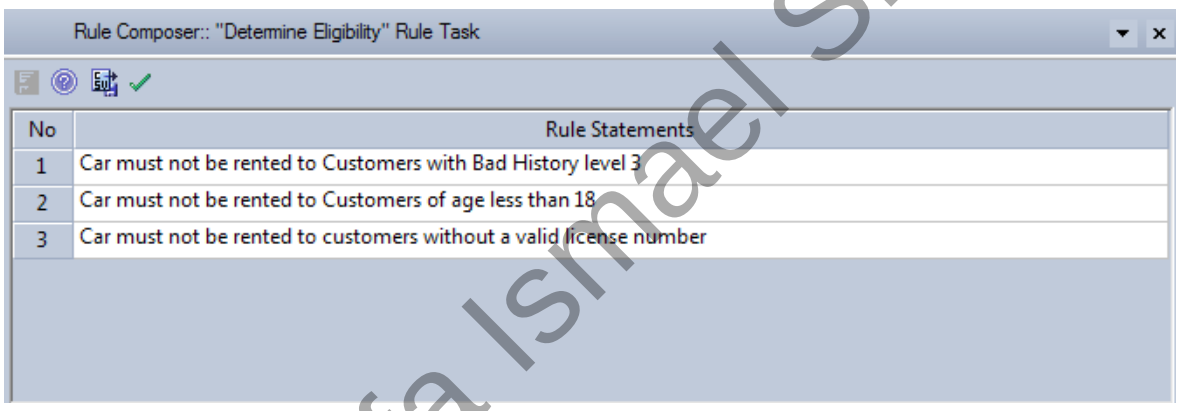
What is a Rule Table?

The Rule Table lists all the Rule Statements associated with a given Rule Task. The rules are written in plain text and should be easy to read and interpret. Let us examine the Rule Table for the Rule Task entitled “Eligibility”. Three rules are listed that will need to be satisfied in order for a customer to be eligible to rent a car. This information can be seen in our Business Rules that were defined in **Step 3**.

A rule can be added to the table by dragging an existing Business Rule element from the Project Browser onto an empty row in the Rule Table. This will create a dependency relationship between the Business Rule element and the Rule Task element.

