

# Logical expression

A logical expression is an expression that uses comparison operators. You can use a logical expression to define a constraint. Cameo Simulation Toolkit uses the expression to validate values at runtime.



## Note

A logical expression must contain a comparison operator ( $=$ ,  $<$ ,  $>$ ,  $<=$ ,  $>=$ , and  $<>$ ) as shown in the following examples:

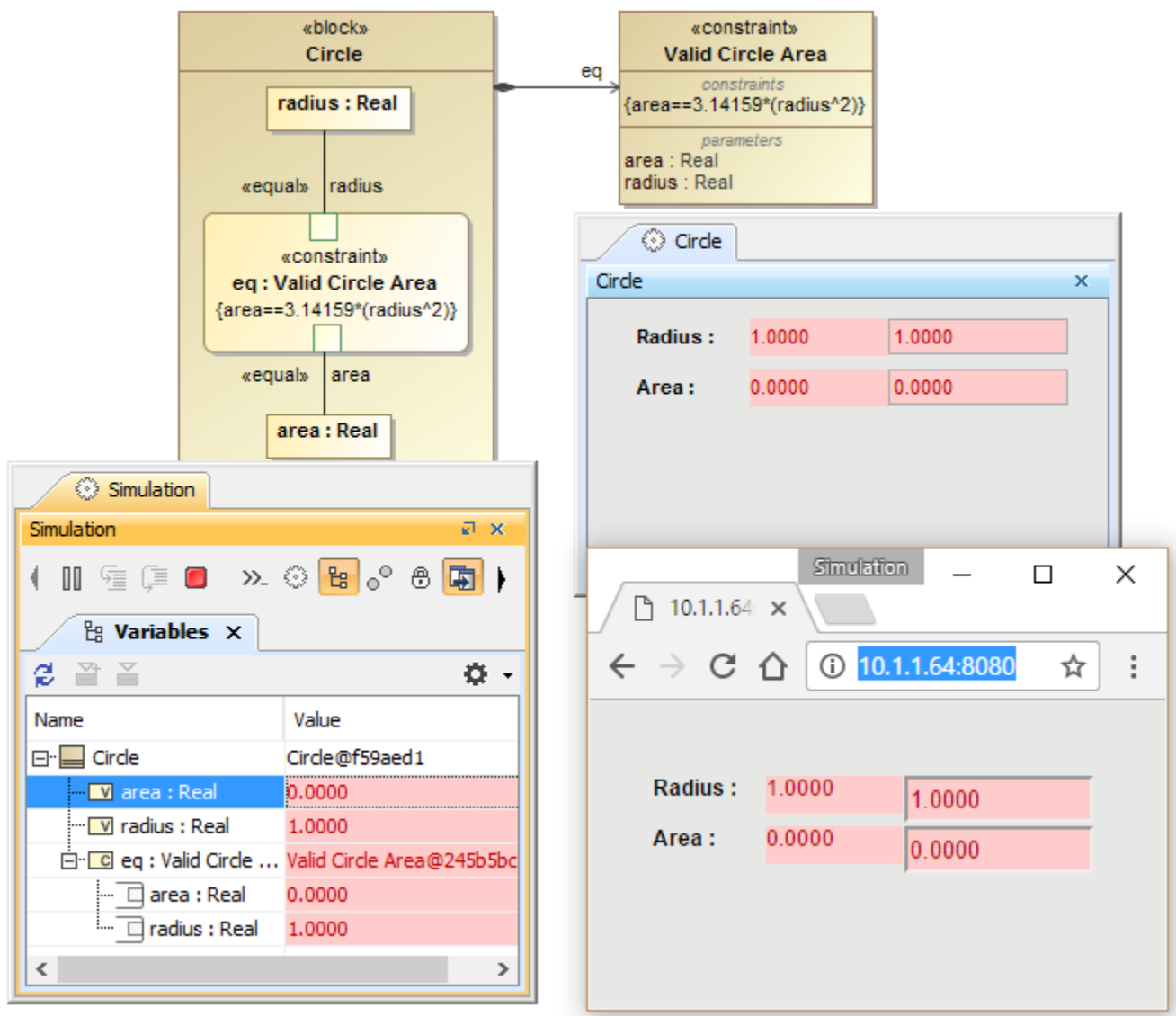
- `val1 >= "500"` is a valid expression.

