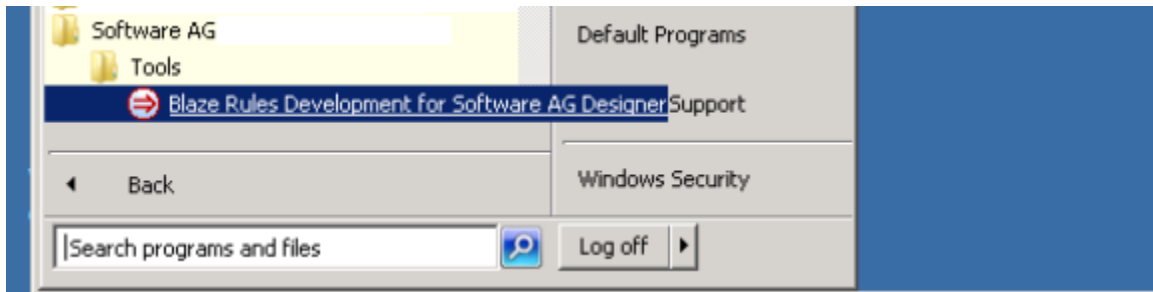
To enable users to work with FICO Blaze Advisor 7.1, Software AG has created a separate installable component, based on Eclipse 3.6, that contains only Blaze Advisor 7.1. This application, known as Blaze Rules Development for Software AG Designer, is available for installation in the Software AG Installer as a separate product node (that is, it is not listed under Designer):



The installation places application files in the same location as your Blaze installation, *Software AG_directory\blaze*.

For detailed information, see "[Installing Blaze Rules Development for Software AG Designer](#)" on page 18.

After installation, the Blaze Rules Development for Software AG Designer application is available from the following Software AG menu:



The first time you start Blaze Rules Development for Software AG Designer, you can accept the default workspace location or specify a different workspace location. When the application opens, the only available perspectives are the Blaze Advisor perspective and the standard Eclipse Java perspective. Online help is available from **Help > Help Contents**. Expand the **FICO Blaze Advisor** help contents to view the available help files.

Note: If you are upgrading to Software AG Designer 9.5.1 or later from version 9.0 SP1 or earlier, see "[Upgrading to Blaze Advisor 7.1](#)" on page 19.

Using Blaze Rules Development for Software AG Designer

Use the Blaze Rules Development for Software AG Designer application to develop all of your Blaze rules projects, as described in this product guide. To work with all other Software AG Eclipse-based products, such as Service Development, UI Development, and Process Development, you must start and work in Software AG Designer. You can have both applications open at the same time.

In addition, it is possible to switch workspaces using the **File > Switch Workspaces** menu command. You can use this feature in Software AG Designer to access and load the rule projects you have developed in the Blaze Rules Development for Software AG Designer application.

About Blaze Advisor Rules

For complex decision making within a business process, you can use *Blaze rules*, which are business rules that you create in FICO Blaze Advisor integrated with Blaze Rules Development for Software AG Designer. A rule is a statement that defines what action should occur for a given set of conditions. For example, for a medical insurance claim, you could define a rule that determines the percentage of coverage for an office visit based on whether the patient was treated by a health care provider who is in or out of the approved network.

In Blaze Advisor, you can group related rules into a *ruleset*, which enables you to maintain a group of rules that work together to achieve a particular business goal. You can also create *templates*, which contain rule definitions and placeholders for actual values that are passed to the rule for evaluation when the business process executes. Rules, rulesets, templates, and other entities that support the rules, such as currency

type or locale settings, are stored in a Blaze Advisor *project*. For more information about Blaze project components, see the FICO publication about developing rules projects.

You can use Blaze rules in webMethods BPM processes by deploying rules from a Blaze Rule project in Blaze Rules Development for Software AG Designer to Integration Server. You can also deploy *Rule Maintenance Applications (RMAs)* to My webMethods Server, where you can edit and deploy the rules to Integration Server.

Deploying Blaze Rule Projects to webMethods Servers

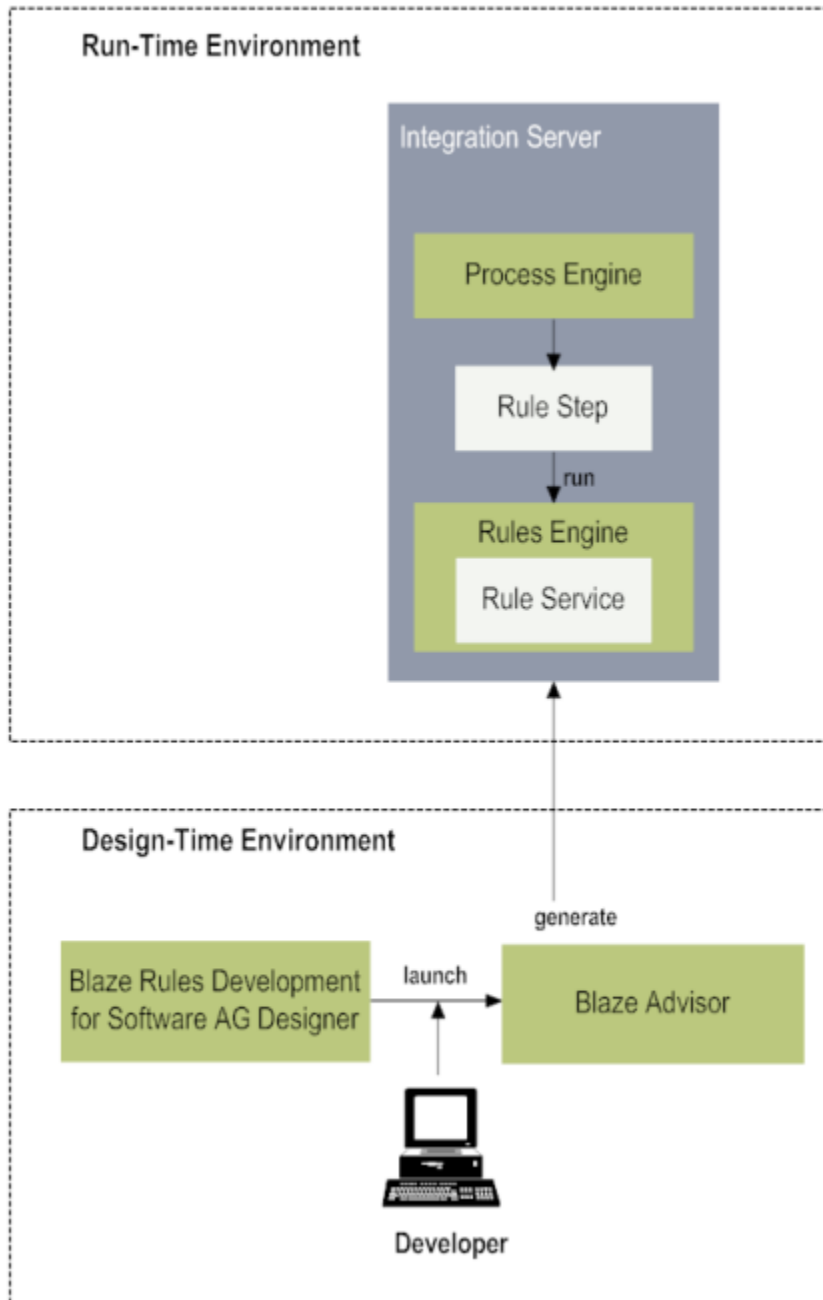
Before you can use Blaze Advisor rules in a webMethods BPM process, you must first deploy the project containing the rules to Integration Server. If you want other users to be able to modify rule instances without having Blaze Advisor installed, you can also deploy RMAs that define the rules to My webMethods Server.

Note: You can only modify rule instances that were created from a template.

Deploying Blaze Rule Projects to Integration Server

When you deploy a Blaze rules project from Blaze Advisor to Integration Server, the deployment creates an Integration Server package that contains the generated *rule services* and the Blaze configuration files to run them. After a rule service is created in Integration Server, you can then use the service as a rule in a process. When creating a process model in Software AG Designer, you can reference the rule service in a *rule step*. At run time to execute the rule step, the process invokes the rule service on Integration Server.

The following diagram illustrates the run-time and design-time environments.



For details about how to deploy rule projects to Integration Server, see ["Using Blaze Rules in Business Processes" on page 31](#). For more information about Blaze Advisor, see the FICO documentation.

