

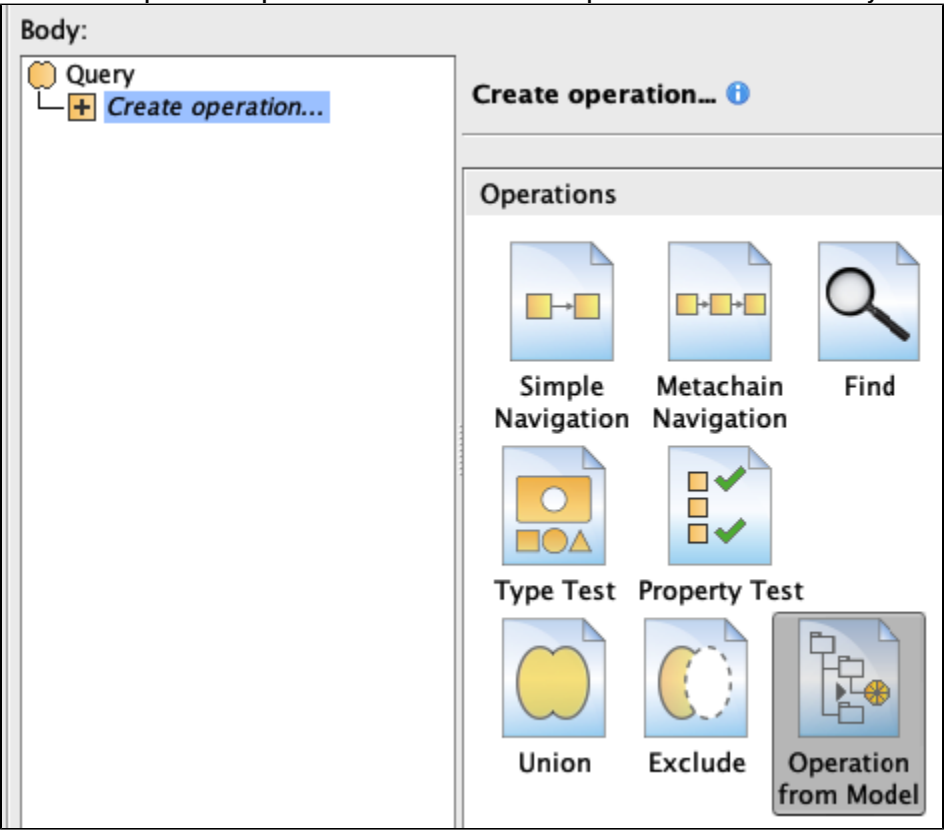
Case 15. Requirements Coverage by Design Elements

Prerequisites:

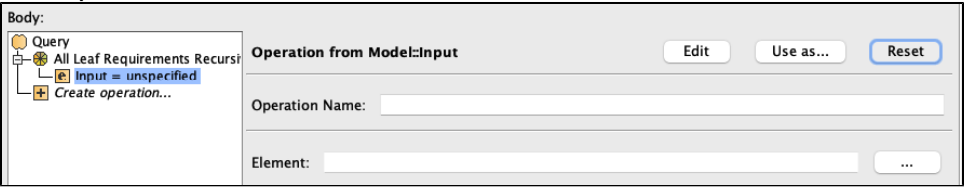
- [Case 15.1. All Leaf Requirements Owned Recursively](#)

