**Lab 11 - Modeling Business Processes**

The policyquoteprocess you create in this lab leverages the existing policyquote.policyquotecalculationprocess (which you created in ruleflow) as a subprocess and adds BPMN2-compliant Service Task and Human Task nodes. From this you learn how to start using the Process Designer to create a business process model.

**Goal:**

Create the policy quote process in the Process Designer (also called Web Designer).

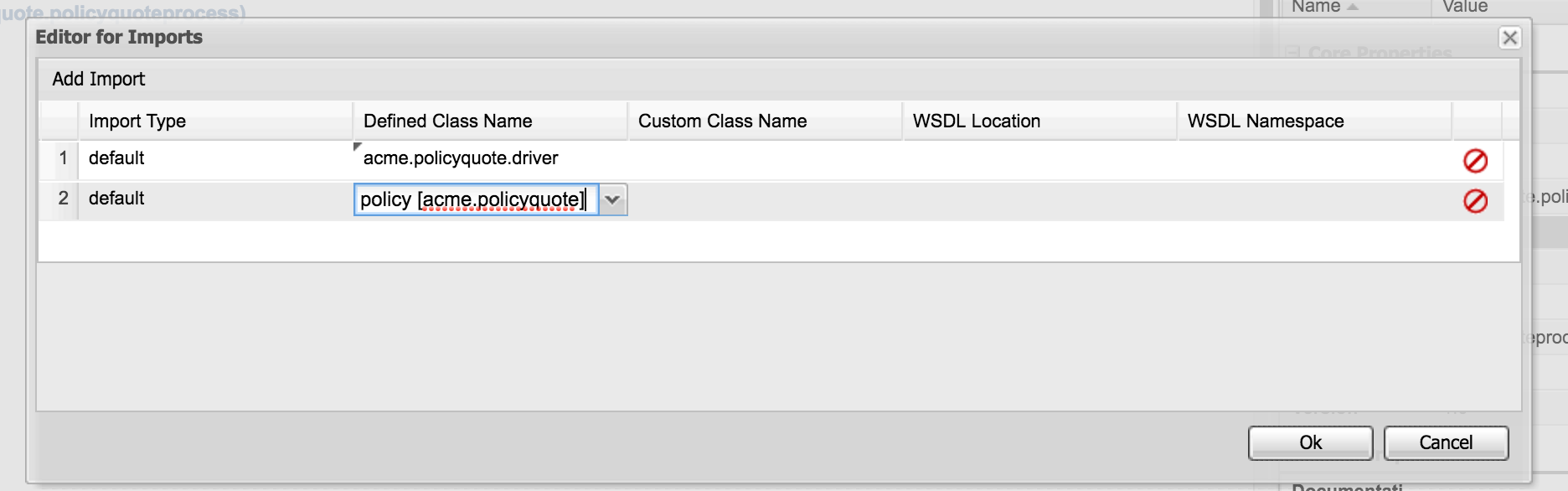
**1. Create the Policy Quote Process**

The business process maintains the state of key data points that must be passed to different activities within the process. JBoss BPM Suite 6 uses variable declarations to hold this information.

For this lab, information from previous labs provides a number of data items. For other data items, including driver name, age, number of accidents, number of tickets, and vehicle year, the process captures this information from form input. The process uses these data items to populate the Policy and Driver facts that are inserted into the JBoss BPM Suite 6 working memory for the knowledge base that the rule engine uses.

