Red Hat JBoss BRMS

Cool Store Demo

(Use case: online shopping cart user interface integration with decision tables, rules, and BPM)

Jason Milliron

JBoss Solution Architect

Eric D. Schabell

JBoss Technology Evangelist (Integration & BPM)

Table of Contents

1Introduction....3
2Setup and Configuration....3
2.1Installation....3
2.2JBoss Developer Studio Configuration....4
2.3Start the JBoss BRMS Platform....6
3Running the Demo....6
3.1Goals of this demo....6
3.2Demo setup from CLI....6
3.3Running demo....7
3.4JBDS IDE unit test (TODO: need to fix arquillian tests)....9
3.5JBoss BRMS Business Central....10
Reset project....10
Build package for deployment....10
3.6Cool Store Demo deployment....10
3.7Cool Store Demo deployment....11

1 Introduction

Interested in fully automating your loan application process but you don't know where to start? Have no fear, with the Red Hat JBoss Business Rules Mangement System (BRMS) you can leverage the power of rules, events, and rule flows in a single integrated suite.

This document will guide you through a coolstore application project using the JBoss BRMS, demonstrating an applicaiton with rules, events, guided rules, a data model, and a rule flow. This is the world that a business analyst lives in.

It will also get you started with an appart project in the JBoss Developer Studio that leverages the coolstore rules, events, and process as a dependency that the web shop application is built upon. This shows how a developer can focus on the application development and just leverage the business analysts work as a dependency.

It will help you demonstrate the various product components and help you to evaluate an application like a online retail shopping cart application.

2 Setup and Configuration

2.1 Installation

You first need to get the project by cloning it from the central location:

git clone git://github.com/eschabell/brms-coolstore-demo.git

Once downloaded, you will have the following folder structure:

- \brms-coolstore-demo
  - \installs – Initially empty, but will contain the BRMS platform downloads.
  - \projects – The web shop project to build the WAR leveraging the coolstore kjar.
  - \support – Additional supporting files used by the demo.
  - \target – Will be created by running init.sh. Contains the fully configured BPM Suite runtime server.
  - \docs – Contains quickstart guide you are reading and architectural overview slides.
  - init.{sh|bat} – Script to install and configure the run time server environment.

Next, download JBoss BPM Suite from the Red Hat Customer Portal (https://access.redhat.com).

Download JBoss BRMS:

1. Under *JBoss Enterprise Platforms*, select the *BRMS* product.
2. Select version *6.0.0.GA* in the *Version* field.

3. Download *Red Hat JBoss BMRS 6.0.0.GA Deployable for EAP 6*
   *(*Please note that this is the deployable distribution, not the standalone one.)

Now copy `boss-brms-6.0.0.GA-deployable-eap6.x.zip`, to the projects *installs* folder. Ensure that this file is executable by running:

```
$ chmod +x <path-to-project>/installs/boss-brms-6.0.0.GA-redhat-2-deployable-eap6.x.zip
```

Download EAP 6 Platform:

4. Under *JBoss Enterprise Platforms*, select the Application *Platform* product.
5. Select version 6*.1.1* in the *Version* field.
6. Download *JBoss Application Platform 6.1.1*.

Now copy `jboss-eap-6.1.1.zip`, to the projects *installs* folder. Ensure that this file is executable by running:

```
$ chmod +x <path-to-project>/installs/jboss-eap-6.1.1.zip
```

Lastly, from the project folder, run the *init.sh* (or *init.bat)* script:

```
$ ./init.sh
```

When the script completes you will have a new folder named *jboss-eap-6.1*, in the projects *target* folder. The folder is a ready to run EAP 6 server with BPM Suite with the following modifications made:

- The *alan, erics* accounts enabled (passwords are all the same, *bpmsuite*) in the *application-roles.properties* and *application-users.properties* file in *brms-coolstore-demo/target/jboss-eap-6.1/standalone/configuration*
- Adjusted system properties*,* copied new *standalone.xml* to *brms-coolstore-demo/target/jboss-eap-6.1/standalone/configuration*
- The coolstore demo repository setup in the product, copied *brms-demo-niogit* to the *brms-coolstore-demo/target/jboss-eap-6.1/bin*
- The coolstore web application has been build into a WAR from the *projects* directory and copied into the *brms-coolstore-demo/target/jboss-eap-6.1/standalone/deployments/brms-coolstore-demo.war*

2.2 JBoss Developer Studio Configuration

In this section, you will configure JBoss Developer Studio. Specifically you will add the JBoss BRMS platform server runtime environment and then import the project that makes up the demo.

Important: It is assumed that you already have JBoss Developer Studio installed. This demo has been tested with JBoss Developer Studio 7.1.

