Template variables

The variables imported to the template are collected by the type of element selected in the package scope. Use the fourth step in the **Report Wizard** dialog to select the package scope. In this example, we take a class diagram with the class name Customer, see figure below, to print it in a report.

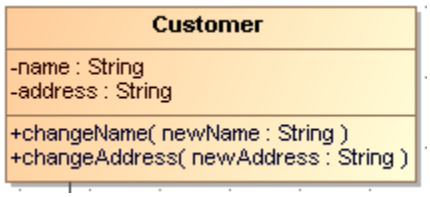


Figure 1: Class Diagram: Customer.

How to print attributes of the Customer class in a report

- Right-click the **Customer** class diagram and open the **Specification** dialog. The element type will appear on the dialog title, and the attribute names will appear on the right-hand side of the dialog box (see the following figure).

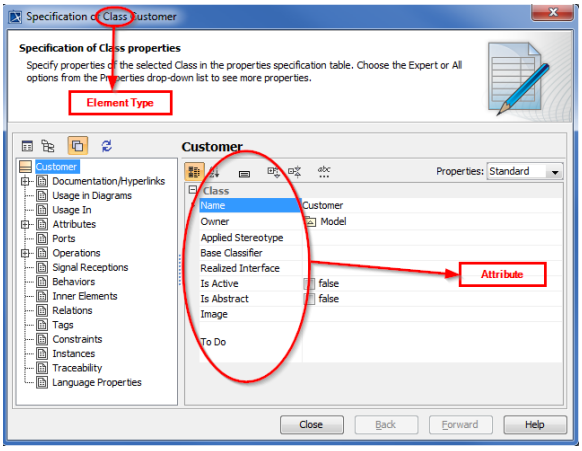


Figure 2: Specification Dialog.

With the following VTL code, you can print the attributes of the Class element:

```
#foreach ($class in $Class)
   Name: $class.name
   Owner:  $class.owner.name
   Visibility: $class.visibility
   Is Abstract: $class.isAbstract
#end
```

The output will be as follows:

```
Name: Customer
Owner:  com
Visibility: public
Is Abstract: false
```

The table below gives and explains the additional properties which are not part of the UML specifications, but retrievable by Report Wizard.

Element Owner	Property Name	Function
Diagram	image	To return the diagram's image.
Diagram	diagramType	To return the diagram's type.

Related pages

- Getting property value
- Retrieving Slot Values

Element	image	To generate the element's image or to print an empty text if the element does not refer to any diagram.
Element	elementType	To return a name of an element's type (metaclass / stereotype) in lowercase letters and without spaces. For example: • return "usecase" for a Use Case • return "class" for a Class • return "callbehavioraction" for a Call Behavior Action • return "flowport" for a Flow Port (a stereotype defined in SysML)
Element	humanName	To return an element's human readable name, which is a string concatenation between the human element type and the element's name.
Element	humanType	To return an element's type in a human readable format.
Element	appliedStereotype	To return an applied stereotype of an element.
Element	activeHyperlink	To return an active hyperlink of an element.
Element	hyperlinks	To return all hyperlinks attached to an element.
Element	toDo	To return an element's ToDo attribute in text format.
Element	elementID	To return an element's ID.
Element	documentation	To return an element's comment/documentation.
Element	presentation	Element To return a presentation element.
Element	typeModifier	To return an element's type modifier.
Element	tags	To return a list of element tags.
Element	text	To return a text representation of an element. In general, a 'text' property returns a string which "textually represents" the element. The string may vary depending on the symbol properties and environment options. This property returns the result from OpenAPI. `RepresentationTextCreator.getRepresentedText (Element element)`
Element	slots	To return a list of element's slots.
Element	elementURL	To return an element's URL.
Classifier	baseClassifier	To return a base classifier of an element.
BehavioredClassifier	realizedInterface	To return a Classifier's Realized Interface.
Package	appliedProfile	To return a list of profiles applied to a package.
Stereotype	metaClass	To return a Meta Class of a stereotype.
DurationConstraint	min	To return a minimum value of the Duration Constraint.

DurationConstraint	max	To return a maximum value of the Duration Constraint.
TimeConstraint	min	To return a minimum value of the Time Constraint.
TimeConstraint	max	To return a maximum value of the Time Constraint.
MultiplicityElement	multiplicity	To return a multiplicity value.
Property	navigable	To return a navigable value.
Property	ownedBy	To return a property's owner.
Lifeline	type	To return a lifeline type.
Message	event	To return a message event.
Message	operation	To return a message operation.
Message	signal	To return a message signal.
Message	number	To return a message number.
Trigger	eventType	To trigger an event type.
RequirementsUseCase	number	To return a use case number.

Some predefined variables, which are not part of the UML specifications, are usable in the templates:

- ***$elements***
  contains a list of all the elements selected from the element scope.
- ***$packageScope***
  contains a selected package from the element scope.
- ***$elementScope***
  contains selected elements from the element scope.
- ***$empty***
  contains a String for empty value, which is specified from the Output Options pane.

humanName and ***humanType*** are locale-specific text. If you plan to generate a report in different languages, try to avoid ***humanType*** and ***humanName***. Instead, use ***elementType*** for a non-locale-specific report.