Deploying RMAs to My webMethods Server

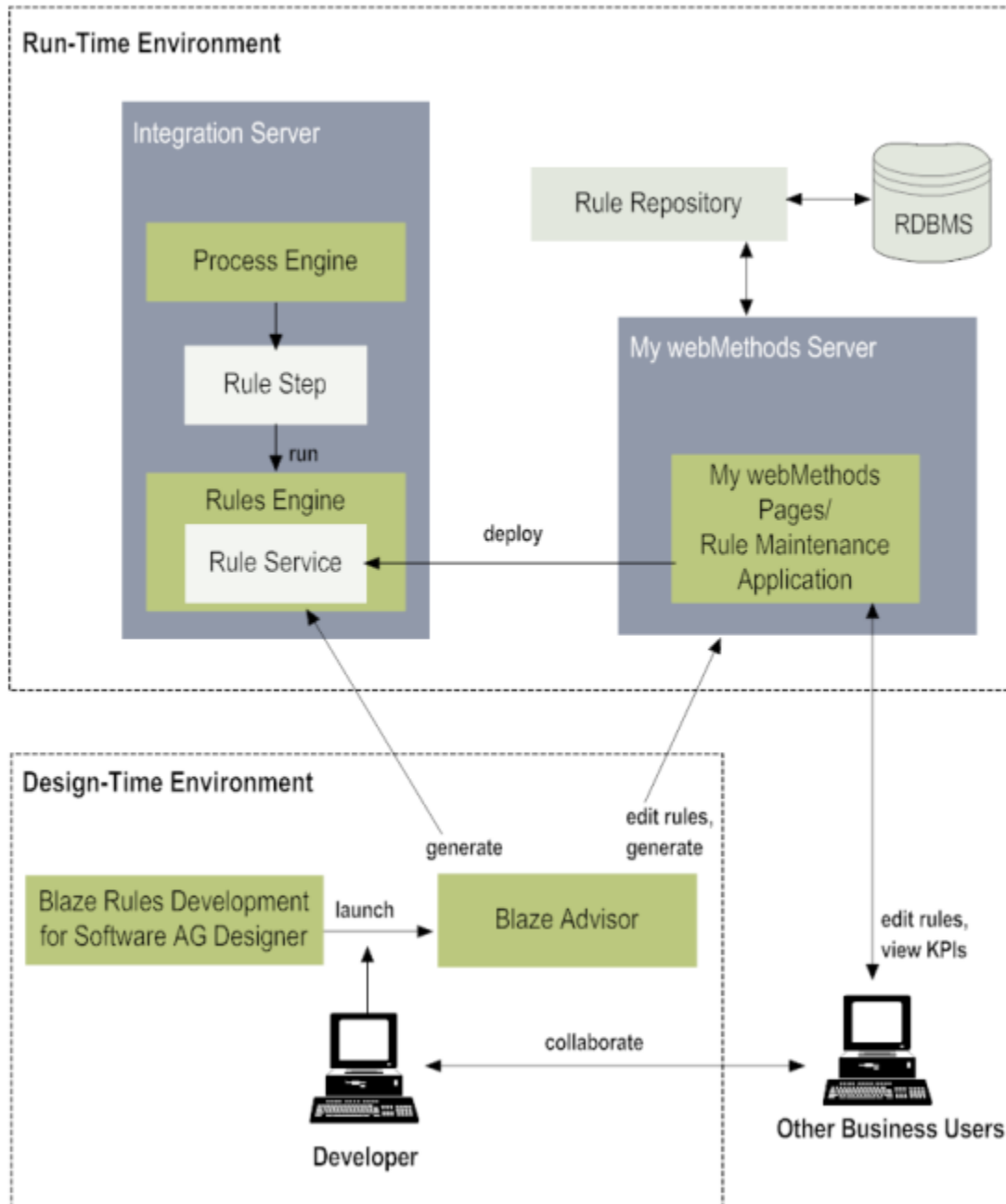
Business users can dynamically update business rules at run time. To enable this feature at design-time, deploy the Blaze rule project as a Rule Maintenance Application (RMA) to My webMethods Server. An RMA is a web application, stored as a .war file,

that contains information regarding Integration Server and the repository database connection used to deploy and manage rules.

Deploying RMAs from Blaze Advisor to My webMethods Server enables other users, such as business analysts, to:

- **Edit rule instances.** In My webMethods, users can edit rule instances that have been created from a template in a manner similar to how you would edit rules in a Blaze project. With the appropriate permissions, these users can then redeploy the rules from My webMethods so that business processes immediately begin using the updated rules.
- **View key performance indicators.** In My webMethods, business analysts can also view key performance indicators (KPIs) that indicate how changes to the rules affect business processes.

The following diagram illustrates this process.



In this scenario, rules and their modifications are stored in a database repository. Blaze Advisor and My webMethods Server connect to the same repository so that changes to rules and rule instances can stay synchronized between design-time and run-time environments. This way, developers who design and create rules can turn the rules over to business owners for day-to-day management.

For details about how to deploy RMAs to My webMethods Server and how to manage rule instances in My webMethods, see ["Before You Can Manage Blaze Rules" on page 41](#) and ["Managing Blaze Rules in My webMethods" on page 49](#). For more information about viewing Blaze rule KPIs, see ["Blaze Rule KPIs" on page 55](#).

Types of Users Who Interact with Blaze Rules

Three types of users can interact with Blaze rules: developers, administrators, and business analysts. The following table describes the typical set of tasks that each type of user may perform:

Typical Task	Developer	Administrator	Business Analyst
Create rules in a Blaze rules project using Blaze Rules Development for Software AG Designer.	•		
Generate and deploy rule projects from Blaze Advisor to Integration Server.	•	•	
Generate and deploy RMAs from Blaze Advisor to My webMethods.	•	•	
Add rules to process steps using Software AG Designer.	•	•	•
View, create, and modify rules, delete rule instances, and view history for rules using My webMethods.	•	•	•
Deploy modified rules from My webMethods to Integration Server.	•	•	•
View KPIs in My webMethods that illustrate how rule changes affect business processes.	•		•

Storing Blaze Rules and Controlling Versions

The Blaze rules that you define in Blaze Rules Development for Software AG Designer are stored in either a database repository or a file repository. Changes to rules are controlled through Blaze Advisor Versioning System.

To use Blaze rules with webMethods components, you must use the following combinations of repository type and versioning tool:

-
- To use Blaze rules with webMethods processes, use either a database or file repository with or without the Blaze Versioning System. However, when you want to manage and deploy the rules from Rule Maintenance Applications (RMAs) to Integration Server, use a database repository.
 - To use Blaze rules with webMethods processes AND dynamically update Blaze rule instances from My webMethods, use a database repository. When you use the Blaze Versioning System, you must use the *private workspace*.

A private workspace allows users to work on files in a workspace on a local directory. Users can update their local workspace to obtain changes other users have checked in to the shared repository. When users are ready for others to have access to their changes, they can check their changes into the shared repository.

You define whether you want to use versioning when you create the repository. For more information about BVS versioning, see *Implementing a Repository With a Blaze Versioning Service* in the FICO Blaze Advisor help, available from the **Help > Help contents** menu in Blaze Rules Development for Software AG Designer.

Before You Can Use Blaze Rules in a webMethods Process

Before you can use Blaze rules in a webMethods BPM process, you must perform the following steps:

1. Install Blaze Rules Development for Software AG Designer and Integration Server. My webMethods Server is required only if you want to deploy RMAs to My webMethods Server. Blaze Advisor is installed as a plug-in when you install Blaze Rules Development for Software AG Designer. For more information, see "[Installing Blaze Rules Development for Software AG Designer](#)" on page 18.
2. Configure Blaze Advisor. For more information, see "[Defining the Blaze Advisor Installation Directory](#)" on page 18.
3. Configure Blaze Rules Development for Software AG Designer to connect to the corresponding Blaze repository database. For more information, see "[Adding JDBC Files](#)" on page 19.
4. Change the logging level of the Blaze Eclipse plug-in. For more information, see "[Changing the Logging Level of the Blaze Eclipse Plug-in](#)" on page 19.
5. Create the Blaze repository. For more information, see "[Creating a Blaze Database Repository](#)" on page 24.

Installing Blaze Rules Development for Software AG Designer

To install Blaze Rules Development for Software AG Designer components using Software AG Installer

1. Make sure that you have installed Integration Server. Install My webMethods Server if you want to deploy RMAs to My webMethods Server.
2. Install the following components to use Blaze Advisor 7.1:
 - **Blaze Rules Development for Software AG Designer.**

This component installs Blaze Rules Development for Software AG Designer and enables you to define Blaze rules for processes and services using Blaze Advisor 7.1. Selecting this component also automatically selects the required supporting components, such as Blaze Advisor libraries.
 - **Integration Server > Blaze Rule Service Support 9.0 SP1**

This component supports Integration Server rule services and Blaze rule integration. You must manually select this component.
 - **My webMethods User Interfaces > Blaze 9.0 SP1**

This component enables you to view and modify rules created in Software AG Designer using the Blaze plug-in. Install this component only if you plan to deploy RMAs to My webMethods Server. You must manually select this component.

For more information about installing webMethods products in general, see the PDF publication.

Defining the Blaze Advisor Installation Directory

The first time you start Blaze Rules Development for Software AG Designer, you are prompted to define the *Software AG_directory* installation directory.

