



Red Hat Process Automation Manager 7.1

Getting started with business processes

Red Hat Process Automation Manager 7.1 Getting started with business processes

Red Hat Customer Content Services
brms-docs@redhat.com

Legal Notice

Copyright © 2018 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Abstract

In this tutorial, you will create and test a mortgage application scenario using Red Hat Process Automation Manager 7.1. The procedures in this document are based on the included Mortgage Process sample project.

Table of Contents

| | |
|--|-----------|
| PREFACE | 3 |
| CHAPTER 1. OVERVIEW | 4 |
| CHAPTER 2. MIGRATING BUSINESS PROCESSES FROM NEW TO LEGACY PROCESS DESIGNER | 5 |
| CHAPTER 3. OPENING THE MORTGAGE PROCESS SAMPLE PROJECT | 6 |
| CHAPTER 4. DATA OBJECTS | 7 |
| 4.1. VIEWING THE MORTGAGE PROCESS DATA OBJECTS | 7 |
| 4.1.1. Viewing the Applicant data object | 7 |
| 4.1.2. Viewing the Application data object | 8 |
| 4.1.3. Viewing the Property data object | 8 |
| 4.1.4. Viewing the ValidationErrorDO data object | 9 |
| CHAPTER 5. CREATING THE MORTGAGEAPPROVALPROCESS BUSINESS PROCESS | 10 |
| 5.1. DELETE THE EXISTING BUSINESS PROCESS | 10 |
| 5.1.1. Validating the mortgage | 10 |
| 5.1.2. Creating outgoing connections | 11 |
| 5.1.3. Defining the validation data | 13 |
| 5.1.4. Calculating the mortgage | 16 |
| 5.1.5. Increasing the down payment | 20 |
| CHAPTER 6. GUIDED RULES | 27 |
| 6.1. DEFINING BUSINESS RULES | 27 |
| 6.1.1. Creating the Validate Down Payment guided rule | 27 |
| 6.1.2. Defining the Validate Down Payment guided rule conditions | 27 |
| 6.1.3. Defining the error conditions | 28 |
| 6.1.3.1. Defining the ValidationErrorDO | 29 |
| 6.1.3.2. Defining the retractValidationErr | 29 |
| CHAPTER 7. GUIDED DECISION TABLES | 31 |
| 7.1. CREATING DECISION TABLES | 31 |
| 7.1.1. Viewing the Mortgage Decision Table | 31 |
| CHAPTER 8. GENERATING AND EDITING FORMS | 32 |
| 8.1. AUTOMATIC FORM GENERATION | 32 |
| 8.2. EDITING FORMS (OPTIONAL) | 34 |
| CHAPTER 9. DEPLOYING MORTGAGE PROCESS APPLICATION | 36 |
| CHAPTER 10. EXECUTING THE MORTGAGE PROCESS | 37 |
| CHAPTER 11. MONITORING THE MORTGAGEAPPROVALPROCESS PROCESS APPLICATION | 39 |
| 11.1. FILTERING PROCESS INSTANCES USING DEFAULT OR ADVANCED FILTERS | 39 |
| 11.1.1. Filtering process instances using default filters | 39 |
| 11.1.2. Filtering process instances using advanced filters | 40 |
| APPENDIX A. VERSIONING INFORMATION | 41 |

PREFACE

As a citizen developer who wants to build rule-based solutions quickly, you can use Business Central in Red Hat Process Automation Manager to design a variety of business processes. A business process is a graph that describes the order in which a series of steps must be executed using a flow chart. A process consists of a collection of predefined node types that are linked to each other using connections. Each of the nodes represents one step in the overall process while the connections specify how to transition from one node to the other.

Prerequisites

- Red Hat JBoss Enterprise Application Platform 7.1.0 is installed. For installation information, see the [Red Hat JBoss EAP 7.1.0 Installation Guide](#).
- Red Hat Process Automation Manager is installed and configured with Process Server. For more information, see [Installing and configuring Red Hat Process Automation Manager on Red Hat JBoss EAP 7.1](#).
- Red Hat Process Automation Manager is running and you can log in to Business Central with the **developer** role. For more information, see [Planning a Red Hat Process Automation Manager installation](#).

CHAPTER 1. OVERVIEW

Red Hat Process Automation Manager enables you to automate your business processes and decisions. For example, a bank offers a housing mortgage loan service. When a customer wants to buy a new property using credit, they contact a broker at the bank who assists in filing for a mortgage loan. The broker collects information about the property and the customer, such as the salary of the customer, social security number, the property sale price, and the requested loan amount. The broker then submits a request on behalf of the customer.

Using Red Hat Process Automation Manager, the housing mortgage department of the bank creates a complete business process for the mortgage loan. Whenever a customer submits a request, a new process instance is created. This ensures consistency in the quality of evaluating each request, provides complete visibility into the status of each request, and makes the process efficient and effective.

CHAPTER 2. MIGRATING BUSINESS PROCESSES FROM NEW TO LEGACY PROCESS DESIGNER

You can migrate a business process from the new process designer in Business Central to the legacy process designer. By default, all business processes open in the new process designer. But once you migrate a business process, it will always open in the legacy designer.



NOTE

Migrating business processes is an irreversible process.

Procedure

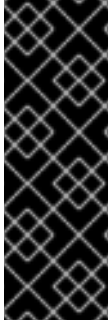
1. Log in to Business Central and go to **Menu** → **Projects**.
2. From **Assets** tab, select the business process. It will open in the new process designer.
3. Click **Migrate** and a warning will appear.
4. Click **Migrate Diagram**.
The process designer will restart and the business process will re-open in the legacy process designer.

Migrating a business process will result in two changes:

- The name of the business process in the process designer will change, for example,
 - Name in new process designer: business-process-name.bpmn - Business Processes
 - Name in legacy process designer: business-process-name.bpmn2 - Business Processes (legacy)
- The extension of the business process will change from *.bpmn to *.bpmn2.

CHAPTER 3. OPENING THE MORTGAGE PROCESS SAMPLE PROJECT

The **Mortgage Process** sample project consists of predefined data objects, guided decision tables, guided rules, forms, and a business process. Using the sample project provides a quick way to get acclimated with Red Hat Process Automation Manager. In a real business scenario, you would create all of the assets by providing data that is specific to your business requirements.




IMPORTANT

The business process application example includes features that are Technology Preview only. Technology Preview features are not supported with Red Hat production service level agreements (SLAs), might not be functionally complete, and are not recommended for production. These features provide early access to upcoming product features, enabling customers to test functionality and provide feedback during the development process. For more information about Red Hat Technology Preview support, see [Technology Preview Features Support Scope](#).

Procedure

Navigate to the Mortgage Process sample project to view the predefined assets.

1. Log in to Business Central and click **Menu** → **Design** → **Projects**.
2. Click  in the upper-right corner of the screen and select **Try Samples**.
3. Select **Mortgage Process** and click **Ok**.

The **Assets** view of the project opens.

CHAPTER 4. DATA OBJECTS

Data objects are the building blocks for the rule assets that you create. Data objects are custom data types implemented as Java objects in specified packages of your project. For example, you might create a **Person** object with data fields **Name**, **Address**, and **DateOfBirth** to specify personal details for loan application rules. These custom data types determine what data your assets and your decision service are based on.

For more information about creating data objects, see ["Creating data objects"](#) in *Designing a decision service using guided decision tables*.

4.1. VIEWING THE MORTGAGE PROCESS DATA OBJECTS

This tutorial utilizes the predefined data objects in the **Mortgage Process** sample project.

The **Mortgage Process** data model is composed of four data objects:

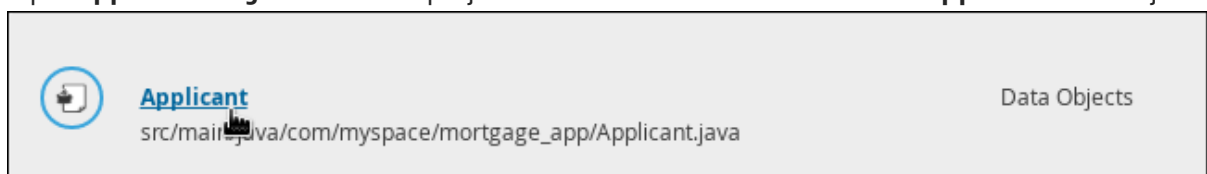
- **Applicant**
- **Application**
- **Property**
- **ValidationErrorDo**

4.1.1. Viewing the Applicant data object

Follow these steps to familiarize yourself with the predefined **Applicant** data object.

Procedure

1. Click **Menu** → **Design** → **Projects**, then click **Mortgage Process**.
2. Input **Applicant.java** in to the project's asset search box and click the **Applicant** data object.



3. Review the **Applicant** data object fields.

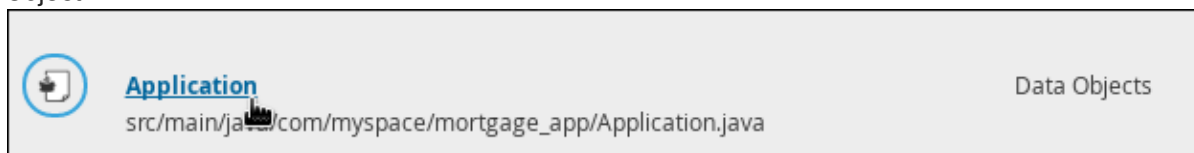
| Applicant.java - Data Objects ▾ | | | | |
|---------------------------------|---------------|--|---------|-------------|
| Model Overview Source | | | | |
| Applicant (Applicant) | | | | + add field |
| Identifier | Label | | Type | |
| address | Address | | String | Delete |
| annualIncome | Annual Income | | Integer | Delete |
| creditrating | Credit Rating | | Integer | Delete |
| name | Name | | String | Delete |
| ssn | SSN | | Integer | Delete |

4.1.2. Viewing the Application data object

Follow these steps to familiarize yourself with the predefined **Application** data object.

Procedure

1. Click **Menu** → **Design** → **Projects**, then click **Mortgage Process**.
2. Input **Application.java** in to the project's asset search box and click the **Application** data object.