## Red background for the labels and text fields in UI

- `val2 equals("500")` is an invalid expression.

If some constraints failed during evaluation, Cameo Simulation Toolkit (CST) will highlight the values in the **Variables** pane with a red background to denote that the values fail the constraint. All parent nodes (Name column) that contain the failed constraints will have a red background (until the root one in the tree view), which indicates failures deeper in the structure tree. You can also see these values in the **Simulation Console** pane.

Furthermore, both the values and validation that are highlighted must always be consistent with the labels and text boxes of both the [UI modeling diagram](#) and the HTML UI during simulation.



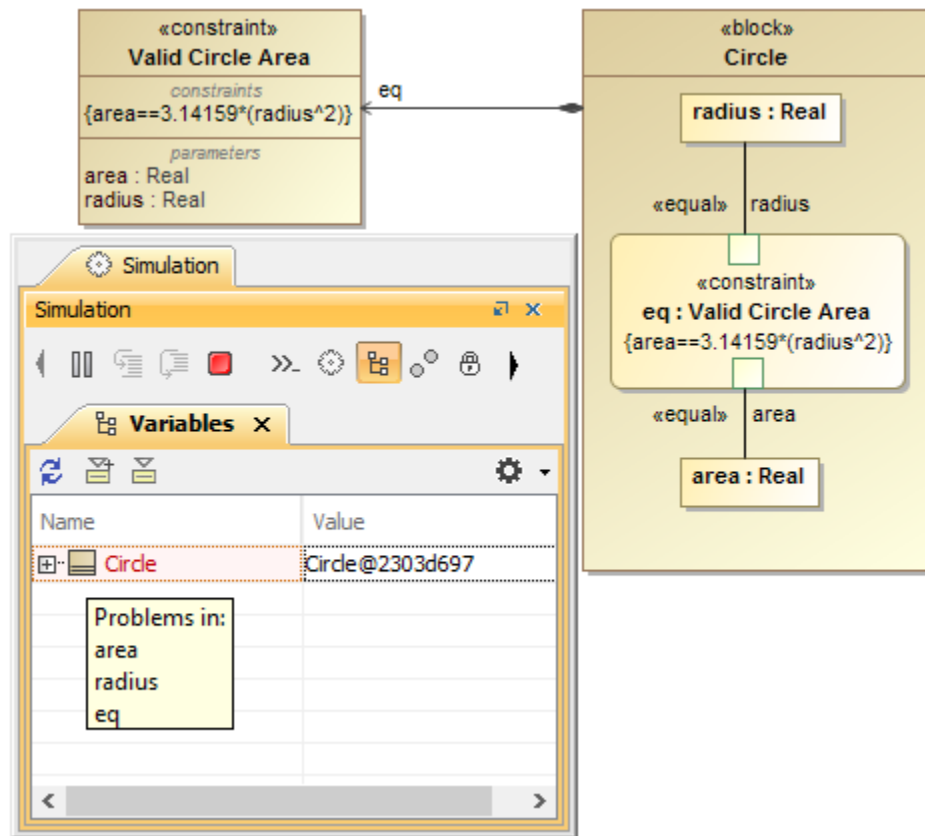
Red backgrounds indicating failures during evaluation.