Launch JBoss Developer Studio

1. Either select or switch to a new workspace by pointing to the *brms-coolstore-demo/projects* folder.

2. If the Welcome to JBoss Developer Studio screen appears, dismiss it by click the Workbench arrow in the upper right.

Install Integration tools for BRMS

1. Select tab at bottom of tab *JBoss Central → Software/Update.*
2. At top of list, select *JBoss Business Process and Rules Development & JBoss Integration and SOA Development (all sub-menu items will be selected). Illustration 1* shows the two integration tools items greyed out due to previous installations. You need to select them and proceed through the installation. You will be told to restart JBoss Developer Studio for the changes to take effect, do this now.

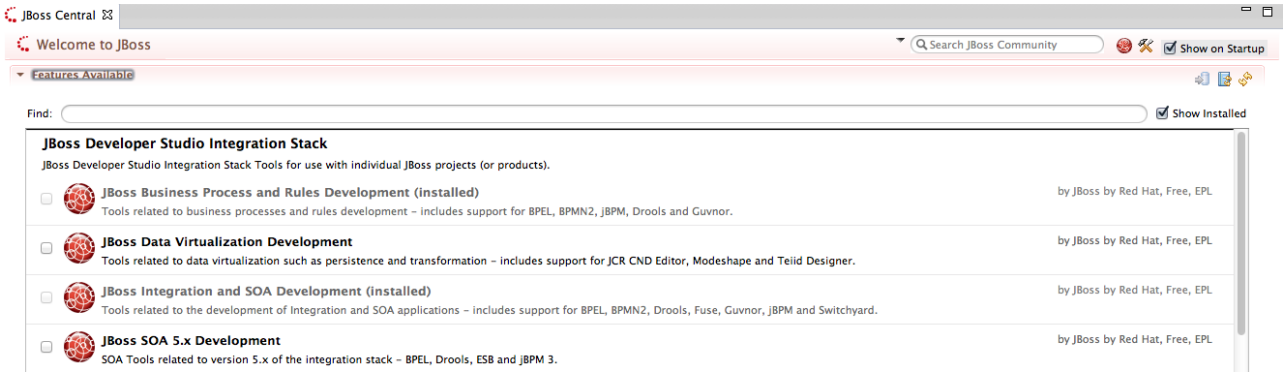


Illustration 1: Select integration stack items for tooling install.

Detect and add a JBoss BRMS runtime environment

1. Select *Preferences* from the *Window* menu.
2. In the left hand side, expand *JBoss Tools* and then select *JBoss Tool Runtime Detection*
3. Select the *Add* button and navigate to the *brms-coolstore-demo/target* folder (**not** brms-coolstore-demo/projects/brms-coolstore-demo/target), then select *OK*.
4. The *jboss-eap-6.1* runtime created earlier should have been found and selected. If so, select *OK*.
5. Select *OK* again to close the Preferences dialog window.

Import Projects

1. Select *Import...* from the *File* menu.
2. Expand the *Maven* folder, and then select *Existing Maven Projects into Workspace*
3. Select the *Browse* button to *Select root directory*, you should be in the *brms-coolstore-demo* folder (if not, then navigate there) and then select *OK*.
4. Make sure the project *brms-coolstore-demo* is selected and then select *Finish*.
5. The project is now imported.
6. You can browse the various elements that make up the Red Hat CoolStore web shopping cart application.

- You can build the WAR for deployment:
  - note the application is leveraging the kjar built from the JBoss BPM Suite in the pom.xml:

```
<!-- KJar file from project. -->
<dependency>
      <groupId>com.redhat</groupId>
      <artifactId>coolstore</artifactId>
      <version>LATEST</version>
</dependency>
```

  - mvn clean install
  - results will be in the *target* directory (brms-coolstore-demo.war)
- the WAR needs to be deployed into the JBoss BRMS Server, either by copying it by hand to the jboss-eap-6.1/standalone/deployments

2.3 Start the JBoss BRMS Platform

In this section, you will start the server from within JBoss Developer Studio.

1. Select the *Servers* view
2. You should see the *JBoss EAP 6.1 Runtime Server* you created earlier.
3. Right click on *JBoss EAP 6.1 Runtime Server* and select *Start* from the pop-up menu. In a few moments your JBoss BRMS Platform will be running.

3 Running the Demo

3.1 Goals of this demo

This demo comes to you from the field, a JBoss Solution Architect Jason Milliron wanted to look at the following:

1. create a JBoss BRMS solution for an online shopping cart.
2. create easy to understand web application where able to apply changes to business logic in JBoss BRMS while viewers see web application updated without restarting application / server.
3. create web application using JavaEE, third party development framework (Vaadin) and CDI technology.