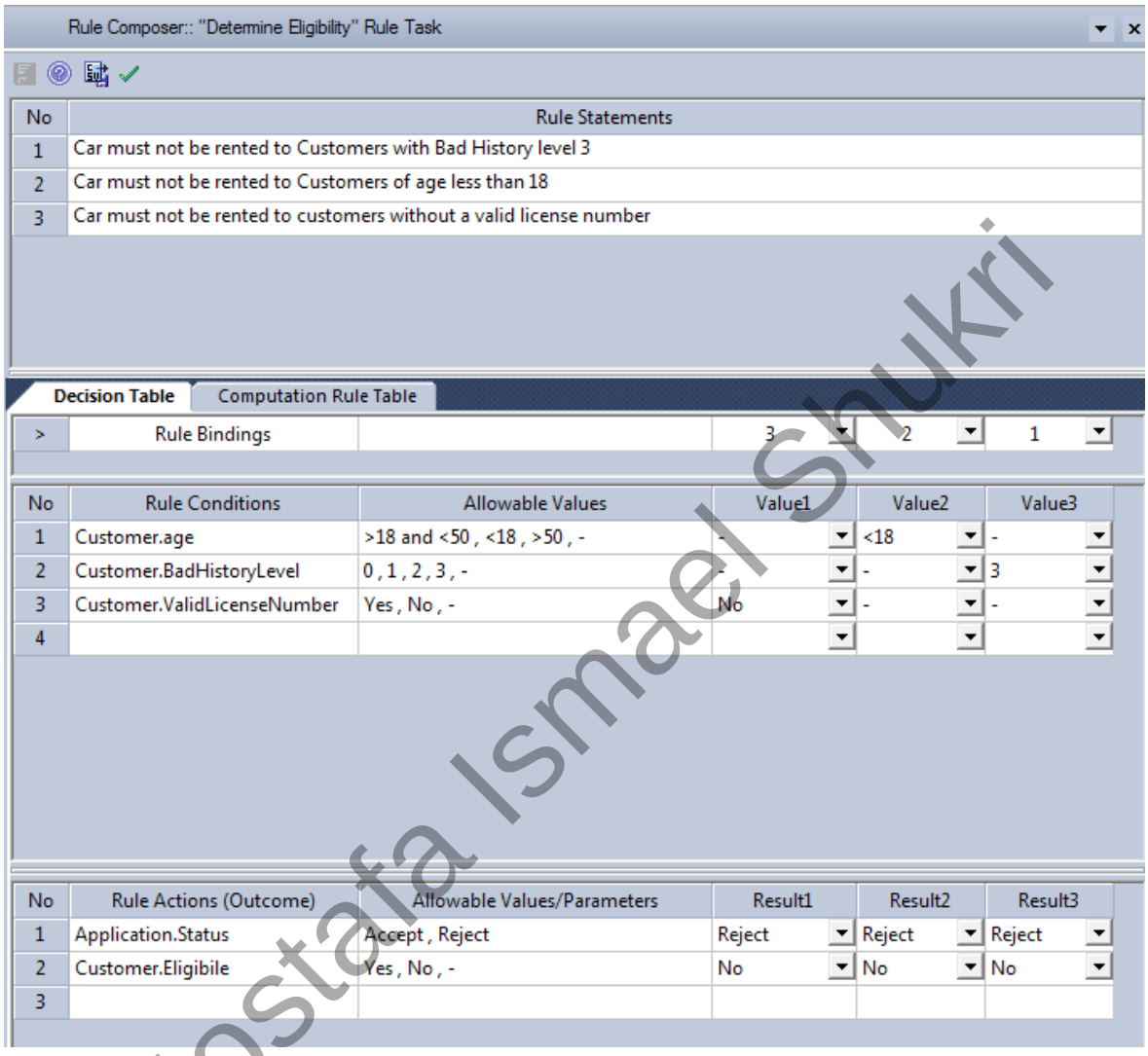
Important Note: *To define a new rule, we have to open the Rule Model diagram and create a new Business Rule element. Similarly removing a rule from this table will not delete the Rule element from the model. Instead, it will remove the dependency relationship between the Rule Task and the Business Rule element.*



Why use a Decision Table?

The Rule Composer's Decision Table:

- Allows you to model conceptual Business Rules at a logical level.
- Lists a range of cause and effect relationships in a matrix.
- Is easy to read and understand.
- Provides a CSV file export, which allows for processing by other tools.
- Helps you create executable code.
- Allows you to create queries and examine multiple paths.
- Displays each cause as a condition down the left hand side.
- Illustrates a set of conditions that will result in a given action.



Note: The "-" symbol denotes the corresponding rule condition does not apply to this rule. For example if a customer does not have a valid license, their age and driving history is not relevant. This technique also allows you to optimize the Table and remove redundant entries.

How To Access The Rule Composer:

1. Open your Rule Flow Model diagram.
2. Select a Rule Task element in the Rule Flow diagram.
3. Right-click on a Rule Task element in the diagram and select "Rule Composer" from the context menu.
4. The Rule Composer View for the the Rule Task will open up in the central work area of Enterprise Architect.

Using the Condition Section:

1. The Business Domain Model defines the business terms (such as Customer) and their associated attributes. From the appropriate Class element in the Project Browser, drag and drop the required condition attribute (such as age) onto the Condition Variable column.
2. Alternatively, the auto complete feature can be used. Click into the condition variable column and press CTRL+Space, The terms in the domain model will be displayed. Select the appropriate one and pressing "." will list all the attributes and operations of the term. Then select the appropriate one.

