TIBCO BusinessEvents® Enterprise Edition

WebStudio Users Guide

Version 6.3.1 | September 2024

Contents

Before You Begin

To maintain uniformity, the following terms have been used in the TIBCO BusinessEvents Studio UI and the product documentation:

- TIBCO ActiveSpaces software version 2.x is referred to as *Legacy ActiveSpaces*.

- TIBCO ActiveSpaces software version 4.6.1 and later are referred to as *ActiveSpaces*.

For details about the supported versions, see the *Readme.txt* file available at the TIBCO BusinessEvents® Enterprise Edition Product Documentation page.

Rule Management Server Prerequisite

In addition to Legacy ActiveSpaces as cluster and cache provider, you can also configure TIBCO BusinessEvents Rule Management Server (RMS) with the following combinations:

Cluster	Cache	Store
Apache Ignite	Apache Ignite	None/Shared Nothing/RDBMS/Store Providers (TIBCO ActiveSpaces and Cassandra)
TIBCO FTL	Apache Ignite	None/Shared Nothing/RDBMS/Store Providers (TIBCO ActiveSpaces and Cassandra)
TIBCO FTL	No cache	TIBCO ActiveSpaces

By default, Apache Ignite is used as the cluster and cache provider.

For more information about configuring these for your RMS project, see *TIBCO BusinessEvents Configuration Guide*.

Third-Party Software Documentation References

For complete details about the third-party software used in the project, see its documentation.

> ℹ **Note:** When you obtain third-party software or services, it is your responsibility to ensure you understand the license terms associated with such third-party software or services and comply with such terms.

Third-Party Software Documentation

Software	Used as	Documentation Reference URL
TIBCO ActiveSpaces 4.6.1 and above	Store provider	TIBCO ActiveSpaces documentation
TIBCO ActiveSpaces 2.x	Cluster and Cache provider	TIBCO ActiveSpaces documentation
Apache Kafka	Channel	Apache Kafka documentation
Confluent Schema Registry	Schema Registry	Confluent documentation
TIBCO Messaging - Schema Repository for Apache Kafka	Schema Registry	TIBCO Messaging - Schema Repository for Apache Kafka documentation
Apache Pulsar	Channel	Apache Pulsar documentation
Apache Cassandra	Store provider	Apache Cassandra documentation
GridGain	Data Center Replication	GridGain documentation
TIBCO FTL	Cluster provider	TIBCO FTL documentation

Software	Used as	Documentation Reference URL
Apache Ignite	Cluster and Cache provider	Apache Ignite documentation
InfluxDB	Metrics store provider	InfluxDB documentation
Grafana	Application metrics visualization	Grafana documentation
Ignite CDC	Data Center Replication	Apache Ignite documentation
Control Plane	Metrics store provider	TIBCO® Platform Documentation
Apache Maven	Native Maven projects	Apache Maven Documentation

TIBCO BusinessEvents WebStudio Concepts

The TIBCO BusinessEvents WebStudio along with the Rule Management Server (RMS) provides an online tool to create and manage business rules and `decision` tables.

Rules Management Server (RMS)

Rules Management Server (RMS) is a server-based component that manages the repository of the TIBCO BusinessEvents projects, lifecycle of the WebStudio artifacts and WebStudio user access.

RMS provides an easy, secure, and scripted deployment lifecycle. RMS is supported on all platforms that support TIBCO BusinessEvents. Users can access RMS remotely. RMS does not have to be installed on users' machines.

Under certain circumstances, the use of RMS is optional. For details, see *TIBCO BusinessEvents Decision Manager User Guide*.

You may require some basic configuration before you use RMS. You also set up the project resources required for business users to create decision tables or business rules for each TIBCO BusinessEvents project. See Configuring RMS Server Properties. Administrators can configure authentication and access control as needed, see Managing Roles.

RMS enables you to check out projects from the repository. The necessary project resources are loaded into the cache and the artifacts are copied into the client's machine. After you are finished working with the projects, you check in the modified artifacts to RMS for approval. For more details about workflow, see WebStudio and RMS User Workflow.

RMS is implemented using TIBCO BusinessEvents. Knowledgeable TIBCO BusinessEvents users can customize it. The example projects are located in the *BE_HOME*`/examples/standard/WebStudio` directory, you can configure the RMS to take projects from SVN location as well. Note that documentation is provided for the product as it is shipped. If you customize RMS, some documentation might not apply.

TIBCO BusinessEvents WebStudio Overview

As the TIBCO BusinessEvents WebStudio software is an online component, business users can create or manage business rules in a web browser.

In the TIBCO BusinessEvents WebStudio software you can define an executable rule (business rule) based on the rule template and the rule template view defined by the developer in TIBCO BusinessEvents Studio. A decision table in TIBCO BusinessEvents WebStudio is defined using the virtual rule function. RMS also supports lifecycle of domain model, where developers creates the domain models in BusinessEvents Studio and can edit them in Studio as well as WebStudio. Rule templates, rule template views, and virtual functions are created in TIBCO BusinessEvents Studio by developers and are stored in the Rule Management Server (RMS) repositories.

Administrators can define access control settings and can approve or reject commit requests originating from TIBCO BusinessEvents Studio and TIBCO BusinessEvents WebStudio users, check on the status of all such requests, and keep track of all project versions.

As an administrator you can also deploy decision tables or business rules when they are ready for use, either in an enterprise archive (EAR) file, or as class files. You can deploy class files for use at startup, or perform hot deployment. For more details, see Generating the Project Deployable Files.

The updates made to a decision table from TIBCO BusinessEvents Studio are visible in TIBCO BusinessEvents WebStudio and vice versa. If you have edited, saved, and committed a decision table in TIBCO BusinessEvents Studio, the changes are displayed in the WebStudio as well in the worklist for approval. Similarly, if you have edited, committed, and approved a decision table in WebStudio, the changes are displayed in TIBCO BusinessEvents Studio after you synchronize the artifacts of the decision table. In case of addition or deletion of a domain model or decision table in TIBCO BusinessEvents Studio, commit the changes in TIBCO BusinessEvents Studio. To see those changes in WebStudio, click the **Refresh** icon on the project explorer on the **Workspace** tab.

> ✓ **Tip:** For a better UI experience, the recommended screen resolution/scale and the browser zoom should be at 100%.

RMS REST API

You can use RMS through a REST-based web service using the APIs provided with WebStudio. Using these APIs you can perform complete WebStudio functionalities:

- for creating, modifying, or deleting artifacts (business rule, decision table)

- for lifecycle operations, such as, to check out, commit, or approve

- for worklist operations, such as, approval, rejection, or delegation of a worklist item

- for validating artifacts or analyzing decision tables.

After starting the RMS server, you can access API documentation, using the following URL:

http://*<hostname>*:*<port>*/WebStudio/apidoc

For example, the default URL to access the WebStudio API documentation is:

http://localhost:8090/WebStudio/apidoc/

Virtual Rule Functions and Decision Tables

In TIBCO BusinessEvents WebStudio, business users add decision table resources to VRFs. The decision table provides the body to the VRF, also known as the VRF implementation.

In TIBCO BusinessEvents Studio, technical users add *virtual rule functions* (VRFs) to a TIBCO BusinessEvents project. A VRF has no body, similar to a Java interface. Its implementation is provided using decision tables authored in TIBCO BusinessEvents WebStudio. VRFs are used in the TIBCO BusinessEvents project like any other rule function; they can be called from rules or other rule functions.

For example:

```
/**
 * @description
 */
virtual void rulefunction Virtual_RF.Applicant_VirtualRuleFunction {
   attribute {
      validity = ACTION;
   }
   scope {
      Concepts.Applicant       applicant;
      Events.ApplicationReceived       applicationreceived;
   }
```

```
    body {
    }
}
```

One VRF can have more than one decision table. If a VRF has more than one decision table, functions in TIBCO BusinessEvents determine how the tables are used.

Rule Building with Decision Tables

As a business user, you check out projects from RMS, build decision tables, and submit them for approval in TIBCO BusinessEvents WebStudio.

> **ℹ Note:** Decision tables are available in TIBCO BusinessEvents WebStudio in the edit mode, only if you have installed the Decision Manager component while installing the TIBCO BusinessEvents Enterprise Edition.

Decision tables provide a graphical way to build complex business rules. You create table columns by dragging and dropping predefined properties onto the decision table editor. The properties belong to ontology resources defined in the TIBCO BusinessEvents project. However, you can only use the properties specified in the VRF. Columns can be created in other ways too. You then define threshold values (conditions) and actions in the cells of the table. Each row can be thought of as one rule in a table made up of many rules. The individual rules are often straightforward, as in the following examples.

Three Rule Conditions:

```
Person.age < Max(20, Parent.age)
Person.creditscore >= Math.function(...)
Person.gender == "female"
```

Three Rule Actions:

```
Application.status = "ACCEPTED"
Application.credit = 4000
sendNotification()
```

However, one decision table can consist of hundreds, even thousands of rules each of which is executed only when its specific conditions are satisfied. In WebStudio a decision table could be displayed in multiple pages to make them easily manageable.

Exception Tables

Each decision table can optionally have another table known as an exception table. The purpose of the exception table is purely organizational: it enables you to separate the business logic of the main decision table (added by business users) from any non-business logic (generally added by technical users). For example, in the exception table, you could capture situations where fields are blank or contain invalid values, and define actions that send notifications or set return values. The rows of a decision table along with the rows of its exception table are considered in an RTC. If you prefer, you can put non-business logic in the main table instead of using an exception table.

Table Analyzer

The Table Analyzer feature analyzes decision tables and reports problems, such as uncovered ranges for conditions, uncovered domain entries, different set of actions for identical conditions.

Rule Template and Business Rule

A rule template is a specialized rule that contains a pre-conditions section which defines the pre-conditions that must be met in addition to the conditions defined by the business user in a business rule.

The Action Context section of a rule template defines all possible actions that can be taken by a business rule (after all conditions are met). Only the action context statements that the WebStudio user selects and defines as commands in the business rules are actually taken (depending on the rule evaluation at runtime). For a business rule execution to succeed, a business rule definition must include all Actions and the order should be maintained. Completing the definition of an action is a WebStudio user task. If bindings are used (and a view) then in WebStudio, the business rule writer has to enter only the binding values to complete the definition. Action context statements are of three types: create, modify, and call, plus arbitrary actions.

Rule Building with Business Rules

As a business user, you check out projects from RMS, build business rules, and submit them for approval in TIBCO BusinessEvents WebStudio.

Business rules provide builder or a user-friendly HTML form to build rules. In a builder you specify the conditions in the "when" section and the actions in the "then" section. You can use artifacts and supported operators for building conditions and actions. In the user-friendly HTML form, you fill up the values for the conditions and actions.

WebStudio and RMS User Workflow

The rules authored in TIBCO BusinessEvents WebStudio are made available to a TIBCO BusinessEvents application after following a complete approval workflow.

Step 1: RMS Project Setup

A TIBCO BusinessEvents user creates a TIBCO BusinessEvents project, adding the ontology, and writing rules that make use of virtual rule functions (VRFs) and Rule Templates.

The TIBCO BusinessEvents project for RMS is stored at the location defined by a property in the RMS server configuration. The RMS project requires an access control file for authorization.

The RMS server must be running so that the login and workflow actions are available in the TIBCO BusinessEvents WebStudio component.

Step 2: TIBCO BusinessEvents WebStudio

A business user starts the TIBCO BusinessEvents WebStudio component, logs on to RMS, and checks out the project. The business user creates one or more decision tables, business rules, or process and saves the modified project locally, then commits them for approval.

Step 3: Approval

An RMS user (with the "approval" access control rights) working in TIBCO BusinessEvents WebStudio receives the request and reviews the checked-in artifacts and then approves or rejects them. The approved artifacts are available for subsequent checkouts or updates.

Step 4: Generating Deployable Files

An RMS user generates deployable files for resources that are ready for deployment. You can generate EAR files or class files.

Step 5: Deployment

EAR files are deployed in the usual way, as explained in TIBCO BusinessEvents Administration. Class files for decision tables and rule template instance files for business rules are deployed by placing them in a configured location recognized by the TIBCO BusinessEvents engines at start. They can also be hot-deployed.

Decision Table Overview

A decision table row represents a business rule. It has one or more conditions and one or more actions.

Each condition cell is equivalent to one condition (one line) in the TIBCO BusinessEvents rule editor condition area. Similarly each action cell is equivalent to one action (one line) in the TIBCO BusinessEvents rule editor action area (that is, one rule). Decision table rules are like business rules. The rule that calls the virtual rule function that implements the decision table participates in inferencing in the usual way. When the virtual rule function is called, the decision table rules are applied.

Conditions and Actions

The columns of a decision table are made up of condition columns on the left, and action columns on the right. Each column represents one condition or one action.

A condition is a test that must evaluate to true before the action is executed. If a decision table rule uses multiple conditions, all the conditions for a row must evaluate to true for the action to execute.

In each row (rule) you define the specific conditions and actions. For example, if a condition column is Age (using a concept property of that name), then each row can define a different age range. The action for each row would define what action to take if a given concept instance contains an age property within the specified range.

If you add a second condition column called Income, then before the action is taken, a concept instance would be tested to see if both the age and the income are within the ranges specified in the rule's conditions.

> ⓘ **Note:** Conditions that are blank, contain an asterisk, or are disabled are ignored, and are treated as if they evaluate to true.

Regular and Custom Conditions and Actions

Regular condition

It is the value of an entity specified in the virtual rule function scope, or a simple comparison with the value - is greater than, is less than, is greater than or equal to, is less than or equal to.

Custom condition

It can use the rule language, standard functions, and data in the scope of the function at runtime (for example, scorecards and global variables). It can contain complex formulas.

Regular action

It sets the value of an entity specified in the VRF scope.

Custom action

The action can use the rule language, standard functions, and data in the scope of the function at runtime to do whatever is desired. For example, the action could be to send an event out to a different local computer for follow-up.

> **Tip:**
> **Using non-literal values in a regular condition or regular action**
>
> Suppose event A and event B are in scope, and event A has property ZZ, and event B has property YY. Both properties belong to the same data type. Suppose you then drag property ZZ to a condition or action column. In the cell, you can then specify a value as `b.YY`. The effect is different depending on the type of column:
>
> - In a condition column, this means: compare the value of property ZZ with the value of property YY.
>
> - In an action column this means: set the property ZZ to the value of YY.

Row Priorities

Each row in a decision table represents a separate business rule. You can control the order in which sets of rows are executed using the row priority setting.

Rows with higher priorities are executed before those with lower priorities, as follows:

- First, all conditions are checked for all rows that have the highest priority. The checking order within a set of rows with the same priority is not determinate.

- Then the rule actions for all of those rows whose conditions evaluate to true are executed. The execution order is not determinate. The runtime engine optimizes rule execution.)

- The process is repeated for all rows with the next highest priority, and so on.

Ten is the lowest priority and one is the highest. Five is the default priority.

Configuring RMS

Artifacts needs to be added to RMS, so that they can be used by TIBCO BusinessEvents WebStudio. Advanced configuration might also be required to enable certain features, such as, SSL authentication.

Adding a Project to RMS

TIBCO BusinessEvents projects, with VRF and rule templates, need to be added to the RMS repository to make them accessible in the TIBCO BusinessEvents WebStudio.

All RMS projects exist in a repository under a *root location* directory and their access control list (ACL) file exist under a *RMS configuration* directory. The default *root location* directory (as shipped) is *BE_HOME*`/examples/standard/WebStudio` and the default *RMS configuration* directory (as shipped) is *BE_HOME*`/rms/config/security` . You can configure the root location by configuring the `ws.scs.rootURL` property and the RMS configuration directory location by configuring the `ws.projects.acl.location` property in the `RMS.cdd` file.

Before you begin
A TIBCO BusinessEvents project created in TIBCO BusinessEvents Studio.

Procedure

1. Under the RMS project root location directory, create a directory with a name appropriate for the project.

    The root location is configurable. See Configuring RMS Server Properties.

2. Copy the TIBCO BusinessEvents Studio project contents that will be used in TIBCO BusinessEvents WebStudio to the project directory.

3. If the project contains custom functions, create the *ProjectName* directory under the directory identified by the `ws.projects.customLib.location` property in the `RMS.cdd` file and then place the custom function jars under it. See RMS Server Configuration Property Reference for more information on the property.

4. Copy the access control file for the project (*RMSProjectName*.ac) to the RMS configuration directory. See *TIBCO BusinessEvents Administration* for authentication topics.

   The RMS configuration location is configurable. See Configuring RMS Server Properties.

