Creating and specifying derived properties

To be included in a MagicDraw project as a property extending a UML element, an attribute must be owned by a DSL customization element and stereotyped as «derivedPropertySpecification».
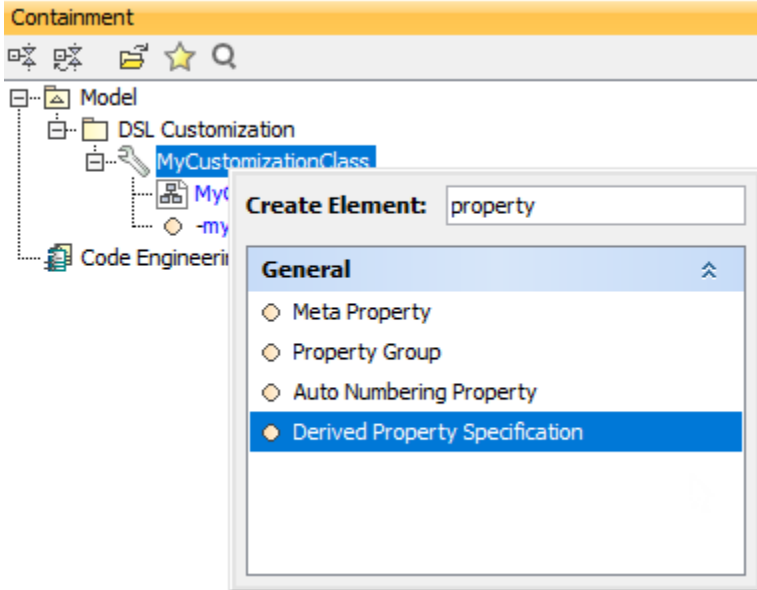
To create and specify a derived property

1. Select the customization element for which you want to create a derived property.
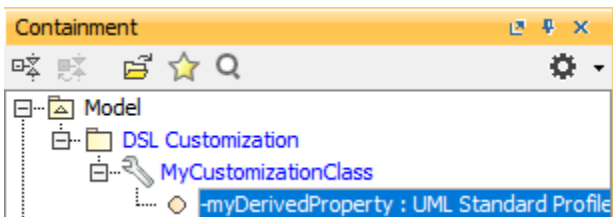
   ⚠ The derived property will be created for the stereotype or metaclass, defined as a customization target in the selected customization element specification.

2. Select **Create Element** > **Derived Property Specification** from the shortcut menu. The attribute is created.

3. Type the attribute name. Your Model Browser should look like this at this stage:

   ⚠ The attribute's name corresponds to the name of the derived property.

   Type derived property names in camel case (i.e., *realizingActivities*).

