

Execution of incomplete or dummy models

On this page

- [AddStructuralFeatureValueAction](#)
- [ReadStructuralFeatureValueAction](#)
- [CallBehaviorAction](#)
- [CallOperationAction](#)
- [SendSignalAction](#)
- [ValueSpecificationAction](#)
- [StartClassifierBehaviorAction](#) and [StateObjectBehaviorAction](#)
- [OpaqueAction](#)

Previous versions of Cameo Simulation Toolkit did not support execution of incomplete or dummy models. From the 18.4 version and above, it will be possible to execute incomplete or dummy models.

Previously, the execution would halt. Now, if any input pin does not have an accompanying value, the Behavior of the Action will not be to halt execution, instead a null value token will be passed to that pin, or when simulating a **CallOperationAction** which does not have a target pin, a runtime object, which is the context of the Activity will be selected as the target. Null Value tokens do not trigger flow animation on triggered behaviors if the input pin multiplicity's lower bound is 0.

The information on this page describes the Behavior of an Action when any of the input or output pins, or any properties of the Action, are not specified or not present.



Supported Actions

The Actions which are currently supported are as follows

- [AddStructuralFeatureValueAction](#)

AddStructuralFeatureValueAction

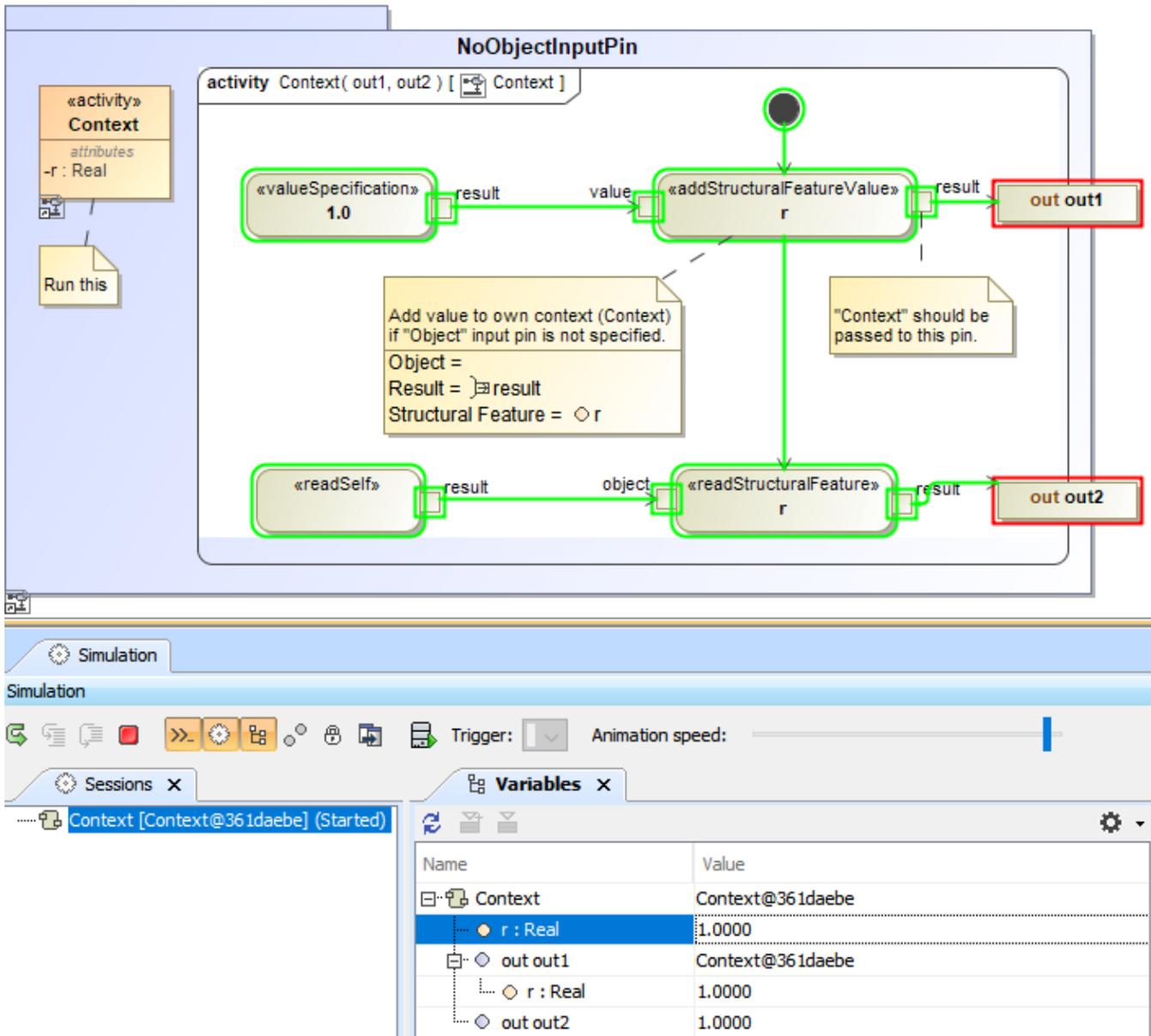
- [CallBehaviorAction](#)