To define the Blaze Advisor installation directory

1. Start Blaze Rules Development for Software AG Designer.
2. You are prompted to specify the *Software AG_directory* installation directory. Click **Next** and enter the following path:
`Software AG_directory\blaze\Advisor71`
3. Click **Next**, read the informational message, and then click **Finish**.
4. You are prompted to restart Blaze Rules Development for Software AG Designer.

Adding JDBC Files

JDBC jar files are required for deploying the rules to Integration Server and for deploying the RMAs to My webMethods Server from the database repository.

To add the JDBC files

1. Download the appropriate JDBC driver .jar file from the database vendor's website.
2. Copy the JDBC driver .jar file to *Software AG_directory/blaze/Advisor71/JdbcJars* folder.
3. If you have any rule-specific custom jar files, copy the jars to the *Software AG_directory/blaze/Advisor71/JdbcJars* folder.
4. In Blaze Rules Development for Software AG Designer: **Window > Preferences > Blaze Advisor**.
5. Click **Add Library**, select the JDBC jar file, and click **OK**.
6. If you have custom jars in the Blaze project, open the file, *Software AG_directory/blaze/Advisor71/lib/templates/softwareag-iwag/mws/bnd.bnd* and append each java package inside the custom jar above the line, `{0}, \.` Save and close the file.
7. Restart Blaze Rules Development for Software AG Designer.

Changing the Logging Level of the Blaze Eclipse Plug-in

To change the logging level of the Blaze Eclipse plug-in

1. Open the *Software AG_directory/blaze/Advisor71/log4j.properties* file in a text editor.
2. Edit the `log4j.rootLogger` property to one of the following logging levels: TRACE, DEBUG, WARN, ERROR, FATAL, or INFO (the default).
3. Save the file.

Upgrading to Blaze Advisor 7.1

With webMethods Product Suite Suite version 8.2 SP2 and earlier, Blaze Advisor was installed as a standalone application. With version 9.0 SP1, Software AG Designer supported the Blaze Advisor 7.1 Eclipse plug-in, and support for the standalone application was removed. As of version 9.5 SP1, there is a special implementation of Blaze Advisor. For more information, see "[Special Considerations Regarding Version Compatibility](#)" on page 10.

Separate upgrade procedures apply depending on whether you are upgrading from version 8.2 SP1 or earlier, or from version 9.0 SP1 to version 9.5.1 or later.

Upgrading from Version 8.2 SP2 and Earlier

Upgrading to Blaze Advisor 7.1 from version 8.2 SP2 and earlier must be performed side-by-side with your previous installation. *Do not attempt to install Blaze Rules Development for Software AG Designer over any earlier version.*

For general information about upgrading webMethods products, see the PDF publication *Upgrading webMethods and Intelligent Business Operations Products*.

To upgrade to Blaze Advisor 7.1 from version 8.2 SP2 and earlier

1. Follow the procedure for installing Blaze Rules Development for Software AG Designer, outlined in "[Installing Blaze Rules Development for Software AG Designer](#)" on page 18.
2. Configure Blaze Advisor in Blaze Rules Development for Software AG Designer, as described in "[Defining the Blaze Advisor Installation Directory](#)" on page 18.
3. Set up the new Blaze 7.1 repository. For more information, see the FICO publication about managing repositories.

Blaze Advisor 7.x does not support the following repository types:

- LDAP (no versioning)
- LDAP (BVS versioning—private)
- LDAP (BVS versioning—shared)
- Database (BVS versioning—shared)

To use these repositories with the current version of Blaze Advisor, you must convert them to one of the supported types. For information about how to convert a repository to another type, see the FICO publication about managing repositories.

Software AG recommends that you migrate content from the repository used by the older Blaze version to the repository used by Blaze Advisor 7.1, instead of upgrading and using the old repository.

4. Migrate content from the old Blaze repository to Blaze Advisor 7.1 repository by exporting the Blaze rules from your previous version and importing them into Blaze Advisor 7.1. For more information, see the FICO publication about migration.
5. Deploy the Blaze rule projects to Integration Server. For more information, see "[Deploying Blaze Rule Projects to webMethods Servers](#)" on page 12.
6. Deploy the RMAs to My webMethods Server. For more information, see "[Deploying RMAs to My webMethods Server](#)" on page 13.

Upgrading from Version 9.0 SP1

Upgrading to Blaze Advisor 7.1 from version 9.0 SP1 must be performed side-by-side with your previous installation. *Do not attempt to install Blaze Rules Development for Software AG Designer over any earlier version.*

Note: This procedure assumes you have an existing Blaze Advisor repository in your version 9.0 SP1 installation.

For general information about upgrading webMethods products, see the PDF publication *Upgrading webMethods and Intelligent Business Operations Products*.

To upgrade to Blaze Advisor 7.1 from version 9.0 SP1

1. Follow the procedure for installing Blaze Rules Development for Software AG Designer, outlined in ["Installing Blaze Rules Development for Software AG Designer" on page 18](#).
2. Define the Blaze Advisor directory in Blaze Rules Development for Software AG Designer, as described in ["Defining the Blaze Advisor Installation Directory" on page 18](#).
3. In Blaze Rules Development for Software AG Designer, in the Blaze Advisor perspective, click **Repository > New Repository Connection** to create a connection to the existing Blaze Advisor repository.
4. To connect to the repository connection:
 - a. Click **Repository > Manage Connections** and select the repository connection that you created in above steps.
 - b. Click **OK**.
5. Proceed to ["Preparing to View Rules for the First Time" on page 28](#).
6. Deploy the Blaze rule projects to Integration Server. For more information, see ["Deploying Blaze Rule Projects to webMethods Servers" on page 12](#).
7. Deploy the RMAs to My webMethods Server. For more information, see ["Deploying RMAs to My webMethods Server" on page 13](#).

2 Setting Up the Blaze Repository

■ Supported Blaze Repository Types	24
■ Creating a Blaze Database Repository	24
■ Preparing to View Rules for the First Time	28

Supported Blaze Repository Types

To use Blaze rules with webMethods components, you must use either a database or a file repository. Aside from other features associated with each repository type as documented in the FICO Blaze Advisor documentation, the type of repository you choose depends on whether you want to be able to update and deploy rules in both Blaze Advisor and My webMethods, or just in Blaze Advisor.

The following table identifies the tasks you are able to perform with each repository type.

With this repository type...	You can...
Database	<ul style="list-style-type: none">■ Maintain rules in both Blaze Advisor and in My webMethods.■ Deploy rules from Blaze Advisor to Integration Server.■ Deploy Rule Maintenance Applications (RMAs) from Blaze Advisor to My webMethods.■ Deploy modified rules from My webMethods to Integration Server.
File	<ul style="list-style-type: none">■ Maintain rules in Blaze Advisor only.■ Deploy rules from Blaze Advisor to Integration Server.

The sections in this chapter describe how to set up a database repository to use with webMethods processes. For information about how to set up a file repository, see the FICO Blaze Advisor documentation.

Important: If you are presently storing Blaze rules in a file repository and you want to be able to maintain rules in My webMethods and deploy modified rule instances from My webMethods to Integration Server, you must migrate the rules from the file repository to a database repository. For details, see the FICO Blaze Advisor documentation.

Creating a Blaze Database Repository

You can create the database schema and generate the repository to hold rules for webMethods processes using two methods:

- Create the database schema and the repository separately. To do so, first create the database schema using the Database Component Configurator and then create the repository in Blaze Advisor. Use this method if you want to create a repository using the default table name `BLAZE_REPOSITORY` that the Database

Component Configurator provides. For details, see ["Creating the Database Schema and Repository Separately" on page 25](#).

- Create both the database schema and the repository at the same time using the generateRepository command-line tool. Use this method if you want to store rule projects in a repository that is based on a different schema from the one that the Database Component Configurator provides. For details, see ["Creating the Database Schema and Repository at the Same Time" on page 25](#).

Creating the Database Schema and Repository Separately

With this method, you will create the database schema using the Database Component Configurator and then create the repository in Blaze Advisor.

To create the database schema and repository separately

1. Create the database schema by creating a Blaze database component using the Database Component Configurator. For more information, see *Installing webMethods and Intelligent Business Operations Products*.

This step creates a table called BLAZE_REPOSITORY. You will use this table name for creating the RDBMS repository in the next step.

2. Create the RDBMS repository in Blaze Advisor as follows:
 - a. Click **Repository > New Repository**
 - b. In the **Repository Type** list, select either **Database Repository (BVS versioning - private)** or **Database Repository (No versioning)**.
 - c. In the Table box of the Database Repository Settings section, type the table name `BLAZE_REPOSITORY`.
 - d. Complete the remaining fields on the Create a Repository window. For more information about these fields, see ["Creating the Database Schema and Repository at the Same Time" on page 25](#).
 - e. Click **OK**.
3. Proceed to ["Preparing to View Rules for the First Time" on page 28](#).

