_designer_excerpts_big

adding_a_role

> (i) **Used on:**
>
> - Lane
> - Modeling Roles

When you open the role editor, it already contains an empty line.

Start typing the name of the role your are seaching for in field **Role**. The field has an autocomplete function: As soon as you start inserting, a drop-down list is displayed containing all available roles that match your input.
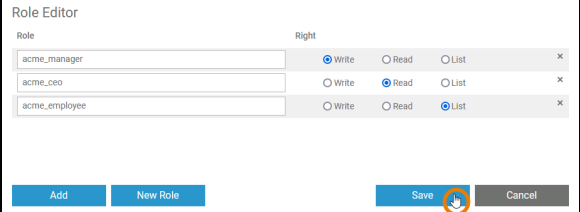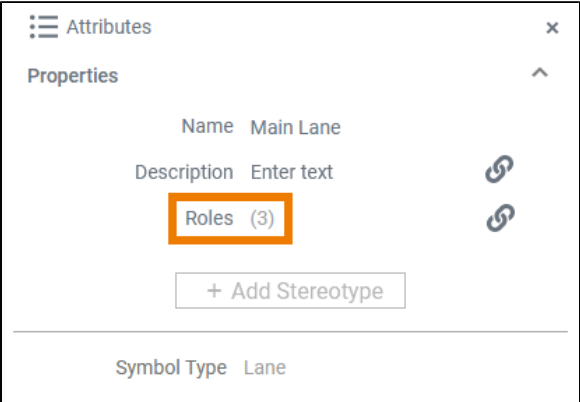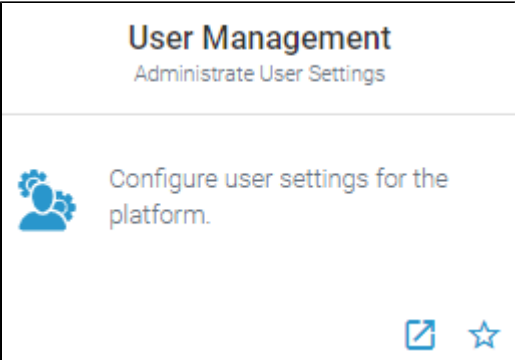
Select the role you want to use.

In the next step you need to assign the rights that will be granted to the role owner.

Available options are:

Right	Description
Write	A user assigned to this role will be able to see and change the instance data for all process steps in the current lane.
Read	A user assigned to this role will be able to display the instance data for all process steps in the current lane. This right is assigned by **default**.
List	A user assigned to this role will only be able to see the instances in the instance table for all process steps in the current lane.

Role Editor	You can continue to add further roles (or create new ones, see below) as multiple roles can be assigned to one lane. When you have finished, save your changes to apply them to the lane.
Attributes panel showing Properties: Name "Main Lane", Description "Enter text", Roles (3), + Add Stereotype, Symbol Type "Lane"	The number of roles assigned to a lane is displayed in the attributes panel.
User Management — Administrate User Settings. Configure user settings for the platform.	You have to open the user management, to assign the new role to users. ✓ Refer to Administration Guide > Managing Roles for further information about role management.

create_connector

ⓘ **Used on:**
- Connectors
- Single Connector
- Using Connectors

Hover with your mouse over the **Connectors** tree node in the service panel. Click the icon to create a new connector.

Alternatively, you can open the context menu of the **Connectors** folder and click **Add > Connector**.

The **Manage Connectors** dialog is displayed.

Select a type of connector from the dropdown list.

Configure the connector in the **Configuration** section, e.g. **OpenAPI** connectors can be configured by simply importing a JSON or YAML description file.

ⓘ Supported OPe

n API versions are 2.0, 3.0.0, 3.0.1, 3.0.2, and 3.0.3. The OpenApi version 3.1 is not supported at the m

oment.

Refer to [Adapters](#) for more information about adapters and their related connector configuration.

When you have selected an appropriate file, click **Next**.

> ⓘ The **Next** button is enabled only when you have selected a file.

If the selected file cannot be loaded, a corresponding error message is displayed in the dialog.

Additionally a (more detailed) toast message is displayed. The following error messages are possible:

Displayed Message	Possible Reason	Solution
The OpenAPI version 'x.y.z' is not supported. Versions 2.0.0, 3.0.0, 3.0.1, 3.0.2, and 3.0.3 are supported.	• You have selected a file in a version that is not supported.	• Select a file in a version that is supported.
The property 'swagger' or 'openapi' is missing.	• The mentioned properties are missing from the file or are misspelled.	• Check if the basic structure of the file is correct. For more detailed information about the basic structure of an OpenAPI file refer to OpenAPI Basic Structure.
The file cannot be loaded.	• The selected file is corrupt.	• Select a valid file.
	• The selected file has the wrong file type.	• Select a file of a supported file type, e.g. a YAML description file.
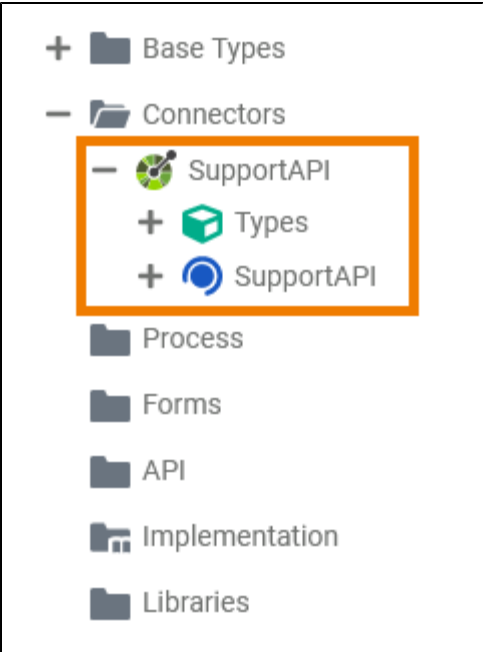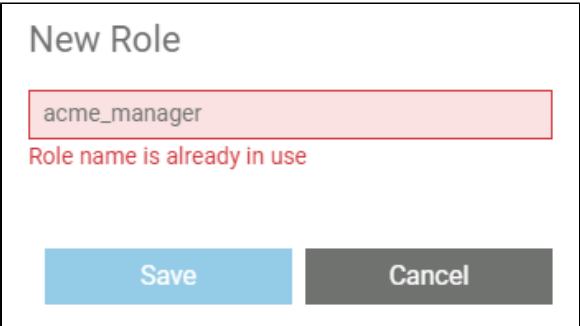
After the selected file has been successfully imported, you can start the actual configuration of the connector.