3. Review the **Application** data object fields.

Application.java - Data Objects ▾

Model Overview Source

Application + add field

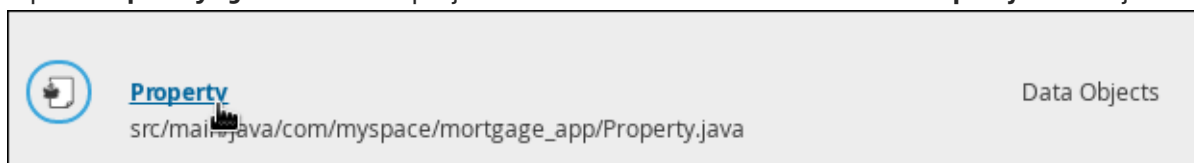
| Identifier | Label | Type | |
|----------------|-----------------------|------------------|--------|
| amortization | Years of amortization | Integer | Delete |
| applicant | Applicant | Applicant | Delete |
| downpayment | Down Payment | Integer | Delete |
| errors | Error details | Validation Error | Delete |
| mortgageamount | Mortgage amount | Integer | Delete |
| property | Property | Property | Delete |

4.1.3. Viewing the Property data object

Follow these steps to familiarize yourself with the predefined **Property** data object.

Procedure

1. Click **Menu** → **Design** → **Projects**, then click **Mortgage Process**.
2. Input **Property.java** in to the project's asset search box and click the **Property** data object.



3. Review the **Property** data object fields.


| Property.java - Data Objects ▾ | | | |
|--------------------------------|--------------------------|------------------------|-----------------------------|
| Model | Overview | Source | |
| Property (Property) | | | + add field |
| Identifier | Label | Type | |
| address | Address of property | String | Delete |
| age | Age of property | Integer | Delete |
| locale | Locale | String | Delete |
| saleprice | Sale Price | Integer | Delete |

4.1.4. Viewing the ValidationErrorDO data object

Follow these steps to familiarize yourself with the predefined **ValidationErrorDO** data object.

Procedure

1. Click **Menu** → **Design** → **Projects**, then click **Mortgage Process**.
2. Input **ValidationErrorDO.java** in to the project's asset search box and click the **ValidationErrorDO** data object.

| | | |
|--|---|--------------|
|  | ValidationErrorDO src/main/java/com/myspace/mortgage_app/ValidationErrorDO.java | Data Objects |
|--|---|--------------|

3. Review the **Property** data object fields.

| ValidationErrorDO.java - Data Objects ▾ | | | |
|---|--------------------------|------------------------|-----------------------------|
| Model | Overview | Source | |
| Validation Error (ValidationErrorDO) | | | + add field |
| Identifier | Label | Type | |
| error | Error and cause | String | Delete |

CHAPTER 5. CREATING THE MORTGAGEAPPROVALPROCESS BUSINESS PROCESS

A business process is a graph that describes the order in which a series of steps must be executed using a flow chart. A business process consists of a collection of nodes that are linked to each other using connections. Each of the nodes represents one step in the overall process while the connections specify how to transition from one node to the other.

The **Mortgage Process** sample project contains a predefined **MortgageApprovalProcess** business process. For this tutorial, you will delete the existing **MortgageApprovalProcess** business process and recreate it to gain a better understanding of creating a business process.

5.1. DELETE THE EXISTING BUSINESS PROCESS

Procedure

You must delete the sample business process before you can create your own.



Procedure

1. Log in to Business Central and click **Menu** → **Design** → **Projects**, then click **Mortgage Process**.
2. Input **MortgageApprovalProcess.bpmn** in to the project's asset search box and click on the **MortgageApprovalProcess** business process.
3. Select **Delete** from the toolbar, then click **Delete** to confirm that you want to delete this asset.

5.1.1. Validating the mortgage

The mortgage validation business process determines whether or not the new application contains the required data before proceeding. If all of the specified data requirements are met, the application will move on to the mortgage calculation business process.

Procedure

1. Log in to Business Central and click **Menu** → **Design** → **Projects**, then click **Mortgage Process**.
2. Click **Add Asset** → **Business Process**.
3. Enter the following values:
 - **Business Process:** **MortgageApprovalProcess**
 - **Package:** Select **com.myspace.mortgage_app**
4. Click **Ok**. The diagram editor opens.
5. In the upper-right corner, click the **Diagram properties**  icon.
6. Scroll down and expand **Process Data** and click  in the **Process Variables** section.

7. Enter the following values:

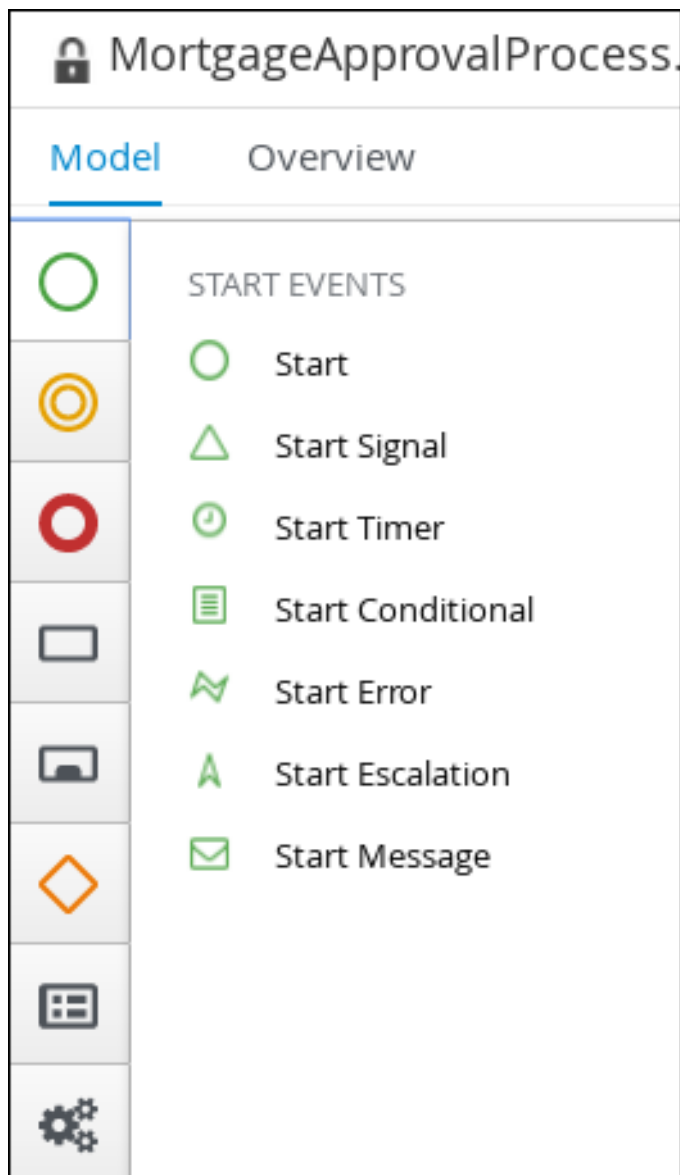
- **Name:** `application`.
- **Data Type:** `Application [com.myspace.mortgage_app]`

8. Click **Save**, then **Save**, to confirm your changes.

5.1.2. Creating outgoing connections

This section describes how to create an outgoing connection from the start event to an exclusive gateway and then from the exclusive gateway to the business rule task. Exclusive gateways are used to make decisions and react to events based on the available data. A business rule task represents a set of rules that must be evaluated.

Red Hat Process Automation Manager contains a predefined selection of node types to simplify business process creation. The predefined node panel is located on the left side of the diagram editor. Click on the various node icons to reveal the available options as shown here.

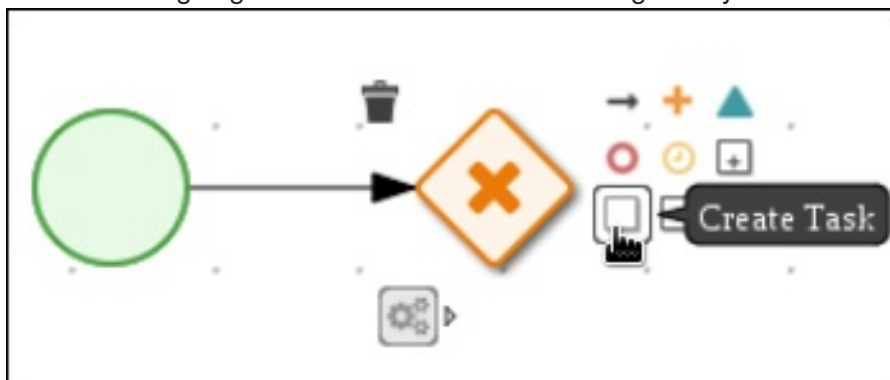


Procedure

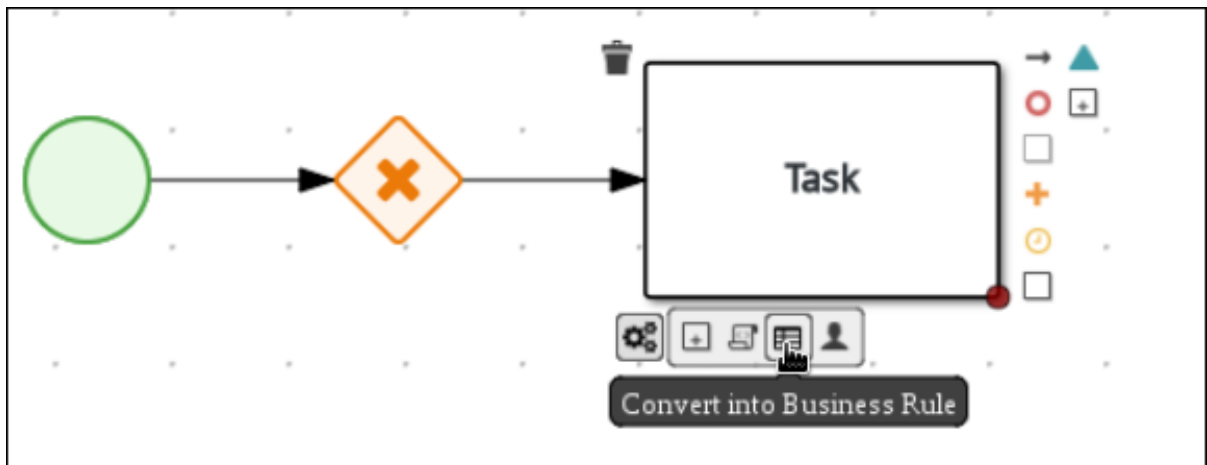
1. Click on the start event and create an outgoing connection from the start event to an exclusive gateway.