   Note: For an operation to be a condition, it must return a value.
3. Double-click on the "Allowable Values" field and enter a range of values that the condition variable can take.
4. For example, our Customer History level is defined as 0,1,2,3. Where 0 is not applicable and level 3 represents a customer with a poor driving history.
5. If the condition variable is of an enumeration type that is defined and listed in the Business Domain Model, then the allowable values field is automatically filled with the appropriate values. (Note: The Allowable Values are saved as constraints for the attributes.)

Using the Action Section

In the Action Variable section, when a specific value of a condition variable calls an operation or changes the value of an attribute, you assign the operation or attribute as an action. To model action variables, follow the steps below:

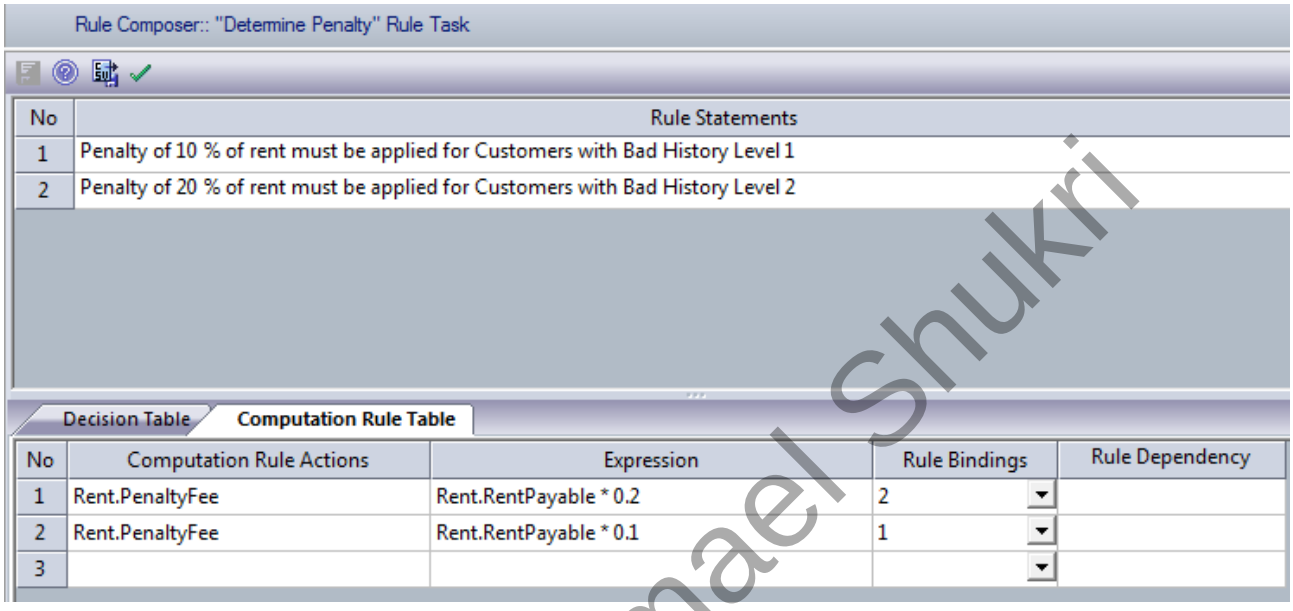
1. From a Class element in Business Domain Model in the Project Browser, drag and drop the required attribute or operation onto the Action Variable field. (Note: This field allows attributes and operation types to be added).
2. Alternatively, the auto complete feature can be used. Click into the condition variable column and press CTRL+Space. The terms in the domain model will be displayed. Select the appropriate one and pressing "." will list all the attributes and operations of the term. Then select the appropriate one.
3. For attributes, double-click on the "Allowable Values/Parameters" field and enter a range of values that the action variable can take is the Edit Allowable Values dialog. If the action variable is of an enumeration type that is defined in the Business Domain Model, then the allowable values/parameters field is automatically filled with the enumeration members. For operations, double-click on the "Allowable Values / Parameters" field and enter parameter values to be passed on calling the operation in Edit Parameters dialog.