When everything is configured to your needs, click **Next** to save the connector.

The connector, his types and interfaces, have been added to your service with the given configuration.

creating_a_role

Used on:

- Lane
- Modeling Roles

Click **New Role** to create a new role and assign it to the lane element.

Enter the name of the new role in the pop-up window.

The input is mandatory.

If the role name is already in use, you will not be able to save.

Use the existing role instead (see **Adding an Existing Role** below).

Save to create the role.

The new role is now added to the role editor and assigned to the current lane.
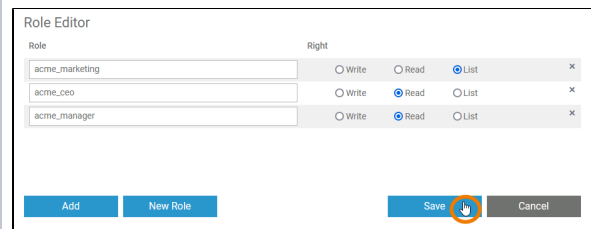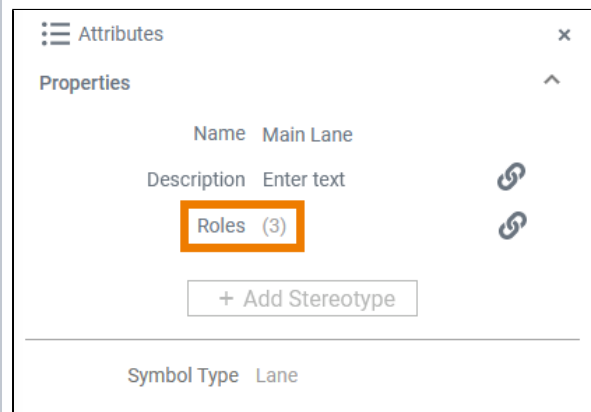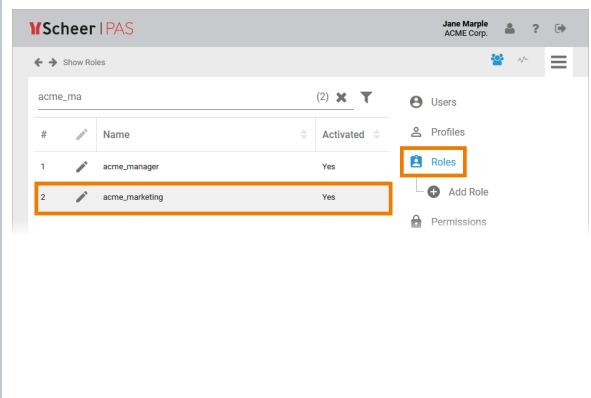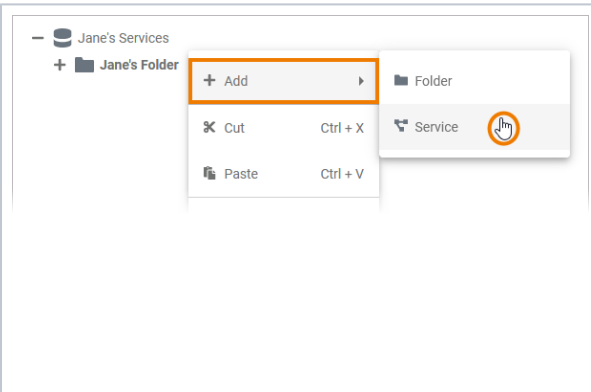
In the next step you need to assign the rights that will be granted to the role owner.

Available options are:

Right	Description
Write	A user assigned to this role will be able to see and change the instance data for all process steps saved in the current lane.
Read	A user assigned to this role will be able to display the instance data for all process steps saved in the current lane. This right is assigned by **d efault**.
List	A user assigned to this role will only be able to see the instances in the instance table for all process steps saved in the current lane.

You can continue to create further roles (or add already existing ones, see below) as multiple roles can be assigned to one lane.
When you have finished, save your changes to apply them to the lane.

The number of roles assigned to a lane is displayed in the attributes panel.

The new role has also been added to the user management.
You have to open the user management, to assign the new role to users.

> ✓ Refer to Administration Guide > Managing Roles for further information about role management.

creating_a_service

> ⓘ **Used on:**
> - Folders
> - Services

Add a service to manage the necessary settings for your microservice. A service can only be created in folders.

To add a service to a folder, right click on the folder and select option **Add > Service** from the context menu.

Create New Service

JanesService

[Save] [Cancel]

Create New Service

Janes First Service

[Save] [Cancel]

Enter a name for the service in the pop-up window **Create New Service**. Click **Save**.

If you want to deploy the service to a container later, please note that on container deployment the service name is automatically changed according to cert ain rules.

⚠ **Naming of Services for Container Deployment**

Example: Service Name: **Janes_First_Service** Container Name: **janes-first-service**

- The two services in this example would get the same container name…
- …. so they would overwrite each other.

> **ⓘ Restrictions on Element Names**
>
> The only restriction for folder names is that they must not be empty. To all other elements in the Explorer, the following name restrictions apply:
>
> Element names...
>
> - ... must not be empty.
> - ... must not contain spaces.
> - ... 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 used as element names: CON, PRN, AUX, NUL, COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8, COM9, LPT1, LPT2, LPT3, LPT4, LPT5, LPT6, LPT7, LPT8, LPT9.

− 🛢 Jane's Services − 📁 Jane's Folder 🔧 **JanesService**	The new service is created and displayed in the tree.
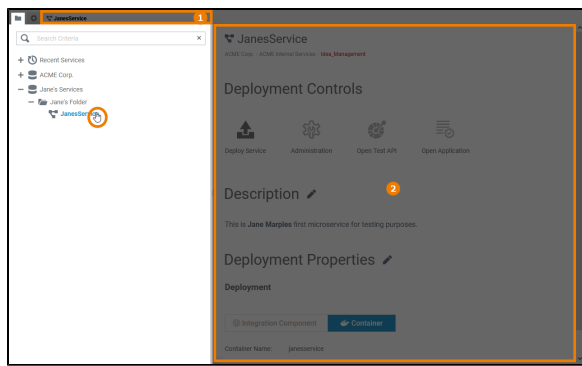
To open a service, click on the service name in the explorer tree.