2. Create an outgoing connection from the exclusive gateway to a task.



3. Convert the new task to a business rule task.

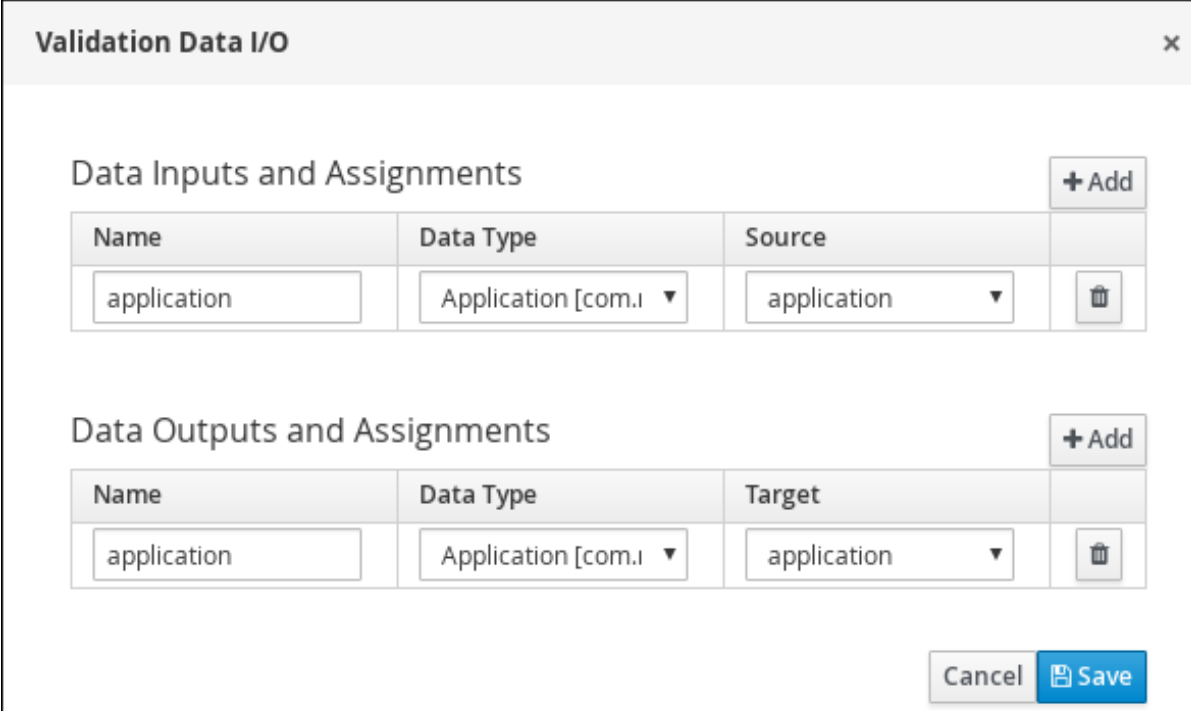


4. Click on the business rules task and enter **Validation** in the **Name** field of the **Diagram properties** panel.
5. Expand **Implementation/Execution**, select **validation** in the **Rule Flow Group** field.
6. Define the following Java expression in the **On Exit Action** field :

```
System.out.println(application.getProperty());
```

7. Scroll down and expand the **Data Assignments** section and click  next to **Assignments**.

8. In the **Validation Data I/O** window, click **Add** to create the following assignments:



Validation Data I/O [X]

Data Inputs and Assignments [Add]

| Name | Data Type | Source | |
|-------------|-----------------------|---------------|-----|
| application | Application [com.i ▼] | application ▼ | [X] |

Data Outputs and Assignments [Add]

| Name | Data Type | Target | |
|-------------|-----------------------|---------------|-----|
| application | Application [com.i ▼] | application ▼ | [X] |

[Cancel] [Save]

9. Click **Save**.

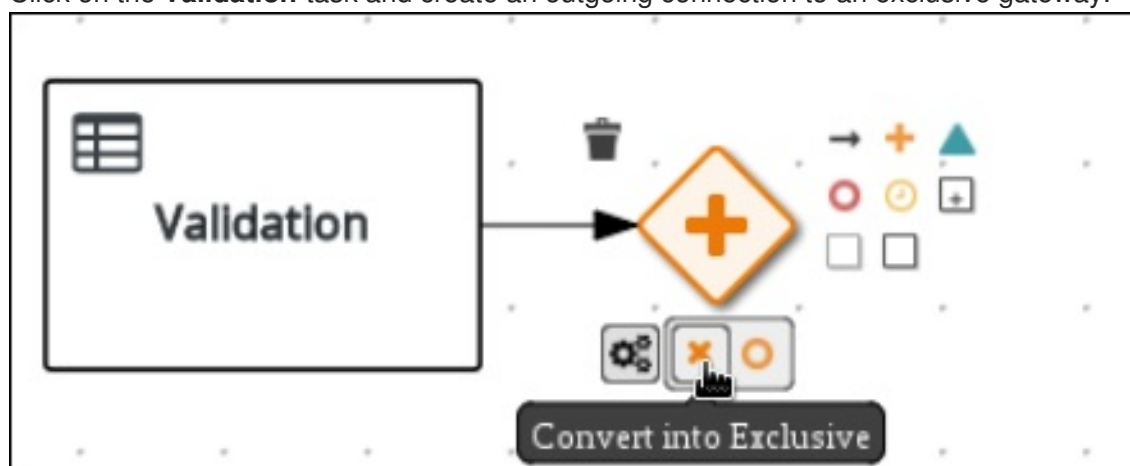
10. In the diagram editor, click **Save**, then **Save**, to confirm your changes.

5.1.3. Defining the validation data

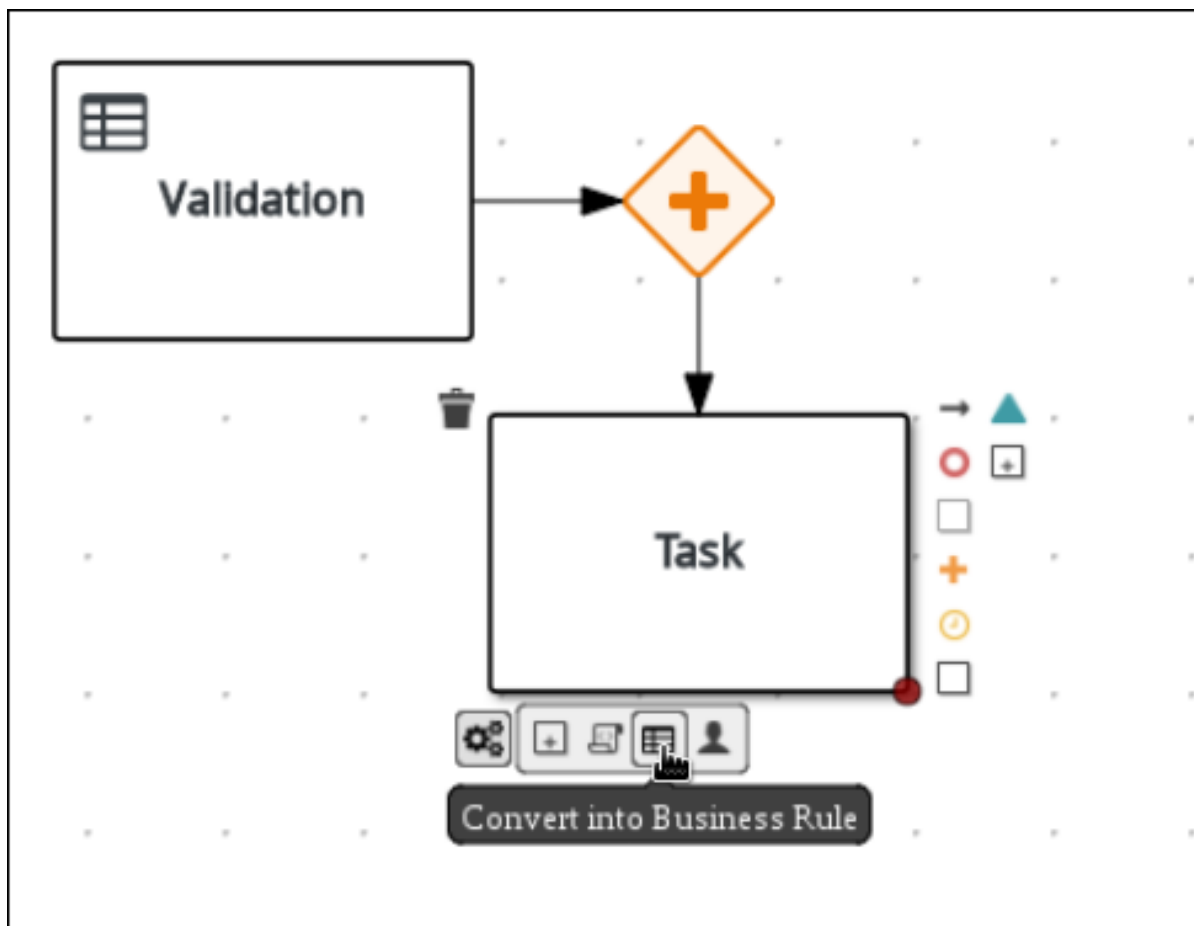
This section describes how to define the data that will determine whether the application data is correct or if there is an error or any missing information.

Procedure

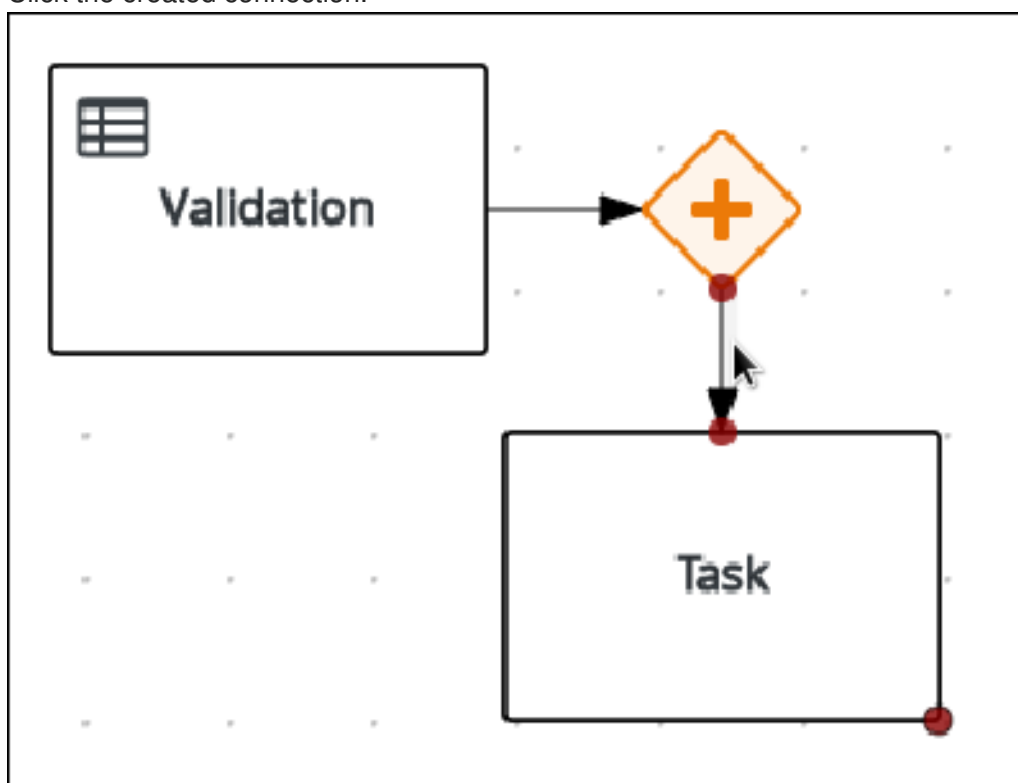
1. Click on the **Validation** task and create an outgoing connection to an exclusive gateway.