Creating the Database Schema and Repository at the Same Time

To simplify the creation of a database repository that will hold rules for webMethods processes, use the generateRepository command-line tool, which Software AG provides. The generateRepository command-line tool creates the database schema and generates the repository at the same time.

The Software AG Installer installs the generateRepository command-line tool in the following location:

```
Software AG_directory  
\blaze\Advisor71\bin
```

Note: You can create the database from Blaze Advisor. However, there is no automatic way to generate the database schema from Blaze Advisor. You need to create the database schema on your own, and then use Blaze Advisor to create the repository.

The following shows the syntax of the generateRepository command-line tool:

```
generateRepository.bat [-options]
```

The options are explained in the following procedure. The generateRepository command uses executables provided by FICO Blaze Advisor. For additional information, see the FICO Blaze Advisor PDF publication *Managing Repositories* located in:

Software AG_directory\blaze\Advisor71\doc\pdf

To create the database schema and repository using the generateRepository command-line tool

1. On the computer where Blaze Advisor is installed, open a command prompt and cd to the directory where the generateRepository batch file is installed, for example:

```
cd Software AG_directory\blaze\Advisor71\bin
```

2. At the command prompt, enter the generateRepository.bat command, specifying the appropriate options for the type of repository you want to create. Be sure to note the values you specify for the options because you will need to use those same values to connect to the repository:

Option	Meaning
-vendor <i>databaseVendor</i>	The vendor that provides the database you are using for the repository. For Oracle, type <code>Oracle</code> For DB2, type <code>ibm</code> For MS SQL Server, type <code>microsoft</code>
-driver <i>JDBC-Driver</i>	The database driver. For example, for Oracle: <code>oracle.jdbc.driver.OracleDriver</code> The location of this driver must be in your CLASSPATH.
-url <i>databaseURL</i>	The URL for the database. For example, for Oracle: <code>jdbc:oracle:thin:@localhost:1521:orcl</code>
-user <i>databaseUserName</i>	A valid user name to connect to the database.
-password <i>databasePassword</i>	The password for the specified user name.

Option	Meaning
<code>-table <i>databaseTable</i></code>	The name to assign to the database table that holds the repository. For example, <code>BLAZE_REPOSITORY</code>
<code>-repository <i>repositoryName</i></code>	The name to assign to the Blaze repository. For example, <code>sharedRepo</code>
<code>-workspaceFolder <i>workspacePath</i></code>	For a private workspace, the path of a local directory to use for the workspace. For example, <code>C:\Blaze\Advisor71\private</code> .
<code>-repositoryType <i>repositoryType</i></code>	The type of repository. Specify either <code>noversioning</code> or <code>private</code> .
<code>-workspaceUser <i>workspaceUserName</i></code>	Optional. For a private workspace, the user name to use to connect to the repository.
<code>-workspacePassword <i>workspacePassword</i></code>	Optional. For a private workspace, the password to use to connect to the repository.

After entering the command, the command-line tool responds with information messages to indicate its progress. After the tool completes the first step of creating the database tools, it issues the following message:

```
The database table tableName has been created successfully.
```

The tool then starts creating the repository and issues the following message after it finishes creating the repository:

```
Command createRepository is executed successfully.
```

The tool then performs some cleanup before finally completing.

3. In Blaze Rules Development for Software AG Designer, in the Blaze Advisor perspective, click **Repository > New Repository Connection** to create a connection to the repository from Blaze Advisor.
4. Specify how you want to connect to the database repository:
 - To connect without versioning, select a **Workspace Type** of **Database**.
 - To connect with BVS versioning-private, select a **Workspace Type** of **File**.
5. Click **Next** and do one of the following:
 - If you chose a workspace type of **Database**, update these fields in the Database Repository Settings wizard page to match the values you specified in the options of the `generateRepository` command when you generated the repository.

Field	Set the value to match this generateRepository option
JDBC Driver	-driver
JDBC URL	-url
User	-user
Password	-password
Repository name	-repository
Table	-table

In the **Accessor class** field, type one of the following:

For Oracle: `com.blazesoft.repository.jdbc.NdOracleJdbcRepositoryAccessor`

For SQLServer or DB2:

`com.blazesoft.repository.jdbc.NdDefaultJdbcRepositoryAccessor`

Select the **Save database user and password** check box if you do not want to enter your credentials each time you connect to the repository.

- If you chose a workspace type of **File**, click the browse button on the Database Repository Settings wizard page to locate and select the workspace that you created using the command-line utility.

Select the **Requires Authentication** check box if you want to enforce user authentication when connecting.

6. Click **Finish**.
7. To connect to the repository connection:
 - a. Click **Repository > Manage Connections** and select the repository connection that you created in above steps.
 - b. Click **OK**.
8. Proceed to "[Preparing to View Rules for the First Time](#)" on page 28.

Preparing to View Rules for the First Time

Before you can open and view rule projects in My webMethods Server, you must carry out the following procedure to update the database connection information in My webMethods Server.

To view rule projects in My webMethods Server for the first time

1. Stop My webMethods Server if it is running.
2. Locate the JDBC jar file for the Blaze Advisor repository you created.
3. Copy the JDBC jar file to *Software AG_directory\MWS\lib*.
4. Open a command session in *Software AG_directory\MWS\bin*.
5. Run the following command to update the My webMethods Server classpath:

```
mws.bat update
```

6. Restart My webMethods Server.

3 Using Blaze Rules in Business Processes

■ Overview	32
■ Creating Blaze Projects	32
■ Using IS Document Types in Blaze Rules	34
■ Guidelines for Blaze Rules Used in webMethods Processes	38
■ Deploying Rules to the Integration Server	39
■ Adding Rule Steps to a Process Model	40

Overview

To use Blaze rules in a process, do the following in Blaze Advisor:

- Create the Blaze project that will contain the rules that you want to use in webMethods processes. For information about how to create Blaze projects, see the FICO Blaze Advisor documentation. For information specific to webMethods processes, see ["Defining Server Connection Properties for Blaze Projects" on page 33](#).
- Configure the Blaze project properties to identify the Integration Server to which to deploy the rules. See ["Defining Server Connection Properties for Blaze Projects" on page 33](#).
- Optionally create Custom Business Object Models that wrap Integration Server document types (IS document types). By creating the Custom Business Object Models, you can then use the IS document types in rules. See ["Using IS Document Types in Blaze Rules" on page 34](#).
- Create the rules you want to use. For information about how to create rules, see the FICO Blaze Advisor documentation. For information specific to rules for webMethods processes, see ["Guidelines for Blaze Rules Used in webMethods Processes" on page 38](#).
- Generate the webMethods deployment to Integration Server so that the rules are deployed to the Integration Server. For more information, see ["Deploying Rules to the Integration Server" on page 39](#).

After you have deployed the rules to Integration Server, you can then use the rules in a process model. For more information, see ["Adding Rule Steps to a Process Model" on page 40](#).

Creating Blaze Projects

A single Blaze project can contain one or more rules or rulesets and/or custom business object models. Each Blaze project that is associated with webMethods requires connection information that Blaze Advisor uses to:

- Deploy Blaze rules to an Integration Server; for more information, see ["Deploying Rules to the Integration Server" on page 39](#).
- Provide access to the Integration Server document types used in custom business object models; for more information, see ["Using IS Document Types in Blaze Rules" on page 34](#).
- Deploy Rule Maintenance Applications (RMAs) to My webMethods Server; for more information, see ["Deploying Rules to the Integration Server" on page 39](#).

Note: You deploy RMAs to My webMethods Server if you want to be able to dynamically update your Blaze rules in My webMethods. You *must* use a database repository to be able to dynamically update your Blaze rules.

You can create as many Blaze projects as you want, and each one can connect to a different Integration Server and My webMethods Server. The connection information that you configure (excluding the password) is stored in the Blaze project properties. When a connection to Integration Server or My webMethods Server is required during a Blaze session, Blaze Advisor prompts for the password. After you have established a connection in a Blaze session, Blaze Advisor maintains the connection until you close the session. If you switch to another project that requires different connection properties, Blaze Advisor prompts for the additional passwords. Blaze Advisor maintains the connections to all servers during the session.

Defining Server Connection Properties for Blaze Projects

The following procedure describes how to define Integration Server and My webMethods Server connection properties for a Blaze project.

Note: Only one Integration Server and one My webMethods Server connection can be defined per project. If you want to change servers for the project at any time, you must modify these connection properties.

To define the server connection properties for a Blaze project

1. In Blaze Advisor, open the **Properties** tab for a project.
2. On the **Properties** tab in the Blaze project, enter the following information:

In this field	Specify...
IS Host	The host name of the Integration Server to which to deploy rules in the project.
IS Port	The port number of the Integration Server to which to deploy rules in the project.
User Name	The user name that Blaze Advisor is to use to log into the Integration Server. The user account must have administrative privileges.
Use SSL	The option that indicates whether or not Blaze Advisor uses SSL for the connection to the Integration Server. Select true to use SSL or false if you do not want to use SSL.