3.2 Demo setup from CLI

The following steps will run through the demo components without using JBDS:

1. setup project as described in Readme by using init.sh.
2. start JBoss EAP
3. Login to business central (*http://localhost:8080/business-central*)
4. build the project in business central (*Authoring → Project Authoring → Tools → Project Editor*

→ *Build & Deploy*)

5. open web application (http://localhost:8080/brms-coolstore-demo), note Checkout button is disabled by design, the demo is about rules changing the discounts of products as you select and put them into your cart.

Illustration 2: Red Hat JBoss BRMS CoolStore

3.3 Running demo

The demo can be manipulated in several ways to show interaction between the GUI and rules based on the Shipping rules:

1. in BRM select the coolstore package
2. open *Business rules assets → Shipping Rules*
3. edit pricing of shipping, last column, to modify how shipping is calculated (for example, increase 4.99 to 5.99, see Illustration 3
4. *Save → add Commit Message*.
5. *Tools → Project Editor → Build & Deploy (no KieScanner)*
    - this pushes a new kjar of the project into your local maven repository.
    - Leverage this by rebuilding the *brms-coolstore-demo.war*
        - *cd brms-coolstore-demo/projects/brms-coolstore-demo*
        - *mvn clean install*
        - *cp target/brms-coolstore-demo.war ../../target/jboss-eap-6.1/standalone/deployments/*
6. *Tools → Project Editor → Build & Deploy (with KieScanner)*
    - Initially you see the KieScanner watching the *coolstore:2.0.0* jar artifact in the server logs.
    - Then when you deploy a new 2.0.1, you see the KieScanner pick up this new version,

but when trying to create a new KieContainer to put these changes into your new KieSession, this version hits an error in red (fix is in next 6.0.1 version):

```
13:29:17,406 INFO  [org.kie.api.builder.KieScanner] (Timer-6) The following artifacts have been
updated: [com.redhat:coolstore:jar:2.0.0]
13:29:20,416 INFO  [org.drools.compiler.kie.builder.impl.KieRepositoryImpl] (http-
localhost/127.0.0.1:8080-4) KieModule was added:MemoryKieModule[ ReleaseId=com.redhat:coolstore:2.0.1]
13:29:21,405 INFO  [org.drools.compiler.kie.builder.impl.KieRepositoryImpl] (Timer-2) KieModule was
added:ZipKieModule[ ReleaseId=com.redhat:coolstore:2.0.1file=/Users/erics/demo-projects/maven-
repository/com/redhat/coolstore/2.0.1/coolstore-2.0.1.jar]
13:29:21,414 INFO  [org.kie.api.builder.KieScanner] (Timer-2) The following artifacts have been
updated: [com.redhat:coolstore:jar:2.0.1]
13:29:21,699 INFO  [org.drools.compiler.kie.builder.impl.KieRepositoryImpl] (Timer-3) KieModule was
added:ZipKieModule[ ReleaseId=com.redhat:coolstore:2.0.1file=/Users/erics/demo-projects/maven-
repository/com/redhat/coolstore/2.0.1/coolstore-2.0.1.jar]
13:29:21,706 ERROR [stderr] (Timer-3) Exception in thread "Timer-3" java.lang.NullPointerException
13:29:21,706 ERROR [stderr] (Timer-3)    at
org.drools.compiler.kie.builder.impl.KieContainerImpl.updateToVersion(KieContainerImpl.java:161)
13:29:21,706 ERROR [stderr] (Timer-3)    at
org.kie.scanner.KieRepositoryScannerImpl.updateKieModule(KieRepositoryScannerImpl.java:214)
13:29:21,706 ERROR [stderr] (Timer-3)    at
org.kie.scanner.KieRepositoryScannerImpl.scanNow(KieRepositoryScannerImpl.java:204)
13:29:21,706 ERROR [stderr] (Timer-3)    at
org.kie.scanner.KieRepositoryScannerImpl$ScanTask.run(KieRepositoryScannerImpl.java:193)
13:29:21,707 ERROR [stderr] (Timer-3)    at java.util.TimerThread.mainLoop(Timer.java:555)
13:29:21,707 ERROR [stderr] (Timer-3)    at java.util.TimerThread.run(Timer.java:505)
13:29:22,811 INFO  [org.drools.compiler.kie.builder.impl.KieRepositoryImpl] (Timer-4) KieModule was
added:ZipKieModule[ ReleaseId=com.redhat:coolstore:2.0.1file=/Users/erics/demo-projects/maven-
repository/com/redhat/coolstore/2.0.1/coolstore-2.0.1.jar]
13:29:22,816 INFO  [org.kie.api.builder.KieScanner] (Timer-4) The following artifacts have been
updated: [com.redhat:coolstore:jar:2.0.1]
13:29:25,024 INFO  [org.drools.compiler.kie.builder.impl.KieRepositoryImpl] (Timer-7) KieModule was
added:ZipKieModule[ ReleaseId=com.redhat:coolstore:2.0.1file=/Users/erics/demo-projects/maven-
repository/com/redhat/coolstore/2.0.1/coolstore-2.0.1.jar]
```

    - This in the future will push the new kjar rules into memory and your next run through the web shopping cart will show the new pricing.