2. Create an outgoing connection from the exclusive gateway to a new business rule task.



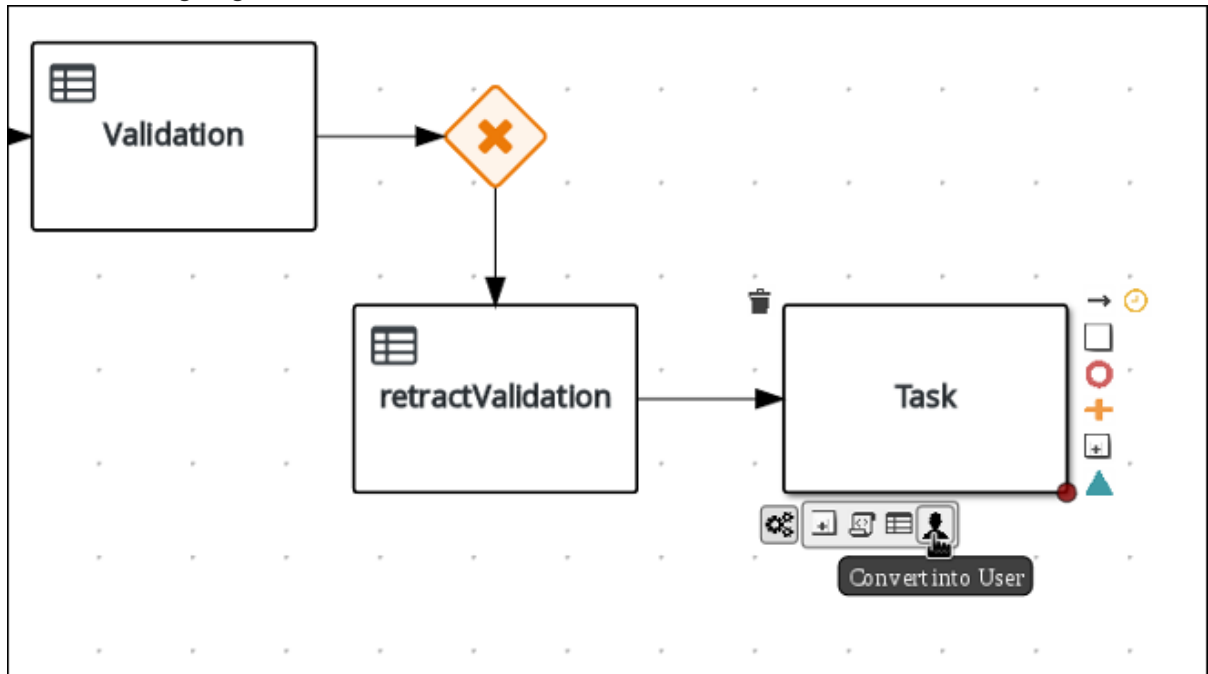
- Click the created connection.




- In the **Diagram Properties** panel, input **Invalid** in the **Name** field.
- Expand **Implementation/Execution**, and enter the following Drools condition expression:
 - Condition Expression: `ValidationErrorD0()`

- **Condition Expression Language: drools**

- Click on the business rule task and enter **Retract Validation** in the **Name** field of the **Diagram properties** panel.
- Expand **Implementation/Execution** and enter **error** in the **Rule Flow Group** field.
- Create an outgoing connection from the **Retract Validation** task to a new user task.



- Click the user task and in the **Diagram Properties** panel, input **Correct Data** in the **Name** field.
- Expand **Implementation/Execution** and enter:
 - **Task Name: CorrectData**
 - **Groups: broker**
- Click  next to **Assignments**. In the **Correct Data Data I/O** window, click **Add** to create the following assignments:
- In the **Correct Data Data I/O** window, click **Add** to create the following assignments:

Correct Data Data I/O

Data Inputs and Assignments

+ Add

| Name | Data Type | Source | |
|-------------|----------------------|---------------|--|
| application | Application [com.i ▼ | application ▼ | |

Data Outputs and Assignments

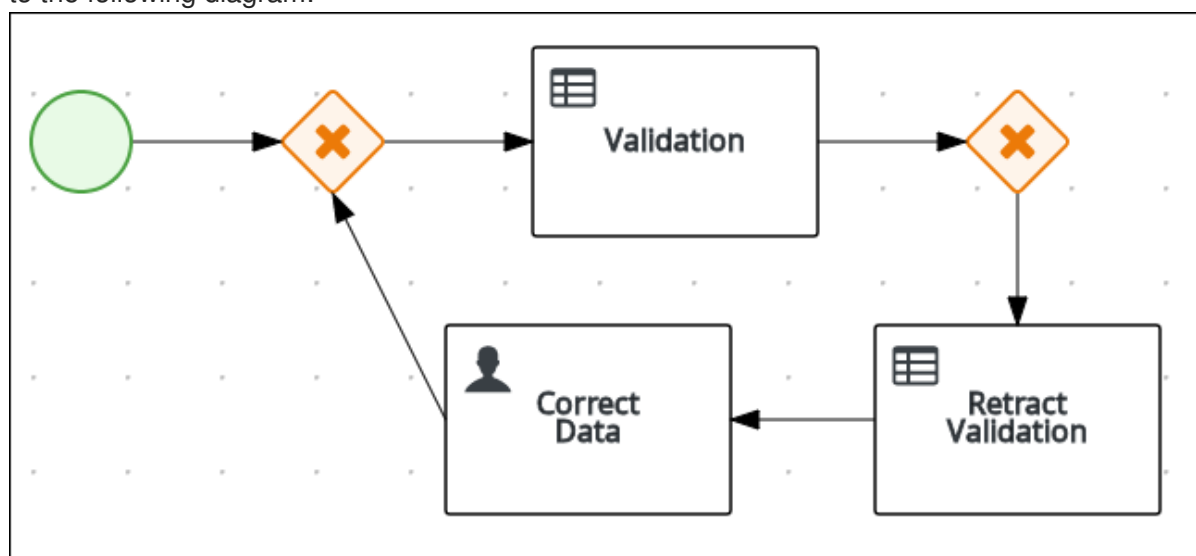
+ Add

| Name | Data Type | Target | |
|-------------|----------------------|---------------|--|
| application | Application [com.i ▼ | application ▼ | |

Cancel

Save

13. Click **Save**.
14. In the diagram editor, click **Save**, then **Save**, to confirm your changes.
15. Connect the **Correct Data** back to the first exclusive gateway. Your workflow should look similar to the following diagram:

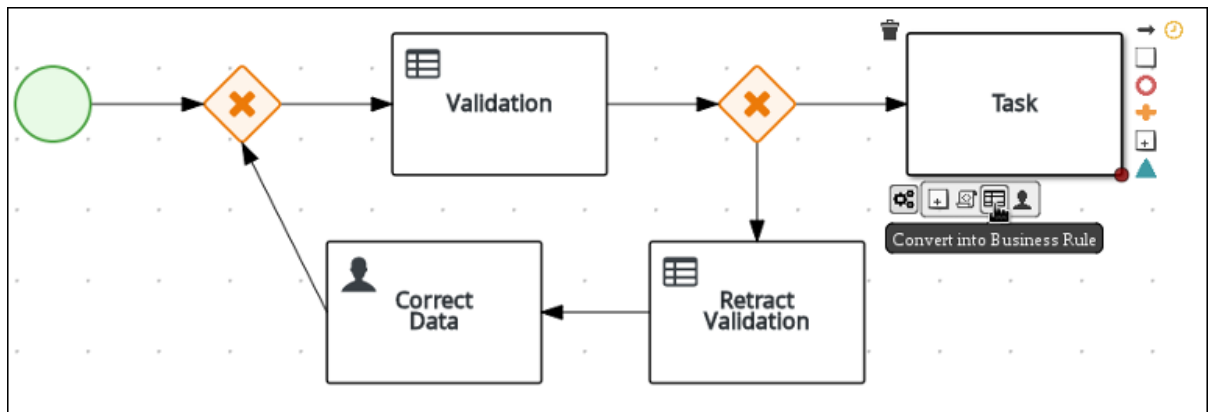



5.1.4. Calculating the mortgage

The mortgage calculation business process determines the applicant's mortgage borrowing limit.

Procedure

1. Return to the second exclusive gateway. Create an outgoing connection to a business rule task.



2. Click the created connection and in the **Diagram Properties** panel, input **Valid** in the **Name** field.
3. Expand **Implementation/Execution**, select and enter:
 - **Condition Expression:** `not ValidationErrorD0()`
 - **Condition Expression Language:** `drools`
4. Click the created business rule task and in the **Diagram Properties** panel, input **Mortgage Calculation** in the **Name** field.
5. Expand **Implementation/Execution**, select and enter `mortgagecalculation` in the **Rule Flow Group** field.
6. Click  next to **Assignments**. In the **Mortgage Calculation Data I/O** window, click **Add** to create the following assignments:

Mortgage Calculation Data I/O ✕

Data Inputs and Assignments + Add


| Name | Data Type | Source | |
|-------------|-----------------------|---------------|---|
| application | Application [com.i ▼] | application ▼ | ✕ |

Data Outputs and Assignments + Add

| Name | Data Type | Target | |
|-------------|-----------------------|---------------|---|
| application | Application [com.i ▼] | application ▼ | ✕ |



Cancel
Save

7. Click **Save**.

8. Click on an empty space on the canvas, scroll down, expand **Process Data**, and click  next to **Process Variables**. Enter the following values:

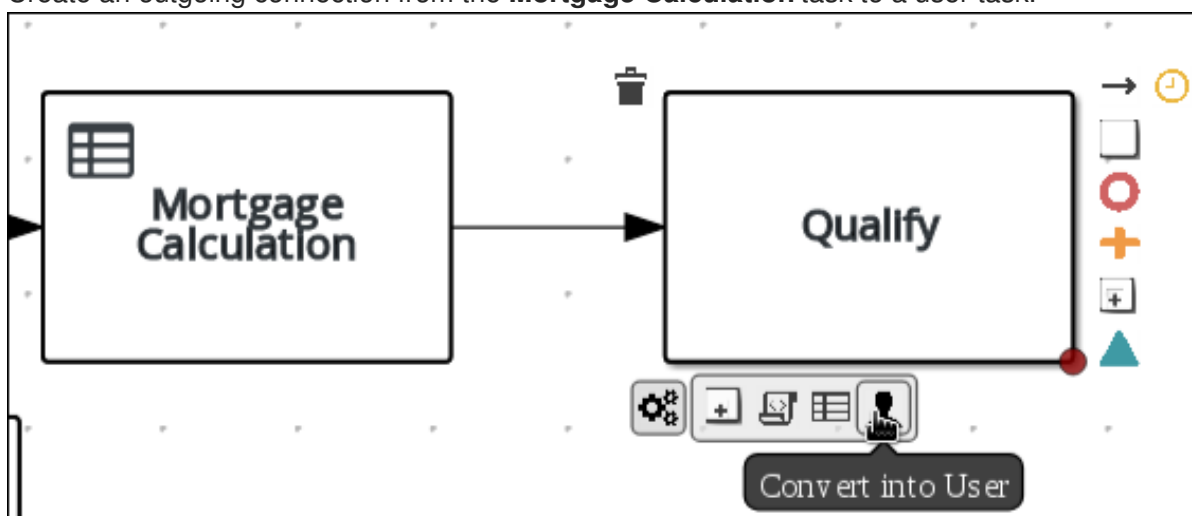
Process Data

Process Variables

| Name | Data Type | |
|-------------|---------------|---|
| Application | Application ▼ |  |
| inlimit | Boolean ▼ |  |


- **Name:** `inlimit`
- **Date Type:** `Boolean`

9. Create an outgoing connection from the **Mortgage Calculation** task to a user task.



10. Click on the user task and enter **Qualify** in the **Name** field.

11. Expand **Implementation/Execution** and enter:

- **Task Name:** `Qualify`
- **Groups:** `approver`
- Click  next to **Assignments**. In the **Qualify Data I/O** window, click **Add** to create the following assignments:

Qualify Data I/O

Data Inputs and Assignments

| Name | Data Type | Source | |
|-------------|----------------------|---------------|---|
| application | Application [com.i ▼ | application ▼ | ✕ |

+ Add

Data Outputs and Assignments

| Name | Data Type | Target | |
|---------|-----------|-----------|---|
| inlimit | Boolean ▼ | inlimit ▼ | ✕ |

+ Add

Cancel

Save

12. Click **Save**. Above the canvas, click **Save**, then **Save**, to confirm your changes.
13. Create an outgoing connection from the **Qualify** task to an exclusive gateway.
 - a. Click on the **GATEWAYS** icon in the node panel.
 - b. Click on **Exclusive** and drag it to the right of the **Qualify** task.
14. Create an outgoing connection from the exclusive gateway and connect it to a user task.
15. Click the connection, name it **in Limit** and define the following Java Condition expression:

```
return inlimit;
```

Implementation/Execution

Priority

Condition Expression

return inlimit;

java



16. Click the user task and define:

- **Name:** Final Approval
- **Task Name:** finalapproval
- **Groups:** manager

17. Click  next to **Assignments**. In the **Final Approval Data I/O** window, click **Add** to create the following assignments:

Final Approval Data I/O
×

Data Inputs and Assignments
+ Add

| Name | Data Type | Source | |
|-------------|----------------------|---------------|---|
| application | Application [com.i ▼ | application ▼ |  |
| inlimit | Boolean ▼ | inlimit ▼ |  |

Data Outputs and Assignments
+ Add

Cancel
Save

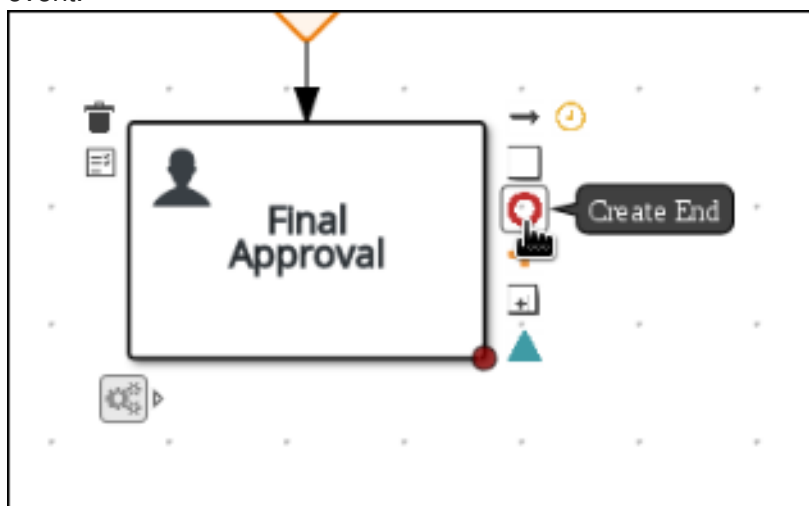
18. Click **Save**. Above the canvas, click **Save**, then **Save**, to confirm your changes.

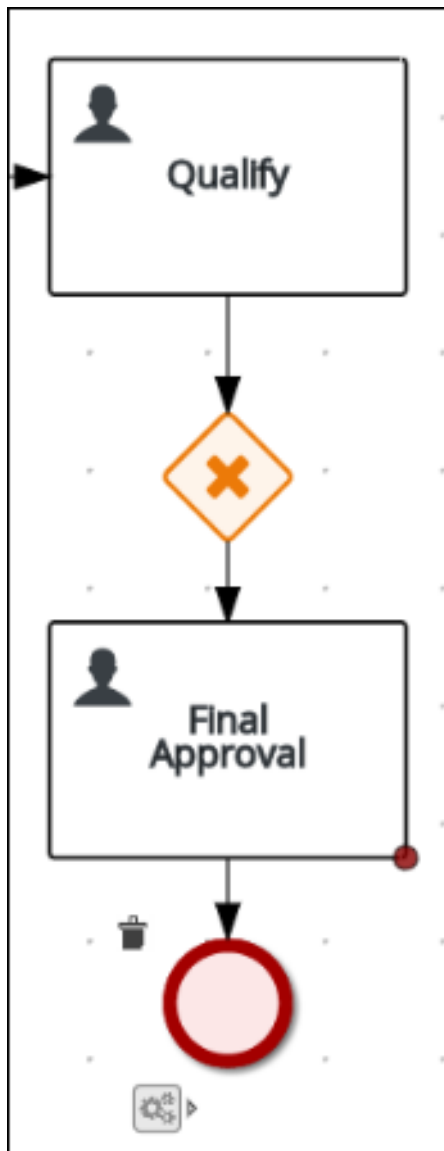
5.1.5. Increasing the down payment

The increasing the down payment business process checks to see if the applicant qualifies for the loan by increasing their down payment. The final result will be either the final loan approval, or loan denial based on the applicant's inability to increase the down payment.