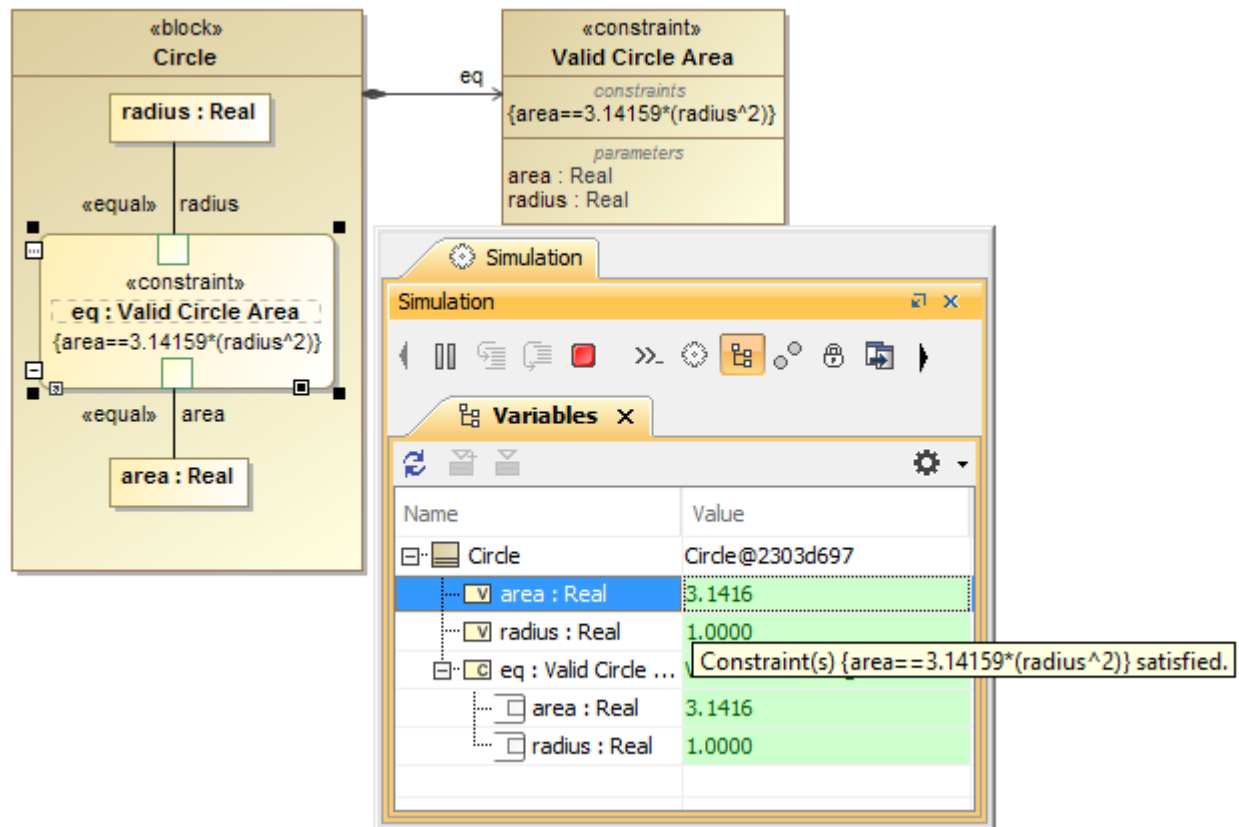
#### Note

Neither green background nor tooltip will be used for "passed" verification status for both Swing mockups and HTML UI.

A tooltip will also appear if you hover the mouse over the failed constraint. Each parent of the failed constraint will appear with a contextual path "dot notation" starting from the node on which tooltip is shown.



The parent node with a red background indicates failures deeper in the structure tree.



Nodes with green background indicating valid values.