1. The Service tab will open.
2. The service details will be displayed on the right side of the tree.
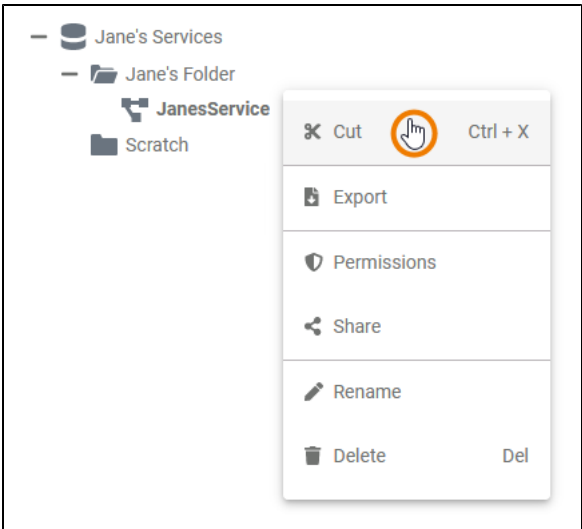
> ✅ To access the service, click on the the explorer tab icon or on the service details area to close the explorer tab. Refer to Service Panel for further information.
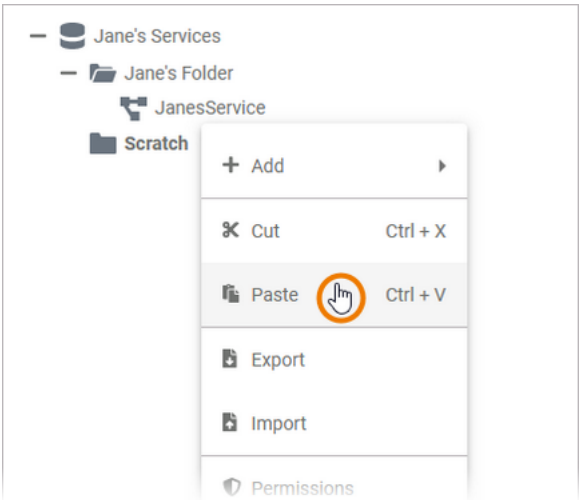
cutting_and_pasting_explorer

> ⓘ **Used on:**
> - Folders
> - Services

If you want to use an element at a different position in the tree, select the context menu option **Cut**.

Open the context menu of the element in which you want to insert the cut element and select **Paste**.

ⓘ Option **Paste** is only enabled if the elements to be inserted fit into

	the tree hierarchy. Pasting elements is only possible in the same namespace.
Jane's Services 　Jane's Folder 　Scratch 　　JanesService	The element is now displayed on its new place in the tree.

designer_development_process

With PAS 23.1, the development process in the Designer has been optimized. To speed up and simplify the development process, we have decoupled development from deployment: Developers now work in a test environment where they can get constant feedback on their changes and can test them directly against the service preview.

The new approach also improves the work of fusion teams on a microservice: Developers can now make a finished feature or an intermediate state of a service available to other team members for testing while continuing to work independently on the same service.

Development/Test System

Development-related Testing

Finished Milestone (Feature, Bugfix...)

Continuous Development

Deployment

Development-independent Testing
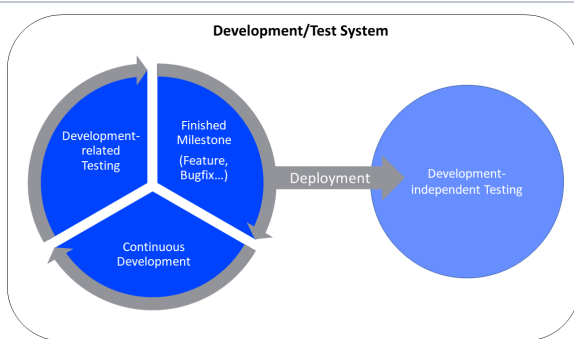
Your benefits:

- You can access a preview of your service without having to deploy it, which speeds up testing and process optimization.
- Your test service is constantly compiled and validated. Errors and warnings are displayed in the validation panel and help you create a flawless service.
- You only need to deploy your service...

- if you want to give other team members the opportunity to test dedicated features or versions of these service.

- - if you want to run regeression test sagainst your test server.
- In the meantime, you continue to develop the service independently of the deployed version.

designer_deployed_service

Deployed Service

Deployment Controls ⬆ Deploy Service ⚙ Administration 🥧 Open Test API 📋 Open Application	The tester opens the deployed service.
# Applications ↳ Test_Application 🔍 ➕ → Start_test	The tester starts the process to create a new instance.
Name _____ Date of birth 📅 **OK**	The deployed version of the form opens.

designer_test_service

Test Service

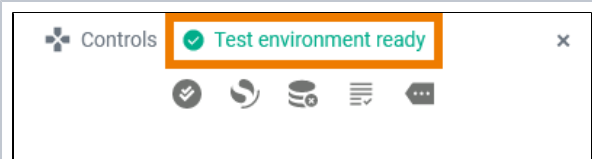|	
⁘ Controls ✅ Test environment ready ✕ ✅ 🌀 🗄 📋 💬	The developer opens the test service.
# Applications ↳ Test_Application 🔍 ➕ → Start_test	The developer starts the process to create a new instance.

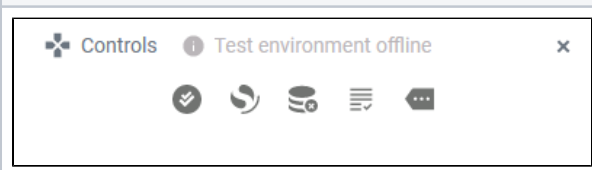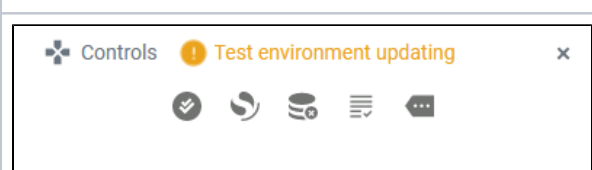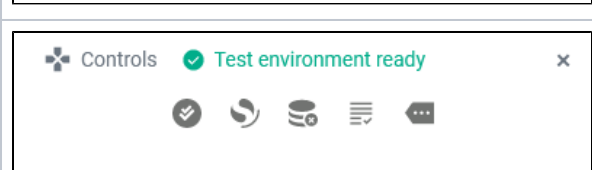|		The adapted form opens.
First Name Last Name Date of birth 📅 Upload Your Profile Image Drag and drop file here or ⬆ Browse **Continue**		

designer_test_environment_status

> ⓘ **Used on:**
> - Developing with Designer from Version 23.1
> - Working with the Test Environment

When you create a new service or open an existing one, you are automatically working in the test environment. Its status is displayed in the **Controls** panel.