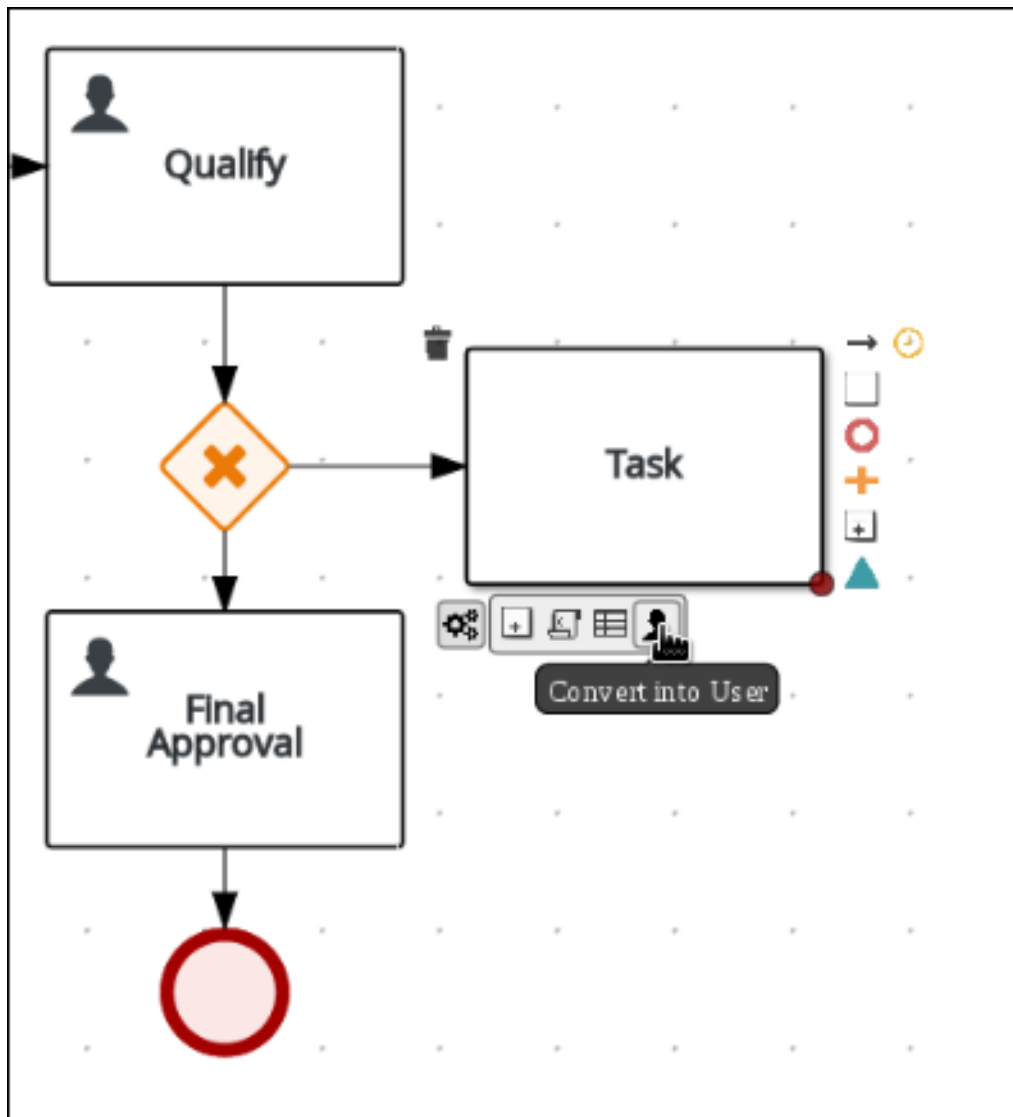
Procedure

1. Create an outgoing connection from the **Final Approval** user task and connect it to an end event.



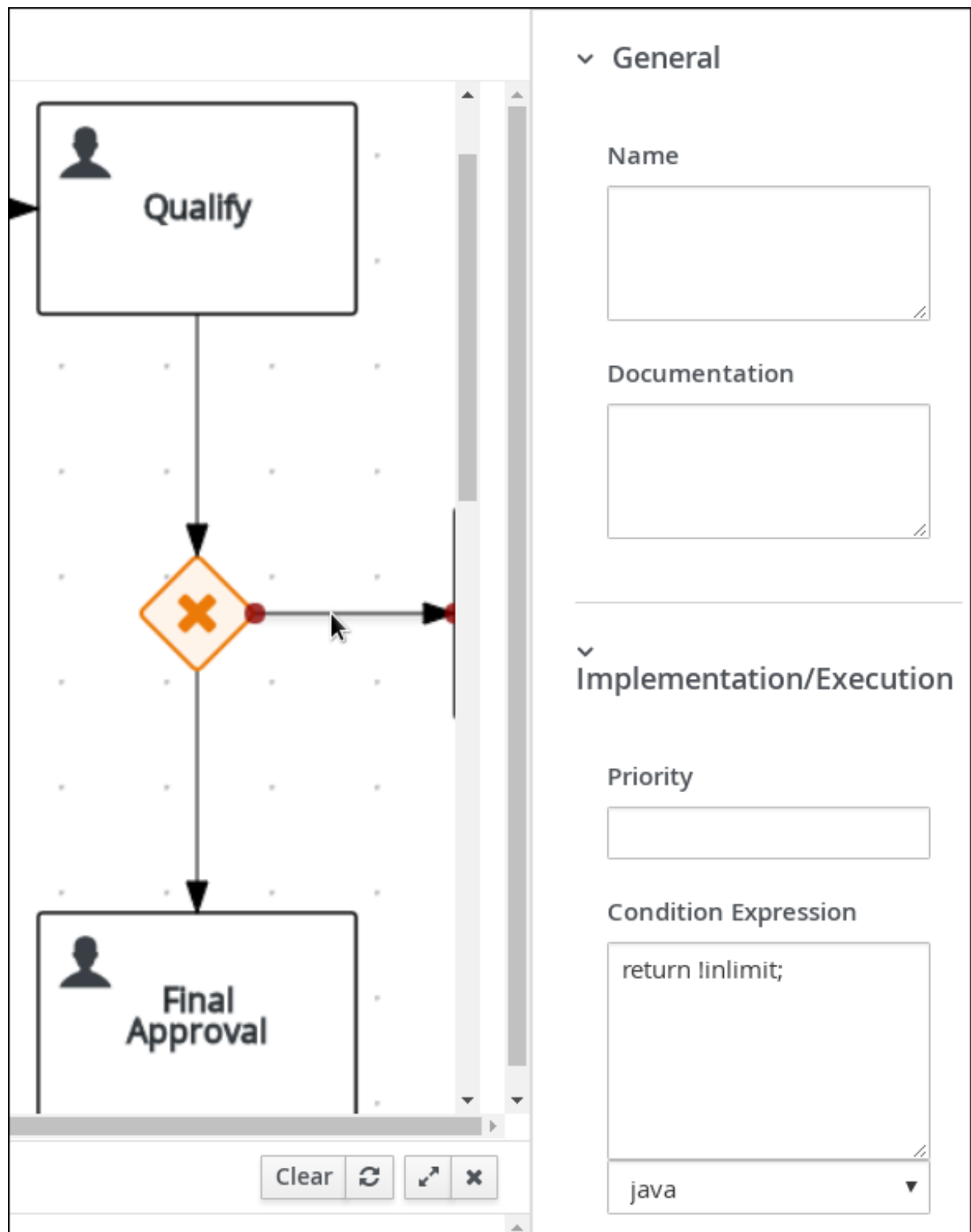



2. Return to the exclusive gateway that connects with the **Final Approval** user task. Create a second outgoing connection and connect it to a new user task.



3. Click the connection, name it **Not in Limit** and define the following Java expression:

```
return !inlimit;
```



4. Click on an empty space on the canvas, scroll down, expand **Process Data**, and click  next to **Process Variables**. Enter the following values:

- **Name:** `incdownpayment`
- **Data Type:** `Boolean`

Diagram properties

Name

MortgageApprovalProcess

Documentation

ID

GettingStartedMortgageApplication.MortgageApprovalF

Package

com.myteam.gettingstartedmortgageapplication

Version

1.0

☐ Ad-Hoc

Process Instance Description


▼ **Process Data**

Process Variables

| Name | Data Type | |
|----------------|---------------------|---|
| Application | Application [cor ▼] | + |
| inlimit | Boolean ▼ | ✕ |
| incdownpayment | Boolean ▼ | ✕ |

5. Click the created user task and define:

- **Name:** Increase Down Payment
- **Task Name:** incdown
- **Groups:** broker

- Click  next to **Assignments**. In the **Increase Down Payment Data I/O** window, click **Add** to create the following assignments:

Increase Down Payment Data I/O

Data Inputs and Assignments

Name


Data Type

Source

application

Application [com.i ▼]

application ▼



Data Outputs and Assignments

Name


Data Type

Target

incdownpayment

Boolean ▼

incdownpayment ▼



Cancel

Save

6. Click **Save**. Above the canvas, click **Save**, then **Save**, to confirm your changes.

7. Create an outgoing connection from the **Increase Down Payment** task to an exclusive gateway.

- Click on the **GATEWAYS** icon in the node panel.
- Click on **Exclusive** and drag it to the right of the **Increase Down Payment** task.

8. Click on the **Create Sequence Flow** icon to create a connection to the exclusive gateway.

25

9. Create an outgoing connection from the exclusive gateway and connect it to an end event. Then, click the connection, name it **Down payment not increased**, and create the following Java expression:

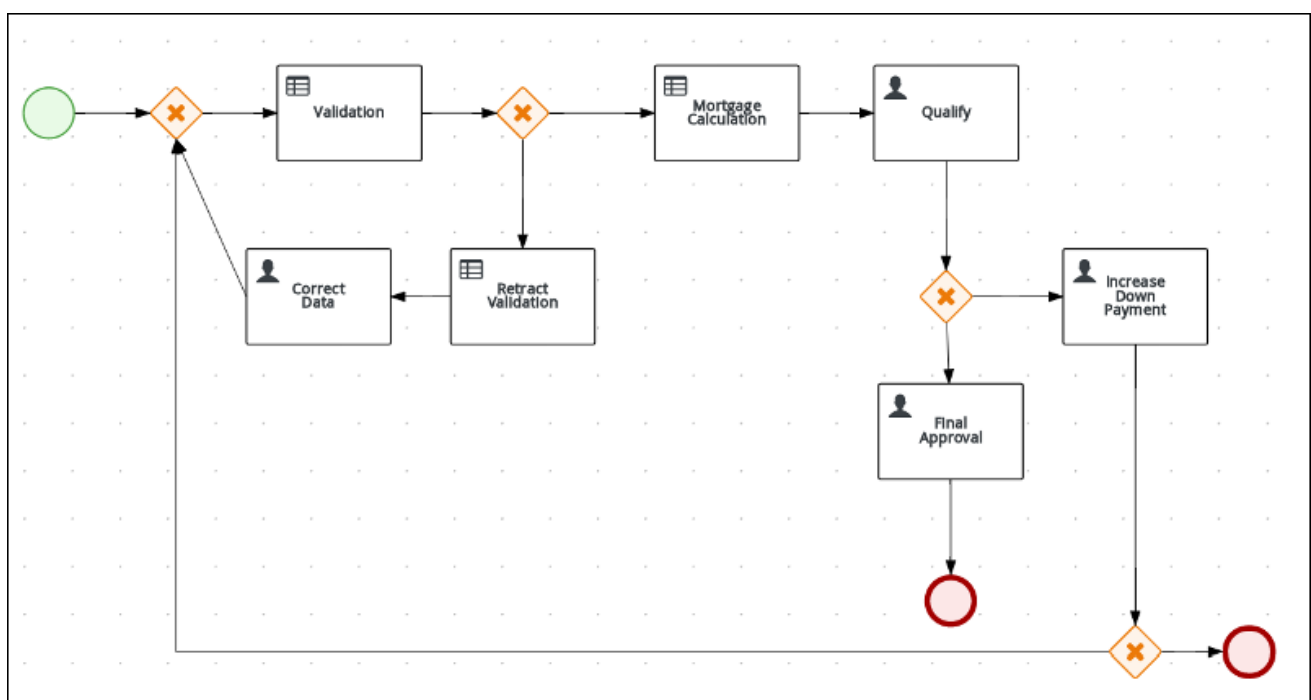
```
return !indownpayment;
```

10. Create an outgoing connection from the exclusive gateway and connect it to the first exclusive gateway. Then, click the connection, name it **Down payment increased**, and create the following Java expression:

```
return indownpayment;
```

11. Above the canvas, click **Save**, then **Save**, to confirm your changes.

The final version of the business process:



CHAPTER 6. GUIDED RULES

Guided rules are business rules that you create in a UI-based guided rules designer in Business Central that leads you through the rule-creation process. The guided rules designer provides fields and options for acceptable input based on the data objects for the rule being defined. The guided rules that you define are compiled into Drools Rule Language (DRL) rules as with all other rule assets.

All data objects related to a guided rule must be in the same project package as the guided rule. Assets in the same package are imported by default. After you create the necessary data objects and the guided rule, you can use the **Data Objects** tab of the guided rules designer to verify that all required data objects are listed or to import other existing data objects by adding a **New item**.

6.1. DEFINING BUSINESS RULES

Define business rules in Red Hat Process Automation Manager using the Guided Rule wizard.

Related information

For more information about Guided business rules, see [Designing a decision service using guided rules](#).

6.1.1. Creating the Validate Down Payment guided rule


Procedure

1. Log in to Business Central and click **Menu** → **Design** → **Projects**, then **Mortgage Process**.
2. Click **Add Asset** → **Guided Rule**, then enter:
 - **Guided Rule: Validate Down Payment**
 - **Package: com.myspace.mortgage_app**
3. Click **Ok** to open the Guided Rule Editor.

6.1.2. Defining the Validate Down Payment guided rule conditions

The conditions that you specify in this section are used to determine whether or not the down payment meets the specified requirements.

Procedure

1. Click  next to the **WHEN** label to open the **Add a condition to the rule** window. Then, select **Application...** and click **Ok**.
2. Click the **There is an Application** label and select **Any of (Or)** from **Multiple field constraints**.
3. Click the **There is an Application [app] with: any of the following:** label, then select **downpayment** from **Add a restriction on a field**.
4. Click **--- please choose ---**, then select **equal to**.

