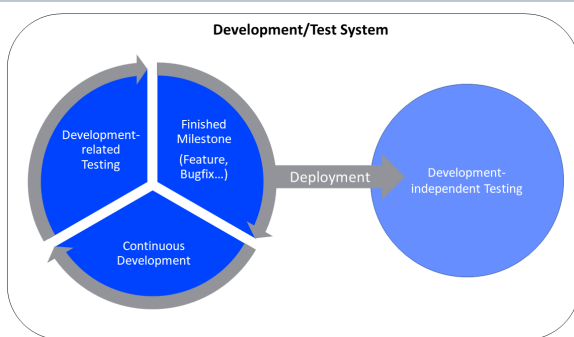


Developing with Designer from Version 23.1

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.



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...

On this Page:

- [New: Test Environment](#)
- [Using the Test Environment](#)
 - [Using the Service Preview](#)
- [Deploying the Service](#)
 - [Executing the Service](#)
 - [Example: Test Service vs. Deployed Service](#)

Related Pages:

- [Going Live with a Designer Service](#)
- [Exposing Designer Services via API Management](#)
- [Designer Tutorials](#)

Related Documentation:


- [Scheer PAS Designer Guide](#)
- [Scheer PAS Administration Guide](#)



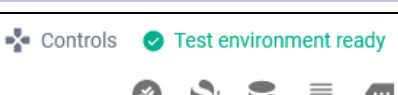
- if you want to give other team members the opportunity to test dedicated features or versions of the service.

◦ if you want to run regression tests against your test server.

- In the meantime, you continue to develop the service independently of the deployed version.

New: Test Environment

| | |
|---|--|
|  | <p>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.</p> <p>The test environment is automatically started and stopped:</p> <ul style="list-style-type: none"> • It is started... <ul style="list-style-type: none"> ◦ 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... <ul style="list-style-type: none"> ◦ 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. |
|---|--|

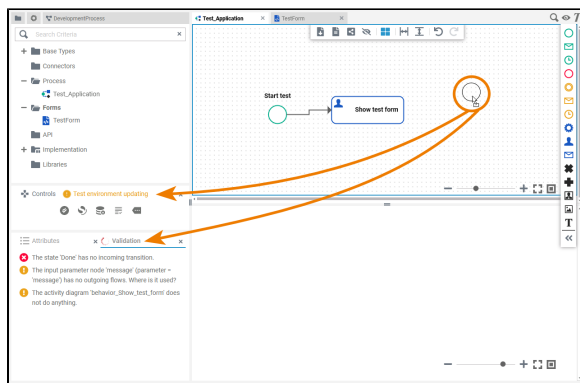
| Status Message | Description |
|---|---|
|  | <p>The test environment has not yet been started or has expired.</p> |
|  | <p>The Designer is preparing the test environment.</p> |
|  | <p>The test environment is ready. You can access the application preview.</p> |

Using the Test Environment






The test environment is initialized automatically when you start working on a service. As soon as you make changes in the Designer, the test environment is updated to reflect your changes. In addition, your actions are constantly compiled and validated. Compilation errors are displayed in the validation panel. Check the panel for errors and warnings to resolve any problems that occur in your service. Go to [Designer Guide > Validation Panel](#) for details.









You can disable the auto validation, refer to [Designer Guide > Managing the Service Details](#) for further information.



✦ Controls
✔ Test environment ready
✕

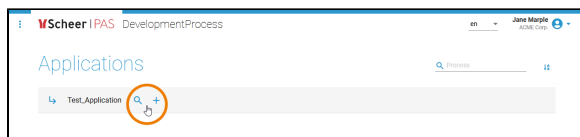






The new **Controls** panel contains options to support you during the development process:

| Icon | Description |
|---|---|
|  | Use this icon to trigger the validation manually. |
|  | Use this icon to open the Log Analyzer and access the logs. You have the choice between service and application logs. <ul style="list-style-type: none"> Application: Access the logs of the Angular server which delivers the application UI. Service: Access the logs of the built xUML service. <div style="border: 1px solid #c6e0b4; padding: 10px; margin-top: 10px; background-color: #f1f8f1;"> <div style="display: flex; align-items: center;">  <div style="margin-left: 5px;"> Refer to Administration Guide > Analyzing Platform Logs for further information about the Log Analyzer. </div> </div> </div> |
|  | <p>The option Clear Compiler cache is now directly available via an icon in the controls panel. Clearing the Compiler cache is helpful if the validation panel shows errors you cannot solve.</p> <p>Using this option reinitializes the test environment.</p> |
|  | Use this icon to open a preview of your service . |
|  | Use this icon to open the additional menu of the controls panel which gives you access to further options (refer to Designer Guide > Controls Panel for details). |

Using the Service Preview

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

<

>

SuMoTuWeThFrSa

APR

1

2345678

9101112131415

16171819202122

23242526272829

30

^

^

^

11 : 25 : 35

v

v

v

✓

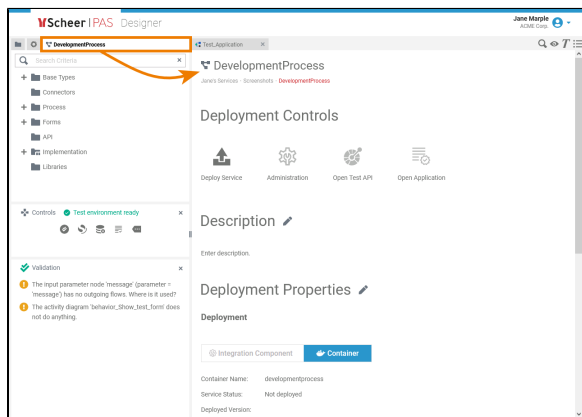
The preview is fully operational.