An add structural feature value Action is a write structural feature Action for adding values to a structural feature. When any input pin is unspecified or empty, or when **Structural Feature** is unspecified or does not exist, this Action's Behavior will have the rules as follows

- [SendSignalAction](#)
- If **Object** input pin is not specified, the runtime context of the Activity execution will be used as the target.
- If **Value** input pin is not specified, this Action will be skipped.
- If **Structural Feature** is not specified or it does not exist in its own context, then this Action will be skipped.
- [StartClassifierBehaviorAction](#) and [StateObjectBehaviorAction](#)

- [OpaqueAction](#)

If an **Object** input pin is empty or unspecified, the runtime context of the Activity execution will be used to add the value to the **Structural Feature** of the context.

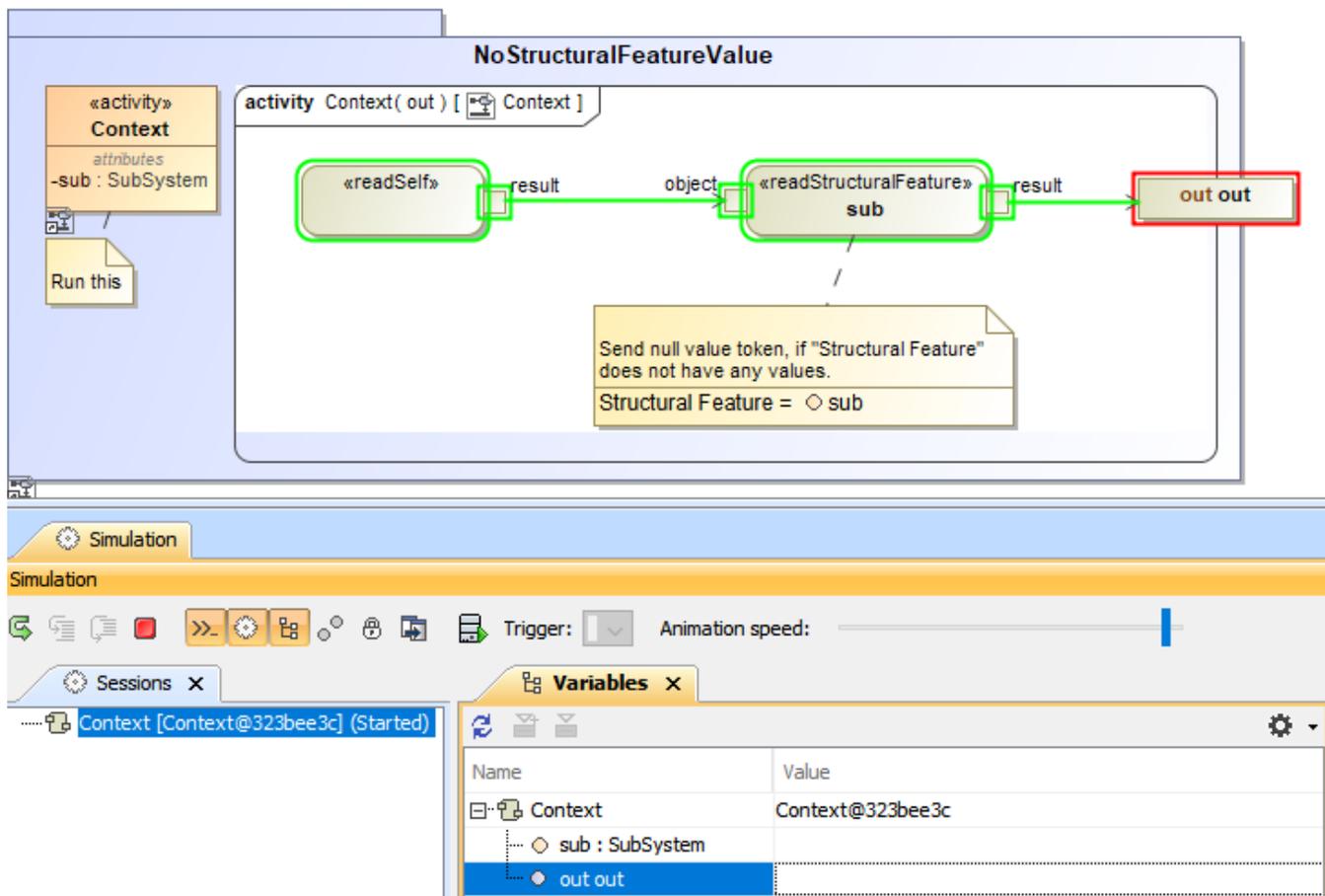


A demo showing the situation if an object pin is unspecified.

ReadStructuralFeatureValueAction

A read structural feature Action is a structural feature Action that retrieves the values of a structural feature. When any of the input or output pins are unspecified or not present, this Action's Behavior will have the rules as follows

- If the **Object** pin is not specified, the runtime context of the Activity execution will be used as the target.
- If **Structural Feature** is not specified or if the **Structural Feature** has no values, then a **null** value token will be sent.