Validate Down Payment.rdr1 - Guided Rules ▾

Model Overview Source Data Objects

EXTENDS - None - ▾

WHEN

1. There is an Application **[app]** with:
any of the following:
downpayment equal to

THEN
(show options...)

Dropdown menu options:
--- please choose ---
equal to
not equal to
less than
greater than
less than or equal to
greater than or equal to
is null
is not null
is contained in the (comma separated) list
is not contained in the (comma separated) list

5. Click , select **Literal value**, and enter **0**.

Validate Down Payment.rdr1 - Guided Rules ▾


Model Overview Source Data Objects

EXTENDS - None - ▾

WHEN

1. There is an Application **[app]** with:
any of the following:
downpayment equal to 0

THEN
(show options...)

6. Click the **There is an Application [app] with: any of the following:** label, then from the **Add a restriction on a field** select **downpayment**.
7. Click **--- please choose ---**, then select **greater than or equal to**.
8. Click , then click **Expression editor**.
9. From the **Choose...** menu, select **app**.
10. From the **Choose...** menu, select **downpayment**.

6.1.3. Defining the error conditions



You must define the conditions and results of the data that is received and processed. In this section, you will define the following two conditions:

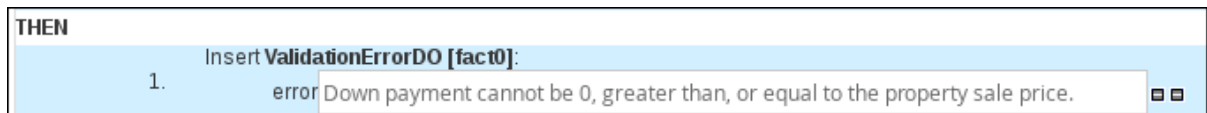
- ValidationErrorDO
- retractValidationErr

6.1.3.1. Defining the ValidationErrorDO

Specify the error message that will be presented to the loan applicant.

Procedure


1. Click  next to the **THEN** label. Select **Insert fact ValidationErrorDO**, and click **Ok**.
2. Click the **Insert ValidationErrorDO** label and select **error** from the **Add field** pull-down menu.
3. Click , then click **Literal value**.
4. In the **error** field, enter: **Down payment cannot be 0, greater than, or equal to the property sale price..**




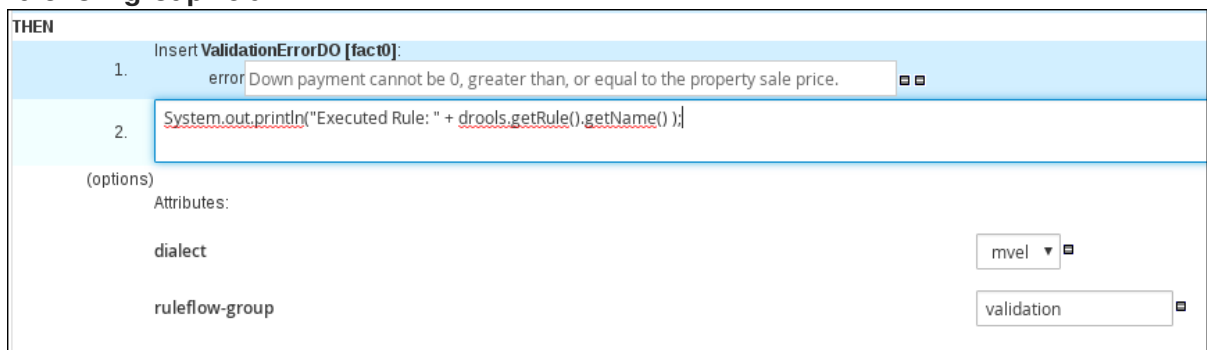
THEN

1. Insert **ValidationErrorDO** [fact0]:

error Down payment cannot be 0, greater than, or equal to the property sale price.

5. Click  next to the **THEN** label. Select **Add free form DRL** and click **Ok**.
6. Enter the following in the *Add free form DRL...* text box:


```
System.out.println("Executed Rule: " + drools.getRule().getName() );
```
7. Click **(show options...)**.
8. Click  next to the **(options)** label to open the **Add an option to the rule** window.
9. Select **ruleflow-group** from the **Attribute** drop-down menu and enter **validation** in the **ruleflow-group** field.



THEN

1. Insert **ValidationErrorDO** [fact0]:

error Down payment cannot be 0, greater than, or equal to the property sale price.

2. `System.out.println("Executed Rule: " + drools.getRule().getName());`

(options)

Attributes:

dialect mvel


ruleflow-group validation

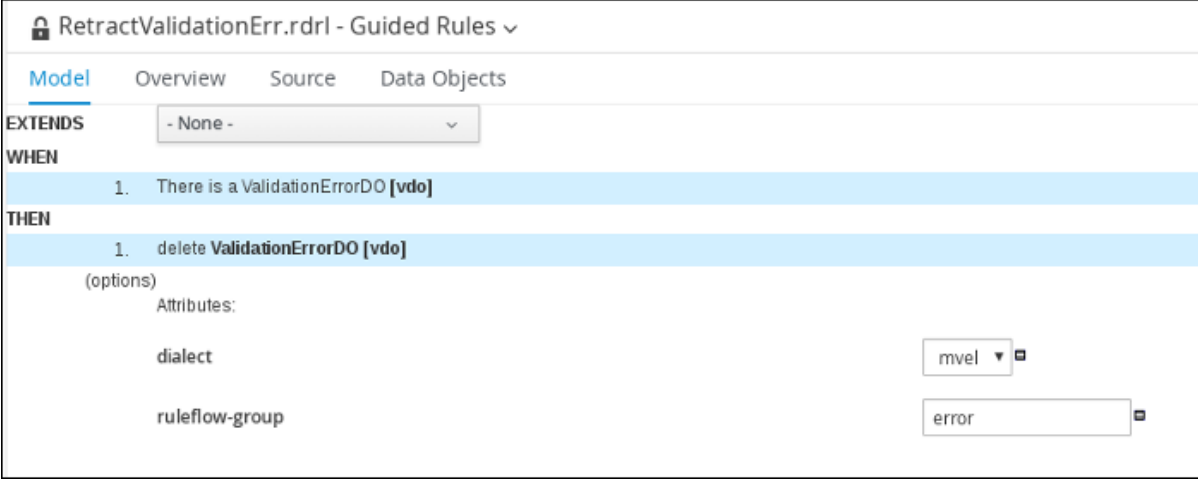
6.1.3.2. Defining the retractValidationErr

Define the error trigger details.

Procedure

1. Click  next to the **WHEN** label to open the **Add a condition to the rule** window. Then, select **ValidationErrorDO** and click **Ok**.

- Click **There is a ValidationErrorDO** to open the **Modify constraints for ValidationErrorDO** window, enter **vdo** in the variable name field, and click **Set**.
- Click  next to the **THEN** label. Select **Delete vdo**, and click **Ok**.
- Click **(show options)**.
- Enter **error** in the **ruleflow-group** field.



RetractValidationErr.rdr1 - Guided Rules ▾

Model Overview Source Data Objects

EXTENDS - None - ▾

WHEN

1. There is a ValidationErrorDO [vdo]

THEN

1. delete ValidationErrorDO [vdo]

(options)

Attributes:

dialect mvel ▾

ruleflow-group error ▾

- Click **Save**, then click **Save** to confirm your changes.

CHAPTER 7. GUIDED DECISION TABLES

Guided decision tables are a wizard-led alternative to uploaded decision table spreadsheets for defining business rules in a tabular format. With guided decision tables, you are led by a UI-based wizard in Business Central that helps you define rule attributes, metadata, conditions, and actions based on specified data objects in your project. After you create your guided decision tables, the rules you defined are compiled into Drools Rule Language (DRL) rules as with all other rule assets.

All data objects related to a guided decision table must be in the same project package as the guided decision table. Assets in the same package are imported by default. After you create the necessary data objects and the guided decision table, you can use the **Data Objects** tab of the guided decision tables designer to verify that all required data objects are listed or to import other existing data objects by adding a **New item**.

7.1. CREATING DECISION TABLES

Decision tables are a way to generate rules driven from the data entered into a spreadsheet. Decision tables are created in Red Hat Process Automation Manager using the Guided Decision Table wizard.

Prerequisites

The business rules have been defined. For more information, see [Section 6.1, “Defining Business Rules”](#).

7.1.1. Viewing the Mortgage Decision Table

The goal of this chapter is to introduce you to the **Mortgage Decision Table**. For this tutorial, you will not create and set the decision table conditions. Instead, review the values and the conditions that are already defined in the **Mortgage Process** sample project's **MortgageDecisionTable** Guided Decision Tables asset.

Procedure

1. Log in to Business Central and click **Menu** → **Design** → **Projects**, then **Mortgage Process**.
2. Scroll down and click the **MortgageDecisionTable** Guided Decision Tables asset.

| Mortgage Decision Table | | | | | | | | |
|-------------------------|-------------|---------------------|------------------------------|---------------------------------------|------------------------|--------------------------|----------|------------------|
| # | Description | ruleflow-group | f1 : Applicant | | f2 : Property | | | f3 : Application |
| | | | Annual Income (Greater Than) | Annual Income (Less Than or Equal To) | Sale Price (Less Than) | Property Age (Less Than) | Location | Mortgage Amount |
| 1 | | mortgagecalculation | 100000 | 200000 | 300000 | 5 | Urban | 200000 |
| 2 | | mortgagecalculation | 50000 | 99999 | 100000 | 10 | Rural | 100000 |

CHAPTER 8. GENERATING AND EDITING FORMS

The following chapter shows you how to automatically generate forms for collecting user data for your mortgage application business process. We have also included a bit more information about editing forms to familiarize you with Red Hat Process Automation Manager form manipulation.

Prerequisites

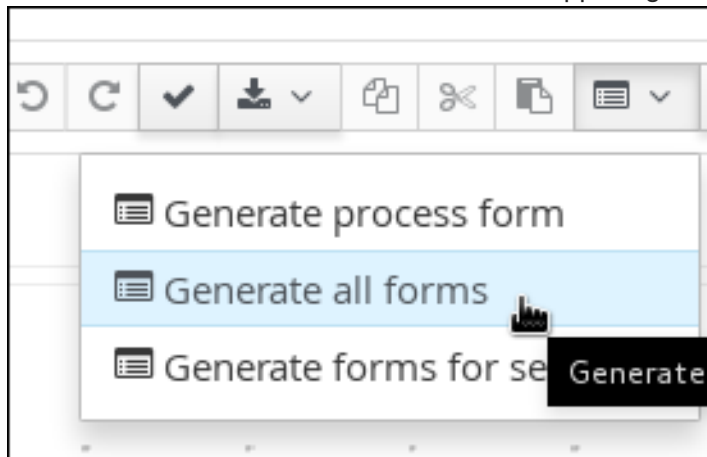
The business process model has been completed. For more information, see [Chapter 5, Creating the MortgageApprovalProcess business process](#).