The test environment is automatically started and stopped:

- It is started...
    - when you open a service and start working on it.
    - when you select option **Open application** .
    - when you trigger a manual validation via option **Start validation** .
- It is stopped...
    - when you have not worked on your service for a certain amount of time. Default setting is 2 hours.
    - when the corresponding project is deleted.

| Status Message	Description
	The test environment has not yet been started or has expired.
	The Designer is preparing the test environment.
	The test environment is ready. You can access the application preview.

designer_test_service_preview

> ⓘ **Used on:**
> - Developing with Designer from Version 23.1
> - Working with the Test Environment

Accessing the test service is possible without deployment. Open the service preview via the controls panel and the service start page will open in a new tab:

Applications

↳ Test_Application 🔍 ⊞

→ Start_test
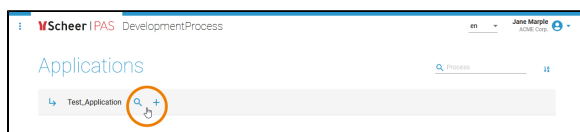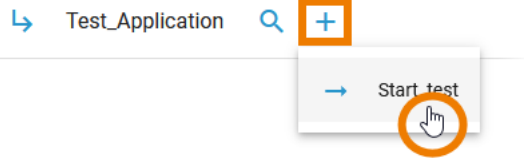
You have direct access to your service(s):

- Start your process ⊞ and create a new instance.
- Open the instance overview 🔍.

Name
Theo Test

Date of birth

APR 2023 ▾   ‹ ›

Su	Mo	Tu	We	Th	Fr	Sa
APR						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						

^   ^   ^

11 : 25 : 35

⌄   ⌄   ⌄

✓

The preview is fully operational.

Use it to check functionality, layout and design of your process and its forms.

ⓘ

deployment

Deployment Properties ✏️

Deployment

⚙️ Integration Component	🐳 Container

Container Name: developmentprocess

Select your deployment target the first time you deploy.

ⓘ **Prerequisites**

- **Container deployment:** Your user must be assigned the **xuml_container_admin** profile.
- **Integration (Bridge) deployment:** You user must be assigned the **integration_user** profile and he must have an integration user account created by an integration administrator.

Refer to Administration Guide > Overview of Standard Profiles.

Enable edit mode ✏️ of the deployment properties and choose between **Integration Component** or **Container**.

Deployment Properties ✓ ✕

Deployment

| Integration Component | 🐳 Container |

Container Name: developmentprocess

Service Status: Not deployed

Deployed Version:

General

Version:	1.0.0
Category:	Development
Enable Validation:	☑
Enable Angular Build:	☑

We recommend to enter a version number and increase the number with each deployment.

✓ **Expert Advice**

By default, each newly created service gets number **0.1.0** assigned. We recommend to increase the version number before redeploying each time you have made relevant changes to the service. You can change the version number in section **General** of the **Deployment Properties**. Follow the concept of sem antic versioning.

In case of deployment problems, comparing the version number and the number of the deployed version can help to find out which version of the service is running.

Save ✓ the deployment properties to continue.

Deployment Controls	Now use the deployment controls to deploy the service to your chosen target:

Deployment Controls

Deploy Service Administration Open Test API Open Application

Now use the deployment controls to deploy the service to your chosen target:

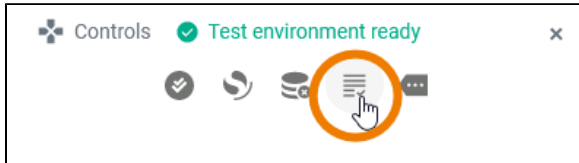| Icon	Description
	Use this option to deploy the service to the selected deployment target.
	Use this option to open the management UI of the selected deployment target: • [PAS Administration](#) for **Container** • [Integration](#) for **Integration Component**
	Use this option to open the test UI for the service-related API.
	Use this option to open the deployed service in a new browser tab.

differences_opening_applications

> ⓘ **Used on:**
>
> • [Running Designer Applications](#)
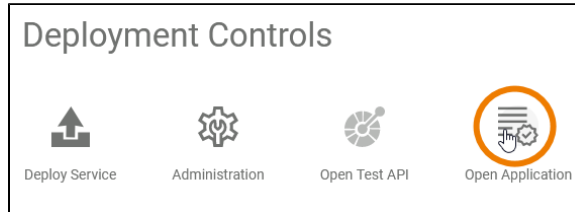> • [Creating a Process Instance](#)

Test and deployed service are both fully functional. The handling of an application is the same for both service types. The only difference is the way how to access the applications start page:

Test Service	Deployed Service

| Open the application via the **Controls Panel** :	Open the application via the **Deployment Controls** on the service details page:
	Of course, to open the deployed service, you must first deploy it. Refer to Deploying Service for more information.