In this field	Specify...
MWS Host	The host name of the My webMethods Server to which to deploy RMAs for the project. For more information, see "Managing Blaze Rules in My webMethods" on page 49.
MWS Port	The port number of the My webMethods Server to which to deploy RMAs for the project. For more information, see "Managing Blaze Rules in My webMethods" on page 49.
MWS User	The user name that Blaze Advisor is to use to log into the My webMethods Server. The user account must be a member of the My webMethods Server Administrator role.

3. Save your changes.

Important: If you are changing the Integration Server connection information for a project and you have created a custom business object model that connects to the old Integration Server, you must use the Custom Business Object Model wizard to specify the connection information of the new Integration Server.

Using IS Document Types in Blaze Rules

To use an IS document in a Blaze rule, you create a webMethods custom business object model in Blaze Advisor. Custom business object models contain business objects that wrap external objects, in this case IS document types.

When you create a Blaze Custom Business Object Model to use in a rule, you can use any user-defined webMethods IS document type as inputs and outputs. These are imported as top-level classes. Fields and records are not top-level classes, and they can be used only as internal content, not as inputs and outputs.

Blaze Advisor does not display the structure of nested documents, so nested document types are displayed as separate referenced documents. Blaze Advisor does not filter the available types for parameters (inputs) and return types (outputs), and will not prevent you from selecting fields and records, but your rule will not function if you do.

To help you distinguish the objects that should not be used as inputs and outputs, a 'zzz' prefix is appended to them. When you generate a webMethods deployment, inputs and outputs are validated; rules that are not properly formed are not deployed.

Before creating custom business object models, see ["Supported Data Types in Blaze Advisor" on page 35](#), below, to determine the data types that Blaze Advisor supports and conversion information. Then see ["Creating Blaze Custom Business Object Models" on page 36](#) for instructions about creating the custom business object models.

Note: If you are working with webMethods products version 8.0 or later, specific conditions apply to returning Decision Table results as an IS document type. For more information, see ["Returning the Results of a Decision Table or Tree as an IS Document Type \(webMethods Version 8.0 and Later\)"](#) on page 37.

Supported Data Types in Blaze Advisor

When you create a Blaze Custom Business Object Model, you use IS document types to create data types. You can use these as parameters (inputs), returns (outputs), and anywhere else you define the data type of a field.

The following default data types in Blaze Advisor are supported for deployment to Integration Server:

- boolean
- date
- integer (becomes long on Integration Server)
- real (becomes double on Integration Server)
- string
- time (becomes date on Integration Server)
- timestamp (becomes date on Integration Server)
- void
- array

When you choose array as a type, you must also select the type of array. You can select multiple array levels in Blaze Advisor. However, Integration Server supports only the following:

- One-dimensional arrays of any supported type (such as an array of integer)
- Two-dimensional arrays of type string

Multi-dimensional arrays other than two-dimensional arrays of type string will not deploy to Integration Server. Errors will occur during deployment.

Blaze Advisor does not support the byte array (byte []) type. Although you can import them without errors into a Custom Business Object Model, byte arrays do not convert correctly from Integration Server to Blaze, or vice versa.

Blaze Advisor does not support all of the Java data types that Integration Server supports. When IS documents use Java data types that are not supported in Blaze Advisor, they are converted to a supported type within the Blaze rule and then converted back to the proper Integration Server data type when rule processing is completed.

Integration Server	Blaze Advisor
byte short int long	integer
	<p>Note: The Blaze 'integer' type is implemented by a Java long type, so using an 'int' in an IS document requires a conversion to run in Blaze, whereas using a 'long' in an IS document does not.</p>
char	string
double float	real
	<p>Note: The Blaze 'real' type is implemented by a Java double type, so double to avoid conversion.</p>

Creating Blaze Custom Business Object Models

This section describes how to create a Blaze custom business object model. After you create a custom business object model, deploy it to Integration Server so that you can use it when you create a Blaze rule that you want to use in a webMethods process.

Note: You must have an Integration Server connection defined and an active connection to the Integration Server to complete this procedure. For more information, see ["Defining Server Connection Properties for Blaze Projects" on page 33](#).

To create a Blaze custom business object model that wraps an IS document type

1. In Blaze Advisor for the Blaze project in which you want to create the custom business object model, configure project properties to connect to the Integration Server. For instructions, see ["Defining Server Connection Properties for Blaze Projects" on page 33](#).
2. In Blaze Rules Development for Software AG Designer, select the project in which you want to create the custom object model, and click **File > New > Business Object Models > Custom Business Object Model**.
3. In the Custom Business Object Model wizard, type a name for the custom business object model in the **Custom Business Object Model** field.
4. Type the following **Class Provider Class**:

```
com.webmethods.blaze.bom.ISDocumentTypeProvider
```

Note: If the **Class Provider Class** value is empty or incorrect, the browse button described in step 6 is not enabled.

5. Click the  **New Argument** button in the **Constructor Arguments** section.

-
6. Click the  browse button.

If you have configured a connection to an Integration Server but are not already connected to it, Blaze Advisor prompts you for the Integration Server password. Enter the password in the Connect to Integration Server window and click **OK**.

Note: If you want to connect to a different Integration Server, you must modify the connection properties for the project as described in "[Defining Server Connection Properties for Blaze Projects](#)" on page 33.

7. In the IS Document Type Import dialog box, select the check box or boxes that correspond with the IS document types you want to use in the custom business object model.
8. Click **OK**.
Blaze Advisor displays the selected IS document types in the **Constructor Arguments** field of the Custom Business Object Model wizard.
9. Type any comments you want to add in the **Comments** field.
10. Click **OK** to save the Custom Business Object Model in the project, then click **Yes** in the confirmation dialog box.

You can right-click the custom business model object to access a context menu that enables you to delete or rename the object, or open the object for modification.

Returning the Results of a Decision Table or Tree as an IS Document Type (webMethods Version 8.0 and Later)

For webMethods product versions 8.0 and later, use the following procedure to return the results of a Decision Table or Decision Tree as an IS document type.

To return the results of a Decision Table or Decision Tree as an IS document type

1. If you have not already done so, specify an existing custom business object model to contain the IS document type. For more information, see "[Creating Blaze Custom Business Object Models](#)" on page 36.
2. Open the Decision Table or Decision Tree template in Blaze Advisor and modify the return type to specify the type of the custom business object model that you want to return.
3. Add a new rule to the Decision Table or Decision Tree template by clicking on the **Additional Rules** icon.
4. Set the Priority level for the newly added rule to:

-999

For more information about this setting, see the section *Specifying the Priority for a Rule* in the Blaze PDF publication *Deploying Rule Services*.
5. Type the following as a rule content in the text area of the newly created rule:





```
if(true) then return inputDoc
```

where *inputDoc* is the name of the input document in the Parameters section in the Decision Table or Decision Tree template. This will return the result of the Decision Table or Decision Tree as an IS document type when you run the Decision Table or Decision Tree in Blaze Rules Development for Software AG Designer.

6. Save your changes to the rule.
7. Recompile the rule service.

Guidelines for Blaze Rules Used in webMethods Processes

When creating rules in Blaze Advisor, be aware that only the following types of Blaze rules are supported for use in webMethods processes:

-  Decision Tables
-  Decision Trees
-  Rulesets
-  Functions

The webMethodsBlaze Advisor OEM license allows a Restricted Maximum Rule Count, with the following conditions:

- A Blaze project can contain a maximum of 250 rules statements.
- A process step can contain a maximum of 250 rules.
- A process step can invoke only one rule service.
- A rule service can call only one Blaze project.
- A rule service cannot invoke another rule service.
- Rules or rule sets for any rule service or business process cannot be separated or sequenced.
- A business function cannot be separated into multiple decision steps.

If you want to use IS document types in your rules, before creating the rule, create custom business object models that wrap the IS document types that you want to use. For more information, see ["Using IS Document Types in Blaze Rules" on page 34](#).

Rule Inputs and Outputs

You must set the rule inputs and outputs in Blaze Advisor. The rule signature you define in Blaze Advisor becomes the signature defining the inputs and outputs of the Integration Server rule service.

The default action of a rule that expects input but receives none is to return a null value. Integration Server rule services do not throw errors, exceptions, or warnings when encountering null values that are passed to Blaze rules. If you want to handle null inputs to rules, you must do so inside the rule itself.

Deploying Rules to the Integration Server

After you create your rules, you must deploy them to Integration Server before you can use them in a process model. To deploy the rules, you generate a deployment from Blaze Advisor to Integration Server.

The first time you generate a webMethods deployment to Integration Server for a Blaze project, a new package is added to the Integration Server. The Integration Server package will have the same name as the Blaze project. The package will contain one or more rule services, depending on the entry points you select when you generate the deployment to Integration Server. The package also contains a wrapper service and the Blaze configuration files needed to run the rule service. Each time you deploy a project to Integration Server, the existing Integration Server package and service are updated.