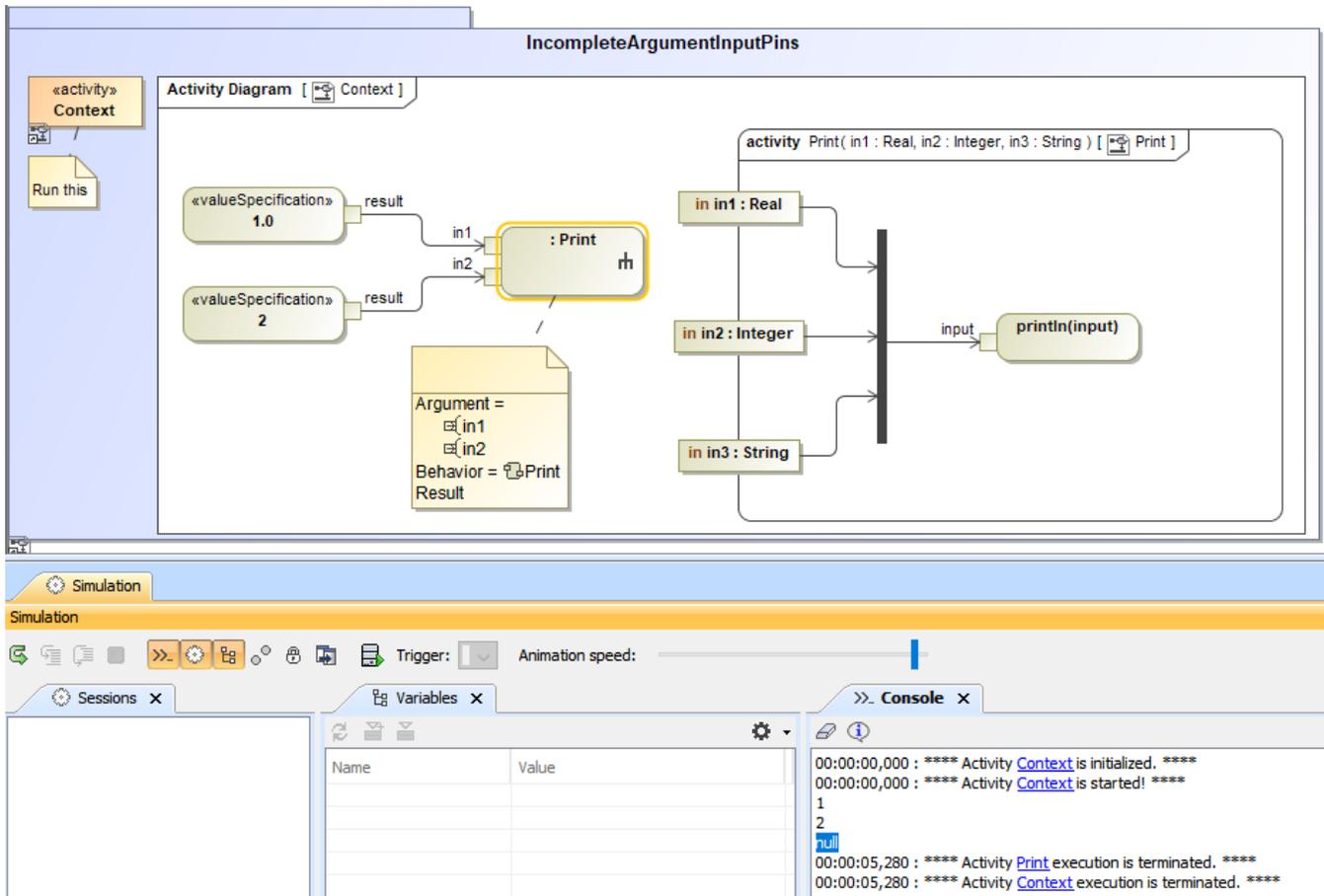


A demo showing the situation if an object pin is not specified.

CallBehaviorAction

Call Behavior Action is a call Action that invokes a Behavior directly rather than invoking an operation that invokes the Behavior. When any of the **Argument** pins are either unspecified or not present, this Action's Behavior will have the rules as follows

- If any of the **Argument** input pins are unspecified, this Action will send a **null** value token for that parameter.
- If none of the **Argument** pins are unspecified, this Action will send **null** value tokens for all the **Arguments**.
- If the **Result** output pin is specified and this Action does not return any results, it will return a **null** value token instead.

