

Modeling Activities

Using the Designer, behavior of classes can be defined by modeling an UML activity diagram.

Adding an Activity Operation

+ Base Types

+ Connectors

+ Process

+ Forms

API

- Implementation

+ Forms

- Ser

ASJS

+ SupportManagerService

Libraries

Add Activity Operation

From the quick actions of a class, select to add an activity operation.

On this Page:

- [Adding an Activity Operation](#)
- [Overview on the Activity Diagram Editor](#)
- [Attributes of an Activity Operation](#)
- [Attributes of an Activity Diagram](#)

Related Pages:

- [Working with the Activity Editor](#)
- [Changing the Attributes of Elements on the Activity Diagram](#)
- [Service Panel](#)
- [Working With Libraries](#)
- [Modeling Data Structures](#)
- [PAS Designer Developer Guide](#)
 - [Drawing an Activity Diagram](#)
 - [Supported UML Elements](#)

Add Activity Operation

checkSupportManager|

Save

Cancel

Enter a name for the operation and click **Save**.

Restrictions on Element Names

A BPMN model name must be unique within one service.

In addition, the following name restrictions apply to all service panel elements:

Element names ...

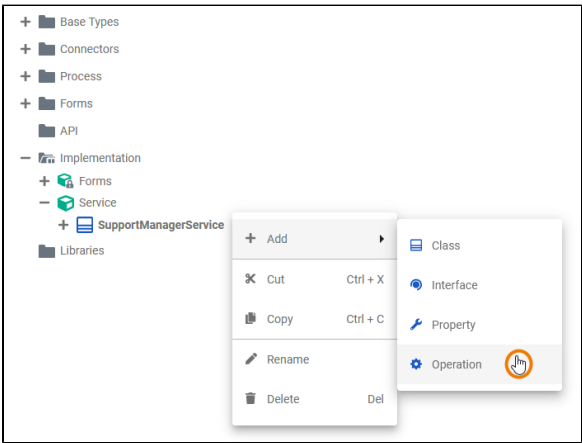
- .. must not be empty.
- .. must not contain spaces. Exception: Spaces are allowed in operation names.

- .. must not start with numbers.
- .. must not end with a period (.).

- .. must not contain one of the following characters : < , > , : , " , ' , / , \ , | , ? , * .
- Furthermore , the following strings must not be u


s	e	d	a	s	e	m	e	n	t	n	a	m	e	s	:	C	O	N	,	P	R	N	,	A	U	X	,	N	U	L	,	C	O	M	1	,	C	O	M	2	,	C	O	M	3	,	C	O	M	4	,	C	O	M	5	,	C	O	M	6	,	C	O	M	7	,	C	O	M	8	,	C	O	M	9	,	L	P
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

T
1
,
L
P
T
2
,
L
P
T
3
,
L
P
T
4
,
L
P
T
5
,
L
P
T
6
,
L
P
T
7
,
L
P
T
8
,
L
P
T
9.



Alternatively,
you can open
the context
menu of the
class and select
the option **Add
Operation**.

Add Operation

 Activity Diagram

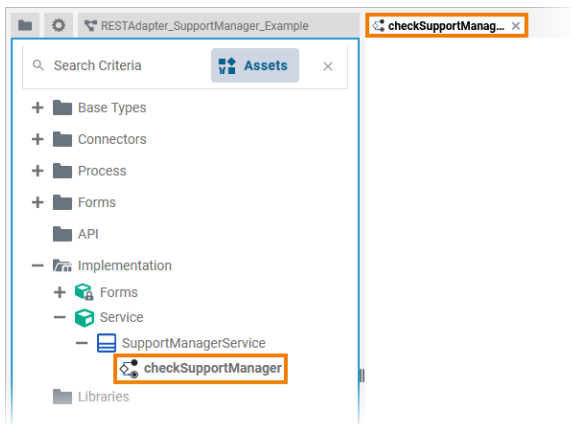
checkSupportManager

Save


Cancel

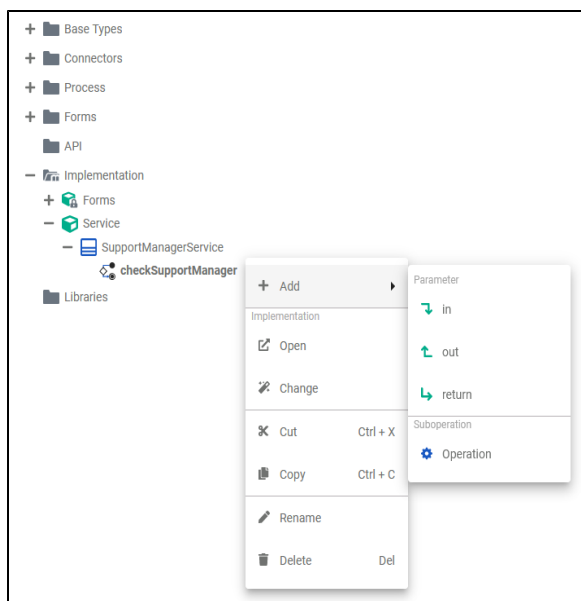
The dialog **Add Operation** opens.

Select **Activity Diagram** from the drop-down list, enter a name for the operation and click **Save**.



The new operation has been added to the class. It opens automatically in a new Designer tab, and you can directly start modeling your activity.

 Go to [Working with the Activity Editor](#) for detailed information.