To deploy rules from Blaze Advisor to the Integration Server

1. In Blaze Advisor, open the project that contains the rules that you want to deploy.
2. Click **Projects > Generate webMethods Deployment**

If you are not already connected to the Integration Server, Blaze Advisor prompts you for the password so that it can connect to the Integration Server. Enter the password in the Connect to Integration Server window and click **OK**.

Note: If you want to connect to a different Integration Server, you must edit the connection properties for the project.

3. In the Entry Point Configuration window in the **Use** column, select the check boxes that correspond to the Blaze rules (entry points) you want to deploy to the Integration Server for use as rule services, and click **OK**.

Note: Rules that have been deployed previously are automatically redeployed. The **Use** boxes corresponding to those services are automatically selected and cannot be deselected. If you no longer want a rule to be available on an Integration Server, you must delete it directly from the Integration Server, rather than from Blaze Advisor.

Important: The Integration Server rule service is regenerated each time a webMethods deployment is generated for a Blaze project. Therefore, any changes you make to a rule service in Blaze Rules Development for Software AG Designer will be overwritten if a change is made to the corresponding project. To apply rule changes to both the 'live' rule on the Integration Server and to the rule definition in Blaze Advisor, dynamically update the Rule Maintenance Applications (RMAs) for the

rules via My webMethods. For more information, see ["Managing Blaze Rules in My webMethods" on page 49](#).

4. If you want to be able to dynamically update the rule from My webMethods, you can deploy the Rule Maintenance Application (RMA) for the rule to My webMethods Server at this time. For instructions, see ["Before You Can Manage Blaze Rules" on page 41](#) and ["Managing Blaze Rules in My webMethods" on page 49](#).

Adding Rule Steps to a Process Model

After deploying Blaze rules to Integration Server, from Software AG Designer you can then reference them in rule steps in a process model.

In Software AG Designer be sure you have configured the Integration Server that contains the Blaze rules so that you can view the services on that Integration Server in Designer's Package Navigator. To use a Blaze rule in a process model, drag a rule service from the Package Navigator view and drop it onto the process editor's canvas to automatically create a rule step. You can also drop a rule service onto an existing activity step to convert it into a rule step. The inputs and outputs of the rule step will be the inputs and the outputs you defined for the rule in Blaze Advisor.

If a step invokes a Blaze rule and you have the appropriate My webMethods permissions, you can click the step name to view the rule on the Rule Maintenance Application panel and modify the rule instance if desired.

For more information about creating process models in Software AG Designer, see the *webMethods BPM Process Development Help*.

Note: You can connect to Blaze Advisor directly from Software AG Designer to create Blaze rules and deploy them to Integration Server. For more information, see the *webMethods BPM Process Development Help*.

4 Before You Can Manage Blaze Rules

■ Requirements for Managing Blaze Rules in My webMethods	42
■ Granting Users Permission to Manage Blaze Rules	42
■ Configuring the My webMethods Server Connection for a Blaze Project	44
■ Updating the Integration Server and Repository Credentials for an RMA	44
■ Checking the Project into the Blaze Versioning System	46
■ Deploying RMAs to the My webMethods Server	46

Requirements for Managing Blaze Rules in My webMethods

Before you can manage Blaze rules in My webMethods, you must meet the following requirements:

- An administrator has set up the Blaze database repository. For more information, see ["Setting Up the Blaze Repository" on page 23](#).
- A My webMethods administrator has granted you the appropriate permissions for managing Blaze rules in My webMethods. For more information, see ["Granting Users Permission to Manage Blaze Rules" on page 42](#).
- A My webMethods administrator has copied a JDBC database jar file and then run the mws.bat update command to update the My webMethods Server classpath. For more information, see ["Preparing to View Rules for the First Time" on page 28](#).
- The Blaze project's connection properties are configured to identify the My webMethods Server to which to deploy Rule Maintenance Applications (RMAs). For more information, see ["Deploying Rules to the Integration Server" on page 39](#).
- For the Blaze rules you want to manage, the RMAs are deployed to My webMethods Server. For more information, see ["Deploying RMAs to the My webMethods Server" on page 46](#).
- The Blaze project that contains the rules to manage are checked into the Blaze Versioning System. For more information, see ["Checking the Project into the Blaze Versioning System" on page 46](#).

Granting Users Permission to Manage Blaze Rules

Before you can take actions against Blaze rules in My webMethods, a My webMethods administrator must assign you the appropriate permissions:

- **Access privileges** that allow you to display the Business Rules page from which you manage Blaze rules.

An administrator assigns access privileges to a user, to a group to which a user belongs, or to a role of which a user is a member.

- **Functional privileges** that allow you to take actions against Blaze rules.

An administrator assigns functional privileges to Blaze projects. As a result, you might be able to perform one or more actions against the rules in one Blaze project but be denied permissions to take actions on rules in other Blaze projects.

To allow users to view the Business Rules page and perform actions against business rules

1. In My webMethods:

Navigate > Applications > Administration > System-Wide > Permissions Management

-
2. On the **Advanced** tab of the Permissions Management page, do the following:
 - a. In the **Resource Type** list, select **webMethods Applications** and, if not already done, move **webMethods Applications** to the **Selected** list.
 - b. Click **Next**.
 3. In the Edit Permissions panel, select the users, groups, or roles you want to provide with Blaze rule permission, and do the following:
 - a. Click **Add Users/Groups/Roles**.
 My webMethods displays the Add Principals panel with the list of defined users displayed by default. You can change the list to show defined groups and roles by clicking the appropriate option in the **Search For** area and then clicking **Search**.
 - b. Move the user/group/role to which you want to grant privileges to the **Selected** list.
 - c. Click **Add**.
 4. Click the value in the **Permissions** column (either **Granted All** or **Custom**).
 My webMethods displays a tree that lists all the possible Access and Functional privileges.
 - a. To allow the user/group/role access to the Business Rules page that lists deployed RMAs, select the following access privilege in the tree:
Access Privileges > Administration > Business > Business Rules
 - b. To allow the user/group/role access to the Analytics Overview page to view key performance indicator (KPI) information for the changes to Blaze rules, select the following access privilege in the tree:
Access Privileges > Monitoring > System-Wide > Analytics Overview
 For instructions for viewing KPI information, see "[Blaze Rule KPIs](#)" on page 55.
 5. To allow the user/group/role to take actions against business rules, select the functional privileges for each Blaze project by performing the following:
 - a. Navigate to **Functional Privileges > Blaze RMA**
 - b. Select the *ProjectName* for which you are defining privileges.
 - c. Define the permissions using the descriptions in the following table.

To allow a user/group/role to...	Under Functional Privileges > Blaze RMA <i>ProjectName</i> , select...
View rules in the project	Read Rules For instructions for viewing rules, see " Viewing Rules in My webMethods " on page 50.

To allow a user/group/role to...	Under Functional Privileges > Blaze RMA <i>ProjectName</i> , select...
Update rules in the project	<p>Modify Rules</p> <p>Deploy Rules</p> <p>For instructions for updating rules, see "Updating a Rule Instance" on page 51.</p>
Create a rule instance in the project	<p>Create New Instance</p> <p>For instructions for creating new rule instances, see "Creating a New Rule Instance" on page 53.</p>
Delete rule instances from the project	<p>Delete Instance</p> <p>For instructions for deleting rule instances, see "Deleting a Rule Instance" on page 54.</p>
View audit information for the changes to rules in the project	<p>View Audit Trail</p> <p>For instructions for viewing audit information, see "Viewing the Version History of a Rule Instance" on page 51.</p>

- d. Click **OK**.
- e. Repeat for each Blaze project.

Configuring the My webMethods Server Connection for a Blaze Project

If you have not already configured the Blaze project to identify the My webMethods Server connection information, you must do so. You define My webMethods Server connection information to indicate where to deploy the RMAs. For instructions, see ["Defining Server Connection Properties for Blaze Projects" on page 33.](#)

Updating the Integration Server and Repository Credentials for an RMA

When you deploy a Blaze project to the My webMethods Server as described in the next section, the RMA that Blaze Advisor creates contains the credentials for connecting to the Integration Server on which the rule service is intended to run and the database

repository to which rule changes are written. If either set of credentials changes (for example, if an RMA should use a different Integration Server or repository or if a connection password changes), you can update the credentials in My webMethods for all of the RMAs that use this information.

Updating the Integration Server

To update the Integration Server credentials for an RMA

1. In My webMethods:

Navigate > Applications > Administration > My webMethods > System Settings > Business Rules Settings

My webMethods displays the Business Rules Settings page.

