Specifying criteria for querying model

Specifying criteria is necessary to

- Associate row and column elements in a dependency matrix.
- Display relations between elements in a relation map.
- Gather the contents of a smart package.
- Calculate/gather the value of a derived property.

A structured expression can also be the body of the executable opaque behavior, metric definition or validation rule, when the StructuredExpression language is selected for specifying that body.

Use one of the following operation types to specify the criterion:

- Using Simple Navigation
- Using Metachain Navigation (indirect relations through chains of properties)
- Find
- Filter
- Union
- Exclude
- Using Implied Relations
- Type Test
- Property Test
- Operations from the model
- Various scripts

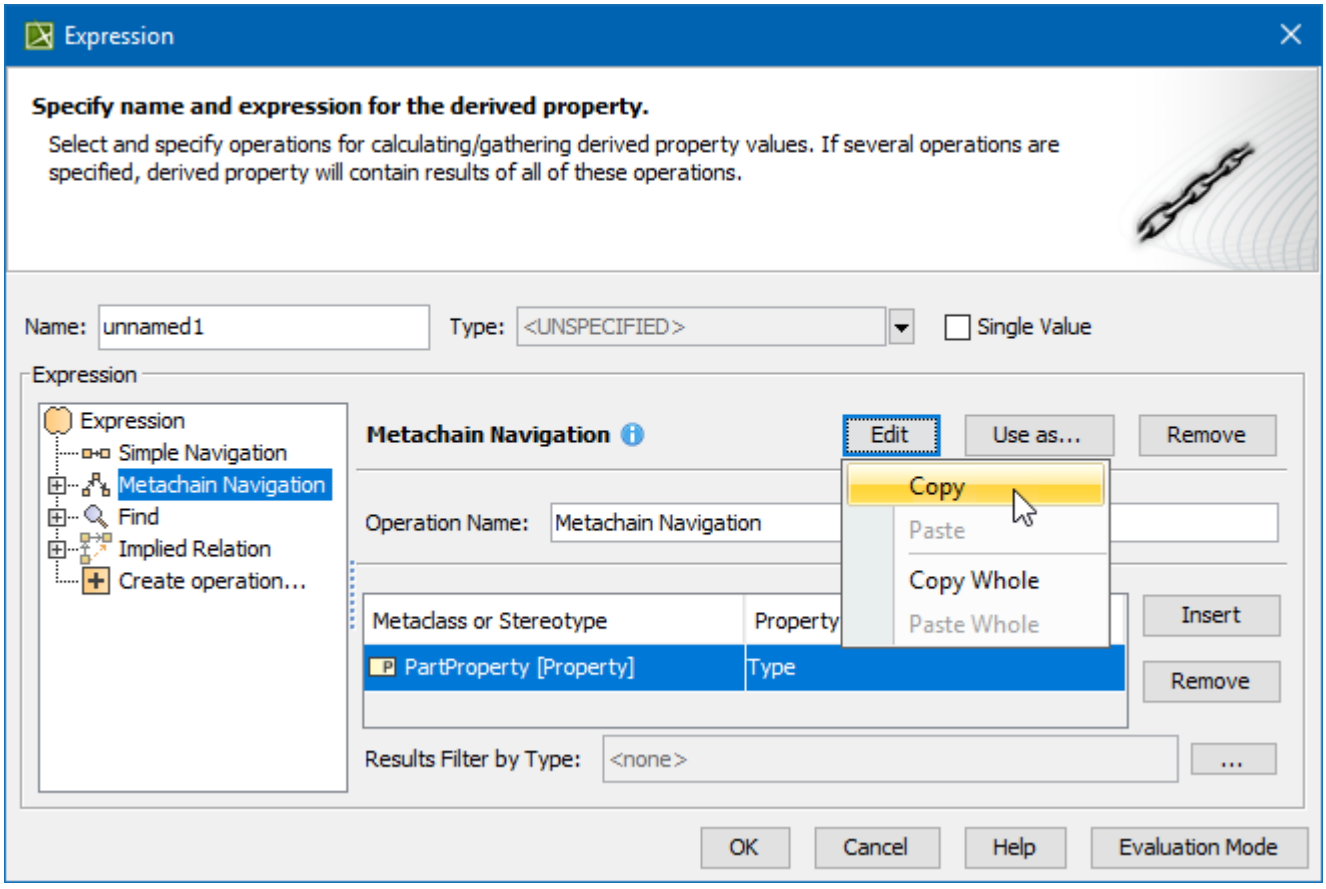
You can also use custom operation types. In addition, a criterion can be an expression combining several operations by passing the results of one operation call to another operation.

If multiple criteria (operations) are specified, the result will be calculated according to all these criteria (operations).

You can copy and paste the selected structured expressions in the **Expression** dialog. Select all or part of the structured expression in one dialog and paste it into the same or another dialog.

You can also copy the structured expression:

- from one smart package and paste it into another;
- from the matrices and paste them into the relation maps and vice versa;
- and use it in the smart package.



Concepts

Read the following description to better understand the material.

Concept	Description
Contextual element	The element that begins the calculation of the result according to the specified criteria. In a dependency matrix, the context is a row or column element. In a relation map, this is every node of the structure. In a table, the context is the table scope value. A smart package is the context itself. The context of a derived property is the target of the customization element that owns this derived property.