To create a Requirements Coverage by Design Elements Legend

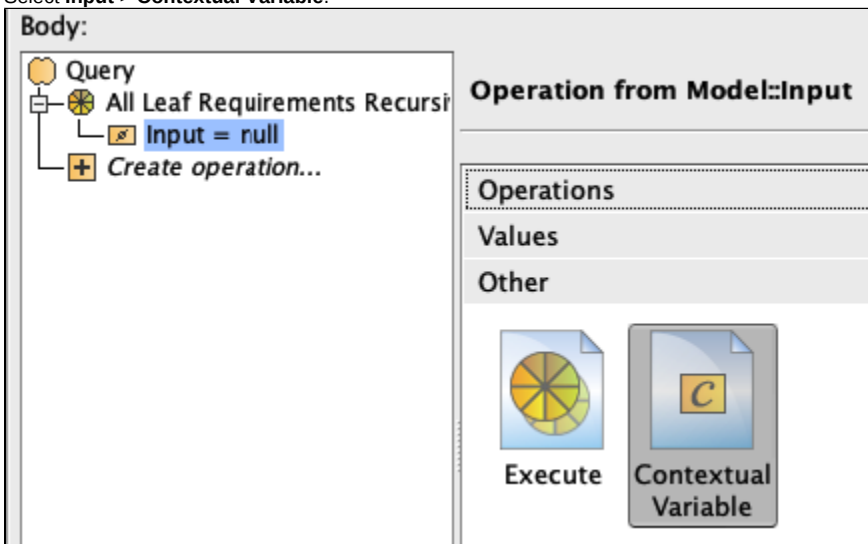
1. Create a [Legend](#).
2. Create a [Legend Item](#) named '0%-25%'.
3. Right-click the **Legend Item** and open its **Specification window**.
4. Click three dots  next to the **Elements by Condition** property to edit it.
5. In the **Elements by Condition** dialog, set the **Element Type** to **Requirement**.
6. Rename **Custom** to **Query**.
7. Select **Create Operation > Operation from Model > All Leaf Requirements Owned Recursively**.



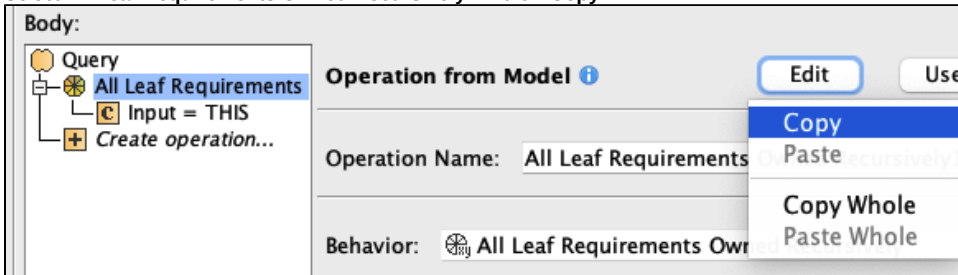
8. Select **Input > Reset**.



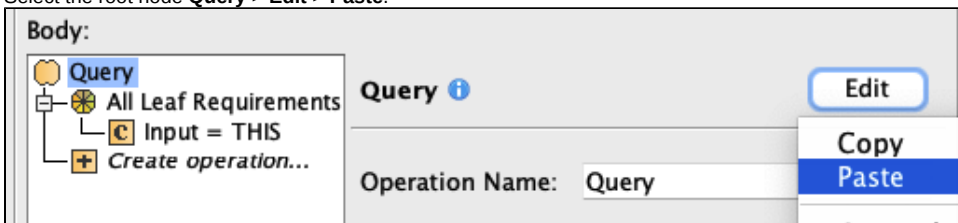
9. Select **Input > Contextual Variable**.



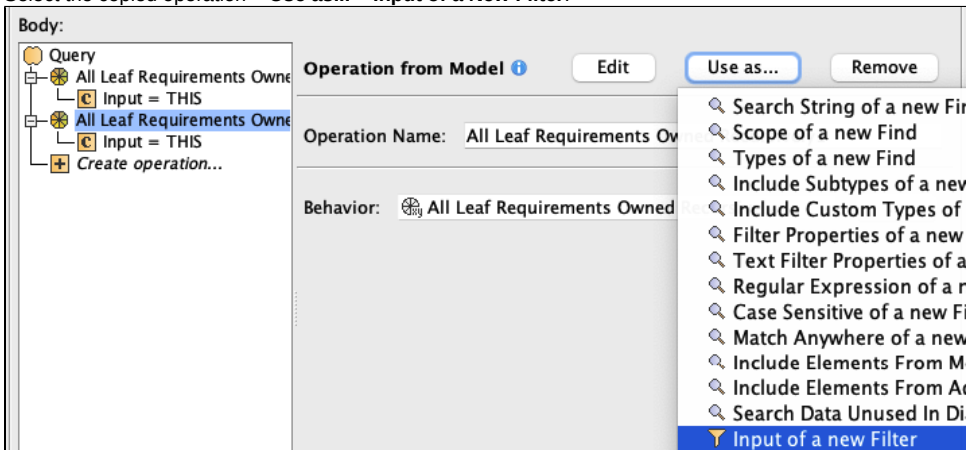
10. Select **All Leaf Requirements Owned Recursively > Edit > Copy**.