Using the Rule Bind Section

The Rule Bind section binds the condition variable and action variable values to the appropriate rule in the Rule Table. Each rule in the Rule Table is numbered to make it easier to read. To bind a value column in the decision table to a rule, select the corresponding rule number in the rule bind cell for that column.

Computational Rule Table

The Computational Rule table enables you to model rules involving computations. For example, our rental car system needs to calculate rent payable and penalties.



The table has the following columns:

- Rule Variable
- Expression
- Rule
- Rule Dependency.

To define a computational rule, follow the steps below:

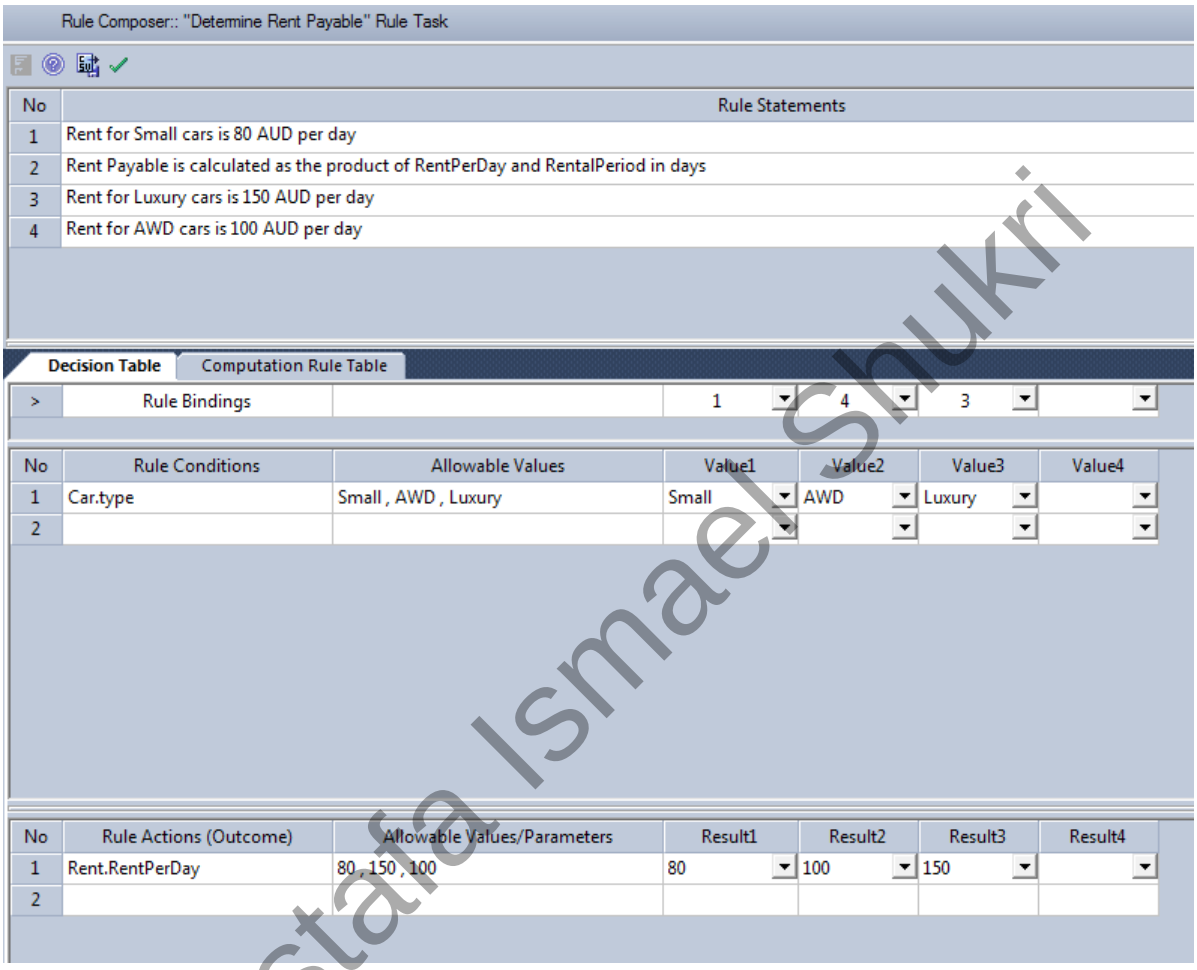
From the Project Browser, drag and drop the appropriate attribute from a Class in the Business Domain Model into the Variable field.

