# **Automatic Instantiation wizard**

### On this page

- Introduction
- Using the Automatic Instantiation wizard

## Introduction

The Automatic Instantiation wizard analyzes the structure of a selected element and collects all available information about that element from the model (attributes, properties, and so on). It evaluates collected information and suggests all possible instances to be created.

This tool allows you to create instances of various entities with just a few clicks. It is useful for working with complex models, and for assembling large systems from parts. For properties having a multiplicity of more than 1, you can create as many instances of the same type as you need, since parallel parts can be added while creating instances. While creating instances automatically, you can:

- Select properties for which the instances will be created.
- · Change types of the instantiated properties.
- · Change names of the instantiated properties.
- · Apply custom stereotypes to the instantiated properties.
- Assign default values for the instantiated properties.

#### The wizard consists of the following areas:

- Steps area: displays all steps of the wizard and shows the current step you are working on.
- Element tree: displays the entire internal structure of the selected element.
- Selection area: allows you to select the elements to be instantiated.
- Instantiated properties area: allows you to change the default values and create several different sets of instances.

	X Automatic Instantiation	×	
	Select parts Select parts to be instantiated. Hold down SHIFT and select a check box to select parts recursively. Change default values of primitive properties or select a subtype of the property in the editor below the tree. Right-click a tree node to create multiple values if the property multiplicity is more than one.		
Steps — area	<ul> <li>I. Select parts</li> <li>2. Select a package</li> <li>3. Create a diagram</li> <li>Wehide [Structure]</li> <li>Wehide [Structure::Engine [Structure::Vehide]</li> <li>WigrossMass : ISO80000-4 Mechanics::Qua</li> <li>WigrossMass : Iso8000-4 Mechanics::Qua</li> <li< td=""><td>antities::mass::mass[kilogram] = 1450.0 e::Vehide] {Value = 4} tructure::Vehide]</td><td>Element tree</td></li<></ul>	antities::mass::mass[kilogram] = 1450.0 e::Vehide] {Value = 4} tructure::Vehide]	Element tree
	<     Instantiated properties Type Value <back next=""></back>	Finish Cancel Help	_

#### Instantiated properties area

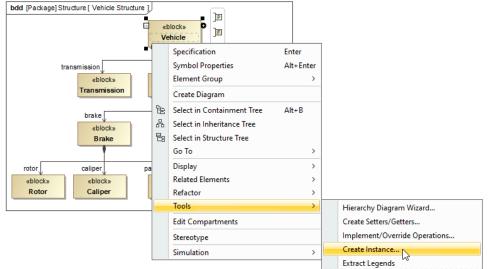
The Automatic Instantiation wizard of the Vehicle Block with the main areas highlighted: Steps area, Element tree, Selection area, and Instantiated properties area.

## Using the Automatic Instantiation wizard

To create and display instance specifications of Blocks in the diagram by using the Automatic Instantiation wizard

1. Select the Block for which you want to create instances.

2. From the shortcut menu, select **Tools > Create Instance** (see the following figure).



3. Follow the steps of the Automatic Instantiation wizard. How to work in each step of the Automatic Instantiation wizard >>

Wou are satisfied with the default values provided in the Automatic Instantiation wizard, you can finish the wizard in the first step without making any changes. The instances will be created with the default values and in your working package. To change default values or create

4. Click Finish, when you are done, read Steps of the Automatic Instantiation wizard. Instance Specifications and new SysML Block Definition Diagram are created for the selected Block (see the figure below). All part instances are grouped under their owners which allows you to visually identify the structure of nested instances in the Containment tree.

🔁 Containment 🗄 Structure 🏄 Diagrams	Instance of the vehicle X		
Containment La 🕫 🛪 🗙	◆ ※   18   10   10 · 1本・ホ・ノロフクス   分面白ま馬  ▲・Z・Δ・Arial		
🖻 🏠 Q 🌼 🗘 🗘			
🖶 - 🄁 Requirements [Requirements.mdzip]	[Package] Structure [Instance of the vehicle ] Shapes of instances		
SysML Block Definition			
Relations     System Diock Demittering     Diagram	«block»		
Brake Assembly	vehicle : Vehicle		
	engine = vehicle.engine		
	transmission = vehicle.transmission wheel = vehicle.wheel		
Wheel Assembly			
⊞ Brake	«block»		
	vehicle.wheel: Wheel whe		
	brake = vehicle.wheel.brake		
	diameter = 16.0 numAxleGearTeeth = 100.0		
🗄 🔚 Tire	numEngineGearTeeth = 50.0		
i∰			
É	vehicle.wheel.brake : Brake vehicle.wheel.tire : Tire vehicle.engine : Engine		
in (a) engine = vehicle.engine	caliper = vehicle.wheel.brake.caliper diameter = "22.0" engineRPM = 0.0		
E. E vehicle.engine : Structure::Engine	pad = vehicle.wheel.brake.pad tireMU = "0.9"		
⊡ ③ engineRPM = 0.0	rotor = vehicle.wheel.brake.rotor		
🖶 🛞 transmission = vehicle. transmission			
	eblock»		
L. 20.0	vehicle.wheel.brake.caliper		
⊡ ③ numAxleGearTeeth = 100.0	caliperFrictionForce = "140"{unit = newton} diameter = "0.038"{unit = metre}		
20. 100.0	partNumber = "		
🖃 🛞 numEngineGearTeeth = 50.0	pressure = "7.8"{unit = megapascal}		
	springForce = "250" {unit = newton}		
	eblocka		
	vehicle.wheel.brake.pad : Pad		
⊕	brakeMU = 0.6		
🗄 🕀 🔁 tire = vehicle.wheel.tire	centerLength = 0.08{unit = metre}		
	thickness = 0.01{unit = metre}		
Grouped instances	width = 0.038(unit = metre)		
	«block»		
	vehicle.wheel.brake.rotor : Rotor		
	outerDiameter = "0.25"{unit = metre}		

The Block Definition Diagram with instances of the vehicle Block is created by using the Automatic Instantiation wizard.

## Related pages

• Steps of the Automatic Instantiation wizard

## Sample model

The model used in the figures of this page is the **VehicleStructure** sample model. To open this sample, download *VehicleStructure.mdzip*.