executing_deployed_service

The deployed service is accessible via option **Open Application** in the deployment controls section. The start page of the deployed service looks exactly the same as that of the test service:
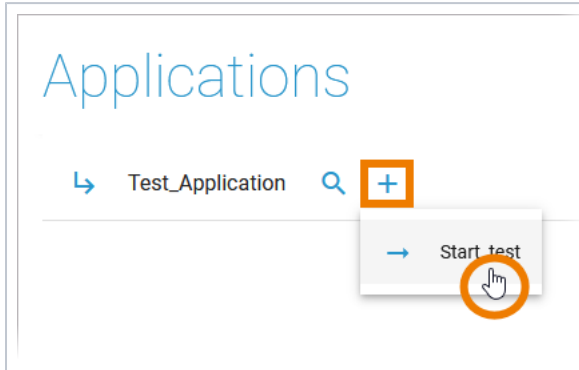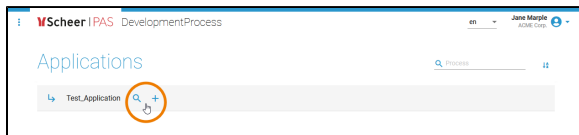
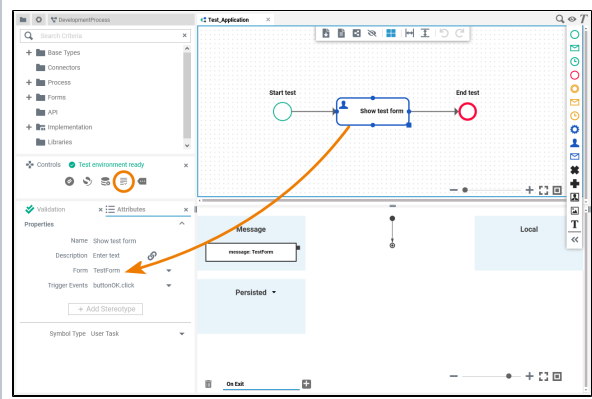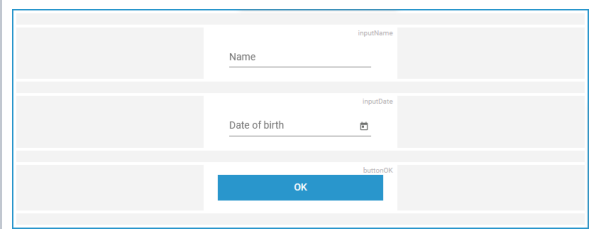|	
	The handling of the deployed service is also the same: • Users can create new instances and execute the process. • Users can access the instance overview.
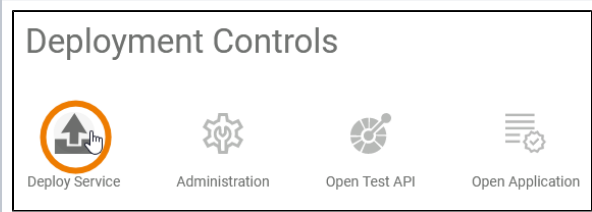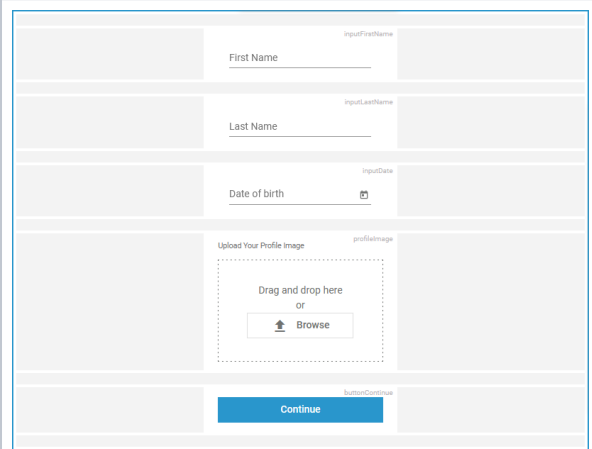
The difference is: The deployed version does not change, even if the developer changes the service in the test environment.

executing_deployed_service_example

A developer created a simple test application, containing only a user task with an assigned test form.

The test form contains two input fields and a button.

The developer deploys this version.

After the deployment of this version, he starts to extend the test form.

field_config_editor

> ⓘ **Used on:**
> - Radio Button Group
> - Select Field
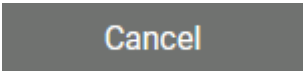
If you click icon 🔗 the **Field Configuration** editor opens:

Use the editor to add and manage the content of your element:

Option	Description
	Click **Add** to insert new rows.
	Use option **Drag** to change the order of the rows.
You can enter the options for this element as value pairs. When entering value pairs, always the first value (**Value**) is saved as key in the data container. The second value (**Label**) is used as notification to the user. The label is displayed in the form. Value pairs may also contain numbers, so that a user can chose a text option, but in the background a calculation is carried out.	
	Enter a **Value** for your element option. It is not possible to assign the same value to different labels.
	Enter a **Label** for your element option. The label is displayed in the form.
	Use this option to set the element option as default.
	Use this option to delete the element option.
	Click **Save** to save your changes.
	Click **Cancel** to discard your changes.

how_to_use_the_timer

 Used on:

- Changing BPMN Element Attributes
- Timer Start Event

Timer Start Event

○ ⦿ Scheduler   ○ Timer

~~Scheduler~~

If you click option 🕐 in the attribute panel, a separate editor will open where you can define the settings of your **Timer Start Event**.

You have two options with different settings:

- **Scheduler** (see below)
- **Timer** (see below)

Timer Start Event

◉ Scheduler ○ Timer

Scheduler

☑ Allow Parallel Executions ⓘ

Year(s) [*] ⓘ

Month(s) [*] ⓘ

Weekday(s) [*] ⓘ

Day(s) [*] ⓘ

Hour(s) [*] ⓘ

Minute(s) [0] ⓘ

Seconds [0] ⓘ

[Time Patterns](#)

[Save] [Cancel]

Scheduler

Use the scheduler if you want to define a pattern for your **Timer Start Event**, for example to start the model every monday, every hour etc.

Setting	Description	Default
Allow Parallel Executions	If checked, each time a scheduler pattern matches, the scheduler activity diagram is executed - even if another execution is already running. If not checked, the next execution cycle is suppressed if an execution diagram is already being run.	Checked
Year(s)	Insert a positive integer or a pattern.	*
Month(s)	Use numbers 1 to 12 or a pattern.	*
Weekdays(s)	Valid input: • Mon or Monday • Tue or Tuesday • Wed or Wednesday • Thur or Thursday • Fri or Friday • Sat or Saturday • Sun or Sunday • a pattern	*
Day(s)	Depending on the month, valid input are numbers from 1 to 31 or a pattern.	*
Hours(s)	Use 0 to 23 or a pattern.	*
Minute(s)	Use 0 to 59 or a pattern.	0
Seconds	Use 0 to 59 or a pattern.	0

Timer Start Event

◯ Scheduler   ● Timer

Timer

Repeat Interval   [   ] ✎ ⓘ

Occurrences   ☑ Always   ⓘ

First Occurrence   [0 seconds   ] ✎ ⓘ

[Time Durations](#)

[**Save**]   [**Cancel**]

Timer

Select the timer if you want to define a cycle for example to start the model after one week, after two hours etc.