4. Edit property values specifying the derived property in the attribute's Specification window.

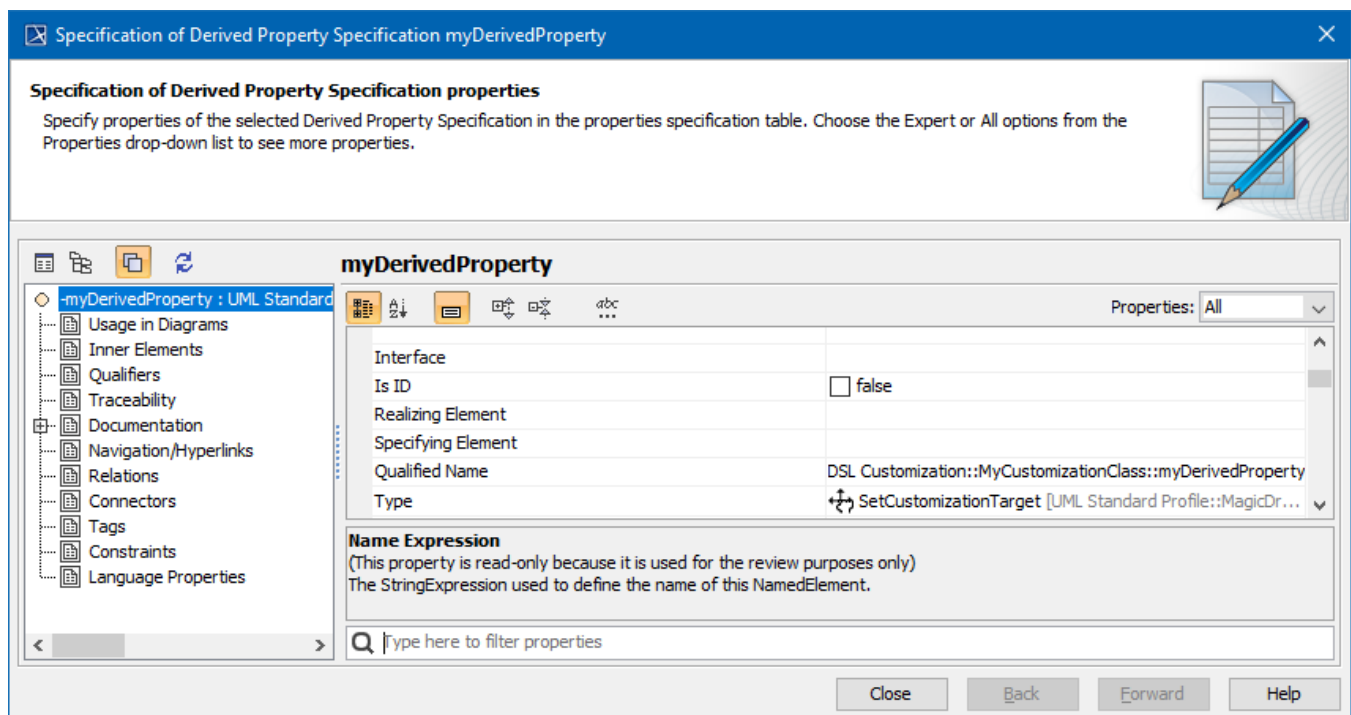   ⚠ All properties are specified in the general specification pane. Some are also specified in the **Tags** specification pane. For detailed information, see the properties description table.

⊘

The following table describes the properties specified for the derived property.

Property	Description
Name	The name of the property. This name will be added and visible in the specification of the element whose type is defined as a customization target.

Type	The type of the derived property values. You may set a model element type or a data type (e.g., string, boolean, or integer from the UML Standard Profile) as a property value. To display classes and components as property values, select the classifier as a type for the derived property value.
Multiplicity	The multiplicity of the derived property values. If the multiplicity is "0..1", the derived property value will be shown as single. If the multiplicity is "0..*", the derived property values will be shown as collection.
Is Read Only	If *true*, the derived property is read-only. You cannot edit the result elements for the derived property value. If *false*, the property is read-write. You can edit the result elements for the derived property according to the expressions defined.
Is Ordered	If *true*, the property values are always displayed in the same order. If *false*, the property values are displayed in nonpredictable order.
Is Unique	If *true*, the property values are unique. The same element is displayed only once. If *false*, the property values may be displayed more than once.
Expression	One or more expressions of the derived property. An expression defines the criterion for selecting result elements. There are ten possible expression types: • Metachain navigation (indirect relations through chains of properties) • Find • Union • Exclude • Filter • Various scripts • Implied Relation • Type Test • Property Test • Operation from Model The criterion for selecting the result elements can also be any combination of the expression types above. For the detailed information, see Defining expressions. Only one Simple Navigation operation criterion can be assigned to the expression of the derived property.
Documentation	The text displayed as the derived property description in the element's Specification window and Properties panel.
customization Target	The specified derived property will be added to one or more element types (stereotypes and/ or metaclasses). The expression calculations begin from the element type of the customization target.

If you do not see the newly created derived property in the element's Specification window, please read the following:

- In the element's Specification window, the specified derived property is invisible by default, i.e., in the **All** properties mode. First, you must switch to this mode.

- If you have specified the derived property for the element (with the **Used UML Properties** property of DSL customization specified) you must add the newly created derived property to the element's visible properties list. For detailed information, please refer to Derived properties visibility.

The Specification window of the derived property

Related pages:

- Extending Metamodel with Derived Properties
- Using Customization Data
- Creating Customization Data
- Defining Expressions