2. To update the Integration Server credentials for a single RMA, do the following:
 - a. On the **RMA Projects** tab, click the link in the **Integration Server** column next to the RMA that has the Integration Server credentials you want to change.
 - b. On the **IS Credentials** tab, in the **Integration Server** list, select the Integration Server that has the credentials you want to change.
 - c. Update the host, port, user name, password, and SSL encryption as necessary.
 - d. Click **Apply**.
3. To update the Integration Server credentials for multiple RMAs at the same time, do the following:
 - a. On the **RMA Projects** tab, click the link in the **Integration Server** column next to the RMA that has the Integration Server credentials you want to change.
 - b. In the **Integration Server** list, select the Integration Server that has the credentials you want to change.
 - c. In the **Projects** list, select the RMAs that should use the updated Integration Server credentials.
 - d. Click **Edit**.
 - e. Update the host, port, user name, password, and SSL encryption as necessary.
 - f. Click **Apply**.

Updating the Database Repository

To update the database repository credentials for an RMA

1. In My webMethods:

Navigate > Applications > Administration > My webMethods > System Settings > Business Rules Settings

My webMethods displays the Business Rules Settings page.

2. To update the repository credentials for a single RMA, do the following:
 - a. On the **RMA Projects** tab, click the link in the **Repository** column next to the RMA that has the repository credentials you want to change.
 - b. On the **Repository Database Credentials** tab, in the **Repository** list, select the repository that has the credentials you want to change.
 - c. Update the database user name, password, URL, and driver information as necessary.
 - d. Click **Apply**.
3. To update the repository credentials for multiple RMAs at the same time, do the following:
 - a. Click the **Blaze Repository** tab.
 - b. In the **Repository** list, select the repository that has the credentials you want to change.
 - c. In the **Projects** list, select the RMAs that should use the updated repository credentials.
 - d. Click **Edit**.
 - e. Update the database user name, password, URL, and driver information as necessary.
 - f. Click **Apply**.

Checking the Project into the Blaze Versioning System

Before you deploy an RMA from Blaze Advisor to My webMethods Server, you must check the project into the Blaze Versioning System. Checking projects into the Blaze Versioning System writes the rules within the project to the database repository so that other users can access and modify those rules.

To check a Blaze project into the Blaze Versioning System

1. In Blaze Advisor, right-click the project that you want to check, and select **Team > Check in**
2. In the Check In window, select the check boxes for the rules that you want to check in and click **Check In**.
3. In Blaze Advisor, right-click the top-level project and click **Check in**.
4. In the **Comment** field of the Check In window, summarize the changes you made and then click **Check In**.

Deploying RMAs to the My webMethods Server

After you check a rule project into the Blaze Versioning System, you can deploy the project as an RMA to My webMethods Server. You deploy RMAs to My webMethods Server to enable other users, such as business analysts, to dynamically modify and

deploy rules and view KPIs that illustrate how changes to rules affect business processes.

You cannot select specific rules within the Blaze project; an RMA for *all* rules within the Blaze project is deployed to My webMethods Server.

If you want to connect to a different My webMethods Server, you must edit the connection properties for the project.

To deploy RMAs from Blaze Advisor to the My webMethods Server

1. In Blaze Advisor, open the project that contains the rules you want to deploy.
2. In the **Project Explorer** view, right-click on the Rules project and select **Open Project Editor**.
3. Click on the Project menu, and then click **Generate Rule Maintenance Application to MWS**.

If you are not already connected to the My webMethods Server, Blaze Advisor prompts you for the password so that it can connect to the My webMethods Server. Enter the password in the **Connect to MWS** window and click **OK**.

Note: The user name is read-only. If you want to change the user name, you must check out the project to remove the read-only status.

Note: You can deploy RMAs from one My webMethods Server to another using webMethods Deployer. If you do so, be sure to update the Integration Server and database repository connection credentials, if necessary, as described in "[Updating the Integration Server and Repository Credentials for an RMA](#)" on page 44.

5 Managing Blaze Rules in My webMethods

■ Overview	50
■ Viewing Rules in My webMethods	50
■ Viewing the Version History of a Rule Instance	51
■ Updating a Rule Instance	51
■ Creating a New Rule Instance	53
■ Deleting a Rule Instance	54
■ Deleting an RMA	54

Overview

The following table lists the tasks you can perform in My webMethods to manage your Blaze rules:

Task	For more information, see...
View business rules	"Viewing Rules in My webMethods" on page 50
Update business rules	"Updating a Rule Instance" on page 51
Create new rule instances	"Creating a New Rule Instance" on page 53
Delete rule instances	"Deleting a Rule Instance" on page 54
View audit information about the changes to rules	"Viewing the Version History of a Rule Instance" on page 51
Delete RMAs	"Deleting an RMA" on page 54

Viewing Rules in My webMethods

The My webMethods Business Rules page displays each RMA that has been deployed from Blaze Advisor. On the Business Rules page, you can open an RMA to view the details of the Blaze rule entities that you can manage via the RMA.

Note: You must have the appropriate My webMethods privileges to view deployed RMAs and business rules information. For more information, see ["Granting Users Permission to Manage Blaze Rules" on page 42](#).

To view Blaze rules in My webMethods

1. From My webMethods:
Navigate > Applications > Administration > Business > Business Rules
My webMethods displays the Business Rules page that lists the RMAs that have been deployed from Blaze Advisor.
2. Click the name of the Blaze project in the list for which you want to view rule instances.
3. In the navigation tree, click on the folder that contains the rules instance you want to view.

Note: The folder structure to access rule instances mirrors the same structure used in Blaze Advisor.

4. Click the name of the rule instance to view its details.

Viewing the Version History of a Rule Instance

Every time a user modifies a rule instance and checks the changes into the repository, My webMethods records the version as an entry on the Business Rules page. You can view the history of changes made to a rule instance on this page.

Note: You must have the appropriate My webMethods privileges to view a rule instance's version history. For more information, see ["Granting Users Permission to Manage Blaze Rules"](#) on page 42.

To view the version history of a Blaze rule instance

1. Follow the procedure in ["Viewing Rules in My webMethods"](#) on page 50 to view the rule instance.
2. In the RMA, in the **Version** drop-down list, select the rule instance for which you want to view the history.
3. Click **History**.

Updating a Rule Instance

From My webMethods you can use the RMA to update a rule instance. To update a rule instance when you are using Database Repository (BVS versioning - private), you need to check it out of the repository, make your changes, and then check the rule instance back in. After making your changes, you redeploy it to Integration Server, which updates the Integration Server rules service that is associated with the Blaze rule instance. As a result, the next time a process that uses the rule service invokes the rule service, it will execute the updated rule.

Note: You can only modify rule instances that were created from a template, and you must have the appropriate My webMethods privileges to update a rule instance. For more information about My webMethods privileges, see ["Granting Users Permission to Manage Blaze Rules"](#) on page 42.

To update a rule instance in My webMethods

1. Follow the procedure in ["Viewing Rules in My webMethods"](#) on page 50 to view the rule instance that you want to update.
2. While viewing the current working copy of the rule instance, click **Check Out**.

Note: If you decide you do not want to make changes, cancel the check out by clicking **Cancel Check Out**.

3. Make your changes to the rule instance.

Note: You can revert your most recent change by clicking **Cancel Edit**.

4. Click **Save** to save your changes.
5. Check the rule instance back in by doing the following:
 - a. Click **Check In**.
 - b. In the **Comment** field, type a comment to describe the changes you made.
 - c. Click **Check In**.
6. Click **Deploy** in the tool bar.

My webMethods displays a panel that lists rule instances that are eligible to be deployed to the Integration Server. The following table describes the information that My webMethods displays in the table:

Column	Description
Deployed	Whether the rule instance is already deployed on the Integration Server.
Template Name	The name of the template from which the rule instance was derived.
Ruleset Name	The name of the ruleset instance.

7. If you want to distribute the changes to all Integration Servers in a clustered environment, click the plus sign (+) to the right of Advanced Settings and then select the **Distribute Rules to a cluster** check box.

Note: For more information about clustering Integration Servers, see the *webMethods Integration Server Clustering Guide*.

Note: To deploy rule instances to a cluster of Integration Servers, each server must use the same process audit database, and the RMA associated with the rules to be distributed must specify one of the servers in the cluster. The table WMRULEDIST contains information about each Integration Server in a cluster.

Note: When you deploy rule instances from My webMethods Server to a cluster of Integration Servers, you can prevent one or more servers in the cluster from receiving these deployments by unregistering their entries in WMRULEDIST. To do so, navigate to the *Software AG_directory/IntegrationServer/instances/instance_name /*

packages/WmRules/config/ directory on each of those servers, open the rules.cnf file in a text editor, set the wm.rules.distribute property to `false`, and then reload the WmRules package.

Note: If any server in a cluster fails during the deployment of a rule instance, the deployment will roll back to the previous state of all servers in the cluster. My webMethods will display the alert message 'Deploy rolled back from all servers' on the Deploy panel.

8. Select the check box to the left of each item you want to deploy.
9. Click **Submit**.

Creating a New Rule Instance

You can create a new rule instance from any rule template that is deployed as an RMA from Blaze Advisor to My webMethods.

Note: You must have the appropriate My webMethods privileges to create a new rule instance. For more information, see ["Granting Users Permission to Manage Blaze Rules" on page 42](#).