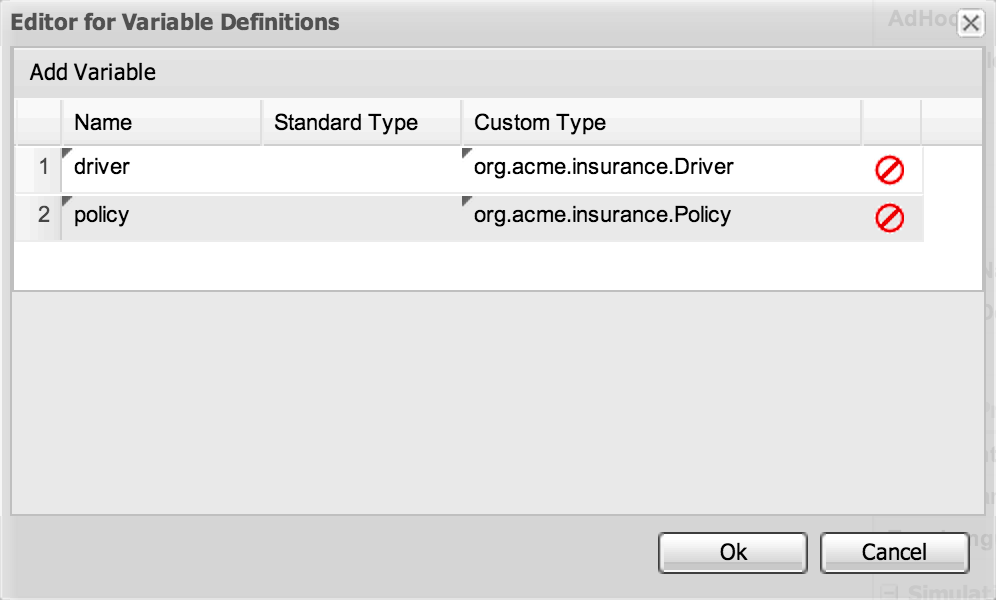
We will now create another super BPM process that is the higher level policy process. Within this super process, we will invoke the process created in Lab10 for policy quote calculation.

1. Add a new BPMN2 process :
   1. Click **Authoring** → **Project Authoring** in the BPM Console of BPM Suite 6.
   2. In **Project Explorer**, click **New Item** → **Business Process**.
   3. On the screen that appears, in the **Name** field, enter **policyquoteprocess**.
   4. Make sure the **package** is set to **acme.policyquote**
   5. Click **Ok**.
2. Add imports:
   1. Click an empty area of the diagram (not on any element).
   2. Open the **Properties** panel to display the properties for the diagram.
   3. Expand the **Core Properties** section.
   4. Click **Imports** and enter the following:

 Figure 1. Process imports

* 1. Click **Ok**.

1. Add process variable declarations:
   1. Click the down arrow on the value for the **Variable Definitions** property.
   2. On the editor screen that appears, click **Add Variable**
   3. Click Defined Types and pick the driver class first and then select the driver and policy class. In the name column, type ‘driver’ and ‘policy’ respectively like in screenshot below.

Figure 2. Process variables

* 1. Click **Ok**.

**2. Add Process Nodes**

1. Add the Script task after the start node
   1. Click the square icon with rounded corners, and drag it to the right of the start event.
   2. With the new task node selected, click the **<<** icon in the upper-right corner of the drawing palette to open the **Properties** panel.
   3. For the task, in the **Name** field, enter **Prepare Data**.
   4. For the **Task Type**, select **Script**.
   5. Under the **Task Type**, click **+** to open **Extra Properties**.
   6. Under **Script Language**, set the value to Java.
   7. Under **Script**, click the **Value** box.
   8. Copy and paste the following script into the screen that appears, and then click **Ok**:

// instantiate and pre-populate driver and policy domain classes

if(driver == null) {

driver = new driver();

driver.setDrivername("Azra");

driver.setAge(22);

driver.setNumOfAccidents(0);

driver.setNumofTickets(1);

kcontext.setVariable("driver", driver);

}

if(policy == null) {

policy = new policy();

kcontext.setVariable("policy", policy);

}

if(policy.getPolicytype() == null)

policy.setPolicytype( "AUTO" );

policy.setDriver(driver);

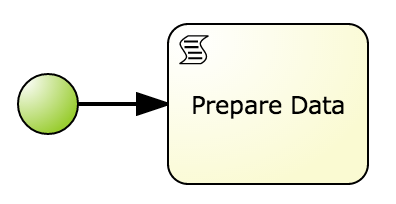
System.out.println("\*\* prepareData: will now add policy & driver to kruntime : "+policy);

/\* insert objects into working memory \*/

kcontext.getKnowledgeRuntime().insert( driver );

kcontext.getKnowledgeRuntime().insert( policy );

