

# Context, runtime object, and runtime value

## On this page

- [Context](#)
- [Runtime object](#)
- [Runtime value](#)
- [Glossary](#)
- [Sample model](#)

When you are executing a model simulation, Cameo Simulation Toolkit creates the context, runtime objects, and runtime values to store the simulated values of the model.

## Context

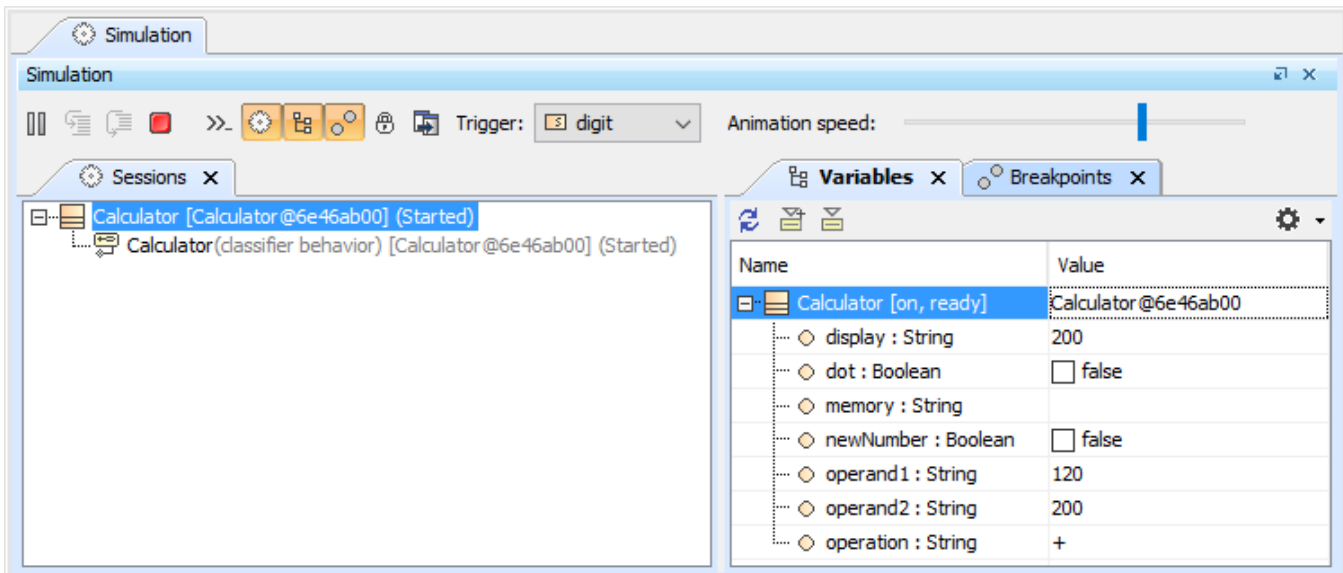
A simulation session is always associated with its context of simulation. The context of a simulation session is a Class or one of its subtypes. When a context element is simulated, a runtime object (of the context's type) will be created to store the runtime values. In the following figure, the context of the selected simulation session is the Calculator class.

## Runtime object

A runtime object is the simulated value of a Class. In other words, it is a runtime instance of a Class and of the context as well. In the following figure, the runtime object of the simulation session context is the "Calculator@6e46ab00" instance. Since the runtime instance is the "Calculator" Class type, it can contain structural features (which correspond to the Class attributes), such as "display" and "operand1".

## Runtime value

A runtime value refers to the value of the structural features mentioned in the Runtime object section, e.g., "200" and "120". However, if the type of a structural feature is a Classifier, its runtime value can also refer to another runtime object of a structural feature type.



The Simulation console showing the context "Class", the runtime object of the simulation session context "Calculator@6e46ab00", and the runtime values of "120" and "200".

## Glossary

Class is a Classifier that describes a set of objects that share the same features, constraints, and semantics (meaning). A class is the most widely used Classifier. It is shown as a solid-outline rectangle containing the Class name and with optional compartments separated by horizontal lines containing features or other members of the Classifier.

Classifier is a category of UML elements that have some common features, e.g., attributes and methods. It describes a set of instances that have common behavioral and structural features (operations and attributes respectively).

Element is the abstract root UML metaclass, it has no superclass in the hierarchy of UML elements. It is the superclass for all metaclasses in the UML infrastructure library.

Instance is a system's entity; it is a concrete manifestation (implementation) of an abstraction. Abstraction could be represented by one or more Classifiers or no Classifiers at all. When an instance manifests a Classifier, it is called instance of that Classifier. For example, Object is an instance of a Class, while the link is an instance of an association. An instance could have a name or be anonymous when the name is not important. The instance name allows you to distinguish it from the other instances within the same naming context (scope). Object is an instance of a Class. It is an individual (thing) with a State and relationships to other objects. The State of an object identifies the values for that object of properties of the Classifier of the object. Structural feature is a typed feature of a Classifier that specifies the structure of instances of the classifier. It specifies that instances of the featuring classifier have a slot whose value or values are of a specified type.

## Sample model



### Related pages

To learn more about working with context, runtime objects, and runtime values, you can visit the following pages

- [Creating values and objects in the variables pane](#)
- [Creating runtime objects from Classifiers](#)
- [Creating runtime objects from Instance Specifications](#)
- [Automatic initialization of context and runtime objects](#)
- [Exporting runtime objects to InstanceSpecifications](#)
- [Recording Verification status of runtime values](#)