Setting	Description	Default
Repeat Interval	Enter a valid time duration expression (see below). Refer to Time Durations in the Bridge documentation for detailed information. This setting defines the time interval between two calls of the timer action.	-
Occurrences	Insert a positive integer to define the count of repetitions. If you want the timer to be endless, check **Always** (default).	Always
First Occurrence	Enter a valid time duration expression (see below). Refer to Time Durations in the Bridge documentation for detailed information. This setting defines the wait interval after service start before the timer is executed for the first time.	0 seconds

Insert duration expression

Year(s)		Y
Month(s)		M
Day(s)		D
Hour(s)		H
Minute(s)		M
Seconds		S

Result:

[Save] [Cancel]

Duration Expression Wizard

The settings **Repeat Interval** and **First Occurrence** contain the option 📝 .

Click this option to open a wizard that supports you to define the necessary duration expression.

Insert duration expression

Year(s)		Y
Month(s)		M
Day(s)	2	D
Hour(s)	3	H
Minute(s)	10	M
Seconds		S

Result: **P2DT3H10M**

[Save] [Cancel]

Insert positive integers in the desired fields.

The result is shown below.

Click **Save** to close the wizard.

Timer Start Event

○ Scheduler ● Timer

Timer

Repeat Interval | P2DT3H10M ✏️ | ℹ️

Occurrences ☑ Always ℹ️

The duration expression is displayed in the corresponding field.

✓ Once you are familiar with the notation of the duration expression, you can also insert it directly in the field.

lane_handling

ℹ️ **Used on:**

- Lane
- Modeling Roles

Main Lane

For a role-based process model, you need at least one main lane.

You can then add sublanes.

To create sublanes, drag the next lane elements onto the header of the main lane element. Drop the new element on the border of an existing lane to position the new sublane.

To create a sublane within a sublane, you can drag the element onto the body of an existing lane.

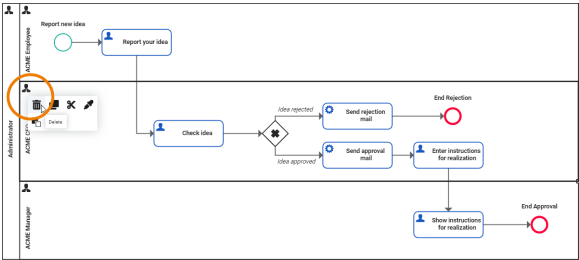Click the image to see an animated version.

To activate the lane el

Main Lane

ement, click its border or icon/header. If the element is active, an indicator is displayed in the lower right corner and the bo

Before Deletion:

After Deletion:

ⓘ **Deletion Behavior of Lane Elements**

If you delete a lane element, it is erased from my

our work area including all content! However, you can use restore the deletion by using **Undo** in the BPMN toolbar.

open_role_editor

☰ Attributes ✕	When you click option 🔗 in the attributes panel, a separate editor will open where you can assign existing roles and create new ones.
Properties ⌃	
Name Main Lane	
Description Enter text 🔗	
Roles (0) 🔗	
+ Add Stereotype	
Symbol Type Lane	

Role Editor	In the role editor, you have two options:
Role Right	
[ ] ○ Write ⦿ Read ○ List ✕	- You can add an already existing role.
If no roles are assigned, the process steps inside the lane can be executed by all users.	- You can create a new role.
[Add] [New Role] [Save] [Cancel]	

ⓘ If no roles are assigned to a lane, the

process steps inside the lane can be executed by all users and a corresponding note is displayed.

permission_administration

In the pop-up window **Permission Administration** you can invite single colleagues or whole groups to access the element.

By default, the creator of the element and everyone sharing the current namespace has write permissions.

ⓘ To limit access to the element you can remove the group **Everyone** completely, or give it read-only permission. If you want to restore permissions for **Everyone**,

1. select option **Invite Group**,
2. search for group **Everyone** and add it again.

If you want to add other users, first select one of the options:

- **Invite Group:** Allow whole user groups to access your element. Available groups are all distribution lists that have been created as a **Role** in the user management.
- **Invite People:** Add single users to your element and grant them read or write permission.

> ✅ Refer to Administration Guide > Managing Roles for detailed information about roles.

Enter the name or email address of the group or person to the search field. Matching results are shown in a list.

Select the entry you want to use.