8.1. AUTOMATIC FORM GENERATION

Red Hat Process Automation Manager enables you to automatically generate forms. For this business process you will automatically generate the **Applicant**, **Property**, and **Application** forms.

Procedure

1. Log in to Business Central and click **Menu** → **Design** → **Projects**, then **Mortgage Process**.
2. From the asset list, select the **MortgageApprovalProcess** business process.
3. Click on the **Form Generation** menu in the upper-right menu and select **Generate all forms**.



4. Click **Menu** → **Design** → **Projects**, then **Mortgage Process**.
5. From the asset list, select the **Applicant** form.
The **Applicant** form should look as follows:

The screenshot shows the 'Form Modeler [Applicant]' window. The top bar contains buttons for 'Save', 'Delete', 'Rename', 'Copy', 'Latest Version', 'Hide Alerts', and a close button. Below the bar, the 'Model' tab is selected, showing an 'Overview' of the form. The form contains three input fields, each with a label and a placeholder text:

- Name**: Placeholder text is 'Name'.
- SSN**: Placeholder text is 'SSN'.
- Annual Income**: Placeholder text is 'Annual Income'.

6. Click **Menu** → **Design** → **Projects**, then **Mortgage Process**.

7. From the asset list, select the **Property** form. The **Property** form should look as follows:

The screenshot shows the 'Form Modeler [Property]' window. The top bar contains buttons for 'Save', 'Delete', 'Rename', 'Copy', 'Latest Version', 'View Alerts', and a close button. Below the bar, the 'Model' tab is selected, showing an 'Overview' of the form. The form contains four input fields, each with a label and a placeholder text:

- Age of property**: Placeholder text is 'Age of property'.
- Address of property**: Placeholder text is 'Address of property'.
- Locale**: Placeholder text is 'Locale'.
- Sale Price**: Placeholder text is 'Sale Price'.

8. Click **Menu** → **Design** → **Projects**, then **Mortgage Process**.

9. From the asset list, select the **Application** form. The **Application** form should look as follows:

The screenshot shows the 'Form Modeler [Application]' interface. At the top, there are tabs for 'Model' and 'Overview', and a toolbar with buttons for 'Save', 'Delete', 'Rename', 'Copy', 'Latest Version', 'View Alerts', and a close button (X). The form is structured as follows:

- Down Payment**: A text input field with the placeholder text 'Down Payment'.
- Years of amortization**: A text input field with the placeholder text 'Years of amortization'.
- Applicant**: A section containing three text input fields:
 - Name**: Placeholder text 'Name'.
 - SSN**: Placeholder text 'SSN'.
 - Annual Income**: Placeholder text 'Annual Income'.
- Property**: A section containing three text input fields:
 - Locale**: Placeholder text 'Locale'.
 - Age of property**: Placeholder text 'Age of property'.
 - Sale Price**: Placeholder text 'Sale Price'.

10. Click on the **X** in the upper-right corner to close the editor.

8.2. EDITING FORMS (OPTIONAL)

The following optional process shows you how to edit the Applicant form that you previously defined as part of this tutorial business process:

Procedure

1. Log in to Business Central and click **Menu** → **Design** → **Projects**, then **Mortgage Process**.
2. Click the **Applicant** data object form and add a new **Address** row.
 - a. Expand the **Form Controls** menu and drag the **TextBox** field in to the existing form.
 - b. Complete the form field properties for the new Address row.

Field properties

Field Type
TextBox

Label
Address

Placeholder
Adress

Max. Length
100

☐ Required

☐ Read Only

☒ Validate on change field value

Help Message ⓘ

Field binding

+ Ok

Cancel

3. Click the three vertical dots in the upper-right corner to edit or delete the row.
4. Click **Save**, then click **Save** to confirm your changes.


CHAPTER 9. DEPLOYING MORTGAGE PROCESS APPLICATION

The following chapter instructs you how to build and deploy a new instance of the **Mortgage Process** application in Red Hat Process Automation Manager.

Prerequisites

The Process Server is deployed and connected to the Business Central.

Procedure

1. Log in to Business Central and click **Menu** → **Design** → **Projects**.
2. Click on the project you want to deploy, for example **Mortgage Process**.
3. Click **Deploy**.
 - If there is no KIE container (deployment unit) with the project name, a container with default values is automatically created.
 - If an older version of the project is already deployed, go to the project settings and change the project version. When finished, save the change and click **Deploy**. This will deploy a new version of the same project with the latest changes in place, alongside the older version(s).
4. To verify the deployment, click **Menu** → **Manage** → **Process Definitions**, and click  .
5. Click on the three vertical dots in the **Actions** column and select **Start** to start a new instance of the process.

CHAPTER 10. EXECUTING THE MORTGAGE PROCESS

Now that you have deployed the project, you can execute the project's defined functionality. For this tutorial you will input data in to a mortgage application form acting as the mortgage broker. The `MortgageApprovalProcess` business process will run and will determine whether or not the applicant has offered an acceptable down payment based on the decision rules that you defined earlier. The business process will either end the rule testing or request that the applicant increase the down payment to proceed. If the application passes the business rule testing, the bank's approver will review the application and either approve or deny the loan.

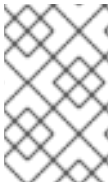
Prerequisites

- The Process Server is deployed and connected to the Red Hat Process Automation Manager
- The **Mortgage Process** application has been deployed
- The user or users working on the tasks must be members of the following groups:
 - **approver** - For the **Qualify** task
 - **broker** - For the **Correct Data** and **Increase Down Payment** tasks
 - **manager** - For the **Final Approval** task

Procedure

1. Log in to Red Hat Process Automation Manager and click **Menu** → **Manage** → **Process Definitions**.
2. Click any where in the **MortgageApprovalProcess** row to view the process details.
3. Click the **Diagram** tab to view the business process diagram in the editor.
4. Click on the **X** in the upper-right corner to close the editor.
5. Click **New Process Instance** to open the **Application** form and input the following values in to the form fields:
 - **Down Payment:** 30000
 - **Years of amortization:** 10
 - **Name:** Ivo
 - **Annual Income:** 60000
 - **SSN:** 123456789
 - **Age of property:** 8
 - **Address of property:** Brno
 - **Locale:** Rura1
 - **Property Sale Price:** 50000

6. Click **Submit** to start a new process instance. After starting the process instance, the **Instance Details** view opens.
7. Click on the **Diagram** tab to view the process flow within the process diagram. The state of the process will be highlighted as it moves through each task.
8. Click **Menu** → **Track** → **Task Inbox**. This will take you to the **Qualify** form.
9. As the approver, you will review the **Qualify** task information, and if approved, select **Is mortgage application in limit?** and click **Complete**.
10. In the **Task Inbox**, click anywhere in the **Final Approval** row to open the **Final Approval** task.
11. Click **Claim** to claim responsibility for the task, then click **Complete** to finalize the loan approval process.



NOTE

The **Save** and **Release** buttons are only used to either pause the approval process and save the instance if you are waiting on a field value, or to release the task for another user to modify.

CHAPTER 11. MONITORING THE MORTGAGEAPPROVALPROCESS PROCESS APPLICATION

The following chapter shows how different bank employees, such as a system administrator or a knowledge worker, might use some of the monitoring capabilities to track an instance of the mortgage approval process.

Prerequisites

The Process Server is deployed and connected to the Business Central.

Procedure

1. Log in to Red Hat Process Automation Manager and click **Menu** → **Manage** → **Process Instances**.
2. In the **Manage Process Instances** window, you can set filters, such as **State**, **Errors**, **Id**, and so on.
3. Select **Completed** in the **State** filter to view all completed **MortgageApprovalProcess** instances.
4. Click on each of the following tabs to get a feel for what type of information is available to monitor a specific process instance:
 - **Instance Details**
 - **Process Variables**
 - **Documents**
 - **Logs**
 - **Diagram**
5. Click **Menu** → **Track** → **Process Reports**. This view contains a variety of charts that can help a senior process manager to gain an overview of all processes based on **Type**, **Start Date**, **Running Time**, and so on to assist with task reporting.

11.1. FILTERING PROCESS INSTANCES USING DEFAULT OR ADVANCED FILTERS

Business Central now provides you with default and advanced filters to help you filter and search through running process instances. You can also create custom filters using the **Advanced Filters** option.

11.1.1. Filtering process instances using default filters

Filter processes instances by attributes such as **State**, **Errors**, **Filter By**, **Name**, **Start Date**, and **Last update**.

Procedure

1. In Business Central, go to **Menu** → **Manage** → **Process Instances**.

2. On the **Manage Process Instances** page, click the filter icon on the left of the page to expand the **Filters** pane.

This pane lists the following process attributes which you can use to filter process instances:

- **State:** Filter process instances based on their state (**Active**, **Aborted**, **Completed**, **Pending**, and **Suspended**).
- **Errors:** Filter process instances by errors.
- **Filter By:** Filter process instances based on **Id**, **Initiator**, **Correlation Key**, or **Description** attribute.
 - i. Select the required attribute.
 - ii. Enter the search query in the text field below.
 - iii. Click **Apply**.
- **Name:** Filter process instances based on process definition name.
- **Start Date:** Filter process instances based on their creation date.
- **Last update:** Filter process instances based on their last modified date.

You can also use the **Advanced Filters** option to create custom filters in Business Central.

11.1.2. Filtering process instances using advanced filters

Use the **Advanced Filters** option to create custom process instance filters. The newly created custom filter is added to the **Saved Filters** pane, which is accessible by clicking on the star icon on the left of the **Manage Process Instances** page.

Procedure

1. In Business Central, go to **Menu** → **Manage** → **Process Instances**.
2. On the **Manage Process Instances** page, click on **Advanced Filters** icon.
3. In the **Advanced Filters** pane, enter the name and description of the filter, and click **Add New**.
4. Select an attribute from the **Select column** drop-down list, for example, **processName**. The content of the drop-down changes to **processName != value1**.
5. Click the drop-down again and choose the required logical query. For the **processName** attribute, choose **equals to**.
6. Change the value of the text field to the name of the process you want to filter.



NOTE

The name must match the value defined in the business process of the project.

7. Click **Save** and the processes are filtered according to the filter definition.
8. Click on the star icon to open the **Saved Filters** pane.
In the **Saved Filters** pane, you can view all the saved advanced filters.

APPENDIX A. VERSIONING INFORMATION

Documentation last updated on Friday, October 12, 2018.