11. Select the root node **Query > Edit > Paste**.



12. Select the copied operation > **Use as... > Input of a New Filter**.



13. Select Predicate > Nested Operation.

Query

All Leaf Requirements Owned Re

Input = THIS

Filter

Input = All Leaf Requirement

Predicate = null

Create operation...

Filter::Predicate

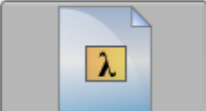
Operations

Type Test

Property Test

Operation from Model

Script

Nested Operation

14. Select Body > Operation from Model > Not.

Body:

Query

All Leaf Requirements Owned Re

Input = THIS

Filter

Input = All Leaf Requirement

Predicate = Nested Operation

Body = null

arg

Create parameter...

Create operation...

Body

Operations

Simple Navigation

Metachain Navigation

Find

Filter

Type Test

Property Test

Union

Exclude

Operation from Model

Script

15. Select A > Reset.

Body:

Query

All Leaf Requirements Owned Re

Input = THIS

Filter

Input = All Leaf Requirement

Predicate = Nested Operation

Body = Not1

A = false

arg

Create parameter...

Operation from Model::...

Edit

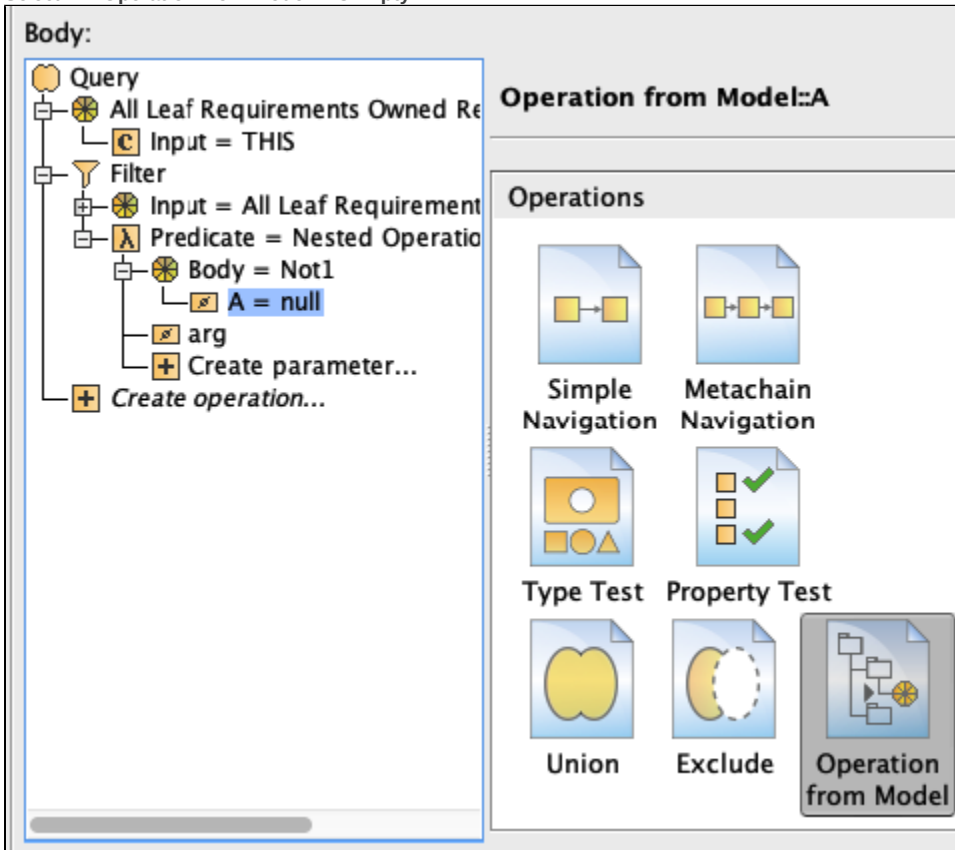
Use as...