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

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

Deploying the Service

The **Deployment Controls** have been moved to the service details page. Click on the **Service** panel tab (service name) to open the details page:



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:** Your 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 [semantic 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





Deploy Service

Administration

Open Test API

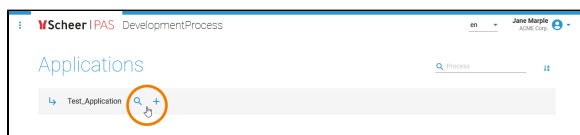
Open Application

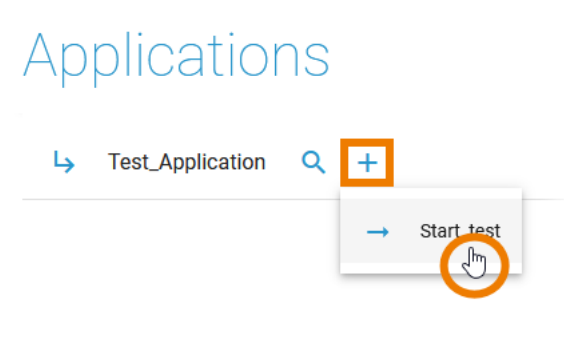
Now use the deployment controls to deploy the service to your chosen target:

| 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: <ul style="list-style-type: none"> 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. |

Executing the Service

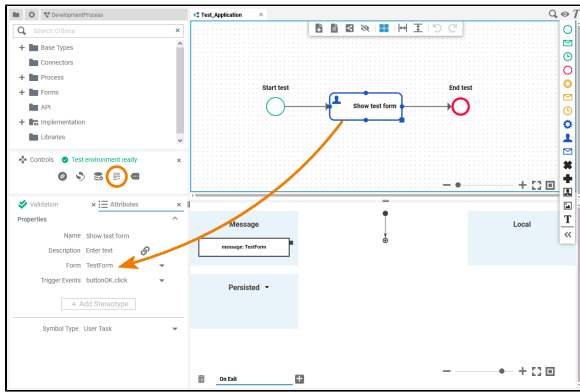
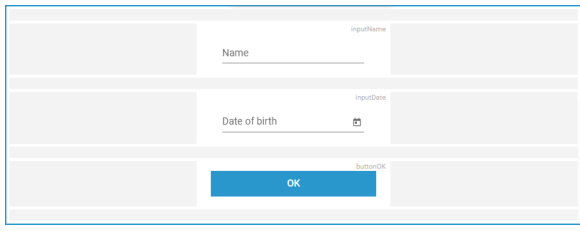
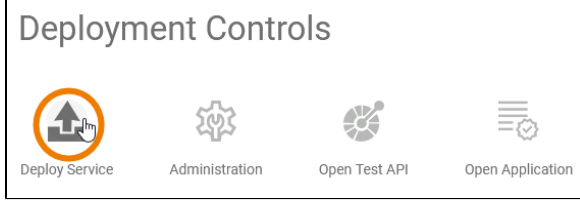
The deployed service is accessible via option **Open Application** in the deployment controls section. The start page of the deployed service looks exactly the same as that of the test service:



| | |
|---|--|
|  | <p>The handling of the deployed service is also the same:</p> <ul style="list-style-type: none"> • 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.

Example: Test Service vs. Deployed Service

| | |
|---|---|
|  | <p>A developer created a simple test application, containing only a user task with an assigned test form.</p> |
|  | <p>The test form contains two input fields and a button.</p> |
|  | <p>The developer deploys this version.</p> |

| | | | |
|--|---|-----------------|--|
| | <input type="text"/> | input.firstName | |
| | First Name | | |
| | <input type="text"/> | input.lastName | |
| | Last Name | | |
| | <input type="text"/> | input.date | |
| | Date of birth | | |
| | Upload Your Profile Image | profileImage | |
| | Drag and drop here or <input type="button" value="Browse"/> | | |
| | <input type="button" value="Continue"/> | buttonContinue | |

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

Test Service

Controls

Test environment ready

Deployment Controls

Deploy Service

Administration

Open Test API

Open Application

p
e
r
o
p
e
n
s
t
h
e
t
e
s
t
s
e
r
v
i
c
e.

Applications

↳ Test_Application



Start test



Applications

↳ Test_Application



Start test



t
h
e
p
r
o
c
e
s
s
t
o
c
r
e
a
t
e
a
n
e
w
i
n
s
t
a
n

Name

Date of birth



OK

First Name

Last Name

Date of birth



Upload Your Profile Image

Drag and drop file here
or



Browse

Continue

T
h
e
a
d
a
p
t
e
d
f
o
r
m
o
p
e
n
s.