Your diagram should look like the following:

Figure 5. Process with two elements

**3. Add a Reusable Subprocess**

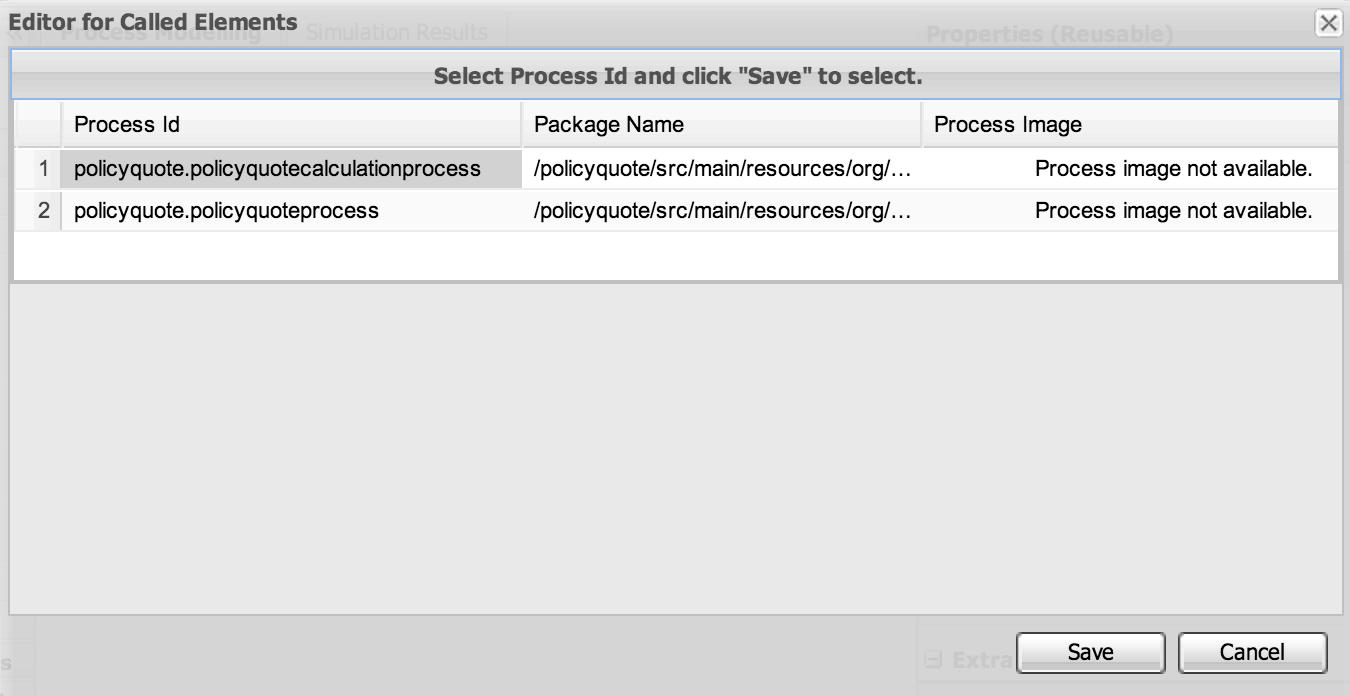
In BPMN2, you use a reusable subprocess to model a global pre-existing process.

1. On the **Object Library** screen, click the **Subprocesses** submenu to locate the **Reuseable** icon:



Figure 6. Reusable subprocess

1. Drag the icon to the right of the **Prepare Data** task.
2. Change the name to **Calculate Policy Quote**.
3. Select the **Called Element** property.
4. On the screen that appears, select **policyquote.policyquotecalculationprocess**:

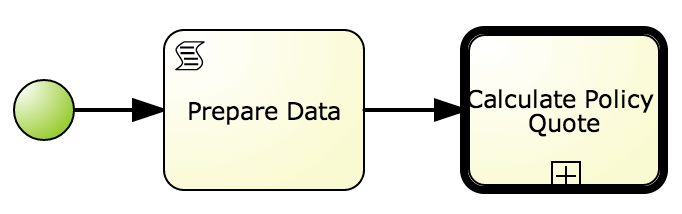
Figure 7. Called element selection

1. Click **Save**.
2. Click the Prepare Data task.
3. From the shortcut menu, drag a sequence flow arrow from the shortcut palette to the new subprocess.