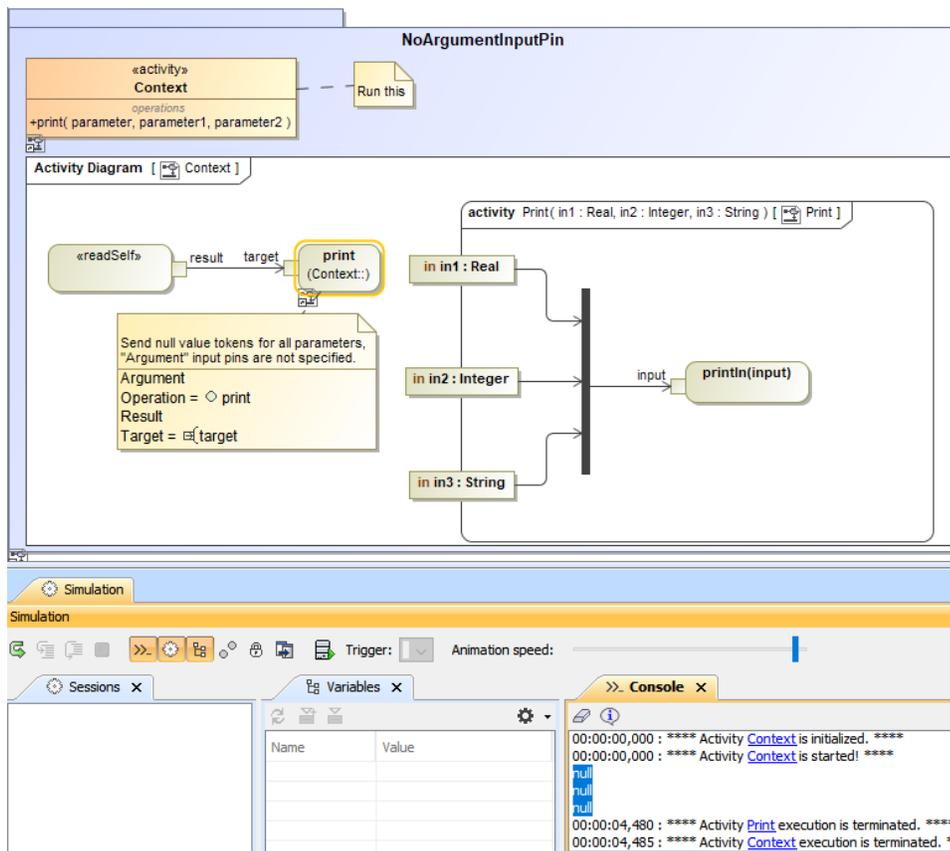


A demo showing the situation if an argument pin is unspecified (A null value token is passed to the pin.).

CallOperationAction

A call operation Action is an Action that transmits an operation call request to the target object, where it may cause the invocation of associated Behavior. When any input or output pin is unspecified or not present, this Action's Behavior will have the rules as follows

- This Action will send a null value token for each unspecified **Argument** input pin.
- In case no **Argument** pin is specified, this Action will send **null** value tokens for all the parameters.
- If the **Result** output pin is specified and this Action does not return any results, it will return a **null** value token instead.
- If the **Target** input pin is not specified, the runtime context of the Activity execution will be used as the target.