5. Restart the RMS server.

What to do next

You can also add a project to the RMS repository while the RMS is running using the existing RMS configurations. During run time, to add a project to the RMS repository, copy the project files to the RMS project root location and its ACL file to the RMS configuration directory. The project is then available for check out from the RMS repository. For more information, see Importing a Project to TIBCO BusinessEvents WebStudio.

Configuring RMS Server Properties

RMS works out of the box on a local machine. Configuration is required to enable access by remote TIBCO BusinessEvents WebStudio and the Decision Manager clients.

You may need to make other changes to the server configuration. For example, if you change the server location or the location of the RMS project repository, you must update the RMS server properties accordingly.

To configure the RMS server properties, edit the RMS.cdd file. You cannot edit a CDD file in TIBCO BusinessEvents Studio outside of its project context. To edit the CDD in TIBCO BusinessEvents Studio, import the BRMS project into your workspace. After that, copy the RMS.cdd file to its installed locations, in the following way

> **Note:** Whenever you change the RMS.cdd file you must restart the RMS server so that it uses the updated values.

See RMS Server Configuration Property Reference for information about each property.

> ℹ **Note:**
> - As with any procedure that changes files, first backup any files that could be affected. The RMS.cdd file is located in two places, in the product as shipped:
>   - *BE_HOME*/rms/project/BRMS/Deployment/RMS.cdd
>   - *BE_HOME*/rms/bin/RMS.cdd
> - After you update the RMS.cdd file in your workspace, overwrite the existing files with the updated ones, in the above locations (or the ones currently in use, if different).
> - Ensure that the CDD files in the BRMS project and the RMS startup directory are kept in sync.

Procedure

1. In TIBCO BusinessEvents Studio, choose **File > Import > Existing TIBCO BusinessEvents Studio Project**.

2. Ensure the **Copy the projects into workspace** check box is selected.

3. Select the following project:

   BE_HOME/rms/project/BRMS

4. In Studio Explorer, double-click **RMS.cdd** to open it in the CDD editor.

5. In the CDD editor Cluster tab, click **Properties**. In the Configuration panel on the right, expand groups to see the individual properties. Update the properties value, as per your requirement, as explained in RMS Server Configuration Property Reference.

6. When you have finished editing, save the RMS.cdd file.

7. In the file system, copy the RMS.cdd file from your workspace to the BRMS project and to the RMS server startup location.:

   The locations are:

   - *BE_HOME*/rms/project/BRMS/Deployment/RMS.cdd
   - *BE_HOME*/rms/bin/RMS.cdd

8. Restart the RMS server.

Enabling Remote Connection to RMS from WebStudio

For production settings, specify the RMS server hostname and port to ensure that TIBCO BusinessEvents WebStudio users can connect to RMS from remote machines.

As shipped, the `tibco.clientVar.Webstudio/hostname` property is set to `localhost`. This setting enables the product to be used on a single machine.

Procedure

1. Import the BRMS project into your workspace and open the `RMS.cdd` file for editing. See Configuring RMS Server Properties for details.

2. In the CDD editor Processing Units tab, click **WS-Inference**. In the Properties panel on the right, expand the WS property group.

3. Specify the TIBCO BusinessEvents WebStudio hostname and port using the following properties:

   - `tibco.clientVar.Webstudio/hostname`

   - `tibco.clientVar.Webstudio/port`

4. Save the `RMS.cdd` file.

5. In the file system, copy the `RMS.cdd` file from your workspace to the BRMS project and to the RMS server startup location. The locations are:

   - *BE_HOME*`/rms/project/BRMS/Deployment/RMS.cdd`

   - *BE_HOME*`/rms/bin/RMS.cdd`

6. Restart the RMS server.

Enabling SSL Authentication in RMS

The SSL authentication process uses certificates that are issued by a certificate authority. The same process applies if the certificates are issued by a certificate generation utility or if self-signed certificates are used.

To establish a SSL connection an application acting as an SSL client contacts an application acting as an SSL server. You can establish either a one-way SSL authentication

or two-way SSL authentication for the application.

Enabling One-way SSL Authentication in RMS

One-way SSL authentication enables the application operating as the SSL client, which in this case is web browser, to verify the identity of the application operating as the SSL server (RMS). The SSL-client application is not verified by the SSL-server application.

Procedure

1. Import the BRMS project in the TIBCO BusinessEvents Studio and copy it into your workspace. Now open the `RMS.cdd` file for editing. See Configuring RMS Server Properties for details.

2. In the **CDD editor Processing Units** tab, click **default**.

   In the **Properties** panel on the right, you can see the `be.channel.deactivate` property.

3. Remove the `/WebStudio/Core/Channels/WS_CH_Secure_WebstudioChannel` value from the `be.channel.deactivate` property.

4. Save the `RMS.cdd` file.

5. In the file system, copy the `RMS.cdd` file from your workspace to the BRMS project and to the RMS server startup location.

   The locations are:

   - *BE_HOME*`/rms/project/BRMS/Deployment/RMS.cdd`

   - *BE_HOME*`/rms/bin/RMS.cdd`

6. Restart the RMS server.

Result

To verify, type the URL `https://localhost/WebStudio/` in a web browser and press **Enter**. No port is required as default SSL port is used. The web browser notifies you about the untrusted certificate from the SSL server.

Enabling Two-way SSL Authentication in RMS

In two-way SSL authentication, the SSL client application, which in this case is a web browser, verifies the identity of the SSL server application (RMS), and then the SSL server application verifies the identity of the SSL-client application.

Procedure

1. Follow the steps mentioned in the Enabling One-way SSL Authentication in RMS to enable the SSL server verification, but do not restart RMS server yet.

2. In Studio Explorer, double-click **BRMS > WebStudio > Core > Transports > WS_ TR_Secure_WebstudioHTTPConnection.sharedhttp** to open it in the resource editor.

3. In the resource editor, click **Configure SSL**.

   The SSL Configuration for HTTPS Connections window is displayed.

4. In the SSL Configuration for HTTPS Connections window, select the **Requires Client Authentication** checkbox and click **OK**.

5. In the BusinessEvents Studio, regenerate the BRMS project EAR file (`RMS.ear`) and save it at *BE_HOME*`/rms/project/`. See *TIBCO BusinessEvents Developer's Guide* for more details.

6. Restart the RMS server.

7. Open your web browser and import the `WSClientStore.p12` certificate file in the web browser from the location *BE_HOME*`/rms/config/security/`. Refer to your web browser's help on how to import certificate in it.

Result

To verify, type the URL `https://localhost/WebStudio/`in the same web browser and press **Enter**. No port is required as default SSL port is used. The web browser displays a confirmation window to select your SSL client certificate and send it to the SSL server. After that the web browser notifies you about the untrusted certificate from the SSL server.

Updating Shared JDBC Connection in Runtime

Edit the `RMS.cdd` file to add the JDBC connection to the backing store from RMS at runtime.

Procedure

1. Import the BRMS project in the TIBCO BusinessEvents Studio and copy it into your workspace. Now open the `RMS.cdd` file for editing.

   See Configuring RMS Server Properties for details.

2. In the **CDD editor Cluster** tab, click the **Properties** in the Cluster pane and expand **node** under the Configuration pane.

3. Expand the **BackingStore** property group and edit the value for backing store properties as required.

   See Backing Store Property Group for more details about the properties.

4. Save the `RMS.cdd` file

5. In the file system, copy the `RMS.cdd` file from your workspace to the BRMS project and to the RMS server startup location.

   The locations are:

   - *BE_HOME*`/rms/project/BRMS/Deployment/RMS.cdd`

   - *BE_HOME*`/rms/bin/RMS.cdd`

6. Restart the RMS server.

Defining Temporary Files Storage Folder for RMS

Every time you restart RMS, a new `httpRoot` is created to store temporary data. You can modify this setting so that the temporary data is stored in the same location after every restart. To specify the location where temporary data must be stored, configure the `com.tibco.be.http.root` property.

Procedure

1. Open the `RMS.cdd` file or the `be-rms.tra` file for editing.

   These files are located at *BE_HOME*`/rms/bin`.

2. Add the `com.tibco.be.http.root` property and set the value to the temporary folder location.

   For example,

   ```
   com.tibco.be.http.root=C:/temp/RMSHTTPRootTmp
   ```

   > ℹ **Note:** Ensure that the specified folder is already present before restarting RMS.

3. Restart RMS.

Result

Now, RMS uses the specified folder (for example, RMSHTTPRootTmp) to store temporary data and does not create a new temporary folder on each restart.

RMS Server Configuration Property Reference

To configure the RMS server properties, edit the `RMS.cdd` file.

See Configuring RMS Server Properties for the related procedure.

Properties not documented in this table are either for internal use or are for other types of configuration explained elsewhere in the documentation.

Common Property Group

Property	Default Value	Description
rms.checkin.revisionId.initValue	10000	Initial value of the revision number to use for project check in
ws.artifact.deploy.location	*BE_HOME*/rms/shared	Name of the directory for storing EAR files that are generated for deployment. Separate directory for each project is

Property	Default Value	Description
		created, where EAR files for the project are stored. See also `be.codegen.rootDirectory`.
`be.codegen.rootDirectory`	`Codegen`	Name of the directory for storing class files generated by the Generate Deployable menu option. The directory location is relative to the `ws.artifact.deploy.location` directory. If this property is not present, then files are stored in the deployment directory. If this property is present but has no value, a directory is created with the default name `Codegen`.
`be.codegen.useLegacyCompilation`	`False`	Set the value to true, if you want to use the file-based legacy compiler to build the EAR file. By default, the value is false, that means EAR files are built in memory.
`ws.projects.customLib.location`	*BE_HOME*`/rms/lib/ext`	Path of the directory for storing custom function libraries for a WebStudio project. The custom function libraries are saved in a folder (with the same name as of the project name) inside the directory identified by the `ws.projects.customLib.location` property. For example, if the value of the `ws.projects.customLib.location` property is `C:/Tibco/BE52/be/6.0/rms/lib/ext`, then the custom function libraries for the MyTestProject project should be saved in

Property	Default Value	Description
		the location `C:/Tibco/BE52/be/6.0/rms/lib/ext/MyTestProject`.
`ws.projects.projectLib.location`	*BE_HOME*/rms/project-lib	Path of the directory for storing project libraries for a WebStudio project. The project libraries are saved in a folder (with the same name as of the project name) inside the directory identified by the `ws.projects.projectLib.location` property. For example, if the value of the `ws.projects.projectLib.location` property is `C:/Tibco/BE52/be/6.0/rms/project-lib`, then the project libraries for the MyTestProject project should be saved in the location `C:/Tibco/BE52/be/6.0/rms/project-lib/MyTestProject`.
`ws.validateDT.ignoreDeployableStatus`	false	Specifies whether to ignore the deployment status of the artifact at server restart and perform decision table validation without regenerating deployables. This property is required when persistence is not enabled for the local computer. When the persistence is enabled, this property is not needed as the deployment status is available across server restarts.

RMS Property Group

Property	Default Value	Description
`rms.project.workspace`	`workspace`	Not used in this release
`rms.project.decisiondata`	`decisiondata`	Not used in this release
`rms.project.deployment`	`deployment`	Not used in this release
`rms.workflowstages.config.file`	*BE_HOME* `/rms/config/RoleWorkflowStages.xml`	Not used in this release
`rms.lockworkflowstages.config.file`	*BE_HOME* `/rms/config/LockWorkflowStages.xml`	Not used in this release Default is BE_HOME /rms/config/LockWorkflowStages.xmL
`rms.roleArtifactTypes.config.file`	*BE_HOME* `/rms/config/RoleApplicableArtifactTypesConfig.xml`	Not used in this release Default is BE_HOME /rms/config/RoleApplicableArtifactTypesConfig.xml
`rms.external.entities.autodetect`		Used internally. Do not change the value.

Authentication and Authorization (ACL) Properties

User authentication topics are common to TIBCO BusinessEvents and the add-on products that use authentication. Common options are fully documented in *TIBCO BusinessEvents Administration*. The relevant properties are also listed as follows for your convenience.

Property	Default Value	Description
`be.auth.type`		Specifies the authentication

Property	Default Value	Description
		mechanism used. Additional configuration is required. The values are: • file • ldap • openldap **Note:** Authentication topics are documented in the *TIBCO BusinessEvents Administration* Guide.
`be.auth.file.location`	*BE_HOME*`/rms/config/` `security/users.pwd`	Location of the authentication file used for file-based authentication. **Note:** Authentication topics are documented in the *TIBCO BusinessEvents Administration* Guide.

`java.security.auth.`
`login.config`
	BE_HOME`/rms/config/`
	`security/jaas-`
	`config.config`
		Location of the JAAS login configuration file. You can substitute a different implementation of the JAAS login module than the one provided.
		
		Note: Authentication topics are documented in the *TIBCO BusinessEvents Administration* Guide.

`ws.projects.acl.location`	*BE_* *HOME* `/rms/config/security`	Location of the directory used for all ACL (authorization) files. Files must be named using the format *RMSProjectName*`.ac.`

RMS-GVs Property Group

This group of properties override the values of global variables that are provided in the TIBCO BusinessEvents Studio BRMS project. These properties are made available as global variables so that the properties can be overridden at runtime, as needed.

You can override the default global variable values in the `RMS.cdd` file for command-line startup. For details about overriding global variable values when deploying with TIBCO Administrator, see *TIBCO BusinessEvents Administration*.

For defining and using global variables in TIBCO BusinessEvents Studio projects, see *TIBCO BusinessEvents Developer Guide*.

Property	Default Value	Description
`tibco.clientVar.RMS/hostname`	`localhost`	Specifies the host name or IP address of the machine where RMS is hosted. Remote clients can connect to the server at this location.
`tibco.clientVar.RMS/port`	`5000`	Specifies the port number of the machine where RMS hosted. This port is used for listening to client requests. See notes for `tibco.clientVar.RMS/hostname`.
`tibco.clientVar.RMS/Approval /adminRole`	`Administrator`	The multiple roles specified here have Administrator role permissions. When you change the default value, the Administrator role still has Administrator permissions. You can also specify project specific administrator roles. To specify project specific admin roles, append the role to the project name separated by a colon (:). Ensure that the role has the approve permission set for the project. For example,