Reset

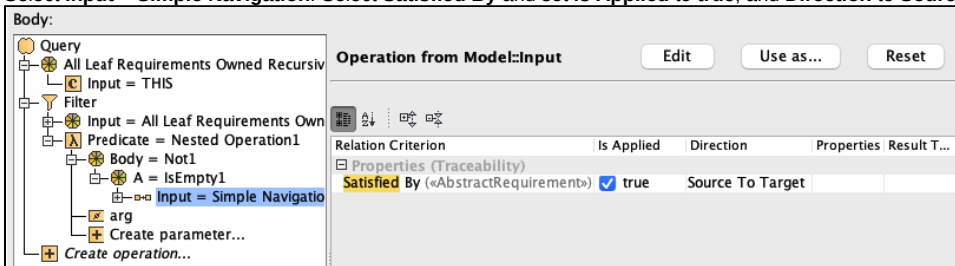
Operation Name:

Value: ☐ false

16. Select **A > Operation from Model > isEmpty**.



17. Select **Input > Simple Navigation**. Select **Satisfied By** and set **Is Applied** to **true**, and **Direction** to **Source To Target**.



18. Select **All Leaf Requirements Owned Recursively > Use as... > arg of a new Script**.

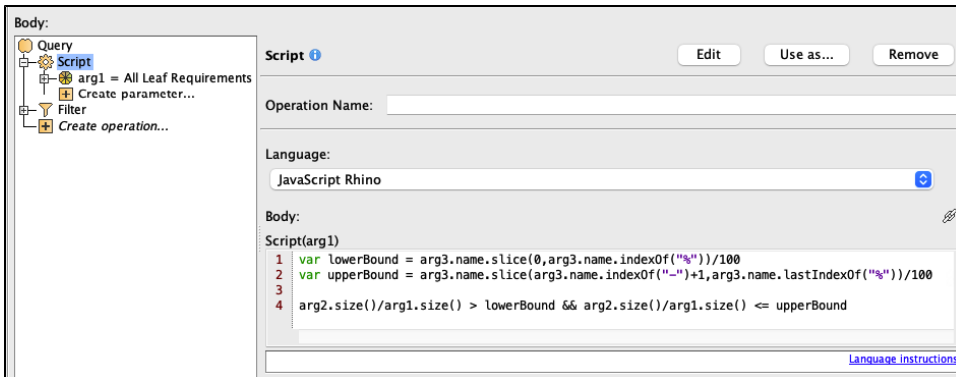
19. From the **Language** drop-down list, select **Javascript Rhino**.

20. Insert the following script as the **Body**:

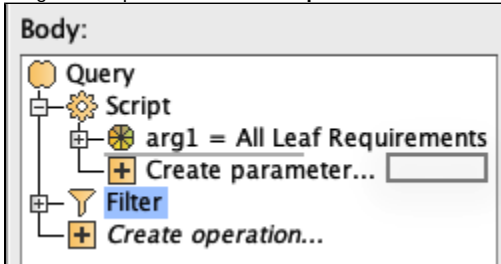
JavaScript Rhino

```
var lowerBound = arg3.name.slice(0,arg3.name.indexOf("%"))/100
var upperBound = arg3.name.slice(arg3.name.indexOf("-")+1,arg3.name.lastIndexOf("%"))/100

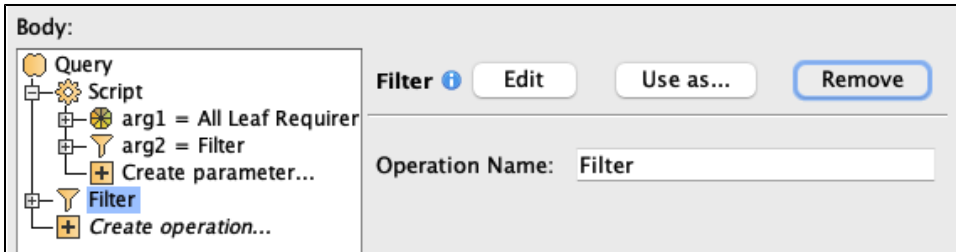
arg2.size()/arg1.size() > lowerBound && arg2.size()/arg1.size() <= upperBound
```