1) In the Expression field, type the expression to be evaluated. [For intellisense /autocomplete suggestions, press CTRL + Space]
2) In the Rule field, type the rule number from the Rule Table of the rule being modeled, to link the table data to the rule.
3) If the rule depends on another rule being satisfied first, Select the number of that rule in the Rule Dependency field.
4) Click on the Save button in the Rule Composer toolbar to save the computational rule.

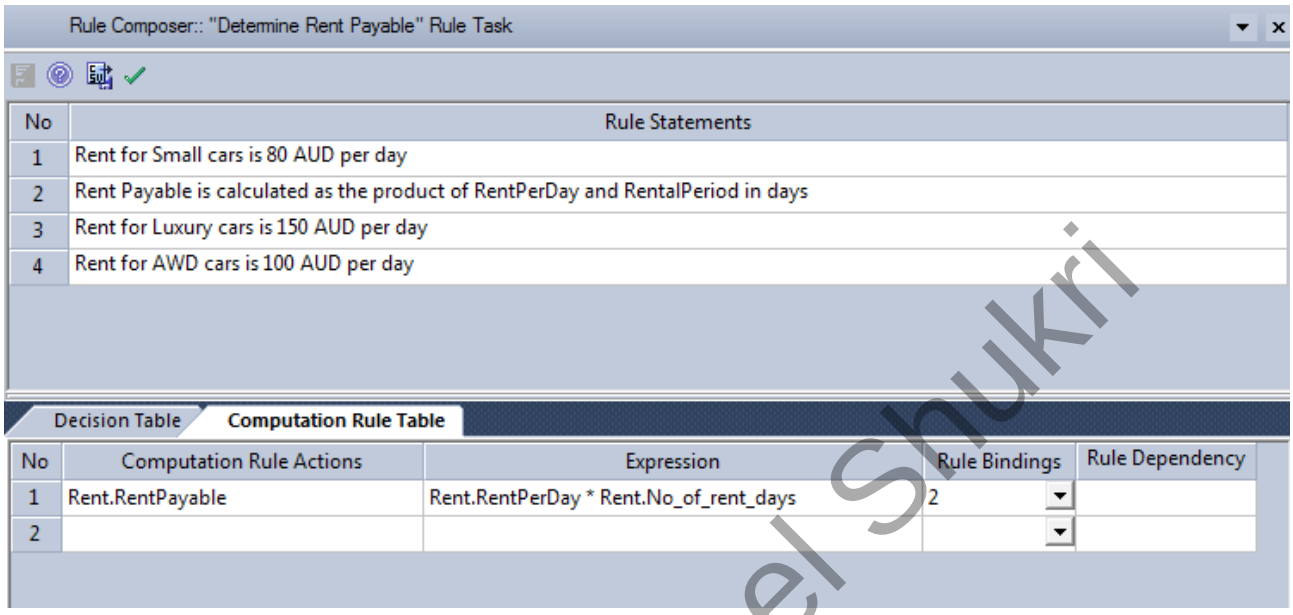
If the computation rule is also a conditional rule, add the condition variable in the Decision table and bind the appropriate rule in the Rule Bind section.

The Completed Rule Table

The “Determine Rent Payable” Rule Task:



The Computational Rule Table for “Determine Rent Payable” Rule Task:



Section Review

What have we learned?

We have used the Business Rule Composer to develop a logical model of the Business Rules defined in natural language. This logical model is used to create executable source code. You have learned how to use the Rule Table, Decision Table and Computational Rule Table to create a logical Model from natural language statements.

The Computational Rule Table has been used to calculate the “Total Amount Payable”.

Where to next?

Next step will be to validate the rules modeled in Rule Composer.