Note:

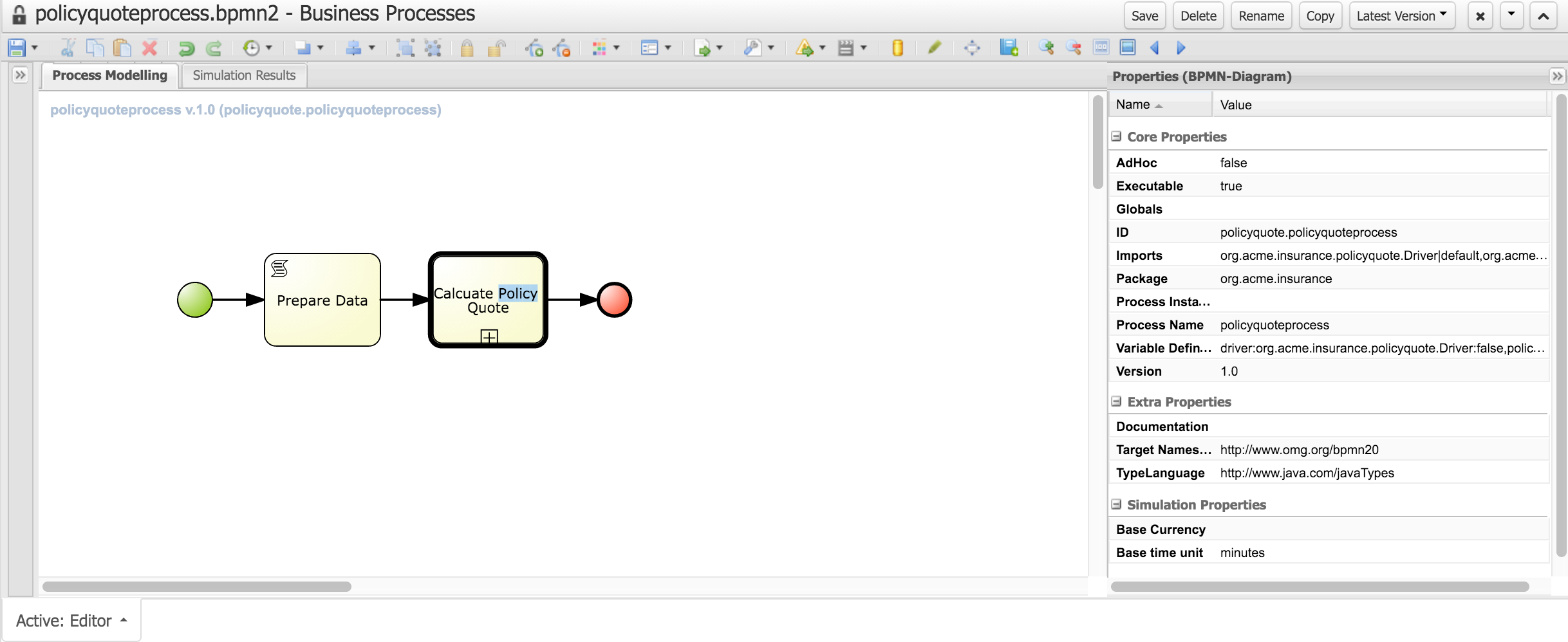
The red indicators change to green when you are at the proper place to release the arrow. . Save the incomplete process to make sure your work is not lost.

Your diagram should look like the following:

Figure 8. Process with three elements

With the subprocess highlighted, select the endnode to finish the process.

The process now looks like this.



At this point, rules in the JBoss BPM Suite 6 engine solely determine the policy quotes that come out of the system. This is a great start, but sometimes you need to handle special cases a certain way to remain competitive. For example, Acme needs you to set up your system so that any policy with a price quote over $500 automatically triggers a manual review. You will do this in the next lab.