7. Clear shopping cart (GUI) and select again items to see price changes take effect.

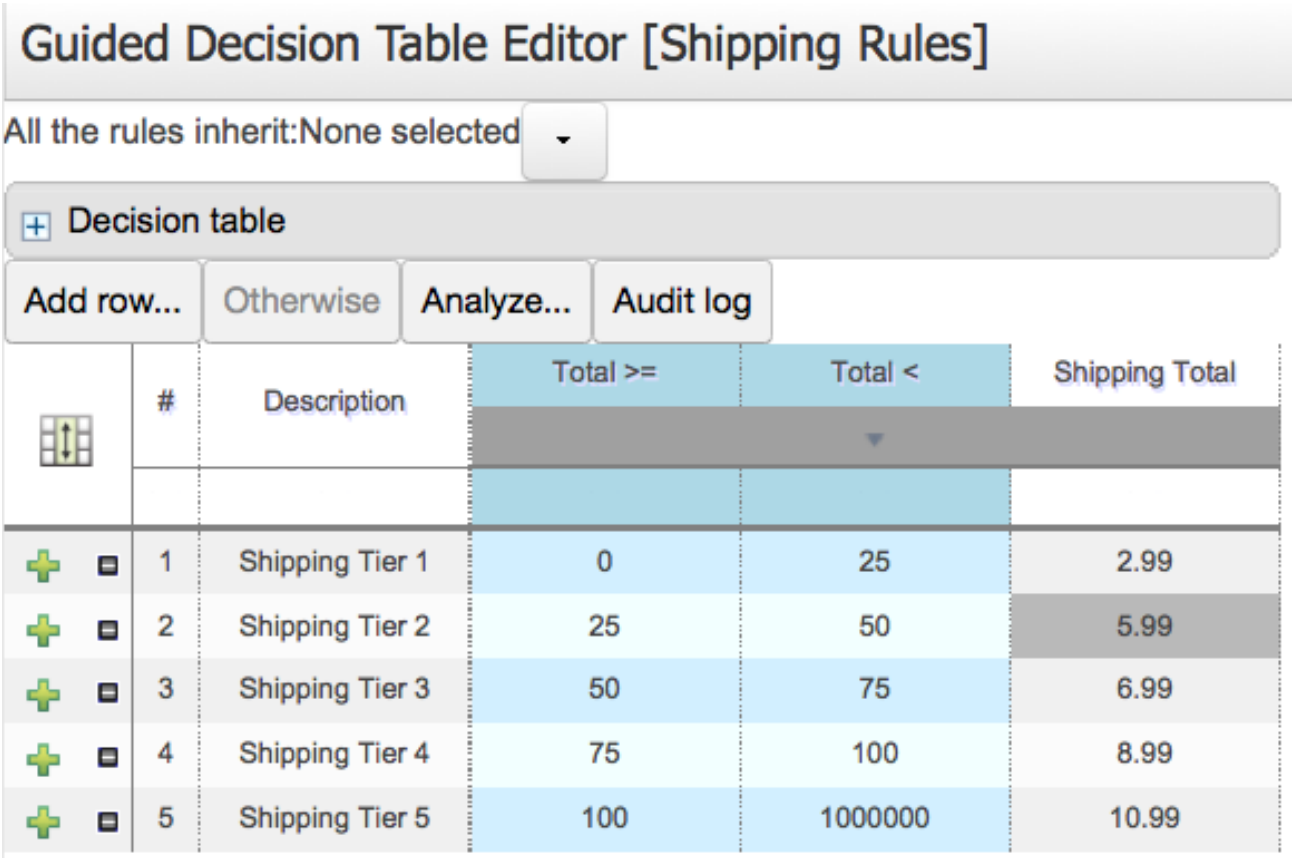


Illustration 3: Adjust shipping pricing in decision table.