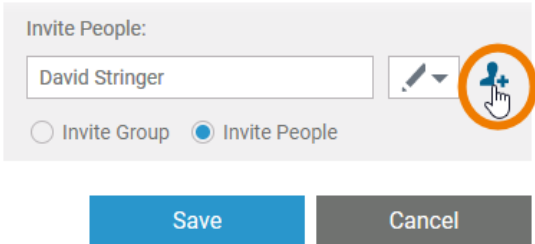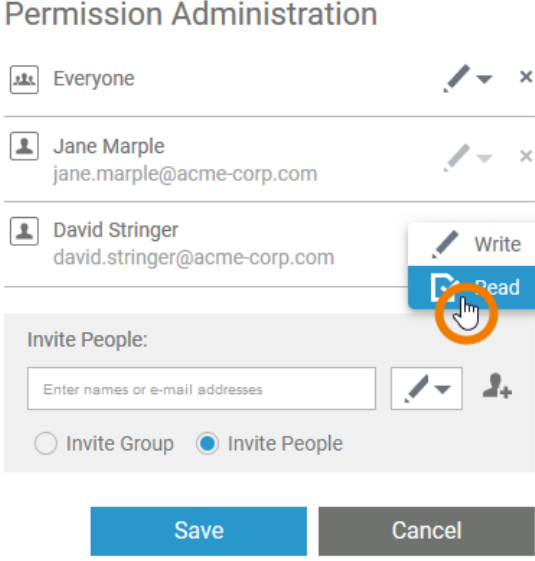|	
	Click option to add the selected user or group.
	Once the user or group has been added, you can specify whether the user or the group should have read or write permission. **Save** your changes.
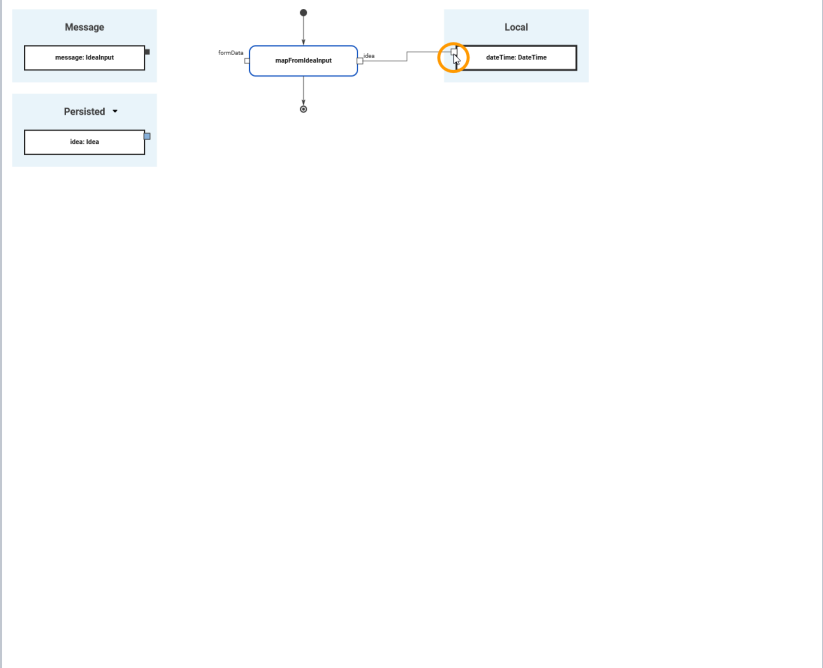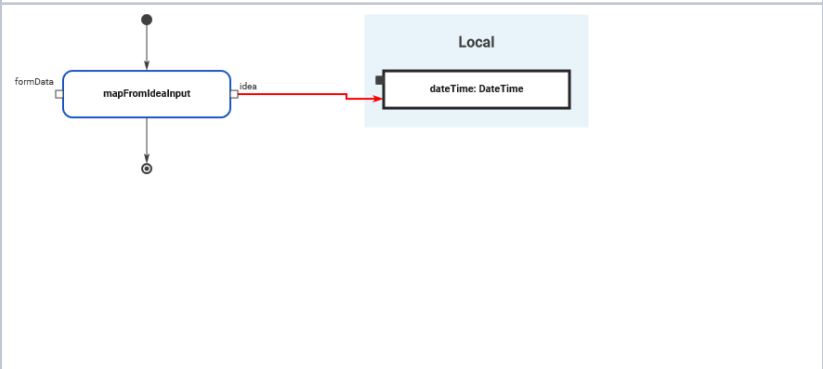
pin_highlighting

> **Used on:**
>
> - Adding Operation Calls
> - Drawing Object Flow Execution

While dragging the object flow, the color of the pins changes to assist you with finding matching connection points.

	When you start dragging an object flow, all pins of the same type and multiplicity are marked blue. Blue pins indicate that this pin is a valid target for the object flow.
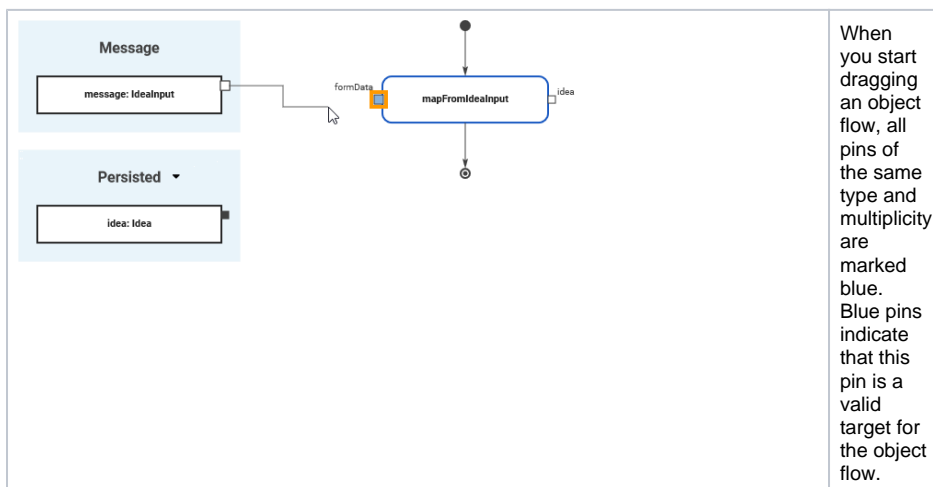
White connection points of variables indicate that you are trying to connect variables that do not have the same type or multiplicity.

The connection point is marked in white when you have reached it with the mouse pointer.

It is possible to draw an object flow to an invalid connection point but the relation will be invalid and displayed in red.

	A pin turns red if an object flow cannot be created at all. You cannot - connect a local variable with another local variable - connect a persisted variable with another persisted variable - draw a connection to a pin that already has an incoming object flow.

validating_error_types

> ⓘ **Used on:**
>
> * Using the Validation
> * Validation Panel

Different types of validation messages are displayed. Both, warnings and errors, are reported by the xUML Compiler, and refer to the implemented execution:

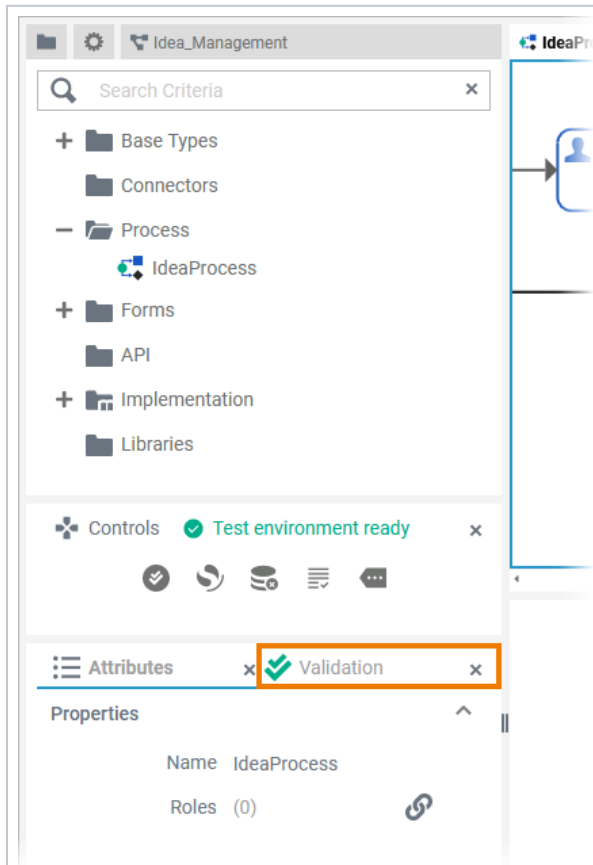| Icon	Error Type	Description
❗	**Warning**	If the compiler reports a warning, you can deploy the service nevertheless. However, you should check the reported items in your service.
❌	**Error**	If an error is reported, deployment is not possible. Check the mentioned parts of the service, and eliminate the cause of the error to enable model execution.

validating_main

The validation results are displayed in the **Validation** panel. It is available in all editors:

- BPMN Editor
- Execution Editor
- Mapping Editor
- Form Editor

If you use the panel preset, the validation panel is the second tab in the lower left corner. To return to the panel preset, go to the user preferences, open the **Designer Options** and select option **Reset Panels**.