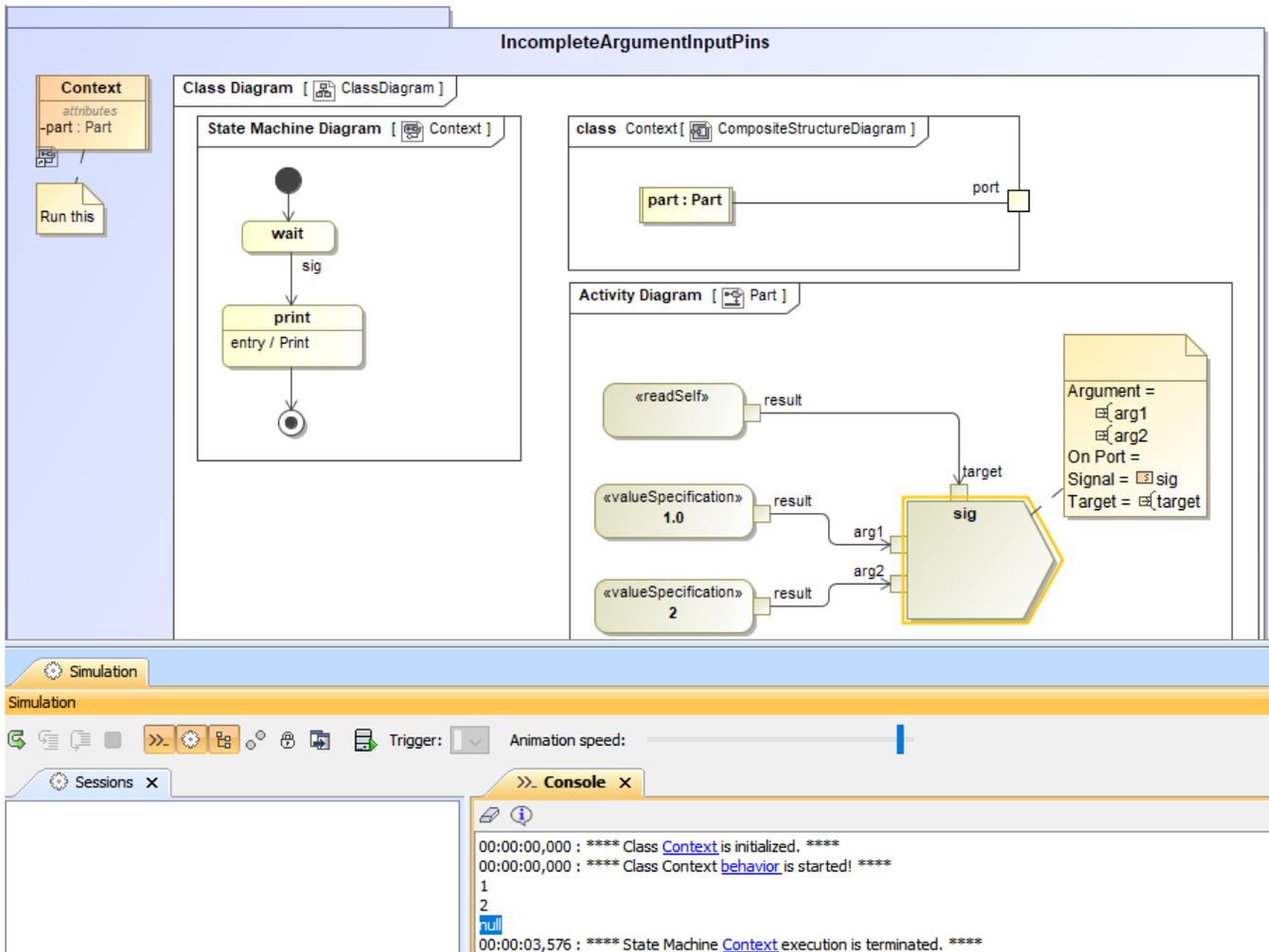


A demo showing the situation if an argument input pin is unspecified (A null value token will be passed.).

SendSignalAction

Send signal Action is an invocation Action that creates a signal from its inputs, and transmits it to the specified target object, where it may cause the firing of a State Machine transition or the execution of an Activity. When either **Argument** pin or **Target** input pin is unspecified or empty, this Action's Behavior will have the rules as follows

- If there are any incomplete **Argument** pins, this Action will send null value tokens for those parameters.
- If **Argument** pin is not present, this Action will send null value tokens for all parameters.
- If **Target** input pin is not present, the runtime context of the Activity execution will be used as the target.