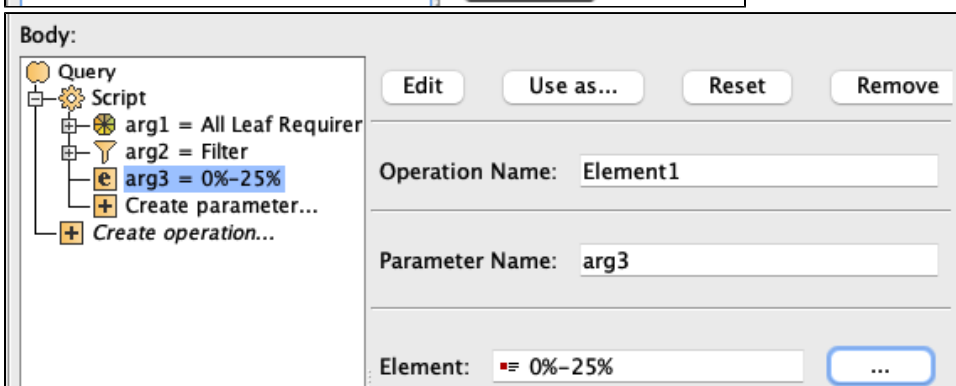
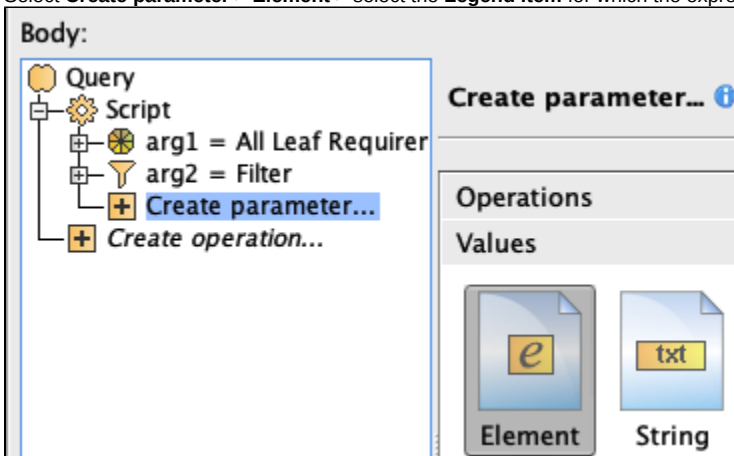
21. Drag-and-drop **Filter** onto **Create parameter** to create a second parameter for the **Script**.



22. Select **Filter** > **Remove**.



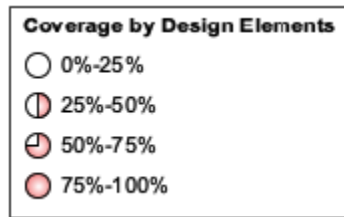
23. Select **Create parameter** > **Element** > select the **Legend Item** for which the expression is defined.





If you cannot see the **Legend Item**, make sure the **Search for** option is set to **Any Element**.

24. Clone the created **Legend Item** and rename the new items accordingly:
- a. 25%-50%
 - b. 50%-75%
 - c. 75%-100%



Sample model

The model used in these examples is the *Case Studies for Querying the Model* sample model. To open this model, you need to download [case studies for querying the model.mdzip](#).