Property	Default Value	Description
| | | ```
tibco.clientVar.RMS/Approval/adm
inRole =
Administrator;CreditCardApplicat
ion:Business-User
``` |
		This shows the worklist items for the CreditCardApplication project, for users with the Business_User roles, while for all other projects, the worklist items show up for users under the Administrator role.
		Ensure that you replace the XML special character in the administrator role name value (if present) with the following characters:
		• "&" by "&"
		• " ' " by "'"

WebStudio Property Group

Property	Default Value	Description
tibco.clientVar.Webstudio/warDir	`rms/bin/WebStudio.war`	Path to the Web archive for the client side representation of the TIBCO BusinessEvents WebStudio
tibco.clientVar.Webstudio/hostName	localhost	Specifies the host name or IP address of the machine where RMS server is hosted for WebStudio
tibco.clientVar.Webstudio/port	`8090`	Specifies the port number

Property	Default Value	Description
		of the machine where RMS server is hosted for WebStudio. This port is used for listening to client requests. See notes for `tibco.clientVar.RMS/hostname`.
tibco.clientVar.Webstudio/Connection/timeout	`60000`	Timeout interval for the HTTP channel. This value is in milliseconds
tibco.clientVar.Webstudio/sessionTimeOut	`30`	Time interval for which an inactive user session will be valid. This value is in minutes
tibco.clientVar.Webstudio/DecisionTable/pageSize	`20`	Number of rows of the decision table to be displayed in a page
ws.validateDT.temp.dir	`rms/temp-dir`	Directory which is used as an interim storage of decision table contents during the validation process

Global Variable Properties for One-Way SSL Between RMS Server and Clients

Property	Default Value	Description
`tibco.clientVar.RMS/ security/securePort`	`8443`	The port to be used for secure HTTP communication in BusinessEvents Studio

Property	Default Value	Description
`tibco.clientVar.RMS/` `security/sslCertificateStore`	*BE_HOME*/rms/ config/security/ SSLKeyStore.ks	Absolute path of the JKS KeyStore which contains the RMS server's public/private keys. If the certificate used is CA signed, include the CA's certificate.
`tibco.clientVar.RMS/` `security/sslCertificateStorePassword`	`changeit`	The password for the KeyStore
`tibco.clientVar.RMS/` `security/secureWebPort`	`443`	The port to be used for secure HTTP communication in WebStudio

Properties for One-Way SSL Between LDAP Server, RMS Server, and Clients

> **ⓘ Note:**
> - For one-way SSL between LDAP server and RMS server, you must also change the value of `be.auth.ldap.port`.
> - For one-way SSL between RMS server and clients, you must also set the value of `be.auth.ldap.port`.

Property	Default Value	Description
`be.auth.ldap.ssl`		If set to true, enables SSL between the RMS server and the LDAP server If set to false or not set, the LDAP authentication uses a non-secure

Property	Default Value	Description
		channel.
`be.auth.ldap.ssl.trustStore`		The absolute path of the KeyStore containing the LDAP server certificate chain
`be.auth.ldap.ssl.trustStorePassword`		The password for the KeyStore
`be.auth.ldap.ssl.trustStoreType`		The KeyStore type for KeyStores other than JKS type, for example, PKCS12

Property for LDAP Authentication

Property	Default Value	Description
`be.auth.ldap.adminPassword`		The password for the LDAP administrator DN (distinguished name) for admin login.

Property for Filtering Based on LDAP Attributes

Property	Default Value	Description
`be.auth.ldap.filter.<attributeName>`		While using LDAP for authentication, you can filter users who get access to TIBCO BusinessEvents WebStudio based on their attributes on the LDAP server. The `<attribute>` in the property is the name of the attribute. Set the value of the

Property	Default Value	Description
		property to the value of the attribute that a user should have to get access to TIBCO BusinessEvents WebStudio.

For example, if you have a large number of LDAP users and they need to be authenticated based on the attribute of their roles, TIBCO BusinessEvents can filter them based on their roles. |

Email Notification Property Group

Email notification properties are used to send email notification when an artifact undergoes a status change in approval workflow, such as, artifact commit and artifact review (Approve, Reject, BuildAndDeploy).

In addition to these properties, you can define any SMTP protocol specific properties depending on the mail server configuration. These properties should be defined with a prefix `ws.notify.prop`, for example, `ws.notify.prop.mail.smtps.auth`, `ws.notify.prop.mail.smtps.ssl.enable`.

Property	Default Value	Description
`ws.notify.enabled`	False	If set to true, send email notification.

If set to false, do not send email notification. |
| `ws.notify.file.location` | *BE_HOME* `/rms/config /notify /UserNotify.Ids` | Specifies the location of the `.IDS` file which contains the mapping of WebStudio user IDs to the email IDs. This field is used in the case of file-based authentication. |

Property	Default Value	Description
ws.notify.ldap. userNotifyIdAttr	userPrincipalName	Specifies the name of the LDAP attribute for the email ID. This field is used in the case of LDAP based authentication.
ws.notify.impl. class	com.tibco.be.ws. notification.impl. EmailNotification	Specifies the implementation class for email notification support
ws.notify.context. impl.class	com.tibco.be.ws. notification.impl. EmailNotificationContext	Specifies the context implementation class for email notification support
ws.notify.message. template.file	*BE_HOME*/rms/config /notify/message.stg	Specifies the location of the message template file, which contains template (subject and content) for the commit, reject, and approve notification emails
ws.notify.mail. domain		Specifies the domain on the email. This is used with the RMS user name to construct the email address to which the notification is to be sent. For example, if RMS user name is "admin" and the property is set to "tibco.com", the email address for the notification is admin@tibco.com.
ws.notify.mail. receiver.cc.emails		Comma-separated list of email IDs or WebStudio user IDs to whom any commit, approve, or reject notifications need to be sent. In the case of CC notifications to the WebStudio users, specify their login IDs. The email IDs are picked up from the .IDS file identified by the ws.notify.file.location

Property	Default Value	Description
		property, if present; otherwise, the email ID is evaluated as *<login-ID>@<mail-domain>*. In the case of LDAP based authentication, the email ID is picked up from the user's LDAP entry.
`ws.notify.prop.` `MAIL_PROTOCOL`	SMTPS	Specifies the email protocol to be used to send the email. SMTP and SMTPS are valid values.
`ws.notify.prop.` `MAIL_SERVER_HOST`		Specifies the host name of the email server
`ws.notify.prop.` `MAIL_SERVER_PORT`		Specifies the port at which the mail server is listening
`ws.notify.prop.` `SENDER_EMAIL`		Specifies the email address to be used for sending the notification email
`ws.notify.prop.` `SENDER_user name`		Specifies the sender's user name for authentication to email server
`ws.notify.prop.` `SENDER_PASSWORD`		Specifies the sender's email password for authentication to email server
`ws.notify.prop.` `mail.smtps.auth`	true	This is a SMTP protocol-specific property. Specifies whether the authentication is checked or not for the email server.

`ws.notify.prop.`
`mail.smtps.ssl.`
`enable`
	true	This is a SMTP protocol-specific property. Specifies whether the email server is SSL enabled.

Hot Deployment Property Group

Use the hot deployment property group to configure JMX connection details in the CDD file and hot deploy the business rule and the decision table directly from WebStudio. You can configure JMX connection details for each project for each environment. Using these properties you can deploy multiple projects in multiple environments. For example, for deploying the CreditCardApplication project in the "QA" environment, set the QA environment in the `CreditCardApplication.ws.applicableEnvironments` property and set the JMX connection details in the `CreditCardApplication.QA.ws.jmx.*` properties.

You can also edit these properties in the WebStudio new user interface under the **Deployment Preferences** tab in the **Settings** page. For more details, see Settings Page Reference.

Property	Default Value	Description
`<ProjectName>.ws.applicableEnvironments`	QA,PROD	Specifies the environment where the artifacts can be deployed. Add this property for each project you want to deploy. For example, if you want to deploy the CreditCardApplication project in the PROD and the QA environment, set the `CreditCardApplication.ws.applicableEnvironments` property value to "PROD,QA".
`<ProjectName>.<EnvironmentName>.ws.jmx.hotDeploy.enable`	true	Specifies whether to enable or disable hot deployment (as part of BuildAndDeploy operation) for the specified project in the specified environment. This works only for cache based projects.
`<ProjectName>.<EnvironmentName>.ws.jmx.inMemory`	false	Specifies whether to enable hot deployment for the inMemory based projects. Do not specify for cache based projects.
`<ProjectName>.<EnvironmentName>.ws.jmx.host`	localhost	The name of the JMX host

Property	Default Value	Description
<ProjectName>.<EnvironmentName>.ws.jmx.port	9990	The unused port number, through which you want to enable the JMX connection
<ProjectName>.<EnvironmentName>.ws.jmx.user		The user name for the JMX connection
<ProjectName>.<EnvironmentName>.ws.jmx.password		The password for the user name for *<ProjectName>.<EnvironmentName>*.ws.jmx.user
<ProjectName>.<EnvironmentName>.ws.jmx.clusterName		The name of the cluster to be monitored
<ProjectName>.<EnvironmentName>.ws.jmx.agentName	inference-class	The name of the TIBCO BusinessEvents agent

Backing Store Property Group

Use the backing store property group to add a shared JDBC connection without the need to build the new EAR file (RMS.ear) for the BRMS project.

Property	Default Value	Description
tibco.clientVar.Webstudio/DB/driverName	oracle.jdbc.OracleDriver	The name of the JDBC driver class
tibco.clientVar.Webstudio/DB/url	jdbc:oracle:thin:@localhost:1521:orcl	The URL to use to connect to the database. You must supply the portions of the URL, for example, the host, port number, and database instance name. .
tibco.clientVar.Webstudio/DB/maxConnections	5	The maximum number of database connections to allocate. The minimum value that can be specified is 1.

Property	Default Value	Description
tibco.clientVar.Webstudio/ DB/userName		User name to use when connecting to the database
tibco.clientVar.Webstudio/ DB/password		Password to use when connecting to the database
tibco.clientVar.Webstudio/ DB/timeout	`zero`	Time (in seconds) to wait for a successful database connection. Only JDBC drivers that support connection timeouts can use this configuration field. If the JDBC driver does not support connection timeouts, the value of this field is ignored. Most JDBC drivers support connection timeouts.

Source Control System (SCS) Property Group

SCS property group enables you to check out a WebStudio project resources from a SCS location and commit the updates.

Property	Default Value	Description
`ws.scs.rootURL`	BE_HOME/examples /standard/WebStudio	Specifies the RMS repository location. This is the root location for RMS project directories.
		If repository is present in source control, then provide the repository URL of RMS projects as the value of `ws.scs.rootURL` .
		If you want to configure multiple RMS project locations, you can specify them separated by commas.

Property	Default Value	Description
`ws.scs.impl.type`	`file`	Identifies the type of repository being referred. The values that configured in the product out-of-the-box are: • file (default) • svn
`ws.scs.repo.dir`	*BE_HOME*`/rms/repo`	The local repository directory on your local computer. All checked out project resources are downloaded at this location.
`ws.scs.command.path`		Absolute path to the source control command line client. For example, if SVN is used then RMS server uses the SVN command line tool to connect to the SVN repository. Any SVN command line tool can be used, one commonly used one being Tortoise SVN CLI.
`ws.scs.default.username`		Default user name to connect to the repository mentioned in the `ws.scs.rootURL property`
`ws.scs.default.password`		Default password, encoded into the Base64 format, to connect to the repository mentioned in the `ws.scs.rootURL property.`For example, `Password123` should be encoded with Base64 format and entered as `UGFzc3dvcmQxMjM=`. **Note:** WebStudio uses the default user name and password (after decoding from Base64 format) to

Property	Default Value	Description
		check out and check in the objects from the SVN repository. You also have the option to enter your SCS user name and password (in the **Preference** portlet under the **Settings** tab) in the WebStudio UI which overrides the default user name and password. Enter the actual password in the WebStudio UI and not a Base64 format encoded one. See Personal Preferences Settings for more details.
`ws.scs.locking.enable`	`false`	You can lock the artifacts for the WebStudio. If the lock is applied on an artifact, the other users cannot modify or delete the artifacts until the lock is released.
`ws.scs.lock.timeout`	`0`	Time in seconds before the lock is automatically released. Set the property value greater than zero to apply timeout on locks.

Cross-Origin Resource Sharing (CORS) Property Group

Using the CORS property group, you can specify which external applications can use the WebStudio APIs.

Property	Default Value	Description
be.http.filter. cors.class	org.apache. catalina.filters. CorsFilter	The filter class name for the CORS filter

Property	Default Value	Description
be.http.filter. cors.param. cors.allowed.origins	*	A list of `Origin` request header values that are allowed to access the resource..
		BusinessEvents matches the value of `Origin` header of the HTTP request with this list. If the `Origin` request header value is present in the list, then the `Access-Control-Allow-Origin` header in the HTTP response returns the `Origin` request header value.
		The default value is "`*`" which means all URLs are valid, and the `Access-Control-Allow-Origin` header in the HTTP response returns "`*`" for all requests.
be.http.filter. cors.param. cors.allowed. methods	GET,POST,DELETE,PUT	A list of HTTP methods that can be used to access the resource, using cross-origin requests. These are the methods which will also be included as part of `Access-Control-Allow-Methods` header in pre-flight response.
be.http.filter. cors.urlpattern.1	/*	The URL pattern that triggers the filter

Integrating Custom Repository with RMS

TIBCO BusinessEvents provides API to integrate your repository with RMS. You can use this repository to store RMS projects.

You can use the `com.tibco.be.ws.scs.impl.repo` API to create implementation classes for your repository to integrate with RMS. TIBCO BusinessEvents provides a sample implementation (SVN repository) of this API at *BE_HOME*`/examples/standard/WebStudio/CustomRepository`.

For details about the `com.tibco.be.ws.scs.impl.repo` API and its classes, see *TIBCO BusinessEvents Java API Reference*.

Procedure

1. Create a Java class extending the base implementation class
   `com.tibco.be.ws.scs.impl.repo.AbstractRepositoryIntegration` and implement
   all its abstract methods.

   The `AbstractRepositoryIntegration` class is available in the `cep-rms.jar` file at *BE_HOME*/rms/lib. Add this JAR file to the classpath of the newly created Java class to
   resolve dependencies.

   For details about the `AbstractRepositoryIntegration` class and its methods, see
   TIBCO BusinessEvents Java API Reference.

   For better understanding of the implementation, refer to the sample implementation
   class created for the SVN repository at *BE_HOME*/examples/standard/WebStudio/CustomRepository/svn/src.

2. If your repository uses a CLI for interaction, install its command line executables.

   For details about the CLI installation for your repository, refer to your repository
   documentation.

3. After you have created the required implementation class, compile them using JDK
   and bundle the class in a JAR file.

4. Copy the JAR file to *BE_HOME*/rms/lib.

5. Edit the `RMS.cdd` file located at *BE_HOME*/rms/bin and set the `ws.scs.impl.type`
   property value to the fully qualified path of the custom implementation class.

   For example, for the provided sample SVN implementation, set `ws.scs.impl.type` as
   `com.tibco.be.ws.scs.impl.repo.svn.SVNIntegration`.

6. If your repository uses a CLI for interaction, in the `RMS.cdd` file, set
   `ws.scs.command.path` property value to the absolute path of the command line
   executable.

7. Save the `RMS.cdd` file and restart RMS.

Basic User Workflow in TIBCO BusinessEvents WebStudio

In TIBCO BusinessEvents WebStudio you can perform all the tasks with ease.

The following is a list of basic tasks that you can perform on the WebStudio UI:

1. Start the rule management server and log in to WebStudio, see Starting RMS and Signing-in to WebStudio.

2. After successful log in, check out a project from RMS repository to work on it, see Checking Out an RMS Project.

3. Generate the EAR file for the project, see Generating the Project Deployable Files.

4. Create a new decision table or business rule, or update an existing decision table or business rule.

   For details about how to create a new decision table, see Creating a Decision Table.

   For details about how to create a new business rule, see Creating a Business Rule.

   For details about decision table editor, see Decision Table Editor Reference.

5. Save and commit the updates to new or existing decision tables and business rules. You can use the **Commit** option in the context menu of the artifact or click the **Commit** button in the editor, see Committing Artifacts for Approval.

6. In the **Dashboard** tab, approve the commit request, see Approving or Rejecting a Commit Request from the Worklist .

7. After the commit request is approved, on the **Dashboard** tab, deploy the approved request at runtime, see Deploying an Updated Artifact to the Application.

Starting RMS and Signing-in to WebStudio

In WebStudio, before you can work on the RMS projects, you need to start and log in to the RMS.

Procedure

1. Start the RMS server by using one of the following ways:

   - Run *BE_HOME*/rms/bin/be-rms.exe (or be-rms.sh, depending on your operating system), with valid arguments. For example, on Windows you would open a command window in *BE_HOME*/rms/bin and execute:

     ```
     be-rms.exe
     ```

     The default cluster and cache provider is Apache Ignite. If you want to use Legacy ActiveSpaces as the cluster and cache provider instead of Apache Ignite for your RMS project, use the following command instead:

     ```
     be-rms.exe -c RMS_LegacyActiveSpaces.cdd
     ```

     If you want to use a TIBCO FTL cluster and an ActiveSpaces store without a cache, use the following command:

     ```
     be-rms.exe -c RMS_FTL.cdd
     ```

   - (Windows and for Apache Ignite provider only) Select **Start > All Programs > TIBCO >** *TIBCO_HOME* **> TIBCO BusinessEvents** *<version>* **> Start Rules Management Server**.

     The server is ready when you see a message in the command window such as:

     ```
     Info [HTTP-Channel-Startup] - [driver.http] Channel server for
     HTTP Channel [Port:5000] successfully started
     ```

2. Type the URL http://*host*:*port*/WebStudio/ in a web browser. Default host is localhost and default port is 8090. Host and port are configurable using properties in RMS.cdd. Ensure that the RMS is running before logging to TIBCO BusinessEvents WebStudio.

3. On the Sign In page, enter user name and password and click **Sign In**.

Result

After successful login, the WebStudio welcome page or the WebStudio Dashboard is displayed. The Welcome page is displayed if no project is already checked out. WebStudio automatically ends any other session of the user and make this session as the active session. For details about the TIBCO BusinessEvents Dashboard, see TIBCO BusinessEvents WebStudio Dashboard Reference.

Status Check for RMS Connection

Once a TIBCO BusinessEvents Studio, a TIBCO BusinessEvents Decision Manager, or a TIBCO BusinessEvents WebStudio client logs in to RMS, a Dashboard background activity periodically polls the RMS server to refresh the worklist items in WebStudio and `rms.heartbeat.relay` property (in `studio.tra`) is used by Studio to configure the heartbeat time.

By default, the delay is of two seconds.

If the server goes down, the menu options for RMS are automatically disabled. However, if you are already performing an RMS operation and the server goes down, the operation fails. The login option is enabled when the server comes up again.

TIBCO BusinessEvents WebStudio Dashboard Reference

The BusinessEvents WebStudio welcome page displays your information as a user. You can then navigate to settings and management page, and also check out the projects from this screen.

Figure 1: WebStudio Welcome Page

Element No.	Element	Description
1	Artifact Explorer	Opens the artifact explorer.
2	Workspace Tab	Workspace where you can modify the manageable artifacts using their respective editors.
3	Dashboard Tab	Click on the **Dashboard** tab to display the worklist items that need approval or that can be deployed. This is the default tab.

Element No.	Element	Description
4	Settings Tab	Set your preferences for users and application. For details, see Settings Page Reference
5	Management Tab	Use this tab for users, user roles, locks on artifacts and reports.