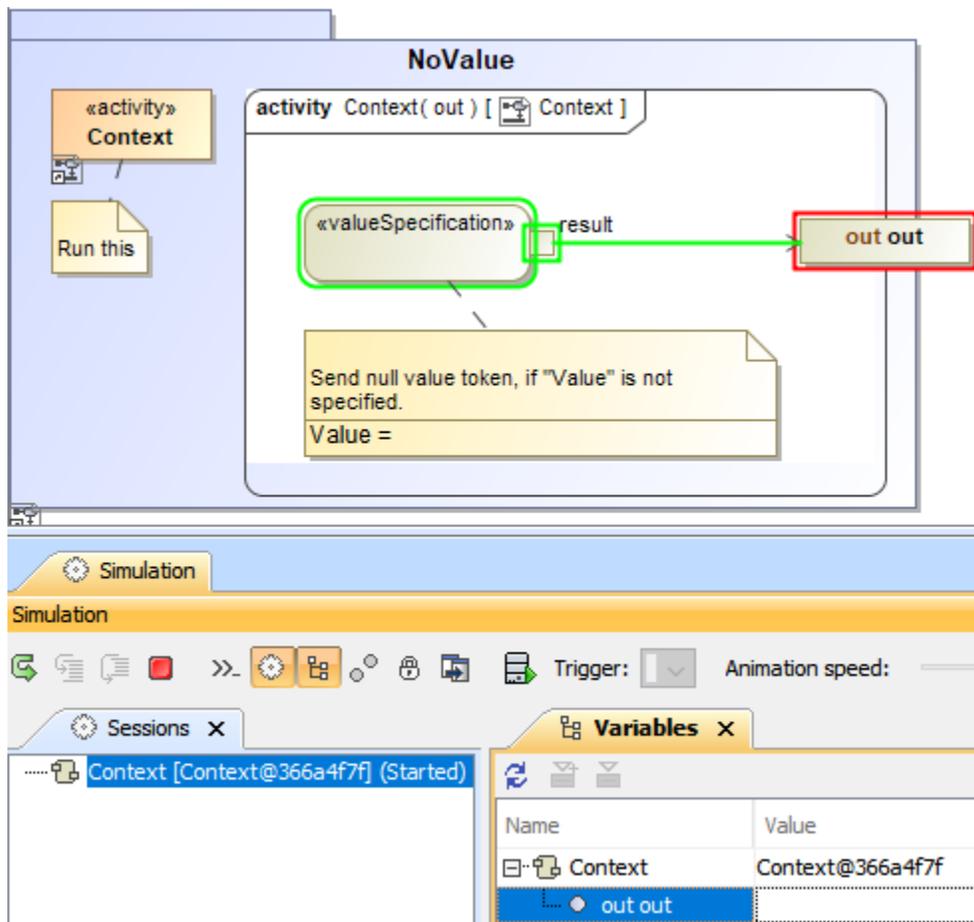


A demo showing the situation when there are incomplete argument pins.

ValueSpecificationAction

A value specification Action is an Action that evaluates a value specification. When the value of a **ValueSpecification** Action is not specified, this Action's Behavior will apply the rule as follows

- If the value of a **ValueSpecification** Action is not specified, a **null** value token will be passed.



A demo showing the situation when the value of a ValueSpecification Action is unspecified (A null value token is passed.).

StartClassifierBehaviorAction and StateObjectBehaviorAction

A start Classifier Behavior Action is an Action that starts the Classifier Behavior of the input. If the **Object** input pin is not present or empty, this Action's Behavior will apply the rule as follows

- If the **Object** input pin is not present or empty, the **StartClassifierBehaviorAction** Action will use the runtime context of the Activity execution as the target.

NoObjectInputPin

«SimulationConfig»
Run Context

```

«SimulationConfig»
autoStart = true
autostartActiveObjects = false
cloneReferences = false
constraintFailureAsBreakpoint = false
executionTarget = Context
fireValueChangeEvent = true
initializeReferences = false
runForksInParallel = true
silent = false
timeVariableName = "simtime"
treatAllClassifiersAsActive = true

```

Run this

```

«SimulationConfig»
autostartActiveObjects = false

```

Class Diagram [ClassDiagram]

```

classDiagram
    class Context {
        -part : Part
    }
    class Part {
        +start()
    }

```

Activity Diagram [Context]

```

graph TD
    Start(( )) --> Start[«start (Part:.)»]
    Start -- result --> ReadStruct[«readStructuralFeature» part]
    ReadStruct -- result --> ReadSelf[«readSelf»]

```

Activity Diagram [StartObjectBehavior]

```

graph TD
    StartObj[«startObjectBehavior»]