To create a new rule instance

1. Follow the procedure in ["Viewing Rules in My webMethods" on page 50](#) to view the rule for which you want to create a new instance.
2. Click **Create New**.
3. Specify the following:

In this field...	Specify...
File Name	The name of the rule instance.
Folder	The RMA folder in which to store the rule instance.
Template	The template from which to create the rule instance.

4. Click **Create File**.

Deleting a Rule Instance

When you have the appropriate My webMethods privileges, you can delete a rule instance from an RMA. For more information about privileges, see "[Granting Users Permission to Manage Blaze Rules](#)" on page 42.

Note: When you delete a rule instance that was previously deployed to Integration Server, process steps that use rule services generated from the rule instance will continue to run even though the rule instance was deleted from the RMA.

To delete a rule instance

1. Follow the procedure in "[Viewing Rules in My webMethods](#)" on page 50 to view the ruleset that contains the rule instance that you want to delete.
2. Select the check box corresponding to the rule instance that you want to delete.
3. Click **Delete** in the tool bar.

Deleting an RMA

You can delete a Rules Maintenance Application in My webMethods Server. After doing so, the RMA will no longer appear in the list of RMAs on the Rules Maintenance Application panel and users will not be able to edit rule instances within the RMA in My webMethods.

This action does not remove information about the RMA's rule definitions from the repository or affect any processes that use those rules.

Note: You must have System Administrator privileges to delete an RMA.

To delete an RMA

1. Log in to My webMethods as the system administrator.
2. In the Configuration section of the Administration Dashboard, click **Install Administration**.
3. Select the check box corresponding to the RMA that you want to delete.
4. Click **Uninstall Selected**.
5. When prompted to confirm the uninstallation, click **Uninstall**.

6 Blaze Rule KPIs

■ Viewing Blaze Rule KPIs	56
---------------------------------	----

Viewing Blaze Rule KPIs

Changes that you make to a Blaze rule and then deploy to Integration Server will affect the processes and services that use the rule. In addition, any key performance indicators (KPIs) that are defined for a process instance will reflect a change in the behavior of the instance as a result of the modified rule. If the process owners who are monitoring KPIs do not know that a rule has been modified, they may misinterpret the changing KPI trends as abnormal when they may in fact be producing expected results.

To correlate KPIs with Blaze rule updates so that process owners can correctly interpret rule alerts and changing KPI trends, webMethodsOptimize offers a Blaze rule KPI called Business Rule Update. Whenever you deploy rules from Blaze Advisor or My webMethods to Integration Server, Optimize reports these events in this KPI. Process owners can use the Business Rule Update KPI to track how changes to a Blazerule affect a process or service.

If you want to use webMethods Universal Messaging instead of webMethods Broker for webMethods messaging, configure Integration Server, Optimize, and Process Engine to specify Universal Messaging as the JMS Server. Configure the JNDI provider alias and JMS connection alias in Integration Server to create the connection to the Universal Messaging server. For more information, see *webMethods Integration Server Administrator's Guide*, *Administering webMethods Optimize*, and *Administering webMethods Process Engine*.

Note: You must have the appropriate My webMethods privileges to view KPI information for Blaze rules. For more information, see "[Granting Users Permission to Manage Blaze Rules](#)" on page 42.

For more information about...	See...
Deploying rules from Blaze Advisor	"Deploying Rules to the Integration Server" on page 39
Deploying rules from My webMethods	"Updating a Rule Instance" on page 51
Defining, customizing, and interpreting KPIs	The PDF publication <i>webMethods Optimize User's Guide</i>

To view KPI information for changes to Blaze rules

1. In My webMethods:
 - Navigate > Applications > Monitoring > System-Wide > Analytics Overview**
2. In the Overview Configuration list, click **Analytics Business View**.

-
3. If necessary, use the search function to execute a search for the information that you want to view.

For more information about conducting a search on this panel, see *webMethods Optimize User's Guide*.

4. In the Detail column of the search results, select the **Include with KPI Summary** check box and then click the **View KPI Summary** button.
5. On the KPI Summary page, select the check box next to the KPI for which you want to view rule creation or update events.

My webMethods updates the KPI Performance section of the page. This section shows an event flag every time a rule is created or updated and redeployed to Integration Server.

6. Click an event flag for more information about the event, including the date and time on which the event occurred.
7. To see details about the changes made to a Blaze rule, see the sections on viewing rules and viewing the version history of a rule instance in "[Managing Blaze Rules in My webMethods](#)" on page 49.

Index

A

access privileges 42
 administrator tasks 16

B

BLAZE_REPOSITORY database table 25
 business analyst tasks 16
 business object model
 creating custom for 36
 using with IS documents 34, 37
 business process, using rules in 32
 Business Rules page
 creating rule instance 53
 deleting rule instance 54
 updating rule instance 51
 viewing rule instance history 51
 viewing rules 50
 Business Rules Settings page
 connection credentials for 45
 connection credentials for database repository 46

C

changing connection information 34
 checking projects into Versioning System 46
 cluster, deploying rules to Integration Server 52
 connection
 Integration Server 44
 My webMethods Server 44
 creating rule instance in My webMethods 53
 custom business object model, creating 36

D

data types supported in 35
 Database Component Configurator, using to create Blaze database schema 25
 database repository 17
 creating 24
 migrating from file repository 24
 required for rule use in process 17
 rule maintenance tasks you can perform 24
 updating connection credentials for an RMA 45
 decision table 38
 returning results of as IS document 37
 decision tree 38

deleting rule instance in My webMethods 54
 deploying RMAs 13, 46
 deploying rule projects 12, 12
 as RMAs 13
 deploying rules from 39
 developer tasks 16
 distributed Integration Servers, deploying rules to 52
 documentation
 using effectively 5

F

file repository 17
 migrating to database repository 24
 required for rule use in process 17
 rule maintenance tasks you can perform 24
 function rule type 38
 functional privileges 42

G

generateRepository command-line tool 25
 granting permission to manage rules 42

H

history, viewing for rule instance 51

I

Integration Server
 defining server connection for 33
 deploying rule instances to cluster 52
 deploying rules from My webMethods 51
 IS document, using in rules 34, 37

K

key performance indicator, viewing for Blaze rule updates 56
 KPI, viewing for Blaze rule updates 56

L

licensing restrictions 38
 limit of rules per project 38

M

migrating from file repository to database repository 24
 My webMethods
 creating rule instance 53

- deleting RMA 54
 - deleting rule instance 54
 - deploying updated rules to Integration Server 51
 - updating rule instance 51
 - viewing KPIs for Blaze rule updates 56
 - viewing rule instance history 51
 - viewing rules 50
 - My webMethods Server
 - defining server connection for 33
 - deleting RMA 54
- O**
- OEM licensing restrictions 38
- P**
- permissions, granting 42
 - private workspace 17
 - process model
 - referencing rules in rule steps 12
 - process, using rules in 17, 32
 - project
 - configuring connection 44
 - configuring properties 33
 - defined 11
 - deploying 12
 - deploying as RMA 13
 - rule limit 38
 - properties for deploying RMAs 33
 - properties for deploying rules 33
 - Properties tab for project 33
- R**
- repository
 - creating database type 24
 - database 17
 - file 17
 - generateRepository command-line tool 25
 - migrating from file to database type 24
 - setting up 24, 26
 - supported types 24
 - updating credentials for an RMA 45
 - requirements for managing rules 42
 - RMA 13
 - rule
 - adding steps to process model 40
 - controlling versions 16
 - defined 11
 - deploying to
 - from 39
 - granting permission to manage 42
 - inputs and outputs 38
 - KPI relationship 56
 - limit per project 38
 - managing
 - requirements 42, 46
 - managing in My webMethods 50, 54
 - creating rule instance 53
 - deleting rule instance 54
 - updating rule instance 51
 - viewing 50
 - viewing rule instance history 51
 - migrating from file repository to dat 24
 - storing in repository 16
 - type supported for use in processes 38
 - types of users who interact with rules 16
 - using in business process 32, 40
 - using IS document in 34
 - viewing instance history 51
 - viewing KPI 56
 - Rule Maintenance Application
 - defined 13
 - deploying 13
 - reason for deploying 13
 - Rules Maintenance Application
 - deleting from My webMethods Server 54
 - deploying 46
 - updating connection credentials 45
 - updating repository connection credentials 45
 - ruleset
 - defined 11
 - supported type for use in processes 38
- S**
- server connection properties 33
 - Software AG Designer, adding rule steps to process model 12, 40
- T**
- template, defined 11
- U**
- updating connection credentials for an RMA 45
 - updating rule instance in My webMethods 51
 - users, types who interact with rules 16
- V**

version control 16
 checking projects into Versioning System 46
 viewing rule instance history 51
Versioning System, checking projects into 46
viewing rule instance history in My webMethods 51
viewing rules in My webMethods 50

W

webMethods Optimize, viewing Blaze rule updates
as KPIs 56