6	total unresolved items	Displays the number of unresolved items that you can either approve, reject, or delegate.
7	total deployable artifacts	Displays the number of deployable artifacts. Click here to access the worklist containing the deployable artifacts.
8	Recent User Activity	You can view all the activities that you have performed in the user timeline. The recent user activity also provides a search option and advance filters to filter out activities on timeline.
9	Project Snapshot	You can checkout the project from the project snapshot.

Importing a Project to TIBCO BusinessEvents WebStudio

To work on artifacts like decision tables and business rules of a project, you must first import a project to TIBCO BusinessEvents WebStudio.

Before you begin
- To import a project to TIBCO BusinessEvents WebStudio you need a ZIP file of the project.

- The name of the project file and the ZIP file must be the same.

- To be able to import a project, the user must have an administrator role permission.

Procedure
1. You can import a project from the Welcome screen or from the **Workspace**.

    - On the Welcome page, click the Upload icon.

    - (On the Workspace tab) Click on the **Checkout/Import Project** icon under the Artifact Explorer pane and click **Import Project**.

    The Import Project dialog opens.

2. In the Import Project window, to select a file, click **Select**.

3. Navigate to the ZIP file location of the project on your local computer and select that.

4. Click **Confirm**.

    > ℹ **Note:** An access control file is generated by default at the default location. You can change permissions in the access control file from the **Management** tab. For details, see Managing Roles.

What to do next
You can check out the imported project in WebStudio. See Checking Out an RMS Project.

Checking Out an RMS Project

To work on any managed artifact of a RMS project in WebStudio, check out the project item from RMS in WebStudio.

When generating an EAR file using the Generate Deployable option, the project resources from the RMS repository and not from the workspace are used.

Depending on your role or roles, you may not have permission to check out all resources or to do all of the documented tasks. For instructions on updating a project you already checked out, see Updating (Synchronizing) a Project.

> **Note:** The checkout dialog does not list the imported 4.x release projects, if the project name contains a dot(.). These imported projects should be renamed inside TIBCO BusinessEvents Studio to remove the dot(.) from the project name.

Before you begin

The project must be present in the RMS repository. For details about how to add project to RMS, see Adding a Project to RMS.

Procedure

1. Log in to WebStudio UI.

   For details about how to login, see Starting RMS and Signing-in to WebStudio.

2. You can check out an RMS project by either of the following methods:

   - On the **Dashboard** tab, click the **Check Out Artifacts** button on the Project Snapshot.

   - On the **Workspace** tab, click the **Checkout/Import Project** icon under the Artifact Explorer pane and click **Checkout Project**.

   - On the **Workspace** tab, click the **Checkout Artifacts** icon.

   The Check Out Artifacts window is displayed.

3. On the Check Out Artifacts window, select the RMS project from the drop-down list. You can then select the artifacts that you want to check out in WebStudio. Click **Confirm** to check out the selected artifacts.

   You can use the search bar to filter out the artifact that you want. You can also use the **Select All** button for ease in selecting artifacts.

Result

The checked out projects and its artifacts are displayed in the project explorer panel under the **Workspace** tab. For more details about the options present on the Artifact Explorer panel, see Artifact Explorer Panel Reference.

RMS Projects and Artifact Management Tasks

The projects and artifacts that were added to RMS can be used in WebStudio to create business rules. You can manage the artifacts in the Workspace tab.

After checking out a project, its artifacts are displayed on the Workspace tab under the Artifact Explorer, see Artifact Explorer Panel Reference.

By using this artifact explorer you can perform various operations on these artifacts to manage them in RMS.

Artifact Explorer Panel Reference

After the successful check out, the business rules and decision tables are displayed in the project explorer panel. To view entire checked out project artifacts in the project explorer panel select the **All Projects** artifact group. In the project explorer you can perform various operations on the artifacts and even check out other projects. You can also group or view artifacts according to your requirement.

Figure 2: WebStudio Project Explorer Panel

Element No.	Element Name	Description
1	Project Explorer Head	Displays the type of artifacts that are displayed on the Project Explorer. In case of the **Show Breadcrumbs** navigation style, this displays the breadcrumbs for the artifacts.
2	Filter	Enter the filter text to filter the artifacts in the project explorer.
3	Recently Opened Artifacts	You can view the list of recently opened artifacts by clicking the down arrow icon.
4	Checkout Artifacts/ Import Project	Click the **Plus** icon to import a project or check out artifacts of a project.
5	Refresh All Projects	Refresh the artifacts
6	Navigation Style	Click the **Menu** icon to select the navigation style for the project explorer. The following navigation styles are available: • **Show resources as a tree** - Shows all the artifacts along with their folder structure in the tree view. • **Show resources as a flattened list** - Displays all the artifacts of the project in a single list. This is the default navigation style. • **Show breadcrumbs** - Shows all the artifacts in the folder structure, but it only displays the artifacts of the selected folder. You can use the breadcrumbs links displayed on the project explorer head to navigate through the project.
7	Project Explorer	Displays list of artifacts of the checked out project. You can use the filter, navigation style, and artifact groups to filter the artifacts list on the project explorer. Click the managed artifacts (business rule and decision table) to open them in their respective editor.

Element No.	Element Name	Description
		Select the managed artifact and click the down arrow ([▾]) icon to open the list of available options for them. The available options are as follows:

Delete

Deletes the artifact. Click **Confirm** on the confirmation screen to delete the artifact. After deletion commit the changes to the project using the tree view.

You can also delete a project from the workspace using the tree view.

Commit

Commits the changes to the artifact for approval. In the Commit Changes window, enter a comment for the changes in **Message** text box. Click the artifact name to review the changes for commit. Click **Confirm** to commit the changes to RMS repository.

Synchronize

Synchronizes the update from the server to your local copy of the artifact. For more details about the fields present on the Synchronize Artifact window, see Synchronize Artifact Details Window Reference.

Revert

Reverts the update (if not committed) done to an artifact. In the Revert window, click the artifact name to review the changes to revert. Click **Confirm** to revert the changes.

Lock

Locks the artifact for the user and other user cannot commit any changes to the table until it is unlocked. This option works only if the `ws.scs.locking.enable` property is set to true in `RMS.cdd`.

Element No.	Element Name	Description
		Rename
		Provide a new name to the artifact. After changing the file name, commit the changes so the artifact is treated as a new artifact.
		History
		View list of the revisions of the artifact that were committed. For each revision, you can click **View Content** to view the revision that was committed or you can click **View Commit** to view the revision details.
		Audit Trail
		Displays the list of actions performed on the artifact by users. You can refine the results using the filters available in the Audit Trail window. For more information on the fields, see Viewing Audit Trail of an Artifact or Project.
		Export
		Exports the artifact to your local computer. Decision table is exported as an Microsoft Excel file (.xls or .xlsx) while business rule is exported as a .ruletemplateinstance file. For details, see Exporting Artifacts from TIBCO BusinessEvents WebStudio.
		Projects
		You can select this option for an artifact to perform actions on the project that it belongs to.
		Groups
		You can select this option for an artifact to edit the artifact group that the it belongs to.
		(Project only) Sync to Repository
		Synchronize the (non-managed) project artifacts changes,

Element No.	Element Name	Description
		that you have made externally after performing check out in WebStudio, into RMS. Select the option to open Sync To Repository screen, which displays the list of artifacts that are synced to the repository. Click **Confirm** to sync artifacts to RMS.
		(Project only) Validate
		Validates decision tables and business rules of the project, for any access control violations or syntax errors. EAR file for the project must be present in the deploy location; otherwise, the validate command fails.
		All the errors are displayed on the **Problems** tab at the bottom of the editor. New errors in syntax are added to these existing errors. Double-click errors to see the problematic view.
		(Project only) Generate Deployable
		Generates deployable files (EAR and class files) using the TIBCO BusinessEvents Studio project files, located in the RMS project folder. The deployables are generated in the location configured in the `RMS.cdd` file by the property `ws.artifact.deploy.location`, see Generated Files Location.
		Select this option to display the Generate Deployable screen. In the Generate Deployable screen enter the value of the following fields and click **Confirm** to generate deployables: • **Project Selected** - Select the project for which you want to generate EAR and class files. • **Generate Classes Only** - Toggle the switch to `ON` to generate only class files for decision tables and business rules. When set to `ON`, the EAR file for the project is not generated.
8	Artifact	Select the artifact group to display the respective artifact type

Element No.	Element Name	Description
	Groups	in the project explorer. The values are: • **All Projects** - Displays all checked out artifacts. • **Business Rules** - Displays only checked out Business Rules. • **Decision Tables** - Displays only checked out Decision Tables. You can also create a custom group where you can add artifacts that you want. Click on the **Create a new artifact group** icon, enter the new group name and click **OK**. You can add artifacts to group using either of the following options: • Drag and drop artifacts from the project explorer to the newly created artifact group to add the artifacts to the group. • Select the managed artifact and click the down arrow to open the list of available options for them. Select **Groups** and select the group to add the artifact.
9	Open Editors	Displays all the open artifact editors in the workspace.

Synchronizing a Project from RMS Repository

After you log into the RMS server, synchronize your local copy of the project, with the most recent changes from the RMS server, before continuing to work on it.

Other users might have checked in changes, additions, and deletions that are now available for check out. There can be other changes too. You can select what updates to accept.

WebStudio intelligently detects if the changes you are making is conflicting with the changes in the repository. WebStudio highlights the Synchronize button

Procedure

1. In the Project Explorer, click the down arrow icon for the required project and select **Synchronize**.

   The Synchronize Project window is displayed with all the artifacts for which the updated version is present on the server.

   Figure 3: Synchronize Project Artifacts

2. Click the artifact to view differences between the server and the local copy.

   The Synchronize artifact details window is displayed. For more details about the elements present on the window, see Synchronize Artifact Details Window Reference.

3. Click **Confirm** to apply the changes and click **Close**.

Result

The synchronization for the artifact is complete and you can see the synchronized copy in the editor.

Synchronize Artifact Details Window Reference

The Synchronize Artifact Details windows helps you to view and synchronize the artifacts from the repository.

Figure 4: Synchronize Artifact Details Window

Element No.	Element Name	Description
1	Strategy	Specifies the synchronizing strategy for the artifact. The values are: • **merge** - Merge your local copy of the artifact with the server copy and you have the option to choose to either keep the server changes or the local changes. • **latest()** - Discard the local copy and keep only the server copy.
2	Latest	Toggle to highlight the update from the server copy
3	Working	Toggle to highlight the update from your local copy
4	Reset	Resets all the toggles and changes that you have done on the Synchronize Artifact window
5	Legends	The legends for the highlighting displayed on the artifact
6	Conflict Resolution	Displays the artifact and then updates are highlighted based on the type of update.

Element No.	Element Name	Description
	Section	Conflicts are displayed when the same field is updated locally and in the server. In conflicts both the values are displayed, from which you can select the value that you want to keep and the value that you want to discard.

Double-click the conflict to open the Conflict Resolution window. Select the toggle for the value that you want to keep and click **Confirm**. |
7	Layout type	Select the type of layout for the artifact.
8	Show table properties	Shows the table properties like Effective Date/Time, Expiration Date/Time, Priority, and Single Row Execution. For more information about these properties, see Decision Table Editor Reference.
9	Show decision table	Use this button to show and hide the artifact.
10	Show exception table	Displays the exception table for the artifact.
11	Columns	Select check boxes for the columns that you want to filter out.
12	Filters	Select check boxes for the rows that you want to filter out.

Synchronizing Artifacts Changes into RMS

Using the **Sync To Repository** option, you can synchronize TIBCO BusinessEvents WebStudio copies of artifacts from the local copies. Using this function you can synchronize those artifacts that you cannot edit using TIBCO BusinessEvents WebStudio. For example, if you update an artifact such as a concept using the TIBCO BusinessEvents Studio, you can update its latest copy to the TIBCO BusinessEvents WebStudio using this option. The project artifacts which can be synchronized are concepts, events, virtual rule

functions, and rule templates.

During synchronization, only additive changes are synchronized to existing WebStudio managed entities, which are referencing those changed artifacts. The WebStudio managed entities are decision tables, business rules, and domain models.

Procedure

1. On the **Workspace** tab, click the **Change the Navigation Style** icon on the Artifact Explorer and select **Show resources as a tree**.

   The project artifacts are displayed in the tree view and project as the root of the tree.

2. Select the down arrow icon for the project whose artifacts you want to synchronize, and select **Sync to Repository**.

   The Sync To Repository window is displayed.

3. In the Sync To Repository window, select the artifacts that you want to sync from RMS and click **Confirm**.

   On successful synchronization, the success message is displayed.

Locking and Unlocking an Artifact

An exclusive lock ensures that while you are modifying the artifact, no other user is able to edit or delete it.

When other user tries to save the edit done to the locked artifact, or lock the locked artifact, WebStudio displays an error stating that the artifact is already locked by a user. After completing work on the artifact, you can release the locks on the artifact so that other users can also work.

Before you begin

Enable the artifact locking mechanism for WebStudio using the `ws.scs.locking.enable` property in the `RMS.cdd` file. See Source Control System (SCS) Property Group.

- **To Lock the artifact** - On the **Workspace** tab, in the Artifact Explorer pane, select the down arrow icon for the artifact you need to lock, and select **Lock**.

- **To Unlock the artifact** - On the **Workspace** tab, in the Artifact Explorer pane, select the down arrow icon for the artifact you need to unlock, and select **Unlock**.

    Alternatively, you can also select the **Auto Unlock on Approve/Reject** property check box to release the lock on the artifact as soon as the commit on the artifact is approved or rejected. The property can be set on the **Settings** tab under the **Personal Preference** settings. See Personal Preferences Settings.

    You can also set up a timeout for locks by using the `ws.scs.lock.timeout` property in `RMS.cdd`, after which the locks are released. See Source Control System (SCS) Property Group.

Reverting Artifact Updates

The Revert command can revert the updates that you made to an artifact in TIBCO BusinessEvents WebStudio and restore it to a previous version.
The Revert command also reverses the Delete command. It can reverse updates made to the following artifacts:

- Decision Tables

- Business Rules

- Domain Models

You cannot undo a revert action. The Revert command restores the artifact to the last version that you have committed to the repository. If multiple changes are made to the artifact and no commit is made to the repository, the Revert command reverses all the changes.

You can make a revert request for one or a selection of artifacts in the Revert Window.

Procedure

1. In BusinessEvents WebStudio, to open the Revert window, use one of the following options:

    - In the Artifact Explorer pane, select the down arrow icon for the artifact or the project containing the artifact whose changes you want to revert and select **Revert**.

    - In the artifact editor, click the **More Options** icon ( ) and select **Revert**.

2. (Optional) Click the artifact name to view the changes made to the committed artifact.

3. Select the check boxes for the artifacts whose changes you want to revert and confirm your action.

Result

The artifact is reverted to the latest committed version in the repository.

Committing Artifacts for Approval

After you complete the additions, changes, or deletions to a project you checked out of RMS, commit the artifacts for approval. You can make a commit request for a selection of artifacts or all artifacts of a project.

Copies of submitted artifacts are saved in cache with the names of the users who submitted them, version numbers, and the status `Committed`. A task is created for the Approver role to review the check-in requests.

A Revision ID is assigned to each commit request. This ID is used in other dialogs to allow you to view the details about the request.

If the project is checked out from the SCS repository, then, similar to the file mode, the committed artifacts are saved in the local cache, till the approval. After the approval, the committed artifacts from the cache are saved to the local repository, identified by the value of the `ws.scs.repo.dir` property in `RMS.cdd`, and checked in to the SVN repository as well. The comments entered for commit are used as check-in comments for the SVN repository, after approval.

Procedure

1. Open the commit window using any of the following methods:

   - In the Project Explorer panel, select the down arrow icon for the artifact or the project containing the artifact whose changes you want to commit and select **Commit**.

   - In the artifact editor, click the **Commit** icon.

2. In the Commit window enter the revision description in the **Message** text box.

   Comments field is not a mandatory field but it is recommended to add comments before committing. This helps the approver to understand the nature of changes.

3. Select the check boxes for the files you want to commit to the RMS project and confirm your action.

Viewing History of an Artifact

You could view the update performed in earlier revisions of the artifact, and also view the update performed to other artifacts in the those revisions.

Procedure

1. On the **Workspace** tab, in the Artifact Explorer pane, select the down arrow icon for the artifact for which you want to view history, and select **History**.

2. The **History** window displays all the previous revisions of the artifact. For each revision you can view the content of the artifact and committed changes.

   - View Commit - Click **View Commit** a revision to open the Commit window which lists all the artifacts which were committed in that revision. Click the artifact name to open the window displaying the content of the artifact.

     The updates on the updated artifact are displayed highlighted in different colors for addition, deletion, and modification. Refer to the legend displayed on the editor for the significance of each highlighted color.

   - View Content - Click **View Content** to view the content of the artifact.

3. Finally, click **Close** to close the History window.

Viewing Audit Trail of an Artifact or Project

You can view a list of all the actions that were performed on a selected artifact or project using the **Audit Trail** option.

Procedure