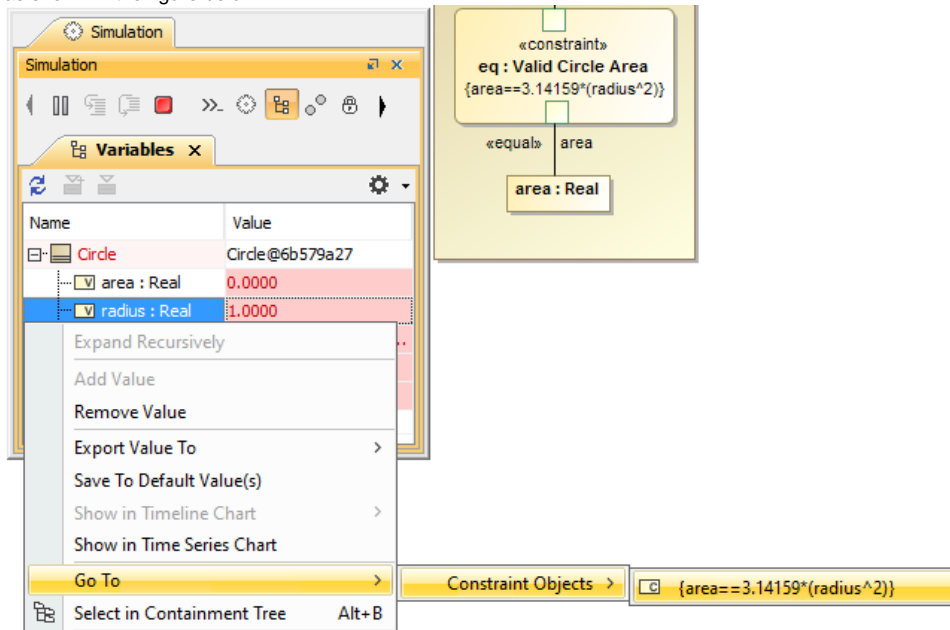
This example uses the [block Circle](#) previously shown in the section [Mathematical equation](#). If we modify the expression of the constraint block **Circle Area** by changing the assignment operator "=" to the equality operator "==", the parametric evaluator will use the expression to validate the values of both area and radius. If the result is false, the values will be highlighted in red (see Figure: The parent node with a red background indicates failures deeper in the structure tree). But, if the result is true, the values will be highlighted in green (see Figure: Nodes with green background indicating valid values).

## Finding a constraint object that binds the value property

You can find the constraint object to which the value of the property is bound in the **Variables** pane.

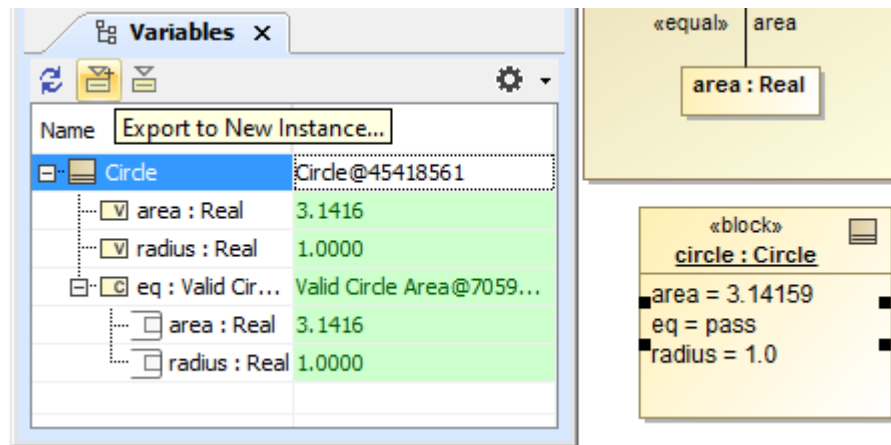
To find a constraint object to which a value property is bound

- Right-click a constraint object row in the **Variables** pane, and select **Go To**. The constraint to which the value of the property is bound will appear, as shown in the figure below.



Finding a constraint object that binds the value property through the shortcut menu.

When you export the object to the instance specification, if the exported object contains a constraint object (an object of a constraint block) and if the expression of the constraint block is a boolean expression, Cameo Simulation Toolkit will set the result of the evaluation, which is a literal boolean, to a slot defined by the constraint property typed by the constraint block.



Exporting an instance as a slot value of the constraint property.

## Related pages

- [Evaluating expressions](#)
- [Mathematical equation](#)