Refer to Customizing Editors and Panels for detailed information about panel management in general.

☰ Attributes	✕	❗ Validation	✕

❌ The state 'Waiting for Check_idea' has no incoming transition.

❌ The argument pin 'type' is not of the called component.

❌ The action 'log' calls an operation not being imported. Most likely reason: one of the packages containing the operation does not have the stereotype 'Repository'.

❌ The operation '' is not a built-in action script operation. (position: -1:-1)

⚠️ The object 'description' is unused, it has no incoming and no outgoing flow.

In the **Validation** panel, validation findings of a service are displayed. The icon in the panel header indicates if there are any validation issues:

✅	Everything is fine. The service is ready to be executed in the test environment or to be deployed via the service details page.

	The validation is in progress. Wait to get the result.
	There are validation issues. You need to resolve these issues before being able to test or deploy the service.

The list is sorted by error types:

1. **Error:** xUML Compiler errors
2. **Warning:** xUML Compiler warnings

 You cannot test or deploy a service that still has validation errors. You need to resolve them first.

validating_using_diplayed_info

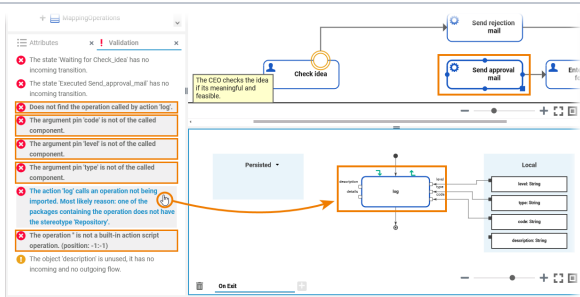 **Used on:**

- Using the Validation
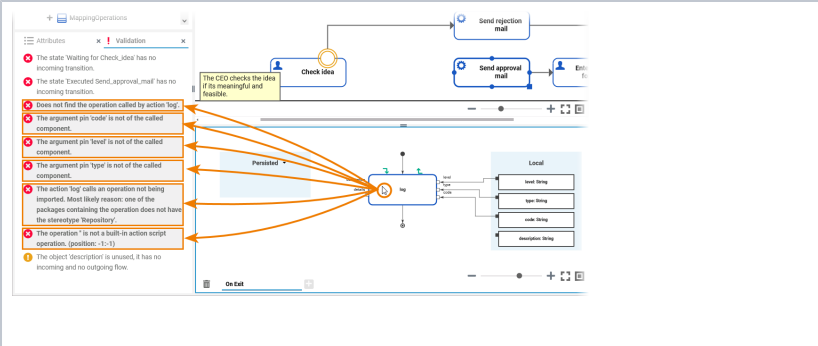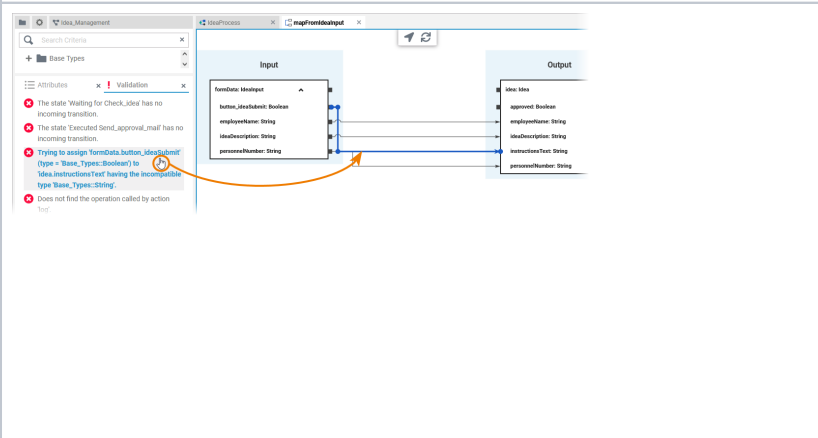- Validation Panel

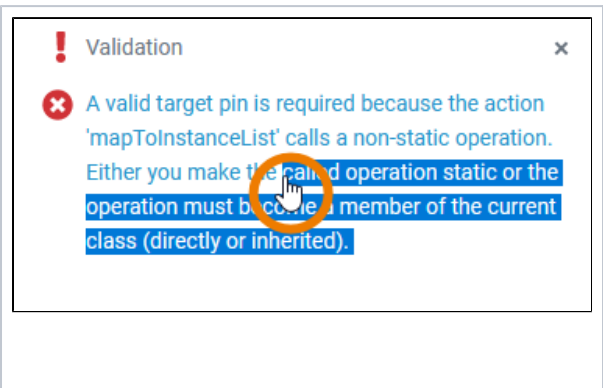Validation messages describe the error, and also provide information on how to solve the problem.

The items in the validation panel are links. If you click on a message...

- ...the related editor opens if not already open.
- ...the affected element is selected on the diagram pane.
- ...all other messages concerning this element are highlighted.

If the message refers to an element on the execution pane, the associated BPMN element on the BPMN diagram pane is also selected.

This also works vice versa: Click an element on the pane to highlight all messages concerning this element.

The handling in the Mapping Editor works the same way: Click a message to highlight the affected element or object flow in the editor.

validating_sharing_validation_messages

> ⓘ **Used on:**
> - Using the Validation
> - Validation Panel

✓ Sometimes you want to share a validation message with another developer.

Select the displayed message (or the part you want to share) with your mouse or a double-click. Now you can copy and paste the highlighted content, e. g. to an email or a chat.