1. You can display the list of actions performed for a project or artifact using either of the following ways:

   - In the Project Artifacts explorer, select the down arrow icon for the supported artifact (decision table and business rule) or project and select **Audit Trail**.

     The Audit Trail window is displayed. For more details on the fields on the window, see Managing Reports.

   - (*Artifacts Only*) Open the business rule or the decision table for editing. In the editor, click **More Options** icon and select **Audit Trail**.

     The Audit Trail window is displayed. For more details about the fields on the window, see Managing Reports.

   - You can view the audit trail of artifacts that you want on the **Management** tab under the **Report Management** section. For more details, see Managing Reports.

Generating the Project Deployable Files

Deployable files are generated using the TIBCO BusinessEvents Studio project files, located in the RMS project's folder.

You can generate deployable EAR files only if you have the permission to check out all the required project resources (ACLs can limit what you can check out). The deployables are generated in the location configured in the `RMS.cdd` file by the property `ws.artifact.deploy.location`, see Generated Files Location.

Procedure

1. In the Project Explorer, click on the **Change the navigation style** icon and select any of these navigation style:

   - **Show resources as a tree**

   - **Show breadcrumbs**

   - **Basic tree navigation (no groups)**

   The project is displayed in the Project Explorer.

2. Select the down arrow icon for the project and select **Generate Deployable**.

3. On the Generate Deployable screen, enter the value of the following fields and click **Confirm** to generate deployable files.

    - **Project Selected** - Select the project for which you want to generate EAR and class files.

    - **Generate Classes Only** - Toggle the switch to ON to generate only class files for decision tables and business rules. When set to ON, the EAR file for the project is not generated.

    The deployable files are generated at the configured deployable location.

Generated Files Location

Location of generated files can be configured using deploy location properties in the `RMS.cdd` file.

See RMS Server Configuration Property Reference for more details about deploy location properties. Below are the locations of the generated files as per the preconfigured locations in the shipped product.

- EAR files are saved in the RMS project directory in the RMS's shared directory. For example, EAR file for CreditCardApplication project is stored at *BE_HOME*`\rms\shared\CreditCardApplication`.

- Project's class files are saved in the `codegen` subdirectory under the RMS project directory in the RMS's shared directory. For example, class files for CreditCardApplication project are stored at *BE_HOME*`\rms\shared\CreditCardApplicatio\codegen`

- Individual decision table's class file and business rule's rule template instance are saved in the RMS project directory in the RMS's shared directory.

Exporting Artifacts from TIBCO BusinessEvents WebStudio

You can work on decision tables and business rules in TIBCO BusinessEvents WebStudio and export them to your local computer.

Before you begin

The navigation style of the project explorer must be a tree structure. To change the

navigation style, on the project explorer, click the **Change the Navigation Style** icon (⋮)
and select **Show resources as a tree**.

Procedure

1. In the project explorer, select the down arrow icon for the project and in the
   available operations, and select **Export**.

   The Export Artifacts from window appears.

2. Search and select the artifacts that you want to export.

   > ℹ **Note:** You can use the **Select All** toggle to select all artifacts at a time.

3. Click **Export**.

   The selected artifacts are exported in a ZIP file.

What to do next

Extract the ZIP file to obtain the decision tables as Microsoft Excel (`.xls` or `.xlsx`) files and
the business rules as rule template instance (`.ruletemplateinstance`) files.

Removing a Project from Workspace

You can remove the project from the WebStudio workspace after completing work on the
project. Once deleted, you can still recover the project if the project is in repository but all
the local changes are lost.

- You can remove a project from the workspace by using either of the following options.

Tab	Steps
The **Dashboard** tab	1. Click **Delete Project** on the snapshot of the project you want to delete. 2. On the Confirmation window, click **Confirm**.
The **Workspace** tab	1. In Artifact Explorer, click the **Change the navigation style** icon and select **Show resources as tree**. 2. Select the down arrow icon for the project you want to delete and select **Delete**. 3. On the Confirmation window, click **Confirm**.

The project is deleted from the workspace and to work again on it, you have to checkout the project again.

Managing Decision Tables

In WebStudio, you can manage existing decision tables of the project or create a new decision table using the virtual rule function.

Creating a Decision Table

A decision table can only be created using a virtual rule function.

For instructions on adding a VRF, see *TIBCO BusinessEvents Developer Guide*.

Procedure

1. In the Project Explorer, select the down arrow icon for the VRF you want to use and click **New Decision Table**.

   The Create Artifact screen is displayed with the project name in the **Project Name** field and Decision Table in the **Artifact Type** field.

2. On the Create Artifact screen, enter the decision table name in the **Artifact Path** field and click **Finish**.

   The new empty decision table is displayed in the work area.

Decision Table Editor Reference

You can create and edit a decision table in WebStudio using the Decision Table editor.

The decision table editor displays the condition and action columns of the decision table and provide various options to edit and manage the decision table. The decision table also displays the auto-generated **ID** for each row (on the left-most column) and **Priority** of each row (on the right-most column). The default **Priority** value for each new row is 5 but if required, you can select a different value from the drop-down. For details about decision tables, see Decision Table Overview.

The following sample screen displays a decision table editor in TIBCO BusinessEvents WebStudio and provides brief description of its elements.

Figure 5: Decision Table Editor Reference

Element No.	Element	Description
1	Table Title	Displays the decision table name
2	More Options	Lists more options for decision table. The options are: **Delete** Deletes the artifact. Click **Confirm** on the confirmation screen to delete the artifact. After deletion commit the changes to the project using the tree view. You can also delete a project from the workspace using the tree view. **Commit** Commits the changes to the artifact for approval. In the Commit Changes window, enter a comment for the changes in **Message** text box. Click the artifact name to review the

Element No.	Element	Description
		changes for commit. Click **Confirm** to commit the changes to RMS repository.

Synchronize

Synchronizes the update from the server to your local copy of the artifact. For more details about the fields present on the Synchronize Artifact window, see Synchronize Artifact Details Window Reference.

Revert

Reverts the update (if not committed) done to an artifact. In the Revert window, click the artifact name to review the changes to revert. Click **Confirm** to revert the changes.

Lock

Locks the artifact for the user and other user cannot commit any changes to the table until it is unlocked. This option works only if the `ws.scs.locking.enable` property is set to true in `RMS.cdd`.

Rename

Provide a new name to the artifact. After changing the file name, commit the changes so the artifact is treated as a new artifact.

History

View list of the revisions of the artifact that were committed. For each revision, you can click **View Content** to view the revision that was committed or you can click **View Commit** to view the revision details.

Audit Trail

Element No.	Element	Description
		Displays the list of actions performed on the artifact by users. You can refine the results using the filters available in the Audit Trail window. For more information on the fields, see Viewing Audit Trail of an Artifact or Project. **Export** Exports the artifact to your local computer. Decision table is exported as a Microsoft Excel (`.xls` or `.xlsx`) file.
3	Save	Saves the changes to the decision table locally, but does not commit to RMS for approval
4	Commit	Commits the changes to the decision table for approval
5	Undo	Reverts last edit done to the decision table
6	Redo	Performs the action again that was reverted using the **Undo** command
7	Validate	Validates a decision table, for any access control violations or syntax errors in the table. EAR file for the project must be present in the deploy location; otherwise, the validate command fails. All the errors are displayed on the **Problems** tab at the bottom of the editor. New errors in syntax are added to these existing errors. Double-click errors to see the problematic view.
8	Effective Date/Time	Specifies the date and time on which the decision table becomes valid in the runtime application

Element No.	Element	Description
9	Expiration Date/Time	Specifies the date and time after which the decision table is no longer valid in the runtime application
10	Priority	Sets a priority for the decision table as desired. When a VRF has multiple implementations (decision tables), the decision table's Priority setting determines the order in which the decision tables executes
11	Single Row Execution	Set the toggle to ON, if you want the decision table to stop after one row (rule) passes the condition tests. Only the actions of that row are considered (until the next time the decision table's rule function is called).
12	Add Column	Opens the Add Column window. The fields available in the Add Column window are: **Custom Type** Toggle the switch to ON to add a custom column or toggle the switch to OFF to add a regular column. **Select Columns** Select the properties to be added as regular column to the decision table. You can select multiple fields and they each of those fields is added as separate column. This field is available only when **Custom Type** is OFF. **Column Name** Enter the custom condition or action. The custom condition or action can use the rule language, standard functions, and data in the scope of the function at run time (for example,

Element No.	Element	Description
		scorecards and global variables). It can also contain complex formulas. **Column Type** Select `Condition` or `Action` to add the selected column as condition or action column. **Property Type** (Read only) Specifies property type of the custom condition or action.
13	Add Row	Add an empty row at the end of the decision table. For your convenience, the newly added row is displayed at the bottom of the selected page of the decision table. Click **Save** to save the newly added row and its content. Once the new row is added, you can also right-click a cell in the row to perform operations like copy-pasting, duplicating, removing, or disabling the row. For specifying conditions with parameters of type Date/Time, you can right-click the cell and set the condition using the Date/Time picker. Select the operator from the **Select** dropdown and set the date and time.
14	Analyze	Opens table analyzer for analyzing the table. For details about table analyzer, see Analyzing a Decision Table
15	Columns	Select check boxes for the columns that you want to filter out.
16	Filter	Select check boxes for the rows that you want to filter out.

Element No.	Element	Description
17	Layout type	Select the type of layout for the artifact.
18	Show table properties	Shows the table properties like Effective Date/Time, Expiration Date/Time, Priority, and Single Row Execution. For more information about these properties, see Decision Table Editor Reference.
19	Show decision table	Use this button to show and hide the artifact.
20	Show exception table	Displays the exception table for the artifact.
21	Errors	The icon displays the number of errors. Click on the icon to open the **Problems** tab
22	Warnings	The icon displays the number of warnings. Click on the icon to open the **Problems** tab
23	Properties	Displays the properties for the decision table, cell, and the rule.
24	Lock Indicator	Indicates whether the artifact (decision table) is locked or unlocked

You can access the **Problems**, **Properties**, and **Test Data** tabs by clicking the **Error/Warnings** or the **Properties** icon. The following sample screen displays these tabs and provides brief description of its elements.

Figure 6: Problems Properties and Test Data of Decision Table Editor Reference

Element No.	Element	Description
1	Problems	Displays the errors and warnings after decision table validation.
2	Properties	Displays the properties for the decision table, cell, and the rule. • The **General** tab will contain information about fields like **Last Modified**, **Version**, and **Implementation Path**. • The **Cell** tab will contain information about whether a particular condition or action cell is enabled or not. You can also add a description about a cell in the **Comments** field. • The **Rule** tab will contain information about fields **Id** and **Priority**. You can also add a description about the rule in the **Comments** field.

Element No.	Element	Description
3	Test Data	Test data is used for validating decision table. Using test data you can verify if the decision table covers all the data that might be passed through the decision table. The test data is not managed in the decision table. The test data must be in the project in your repository.

Analyzing a Decision Table

In the Table Analyzer you can set example condition values and perform various validation checks on the currently displayed decision table.

> **ℹ Note:** Table analyzer supports the Not Equal To (!=) operator.

Procedure

1. On the Workspace tab, open the decision table you want to analyze.

2. Click **Analyze** to view Table Analyzer.

3. In the Table Analyzer section, controls are created dynamically for setting values and ranges for each condition in the table. Enter or select the example values for which you want to analyze the table.

   For example, if the condition is a range, for instance "< 40" or "> 10 && < 100", then the corresponding control in the Table Analyzer section are textboxes for the minimum and maximum value of the range. You can then select a range of values within the range.

4. Click the **Analyze** icon 🖼 to analyze the decision table based on the selected values.

   A pop-up window displays the count of any issues. The Problems section at the bottom displays more information on issues. Click the issue to highlight the row.

5. You can click the **Show Coverage** icon ⊞ in the table analyzer section to highlight all rows that meet the criteria. In page view, browse through the pages to see the highlighted rows.

   For example, if you select `Male` for `Gender`; `true` for `Eligible`; and `50000-60000` for `Income`, clicking **Show Coverage** displays all rows that have `Male`, `true`, and an income between 50000 and 60000.

Validating a Decision Table

In TIBCO BusinessEvents WebStudio, validate a decision table, for any access control violations or syntax errors in the table, using the validate command.

> **ℹ Note:** The Validate icon is activated when the decision table is opened in the editor. EAR file for the project should be present in the deploy location; otherwise, the validate command fails. If the EAR file is not present at the deploy location, then you receive the following error while validating the decision table:
>
> ```
> EAR for projects: <project_name> either not present or not in sync
> at deploy location.
> ```

Procedure

1. Select the decision table and click the **Validate** icon ☑ to validate the table.

   All the errors are displayed in the **Problems View** tab at the bottom of the application. New errors in syntax are added to these existing errors.

2. Double-click errors to see the problematic view.

   Take any needed corrective actions and then validate the table again until all errors are resolved.

Validating a Decision Table Using Test Data

Using test data you can verify if the decision table covers all the data that might be passed through the decision table. This enables you to add more rules to the decision table, if the

decision table is not able to cover the test data.

Before you begin

Create the test data for at least one argument from BusinessEvents Studio. See *TIBCO BusinessEvents Developer's Guide* for more information on creation of test data. The test data must be in the project in your repository.

> ℹ **Note:** The test data are not managed in the RMS repository.

Procedure

1. In the **Table Analyzer**, click the **Check Test Data Coverage** icon 📑.

    > ℹ **Note:** You can select the test data for only a single argument at a time.

    The Check Test Data Coverage window is displayed with the available test data.

2. Select the test data, for which you want to validate the decision table and click **OK**.

    The table analyzer validates the decision table based on the test data and displays the **Test Data Coverage** tab at the bottom pane. The Test Data Coverage pane highlights the rows of test data, which are covered in the decision table.

Column Options for Decision Tables

Hover over the column name and click the **More Options** icon to perform operations on columns in a decision table.

Figure 7: Decision Tables Column Options

Element No.	Element	Description
1	Pin Column	Use this option to pin the selected column to the left or the right.
2	Autosize This Column	Use this option to resize the column to accommodate different sized entries in that column.
3	Autosize All Columns	Use this option to resize all the column to accommodate different sized entries in them.
4	Reset Columns	Use this option to reset the values in the cells to their last saved value.
5	Remove Column	Removes the column from the decision table.
6	Field Settings	Select this option and in the Column Field Settings window that appears, set the following:

Element No.	Element	Description
		• **Column Alias**: Sets a simple column alias for the column. **Note:** To see the column alias, select the **Show Column Alias If Present** check box on the **Settings** tab. • **Default Text**: Sets the default text for every new row that is added under the column. If you want to enter the default value in exiting empty cells, toggle the **Include existing rules** switch to ON position.
7	Select	Select the check boxes for unique cell values in the column to filter out entries that match the selected values.
8	Filter	Select check boxes for the columns that you want to filter out.

Exporting a Decision Table

You can export decision tables from TIBCO BusinessEvents WebStudio to Microsoft Excel spreadsheets.

Procedure

1. You can run an export command in either of the following ways:

   • Select the down arrow icon for the decision table you want to export and in the Project Explorer and select **Export**.

   • Open the decision table you want to export. Click the **More Options** icon in the toolbar and select **Export**.

   The Save As dialog box is displayed for the exported file.

2. Navigate to the directory in which you want to save the excel (`.xls` or `.xlsx`) file and provide a file name.
   By default, decision tables are exported in the `.xlsx` format.

3. Click **Save**.

Importing a Decision Table

You can import correctly formatted Microsoft Excel file to create a decision table or to merge to an existing decision table in TIBCO BusinessEvents WebStudio.

> **ℹ Note:**
> - Only import from the Microsoft Excel files is supported. You cannot import from CSV (comma separated values) files or tab-delimited text files. You can open such files in Excel and save as Excel format binary files.
>
> - Before importing the Microsoft Excel file, ensure that cells are not protected (locked). The protected cells, after the import, appear disabled in WebStudio.

Figure 8: Importing a Decision Table Flowchart

Procedure

1. Select the down arrow icon for a virtual rule function in the project explorer and select **Import**.

   The Import dialog box is displayed.

2. In the Import dialog box, click **Select Files** to browse and **Open** the excel file that you want.

   The **Name** field is automatically populated with the excel file name.

3. In the **Name** field, you can either provide an existing table name or a new name.

   - If table already exists - Select **Overwrite** or **Merge** based on your requirement and click **Confirm**. If you have selected **Overwrite**, the decision table with the same name is overwritten. However, if you have selected **Merge**, the Synchronize window is displayed where you can view the modified values for merging and click **Confirm**. For more details about the elements present in the window, see Synchronize Artifact Details Window Reference.

   - If table is new - Verify the **Parent Folder** and click **Confirm**.

Supported Operators

TIBCO BusinessEvents WebStudio supports several operators for Decision Tables that you can use in the expressions.

Named or Scoped Conditions must Evaluate to Boolean

For logical operators, named conditions will have to use them explicitly from the second operand onwards.

For example, if the LHS of the condition is the column name, bankuser.Age, the following are valid entries:

```
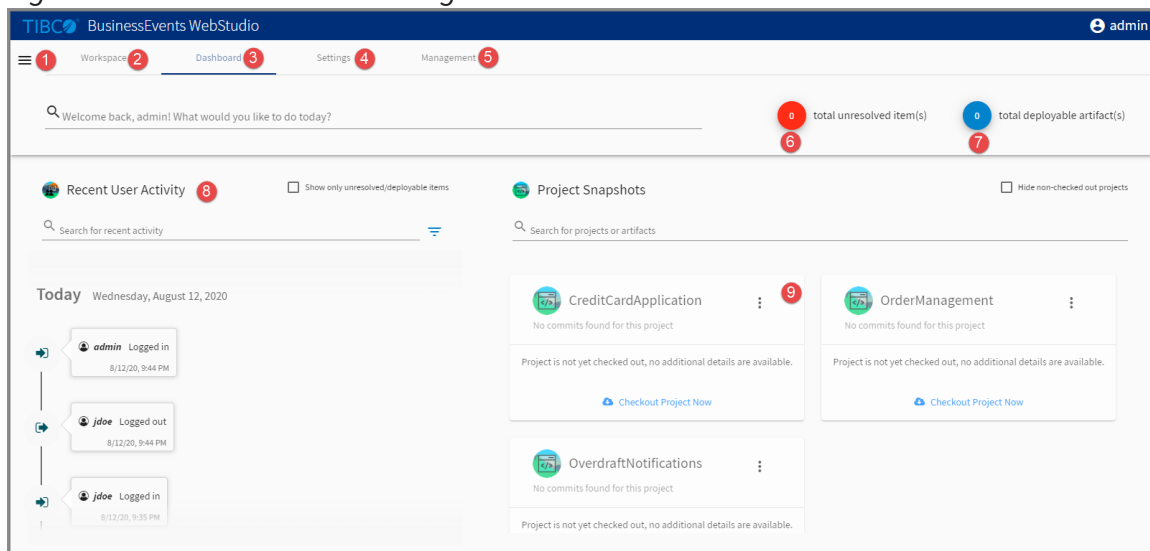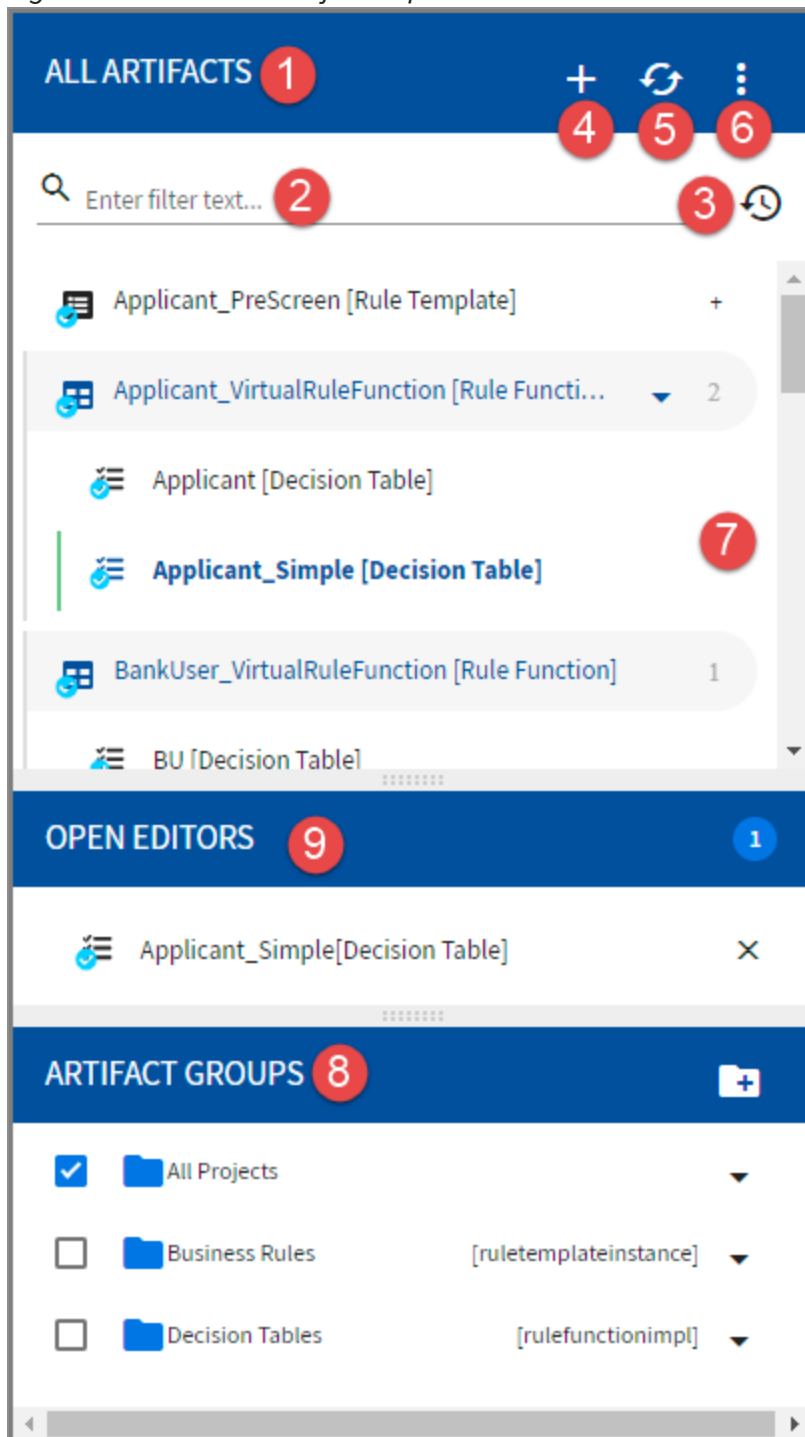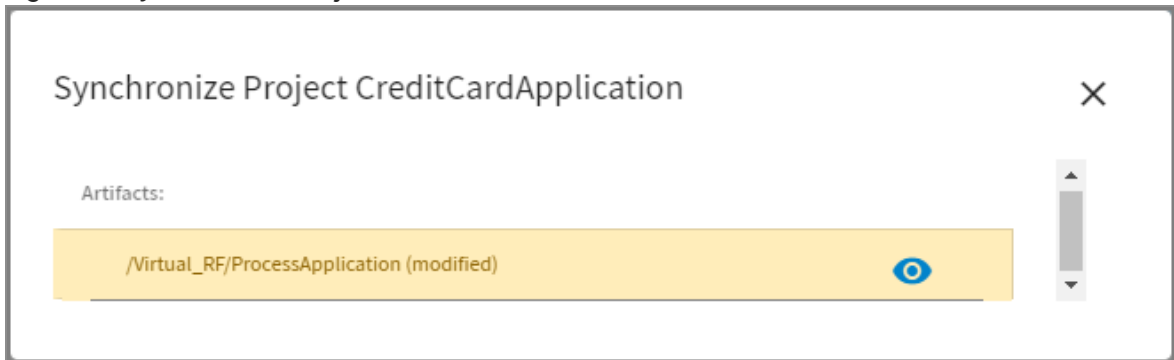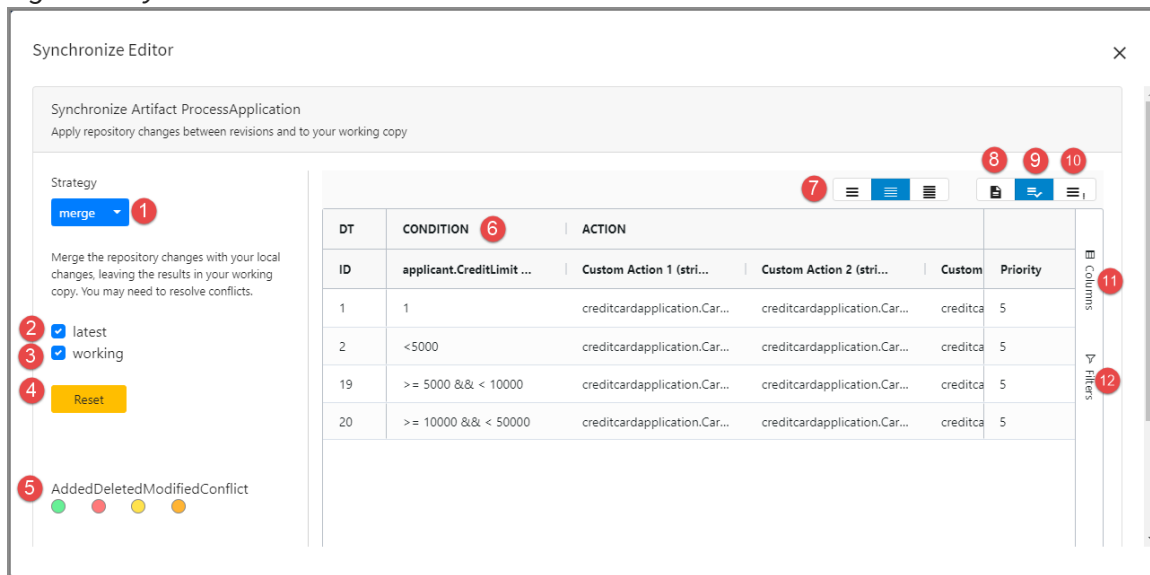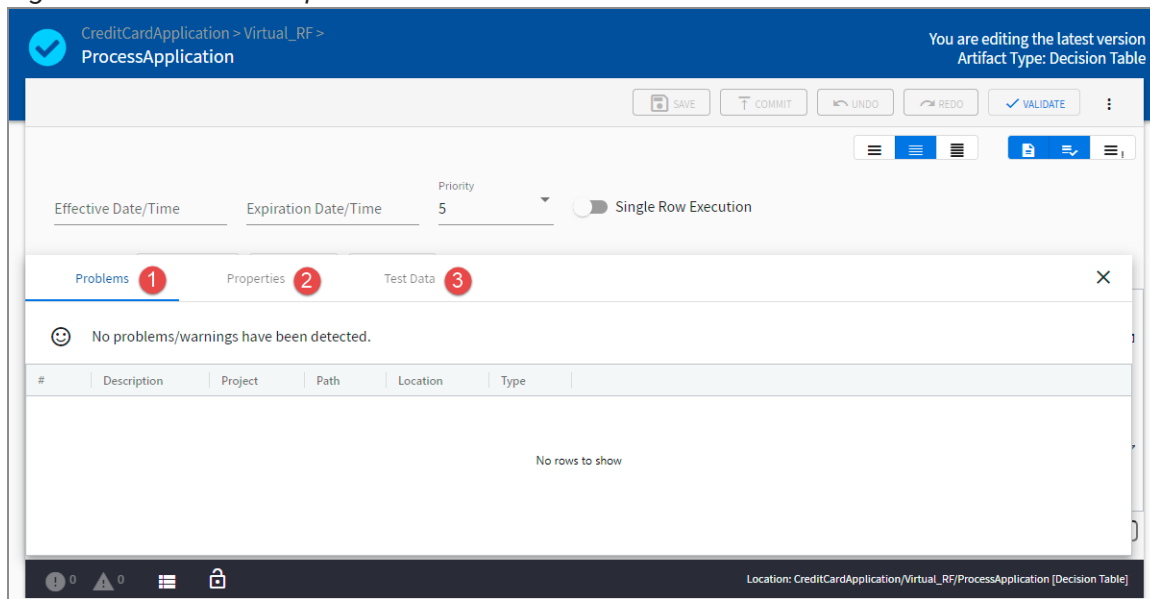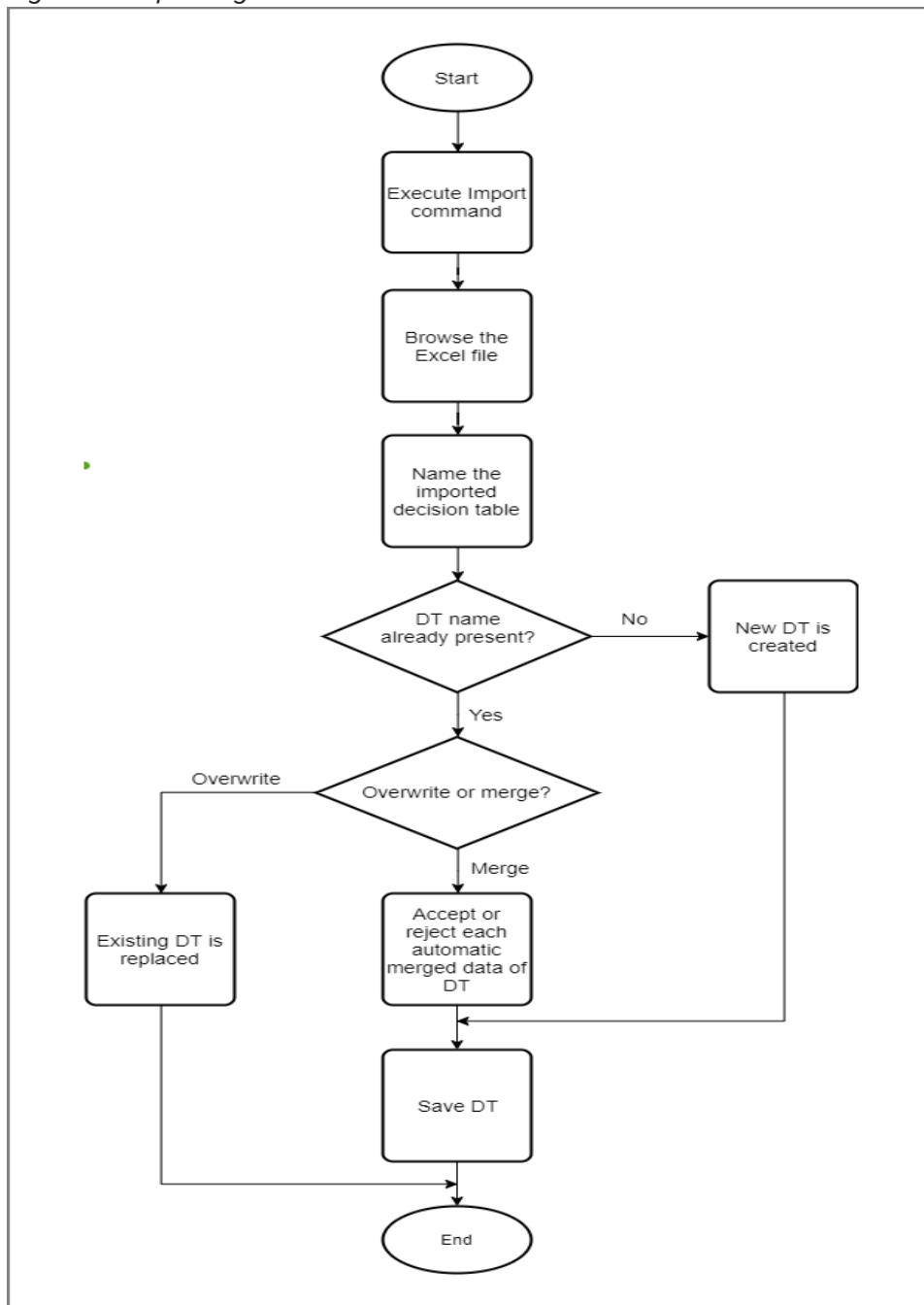50 || ==60 , 20 || >=80
```

Custom Conditions must evaluate to Boolean

Same as above, except that you must specify the entire Boolean expression in the cell, including LHS.

Actions or Custom Actions

Named actions perform assignment of the cell expression to the column name property.

Custom actions are treated as normal strings and can be anything that you would enter in the THEN section of rules (LHS included).

Supported Operators

Operator	Description
==	Equals to
!=	Does not equal to
<>	Does not equal to
>	Greater than
>=	Greater than or equals to
<	Less than
<=	Less than or equals to
||	Logical OR
&&	Logical AND
+	Addition or String concatenation (depending on context)
!	Logical NOT (only with boolean)
–	Subtraction
*	"Don't Care" (that is, ignore), or multiplication (depending on context)
/	Division
=	Assignment
.	Scope resolution
()	Operator precedence order or function call (depending on context)

Operator	Description
[]	Array declaration or array indexing (depending on context)
{}	Block resolution or array Initialization (depending on context)
++	Increment (used only with custom actions)
--	Decrement (used only with custom actions)

Business Rules

Business Rules artifact allows non-technical business users to build complex business rules in TIBCO BusinessEvents WebStudio.

Creating a Business Rule

A business rule can only be created using a rule template.

For instructions on adding a rule template, see *TIBCO BusinessEvents Developer Guide*.

Before you begin

Ensure that the **All Projects** option is selected under the Artifact Groups in the project explorer. Thus, rule templates are visible in the project explorer.

Procedure

1. In the Project Explorer, select the down arrow icon for the rule template you want to use and click **New Business Rule** option.

   The Create Artifact window is displayed with the **Project Name** and Rule Template Instance in the **Artifact Type**.

2. In the Create Artifact window, enter the business rule name in the **Artifact Path** and click **Finish**.

   The **Project Name** and **Artifact Type** fields are pre-filled with appropriate values.

Result

The new business rule is displayed in the work area for editing. The business rule editor depends on the rule template implementation in the application:

- If the rule template has a rule template view associated with it, an easy to use form is displayed to specify conditions and actions.

- If the rule template do not have a rule template view associated with it, a builder interface is displayed to add condition and actions. For details, see the following topics:

    - [Adding a Conditional Clause by Using the Condition Builder](#)

    - [Adding an Action Using the Command Builder](#)

    - [Business Rule Editor Reference](#)

Business Rule Example

In the bundled example `CreditCardApplication`, there are two rule templates: `SpecialOffers` and `Applicant_PreScreen`. The `Applicant_PreScreen` rule template has a rule template view (`PreScreenTemplateView`) associated with it. The `PreScreenTemplateView` artifact contains the HTML code for presenting the form based on fields defined in the `Applicant_PreScreen` rule template. Thus, when you create a new business rule from the `Applicant_PreScreen` rule template, a simple HTML form opens up based on the HTML code defined in `PreScreenTemplateView`. However, if you create a business rule from the `SpecialOffers` rule template, a builder interface opens up which helps to create more complex rules.

Adding a Business Rule Using the HTML Form

If a rule template view is defined for the rule template, then you can provide values for the bindings, defined in TIBCO BusinessEvents Studio, in a HTML form.

The form interface is the visual presentation of the rule template defined by rule template views. See *TIBCO BusinessEvents Developer Guide* for more details about rule template views.

Procedure
1. On the Workspace tab, in the Artifact Explorer pane, select the down arrow icon for the rule template you want to use and select **New Business Rule**.

2. Type the business rule name in the dialogue box and click **Finish**.

   If the rule template view is defined for the rule template, then the HTML form is presented in the business rule editor. Based on the data types in the rule template view, the HTML element are generated for the HTML form.

   Rule Template Data Type and HTML Field Type Mapping

Rule Template Data Type	HTML Field Type
Integer	Textbox
Double	Textbox
String	TextBox/TextArea (based on the **Select widget for string bindings in Rule Template Instance** field in the **Personal Preferences** settings). If you specify the `WidgetType` as TextBox or TextArea in the binding itself, you cannot change it in the **Personal Preferences** setting.
Boolean	Check box
String (with associated domain model)	Dropdown list
Datetime	Datetime widget
Long	Textbox

3. Enter the threshold values in the input fields of the HTML form.

4. Click **Save** to save updates to the business rule.

What to do next

Validate the business rule based on the value entered and bindings defined for the rule template, see Business Rule Editor Reference.

Adding a Conditional Clause by Using the Condition Builder

The When section lets the user define additional conditions that must be met, in addition to the precondition defined in the rule template.

The builder determines the scope of the rule template and allows you to define individual conditional clauses based on that scope. Each conditional clause has one or more sub-clause, and a match operator to determine whether the conditional clause is true based on the sub-clause values.

If you require to move any condition up or down the order then you can drag-and-drop the condition to move it. Move the mouse pointer over any condition till the pointer is converted to a drag icon. Now, you can drag that particular condition and drop it within the same or different filter.

Procedure

1. In the business rule editor, under the **When** section, click the artifact-name link and select the artifact you require.

2. Click the operator-name link to select another operator, or a child artifact of the previously selected artifact, as you require. See Business Rule Operators for a list of supported operators.

3. Enter the conditional value for the artifact in the empty box.

4. Click the **Add** icon . to another condition to the conditional clause, or click the **Remove** icon ▫ to remove the existing condition from the conditional clause, if you want.

5. Click the **Add Sub-clause** icon ▫ to add a conditional sub-clause to the conditional clause if you want.

6. Click the match-operator link to select the matching condition for which the conditional clause evaluates to true. The values are:

Option	Description
Match Any	Conditional clause is true if any of the condition evaluates to true
Match All	Conditional clause is true only if all of the conditions evaluates to true
Match None	Conditional clause is true only if none of the conditions evaluates to true

7. Click the **Save** icon to save updates to the business rule.

What to do next

Adding an Action Using the Command Builder

Business Rule Operators

Business Rule builder operators consists of the conditional clause operators and the action builder operators. The operator list changes as per the parent entity type in the builder.

Condition Builder Operators

Operator Value on WebStudio Client	Parent Type	Operator Value on RMS Server
Matches Other Field	• Concept / Event • Data / Time / Datetime • Integer / Float • Text / String • Boolean	== *<field>*

Operator Value on WebStudio Client	Parent Type	Operator Value on RMS Server
Differs From Field	- Concept / Event - Date / Time / Datetime - Integer / Float - Text / String - Boolean	!= *\<field\>*
Is Null	- Concept / Event - Date / Time / Datetime - Integer / Float - Text / String	== null
Is not Null	- Concept / Event - Date / Time / Datetime - Integer / Float - Text / String	!= null
Greater Than	Integer / Float	>
Greater Than Field	Integer / Float	> *\<field\>*
Greater Than Equal To	Integer / Float	>=
Greater Than Equal To Field	Integer / Float	>= *\<field\>*
Less Than	Integer / Float	<

Operator Value on WebStudio Client	Parent Type	Operator Value on RMS Server
Less Than Field	Integer / Float	< *<field>*
Less Than Equal To	Integer / Float	<=
Less Than Equal To Field	Integer / Float	<= *<field>*
Equals	• Integer / Float • Text / String • Boolean	==
Not Equals	• Integer / Float • Text / String • Boolean	!=

Action Builder Operators

Operator Value on WebStudio Client	Parent Type	Operator Value on RMS Server
Set To	• Concept / Event • Date / Time / Datetime • Integer / Float • Text / String	=
Set To Field	• Concept / Event • Date / Time / Datetime • Integer / Float	= *<field>*

Operator Value on WebStudio Client	Parent Type	Operator Value on RMS Server
	• Text / String	
	• Boolean	
Set To Null	• Concept / Event	= null
	• Integer / Float	
	• Text / String	
Increment By	Integer / Float	+=
Increment By Field	Integer / Float	+= *<field>*
Decrement By	Integer / Float	-=
Decrement By Field	Integer / Float	-= *<field>*
Set To True	Boolean	= true
Set To False	Boolean	= false

Adding an Action Using the Command Builder

The **Then** section defines additional actions to be taken based on the conditions defined.

The builder lets the user define actions based on the statements defined in the `actionContext` section of the rule template.

> **Note:** Ensure that all the actions in the business rule are defined and the order should match that of the rule template. Otherwise, the business rule will be invalid.

Procedure

1. In the business rule editor, under the **Then** section, click the **Action** dropdown list to select the actions available for the rule template.

2. Click the artifact-name link and select the artifact you want modify.

3. Click the operator-name link and select the command for the artifact. For list of available operators, see Business Rule Operators.

4. In the text box, enter the value that you want to set for the artifact.

5. Click the **Add** icon ₊ to add another parameter for the action, or click the **Remove** icon ₋ to remove the parameter for action, if you want.

6. Click **Save** to save updates to the business rule.

Business Rule Editor Reference

You can create and edit a business rule in WebStudio using the business rule editor.

The following sample screen displays a business rule editor in WebStudio and the following table provides brief description of its elements.

Figure 9: Business Rule Editor Reference

Element No.	Element	Description
1	Table title	Displays the business rule name
2	Save	Saves the changes to the business rule in cache, but does not commit to RMS
3	Commit	Commits the business rule changes to RMS
4	Undo	Reverts last edit done to the business rule
5	Redo	Performs the action again that was reverted using the **Undo** command
6	Validate	Validates a business rule, for any error. EAR file for the project must be present in the deploy location; otherwise, the validate command fails. All the errors are displayed on the **Problems** tab at the bottom of the editor. New errors in syntax are added to these existing errors. Double-click errors to see the problematic view.
7	More options	Lists more options for business rule . The options are: **Delete** Deletes the artifact. Click **Confirm** on the confirmation screen to delete the artifact. After deletion commit the changes to the project using the tree view. You can also delete a project from the workspace using the tree view. **Commit** Commits the changes to the artifact for approval. In the Commit Changes window, enter a comment for the changes in **Message** text box. Click the artifact name to review the changes for commit. Click **Confirm** to commit the changes to RMS repository.

Element No.	Element	Description
		Synchronize
		Synchronizes the update from the server to your local copy of the artifact. For more details about the fields present on the Synchronize Artifact window, see Synchronize Artifact Details Window Reference.
		Revert
		Reverts the update (if not committed) done to an artifact. In the Revert window, click the artifact name to review the changes to revert. Click **Confirm** to revert the changes.
		Lock
		Locks the artifact for the user and other user cannot commit any changes to the table until it is unlocked. This option works only if the `ws.scs.locking.enable` property is set to true in `RMS.cdd`.
		Rename
		Provide a new name to the artifact. After changing the file name, commit the changes so the artifact is treated as a new artifact.
		History
		View list of the revisions of the artifact that were committed. For each revision, you can click **View Content** to view the revision that was committed or you can click **View Commit** to view the revision details.
		Audit Trail
		Displays the list of actions performed on the artifact by users. You can refine the results using the filters available in the Audit Trail window. For more information on the fields, see Viewing Audit Trail of an Artifact or Project.
		Export

Element No.	Element	Description
		Exports the artifact to your local computer. Business rule is exported as a `.ruletemplateinstance` file.
8	When section	The **When** section lets the user define additional conditions that must be met, in addition to the precondition defined in the rule template. For more information, see Adding a Conditional Clause by Using the Condition Builder.
9	Then section	The **Then** section defines additional actions to be taken based on the conditions defined. For more information, see Adding an Action Using the Command Builder
10	Errors	The icon displays the number of errors. Click on the icon to open the **Problems** tab