    Note[Use own context, if "Object" input pin is not specified.  
Object =]
    StartObj --- Note

```

Activity Diagram [Part]

```

graph TD
    Print[println("Part is stated.");]

```

Simulation

Simulation

Trigger: [v] Animation speed: []

Sessions x

- Context [Context@75a2d204] (Paused)
 - Context [Context@75a2d204] (Paused)

Variables x

Name	Value
Context	Context@75a2d204
part : Part	Part@67fc586e

Console x

```

00:00:00,000 : **** Class Context is initialized. ****
00:00:00,000 : **** Class Context behavior is started! ****
Part is stated.
00:00:01,866 : **** Activity Part execution is terminated. ****

```

A demo showing the situation when the object pin is either not present or empty.

OpaqueAction

Opaque Actions are a type of Action that can be used to represent implementation information. They can also be used as placeholder Actions until you determine the specific type of Action to use. If the opaque Action does not return a value to the **OutputValue** pin, this Action's Behavior will apply the rule as follows

- If the opaque Action does not return any value to the **OutputValue** pin, a **null** value token will be passed to the pin at runtime.

«activity» Context

activity Context(out) [Context]

print("just print");

outputValue

out out

if "Output Value" output pin is specified and this action does not return any results, send null value token.
Output Value = outputValue

Simulation

Simulation

Trigger: Animation speed:

Variables

Name	Value
Context	Context@6392...
out out	

Console

```

00:00:00,000 : **** Activity Context is initialized. ****
00:00:00,000 : **** Activity Context is started! ****
just print
00:00:00,738 : **** Activity Context execution is terminated. **

```

If an opaque Action does not return any value to the OutputValue pin, a null token will be passed to the pin.

Information "null value token" is an **ObjectToken** with has no values defined.