Once the operation has been created, you can use the quick actions and the context menu to manage it.

You can:

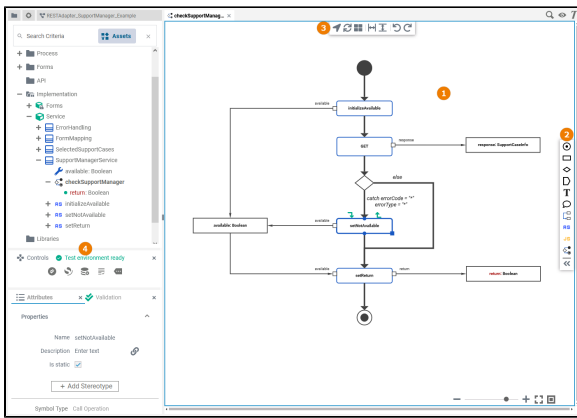
- add a parameter
 - in
 - out
 - return
- open the implementation of the activity diagram
- change the implementation from activity diagram to action script or mapping diagram
- cut the operation
- copy the operation
- paste the operation (available if **Copy** or **Cut** option have been used before)
- rename the operation
- delete the operation

✓ Refer to [Implementation and Modeling Data Structures](#) for more information on your options here.

Refer to [Modeling Data Structures](#) for more information on how to create classes and operations.

Overview on the Activity Diagram Editor

When you create a class operation that is implemented by an activity diagram, the activity diagram editor opens:



Use the various functionalities of the Activity Diagram Editor to model your activity:

Name	Description
1 Diagram Pane	The diagram pane is where you model your activity diagram. Go to Working with the Activity Editor for detailed information about the features of the activity diagram editor.
2 Elements Toolbar	The elements toolbar contains all UML elements that you can create on the diagram pane. Go to Working with the Activity Editor for further details about the usage of the toolbar. In the PAS Designer Developer Guide > Supported UML Elements all UML elements are explained in detail.
3 Model Toolbar	The model toolbar assists you during modeling on the diagram pane. Go to page Working with the Activity Editor for an overview of the available options of the model toolbar.
4 Designer Panels	In the activity diagram editor, the following panels assist you during modeling: <ul style="list-style-type: none"> • Service Panel: In the Service panel you can manage the contents of your whole service such as you BPMN and data model, forms and used libraries. Go to Service Panel and Working With Libraries for detailed information. • Attributes Panel: Use the Attributes panel to change the settings of the activity diagram elements and the diagram pane. Go to Changing the Attributes of Elements on the Activity Diagram for detailed information. • Validation Panel: The Validation panel supports you during modelling by displaying notes for invalid actions in your models or forms. It also gives advice on how to fix the errors. Go to Validating and Testing a Service for detailed information. • Overview Panel: In the Overview panel, your model and a blue frame are displayed, representing the content shown on the diagram pane. Use the panel to keep the overview of your BPMN model - and to navigate within. Go to Working with the Activity Editor for further information about the Overview panel.

Attributes of an Activity Operation

Select an activity operation in the **Implementation** folder of the **Service** panel to display its attributes in the **Attributes** panel. You can also edit them there.

Activity operations have the following attributes:

Attribute	Description	Possible Values / Example
-----------	-------------	---------------------------

RESTAdapter_SupportManager_Example

Search Criteria

Assets

+ Base Types

+ Connectors

+ Process

+ Forms

API

- Implementation

+ Forms

- Service

+ ErrorHandling

+ FormMapping

+ SelectedSupportCases

- SupportManagerService

- checkSupportManager

return: Boolean

Libraries

Attributes

Properties

Name

checkSupportManager

Description

Enter text

Is static

☒

+ Add Stereotype

Name	<div>Click here to change the Name of the related operation.</div> <div>Activity operation names must follow certain naming rules. They<ul style="list-style-type: none">must not contain blanksmust not start with a numbermust not contain special characters</div>	checkSupportManager
Description	<div>If you want to insert or change a description for the respective activity operation, click here to open a text editor where you can enter and format your text.</div>	
Is static	<div>Specify if the operation is static (default) or not.</div> <div><ul style="list-style-type: none">Static activity operations can be called without creating an instance of the related class. They get all necessary data via their input parameters.Wanting to call a non-static activity operation, you need to create a local instance of the related class, and call the operation on that object. This is called self context.</div> <div>For more information, also refer to Adding Operation Calls.</div>	<div><div>true</div><div>The activity operation is static (default) and can be used outside the context of the related class.</div></div> <div><div>false</div><div>The activity operation is non-static and needs a self object as an input.</div></div>

		Stereotype Via button Add Stereotype , you can add a stereotype to an activity operation. By adding a stereotype, you can extend the attributes of a activity operation with additional properties.	REST
--	--	--	------

Attributes of an Activity Diagram

<div> <div>☰ Attributes</div> <div>×</div> <div>Properties</div> <div>^</div> <div> <div>Name</div> <div>checkSupportManager</div> </div> <div> <div>Type</div> <div>Service.SupportMana...</div> </div> </div>	<p>When you click on the diagram pane in the Activity editor, the following attributes of the current activity diagram are displayed in the Attributes panel. All attributes are read-only and cannot be edited.</p>
---	---

Attribute	Description	Possible Values / Example
Name	Displays the name of the current activity diagram.	<code>checkSupportManager</code>
Type	Path within the implementation folder where the corresponding activity operation resides.	<code>Service.SupportManagerService</code>