11	Warnings	The icon displays the number of warnings. Click on the icon to open the **Problems** tab
12	Properties	Displays the properties for the decision table, cell, and the rule.
13	Lock Indicator	Indicates whether the artifact (decision table) is locked or unlocked

Domain Models

A domain model specifies the values that you might find useful for defining ontology item properties. For example, instead of typing text in a certain table cell, you can pick a value from a list or enter a value within a predefined range.

You can define and edit domain models using TIBCO BusinessEvents Studio. You can also edit them using TIBCO Cloud Events WebStudio. For more information, see Editing Domain Models.

Domain Models in Decision Table Cells

Using domain models, you can easily fill values for cells in a decision table by choosing from a list or by entering values from a predefined range.

All domain model values can have optional descriptions. A preference determines whether domain model values or their descriptions appear in decision table cells. For some applications displaying descriptions can make the table easier to understand.

Users will find it is easier to work with the description than the code. As another example, for a Boolean data type, the description can provide words such as Accepted and Rejected for the values True or False. At runtime, the actual value is used.

Editing Domain Models

Using RMS, you can manage the lifecycle of domain models, and edit them in TIBCO BusinessEvents WebStudio.
You can create a domain model only in BusinessEvents Studio; however, you can edit them in Studio as well as WebStudio. The updates are committed in the RMS and are available to all user after the approval.

See *TIBCO BusinessEvents Developer's Guide* for more information on domain models.

Before you begin

Ensure that **Project** is selected under the **Groups** section in WebStudio.

Procedure

1. In WebStudio, click the **Workspace** tab.

2. On the **Workspace** tab, select the **All Projects** artifact group to display domain models in the artifacts list.

3. In the Artifact Explorer, select the domain model that you want to update.

   The Domain Model editor opens up on the right panel.

4. Perform either of the operations to update the domain model:

   - Click **Add domain entry** icon to add a new row to the domain entry list.

   - Select any of the existing domain entry and click the **Duplicate** icon to create a duplicate entry.

   - Double-click on any entry to edit the domain entry value and its description.

   - Click the **Inherits from** drop-down and select any other domain model in your project, from which you want to inherit domain entries.

   - Select an existing domain entry and click the **Delete** icon to remove that entry.

5. Click **Save** to save the changes.

6. Click **Validate** to check for any issues.

7. Click **Commit** to commit the changes to the RMS.

Approval Workflow Overview

After updating the artifacts (decision tables or business rules), submit them for approval. All business user actions, such as committing or deleting artifacts are committed to the RMS project only after a approver approves those updates.

The approval permissions are set at the project level using the ACL file for the project. Allow the `approval` action type for the user role in the ACL file to the grant approval permission. See *TIBCO BusinessEvents Administration* for details about setting permission for a user role at the project level.

> **Note:** This manual documents the behavior of the product as shipped. The workflow, the roles used, and the permissions granted to the roles are all configurable. However, the general flow is likely to be similar to that described in this guide.
>
> Use of RMS—and therefore the approval process—is required when TIBCO BusinessEvents WebStudio is used.

See Committing Artifacts for Approval for business user procedures.

See Approving or Rejecting a Commit Request from the Worklist for approver procedures.

Who Can Approve or Reject Commit Requests

In TIBCO BusinessEvents WebStudio users with Administrator role permissions can approve or reject commit requests, or delegate approval. They can also delegate the task of approving or rejecting commit requests to any specified user role and not to a specific user. In this case, the approval or rejection requests will be sent to all the users in that role. See Delegating a Workitem from the Worklist.

Approval Status

COMMITTED

When a user commits one or more resources for approval, the status of the request is set to Committed. The approver then sets the status as appropriate.

APPROVE

The request was approved. If the submission request was to add or change an artifact, that artifact is copied to the RMS project. If the submission request was to delete an approved artifact, that artifact is deleted from the RMS project.

REJECT

The change was rejected. If the submission request was to add or change an artifact, that artifact is not copied to the RMS project. If the submission request was to delete an approved artifact, that artifact is not deleted. It is up to the business user to make changes in their local project accordingly.

BUILDANDDEPLOY

The artifact is built and hot deployed. The artifact after approval is built and hot deployed at the location specified by a property in the `RMS.cdd` file. See Generating the Project Deployable Files.

Dashboard Reference for Working with Unresolved Artifacts

On the Dashboard tab, you can access the worklist containing the unresolved artifacts to approve or reject a commit request, delegating a workitem, and comparing committed and previous versions of artifacts.

On the **Dashboard** tab, you can access the worklist containing the unresolved artifacts from multiple locations as shown in the following image:

Figure 10: Unresolved Workitems from the Worklist

Item No.	Description
1	This icon displays the total number of unresolved items for all projects
2	User activity timeline displays entry for each unresolved artifact
3	This icon displays the number of unresolved worklist items for that project

Approving or Rejecting a Commit Request from the Worklist

You can approve or reject requests after they are committed to the RMS.
The **Dashboard** tab displays unresolved artifacts that you can approve or reject. For details, see Dashboard Reference for Working with Unresolved Artifacts.

Before you begin

You should have the permission for the approval action type in the project's access control file.

Procedure

1. In TIBCO BusinessEvents WebStudio, click the **Dashboard** tab.

2. On the **Dashboard** tab, open the worklist of unresolved items in one of the following ways:

   - Click the icon specifying the total number of unresolved items on the top of page.

   - In the user activities timeline, scroll down to the unresolved artifact entry.

   - Click the icon specifying the number of unresolved worklist items on the project card.

3. In the worklist, expand the entry for the commit that you want to approve or reject by clicking its name for example `/Virtual_RF/ProcessApplication (modified)` to see the modifications made to it. For details, see Comparing the Committed and Previous Version of Artifacts.

4. Click **Approve** or **Reject**, depending on the action you want to take for the application changes.

5. Click **Finish**.

Delegating a Workitem from the Worklist

A user with the Administrator role can delegate a workitem to one or more roles. Users belonging to any of these roles can see the workitem in their worklists.
The **Dashboard** tab displays unresolved artifacts that you can approve or reject. For details, see Dashboard Reference for Working with Unresolved Artifacts.

Before you begin

Artifacts can be delegated to only those users who have the permission to approve or reject the committed artifacts.

Procedure

1. In TIBCO BusinessEvents WebStudio, click the **Dashboard** tab.

2. On the **Dashboard** tab, open the worklist of unresolved items in one of the following ways:

    • Click the icon specifying the total number of unresolved items on the top of page.

    • In the user activities timeline, scroll down to the unresolved artifact entry.

    • Click the icon specifying the number of unresolved worklist items on the project card.

3. In the worklist, expand the entry for the commit that you want to delegate.

4. Click **Delegate**.

    The Delegate window opens.

5. From the **Select Role/s To Delegate To** list, select the users to whom you want to delegate the unresolved artifacts.

6. Confirm the selected roles.

Comparing the Committed and Previous Version of Artifacts

Before approving or rejecting a commit request, you can view updates that are made to the committed version before approving or rejecting the commit request.
The **Dashboard** tab displays unresolved artifacts that you can approve or reject. For details, see Dashboard Reference for Working with Unresolved Artifacts.

Procedure

1. In TIBCO BusinessEvents WebStudio, click the **Dashboard** tab.

2. On the **Dashboard** tab, open the worklist of unresolved items in one of the following ways:

    • Click the icon specifying the total number of unresolved items on the top of page.

    • In the user activities timeline, scroll down to the unresolved artifact entry.

    • Click the icon specifying the number of unresolved worklist items on the project card.

3. In the worklist, expand the entry for the commit that you want to compare.

4. Click the artifact name to view changes made to the artifact.

   The updated artifact displays color-coded updates for addition, deletion, and modification. Refer to the legends displayed on the editor for the significance of each highlighted color.

Deleting Workitems from the Worklist

From your worklist, you can delete workitems that are either in the approved or rejected status.
The **Recent User Activity** timeline shows the workitems that are eligible for deletion. The following figure shows an example:

Figure 11: Deleting Workitems from the Worklist

Before you begin

The workitem that you want to delete should be in the approved or rejected status.

Procedure

1. Scroll down to the workitem that you want to delete in the **Recent User Activity** timeline. If you want to see its status, expand it.

   > **ℹ Note:** You can delete approved workitems also from the project snapshot by clicking the deployable artifacts icon.

   .

2. Click **Delete** and in the confirmation window, confirm your action.

Deploying an Updated Artifact to the Application

After committing edits to an artifact, you can deploy the update to the running application. On the **Dashboard** tab, you can access the worklist containing the deployable artifacts from multiple locations as shown in the following image:

Figure 12: Deploying an Artifact Update to the Application

Item No.	Description
1	This icon displays the total number of deployable items for all projects
2	User activity timeline displays entry for each deployable artifact
3	This icon displays the number of deployable artifacts for that project

Before you begin

Ensure that you have generated deployable files for the project. For details, see Generating the Project Deployable Files

Procedure

1. In WebStudio, click the **Dashboard** tab.

2. On the **Dashboard** tab, open worklist of deployable items in one of the following ways:

   - To open the worklist, click the **total deployable artifact(s)** icon.

   - In the user activities timeline, scroll down to the deployable artifact entry.

   - Click the icon specifying the number of deployable artifacts on the project card.

3. In the worklist, expand the entry for the commit that you want to deploy.

4. Click **Deploy** for the artifact to deploy its changes to the application.

5. On the Deploy Artifact page, select the environment for deployment (for example, QA) and enter the comment.

6. Confirm your action.

WebStudio Management

WebStudio provides options to manage user and roles, and set multiple configuration in UI. You can utilize the **Settings** and **Management** tabs to perform these tasks.

Settings Page Reference

On the Settings page you can change user settings like operator preferences and application settings like operator preferences, notification preferences, and deployment preferences.

User Settings

This section provides options for personal preferences that you can set for WebStudio. For more details about these options, see Personal Preferences Settings.

Applications Settings

Using the Application Settings, you can restrict the operators, and configure project-level email notifications for each project. This section further categorizes the settings in the following preferences:

- **Operator Preferences** - Using the Operator section, you can restrict the operators available for creating the Business Rule using the Builder interface in WebStudio. For more details about these options, see Operator Settings.

- **Notification Preferences** - Using the notification section, you can configure email notifications based on the project action. For more details about these options, see Notification Settings.

- **Deployment Preferences** - Using the Deployment Preferences, you can configure hot deployment settings for the projects without restarting RMS.

> **Note:** These configurations are not applied to the `RMS.cdd` file but are available only for that RMS session.

 Click the **Add** icon to add a new hot deployment preferences for a project. Click the **Remove** icon to delete an existing deployment preference.

Hot Deployment Configuration Properties

Fields	Mapped `RMS.cdd` Property
Name	`<ProjectName>.ws.applicableEnvironments`
Project	Select the project for which hot deployment preferences are configured
Enable	`<ProjectName>.<EnvironmentName>.ws.jmx.hotDeploy.enable`
Host	`<ProjectName>.<EnvironmentName>.ws.jmx.host`
Port	`<ProjectName>.<EnvironmentName>.ws.jmx.port`