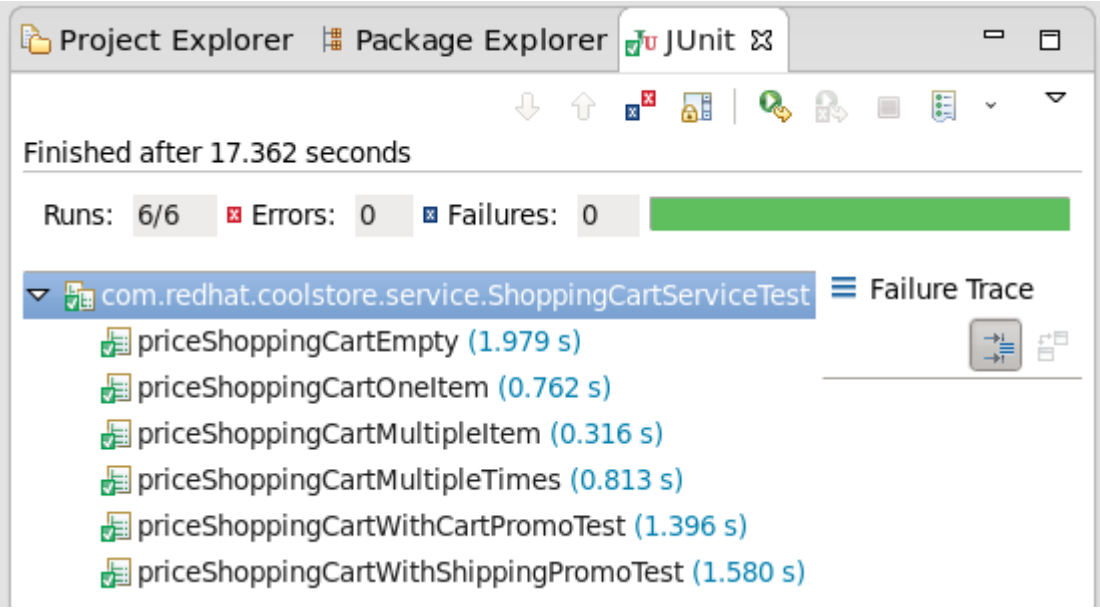
3.4 JBDS IDE unit test (TODO: need to fix arquillian tests)

Unit Testing

The following will run the demo process with two unit tests:

1. Right-click the *brms-coolstore-demo* project and select *Properties*.
2. Select the *Maven* option in the tree on the left.
3. Set Active Maven Profile to *arq-jbossas-remote*, ensure "Resolve dependencies from Workspace projects" is checked. Click *Yes* when asked to update the project configuration. Click OK button to close Properties dialog.
4. Select the file *brms-coolstore-demo/Java Resources/src/test/java/com.redhat.coolstore.service.ShoppingCartServiceTest.java*, right click, *Run As...*, *JUnit Test.*

*Illustration 4: **ShoppingCartService**Test run*



3.5 JBoss BRMS Business Central

The poject was already imported so now we can look closer at web designers viewing, rules asset viewing, package building and deploying, and business central viewing.

Reset project

The business central dashboard is available on http://localhost:8080/business-central once the JBoss BRMS server has been fully started.

1. To start fresh you can at any time:
    - stop the JBoss BRMS server
    - *rm -rf brms-coolstore-demo/target/jboss-eap-6.1/bin/.niogit*
    - *cp -rf brms-coolstore-demo/support/brms-demo-niogit brms-coolstore-demo/target/jboss-eap-6.1/bin/.niogit*
    - start the JBoss BRMS server
        - *./target/jboss-eap-6.1/bin/standalone.sh*

Build package for deployment

To build your project (kjar) and deploy:

1. *Authoring → Projet Authoring → Tools → Project Editor → Build & Deploy* button.
2. Watch for successful build popup messages (should be green if all goes well).

3.6 Cool Store Demo deployment

In this section, you will deploy the application to the server from within JBoss Developer Studio.

1. Select the *Servers* view
   If it is currently not open, select *Show View* --> *Other*... from the *Window* menu and search for the *Servers* view.
2. If the server is not running, right click on *JBoss EAP 6.1 Runtime Server* and select Start from the pop-up menu. In a few moments your JBoss Enterprise BRMS Platform will be running.
3. Right-click on *JBoss EAP 6.1 Runtime Server* and select *Add and Remove*... from the pop-up menu.
4. Select *brms-coolstore-demo* from the list of Available resources and click the *Add* > button to move it to the Configured resources section.
5. Click *Finish*.

3.7 Cool Store Demo deployment

1. After building and deploying the CoolStore web shopping cart application.
2. Open up your Web browser of choice and navigate to http://localhost:8080/brms-coolstore-demo, should see Illustration 2.