User Name	`<ProjectName>.<EnvironmentName>.ws.jmx.user`
Password	`<ProjectName>.<EnvironmentName>.ws.jmx.password`
Cluster Name	`<ProjectName>.<EnvironmentName>.ws.jmx.clusterName`
Agent Name	`<ProjectName>.<EnvironmentName>.ws.jmx.agentName`
InMemory	`<ProjectName>.<EnvironmentName>.ws.jmx.inMemory`

For more information on the hot deployment settings present in the CDD file, see Hot Deployment Property Group.

Personal Preferences Settings

Set preferences for TIBCO BusinessEvents WebStudio using the options available on the **Personal Preferences** tab.

After updating the values, click **Apply Changes** to set new WebStudio preferences.

Field	Default Value	Description
Number of Recently Opened Items	10	The number of recently open items to be displayed in the **Recently Opened** artifact list on the artifact explorer. See Artifact Explorer Panel Reference.
SCS User Name	jdoe	The user name to connect to the SVN repository mentioned in the `ws.scs.rootURL` property of the `RMS.cdd` file. This user name overrides the default user name provided in the `ws.scs.default.username` property of the `RMS.cdd` file. See Source Control System (SCS) Property Group for more details about the `RMS.cdd` properties.
SCS User Password	jdoe	The password for the **SCS User Name** field to connect to the SVN repository mentioned in the `ws.scs.rootURL` property of the `RMS.cdd` file. This password overrides the default password provided in the `ws.scs.default.password` property of the `RMS.cdd` file. See Source Control System (SCS) Property Group for more details about the `RMS.cdd` properties. Unlike the `ws.scs.default.password` property of the `RMS.cdd` file, this field takes the actual password and not the Base64 formatted password.
View Project	List	The structural view of the project artifacts under the Artifact Explorer in workspace.
Decision Table Auto Size Strategy	-	Specifies the autofit approach for the columns of decision tables. Select the required value and click **Apply Changes**. Reload the decision table for the new autofit approach to take effect. • `Auto Size All`- Resizes all columns to autofit the column heading as well as the column values

Field	Default Value	Description
		• `Auto Size Values Only (Skip Headers)` - Resizes all columns to autofit the column values (not headers) • `Size Columns to Fit Screen` - Resizes all columns to autofit all of them in the current screen
Decision Table Page Size	20	The number of rows of the decision table to be displayed in a page. Reopen the decision table, or disable and then again enable the page view, to reflect the updated page size for the decision table. The minimum page size is limited to 10.
Default Rule Template Instance Filter	Match Any	Specifies the default logical operator for new conditions in the condition builder of the business rule.
Select widget for string bindings in Rule Template Instance	TextArea	Choose between TextBox and TextArea for string bindings in rule template instances. To enter multiple-line values, select the TextArea option. The setting is applicable across all rule template instances in all the projects. You can use TIBCO BusinessEvents Studio to set the binding values in rule template instances of a particular project. For the string binding that you want, in the binding element, set an attribute with `widgetType` key and value `TextBox` or `TextArea`.
Auto UnLock on Approve/Reject	selected	If selected, the property enables the artifact to be unlocked once they are either approved or rejected for the committed update
Group Related Artifacts	selected	If selected the business rules are grouped under their parent rule templates, and decision tables are grouped under their parent virtual rule function. If the check box is clear, the decision table and rule templates are not grouped and are listed individually.

Field	Default Value	Description
Allow Custom Domain Values	selected	If selected, the custom values can be entered for the attributes associated to the domain. If the check box is cleared, you can only select from the domain values list using the drop-down.
Show column alias if present	cleared	If selected then column alias are displayed for the decision table columns. Column alias can be defined using the column option **Field Settings**.

Operator Settings

Using the **Operator Preferences**, you can restrict the operators available for creating business rules using the Builder interface in the TIBCO BusinessEvents WebStudio.

The Operator Preferences section consists of three columns.

Preference	Description
Field Type	Lists the type of the fields available in the condition and command builder. Select the field type for which you want to restrict the available operators. The available field types are: - String - Integer - Long - Double - Boolean - DateTime - Concept/Event
Filter Operators	Lists the operators available for the selected **Field Type** in the condition (When) builder.

Preference	Description
	Select the operators that you want to display in the condition builder for the selected Field Type and click **Apply**.
	The operator with a clear checkbox does not appear in the Builder interface while creating a conditional clause for the business rule.
Command Operators	Lists the operators available in the command (Then) builder for the selected **Field Type**.
	Select the operators that you want to display in the command builder for the selected Field Type and click **Apply**.
	The operator with the clear checkbox does not appear in the Builder interface while defining actions for the Business Rule.

Notification Settings

Using the **Notification Preferences**, you can configure email notifications based on the project action.

Projects

All the projects in the rule management server are listed under this column. You can configure email notifications for each project.

Actions

You can select the actions for which you want to send email notification for the selected project. The **Apply** button becomes active if there are any changes in the actions for the project. Click **Apply** to apply all the email notification configurations.

Email ID

You can add, edit, or delete email IDs to which you want to send notifications for the project actions. Click the **Add** or **Delete** icon to add or delete an email ID. The **Apply** button becomes active if there are any changes in the email list for the project. Click **Apply** to apply all the email notification configurations.

Managing Users

A user with an administrator role can add users, delete users, change passwords for users, and edit user roles.

Procedure

1. In the TIBCO BusinessEvents WebStudio, select the **Management** tab.

2. Select **Users** under the **Permission Management** menu.

3. To add users, delete users, change user passwords, and edit user roles, perform the relevant steps:

Purpose	Steps

Creating a user	a. Click the **Create User** icon.
	
	b. In the Add User window, fill in the following fields:
	
	    • Username
	    • Password
	    • Confirm password
	    • Assign Role to User
	
	c. Confirm your actions.

Deleting a user	a. Select the user you want to delete. b. Click the **Delete User** icon.
Changing user password	a. Select the user whose password you want to change. b. Click the **Change Password** icon. c. In the Change Password window, enter the new password and confirm it. d. Confirm your actions.
Editing user roles	a. Select the user whose roles you want to edit. b. In the **Roles** section, based on your requirement, select or clear actions for the selected user.

4. Click **Apply**.

Managing Roles

Users with the administrator role can add, edit, or delete roles for the projects present in the TIBCO BusinessEvents WebStudio. They can also define the actions that a user with a particular role can perform on the different types of artifacts.

Procedure

1. In the TIBCO BusinessEvents WebStudio, select the **Management** tab.

2. Select **Roles** under the **Permission Management** menu.

3. Perform an action based on your requirement:

Purpose	Steps
Adding roles to a project	a. On the **Roles** section, click the **Add role to the project** icon. b. Enter the name for the new role. c. Confirm your action.
Removing roles from a project	a. Click the **Remove role from project** icon.
Defining actions that can be performed by user roles	You can change artifact-specific permissions for roles in specific projects. a. From the **Projects** section, select the project for which you want to edit the role permissions. b. From the **Roles** section, select the role for which you want to edit the permissions. c. In the **Actions** section, based on your requirement, select or clear actions for a user role.

Managing Locks

Users with administrator roles can release artifacts locked by other users. For example, when the user who has locked the artifact is not available and the artifact needs to be updated, the administrator can make the artifact available for an update.

Before you begin
Locking must be enabled by setting the `ws.scs.locking.enable` property to `true` in the CDD file.

Procedure

1. In the TIBCO BusinessEvents WebStudio, select the **Management** tab.

2. From the **Permission Management** pane, select **Locks**.

3. In the **Projects** section, select the project whose artifacts you want to release.

4. From the **List of Locked Artifacts** section, select the artifact you want to release.

5. From the **Artifact Details** section, clear the **Lock** checkbox and click **Apply**.

   Alternatively, under the **List of Locked Artifacts** section. select the **Unlock All** check box and click **Apply** to unlock all artifacts of the selected project.

Managing Reports

Users with administrator roles can generate reports on user activities

Procedure

1. In the TIBCO BusinessEvents WebStudio, select the **Management** tab.

2. Select **Audit Trail** under the **Report Management** menu.

3. Apply filters that you want from the following:

Field	Description
Project	Select the project for which you want to see the actions performed
Artifact	Enter the full name of the artifact along with its path to view audit trail for that artifact. For example, enter `/Virtual_RF/Applicant` to display results for actions performed on the Applicant decision table.
Action Types	Select the type of actions for which you want to see the audit trail
After Date	Select the date and time to display actions that were performed after the selected date and time
Before Date	Select the date and time to display only those actions that were performed before the selected data and time
LogFile	Select the **LogFile** check box to save the results in a log file instead of displaying the results on the screen

4. To save the filters and generate a report, click **Apply**.

Changing Language of the TIBCO BusinessEvents WebStudio

You can change the language of the new user interface in TIBCO BusinessEvents WebStudio from American English (en_US) to other supported languages.

The following languages are currently supported by TIBCO BusinessEvents new user interface:

- Arabic (right to left)
- Chinese
- French

- German

- Italian

- Korean

Procedure

1. Clear the cache for the browser in which you are running TIBCO BusinessEvents WebStudio.

2. Open the browser settings.

3. Choose your preferred language from the list of available languages supported by TIBCO BusinessEvents WebStudio.

4. Refresh the TIBCO BusinessEvents WebStudio URL.

TIBCO Documentation and Support Services

For information about this product, you can read the documentation, contact TIBCO Support, and join TIBCO Community.

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the Product Documentation website, mainly in HTML and PDF formats.

The Product Documentation website is updated frequently and is more current than any other documentation included with the product.

Product-Specific Documentation

The documentation for this product is available on the TIBCO BusinessEvents® Enterprise Edition Documentation page.

To directly access documentation for this product, double-click the file at the following location:

TIBCO_HOME`/release_notes/TIB_businessevents-enterprise_6.3.1_docinfo.html`

where *TIBCO_HOME* is the top-level directory in which TIBCO products are installed. On Windows, the default *TIBCO_HOME* is `C:\tibco`. On UNIX systems, the default *TIBCO_HOME* is `/opt/tibco`.

Other TIBCO Product Documentation

When working with TIBCO BusinessEvents Enterprise Edition, you may find it useful to read the documentation of the following TIBCO products:

- TIBCO ActiveSpaces®: It is used as the cluster, cache, or store provider for the TIBCO BusinessEvents Enterprise Edition project.
- TIBCO FTL®: It is used as the cluster provider for the TIBCO BusinessEvents Enterprise Edition project.

How to Access Related Third-Party Documentation

When working with TIBCO BusinessEvents® Enterprise Edition, you may find it useful to read the documentation of the following third-party products:

- Apache Ignite

- Apache Kafka

- Confluent Kafka Schema Registry

- TIBCO Messaging - Schema Repository for Apache Kafka

- Apache Pulsar

- GridGain

- Apache Cassandra

- Grafana

- InfluxDB

- OpenTelemetry

- Control Plane

- Apache Maven

How to Contact Support for TIBCO Products

You can contact the Support team in the following ways:

- To access the Support Knowledge Base and getting personalized content about products you are interested in, visit our product Support website.

- To create a Support case, you must have a valid maintenance or support contract with a Cloud Software Group entity. You also need a username and password to log in to the product Support website. If you do not have a username, you can request one by clicking **Register** on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to

gain full value from TIBCO products. In addition, users can submit and vote on feature requests from within the TIBCO Ideas Portal. For a free registration, go to TIBCO Community.

Legal and Third-Party Notices