SAP Build Process Automation

Generated on: 2024-02-26 21:27:47 GMT+0000

SAP Build Process Automation | Cloud

PUBLIC

Original content: https://help.sap.com/docs/PROCESS_AUTOMATION/a331c4ef0a9d48a89c779fd449c022e7?locale=en-US&state=PRODUCTION&version=Cloud

What Is SAP Build Process Automation?

SAP Build Process Automation is a citizen developer solution to adapt, improve, and innovate business processes with no-code workflow management and robotic process automation capabilities.

SAP Build Process Automation enables business users and technologists to become citizen developers. With powerful yet intuitive low-code and no-code capabilities, the solution supports you in driving automation by tapping into the expertise of citizen developers.

Features

The solution offers the following key features:

- Build or adapt processes with an intuitive graphical interface.
- Create forms-based workflows using drag-and-drop functionality.
- Develop and manage decision logic in tabular, spreadsheet-like decision tables.
- Automate repetitive tasks within existing process flows using robotic process automation.
- Create intelligent actions and recommendations using machine learning capabilities.
- Work efficiently from a unified launchpad and task center.
- Hand over projects to professional developers, who can embed actions and advanced workflows into projects initiated by citizen developers.
- Support real-time, event-driven transparency into comprehensive processes and process instances with process visibility dashboards.

In addition, SAP Build Process Automation offers prebuilt content and features – such as bots, process steps, business rules, and workflow components.

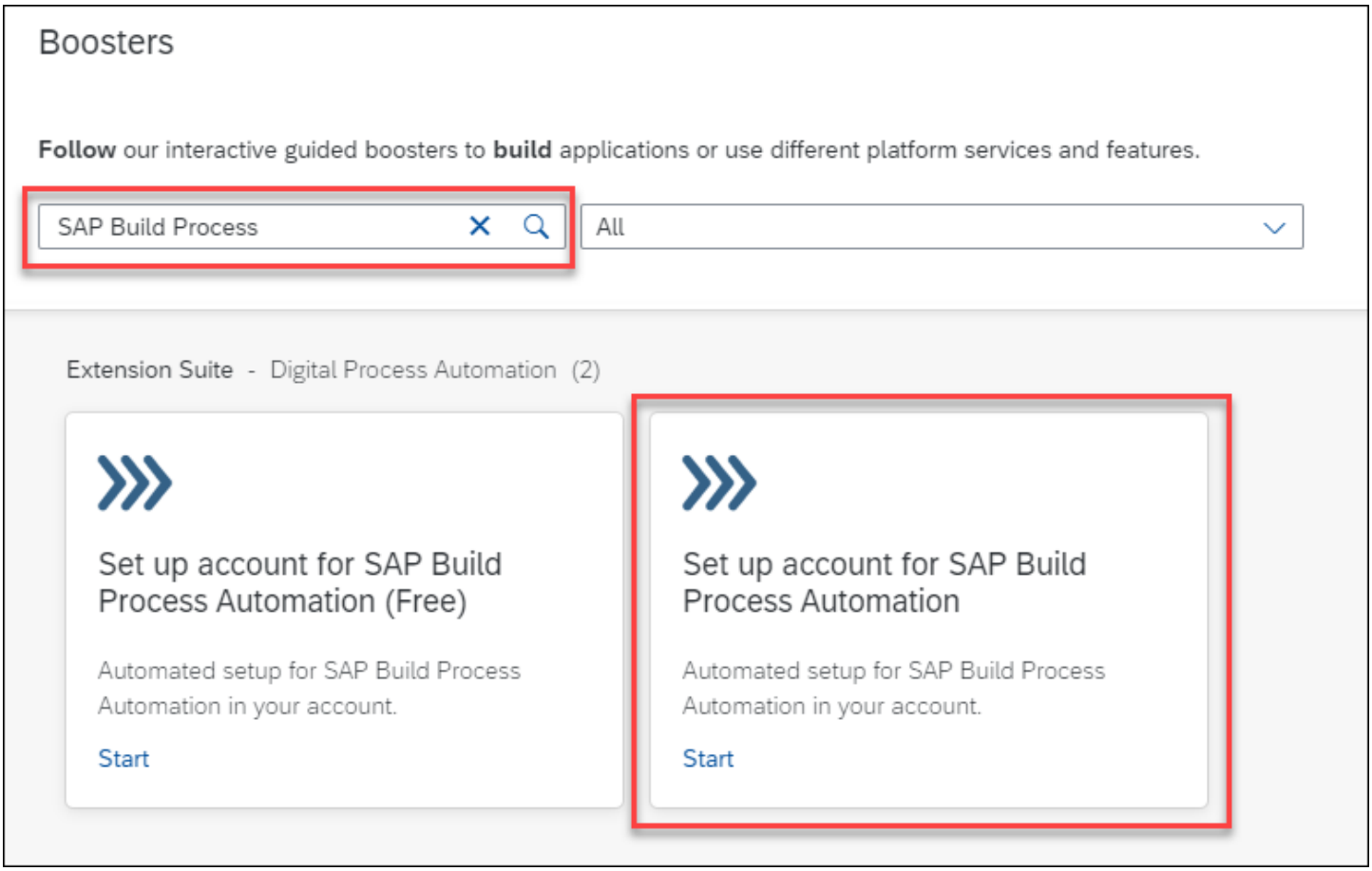
Related Information

Supported Languages
Quotas, Restrictions, and Limits
Known Limitations

Use SAP Build Process Automation

SAP Build Process Automation is an SAP BTP service that allows you to create, run, automate, and monitor your business processes on one interface using low-code/no-code capabilities.

You can subscribe to SAP Build Process Automation using either the standard (paid) or free plan, with active subscriptions added to your SAP BTP subaccount.



For more information about subscribing to SAP Build Process Automation and accessing the service, see Initial Setup.

And for more information about business processes and the role of citizen developers, take our SAP Learning journey:

Utilize Low-Code/No-Code Applications and Automations for Citizen Developers.

Already subscribed to SAP Build Process Automation?

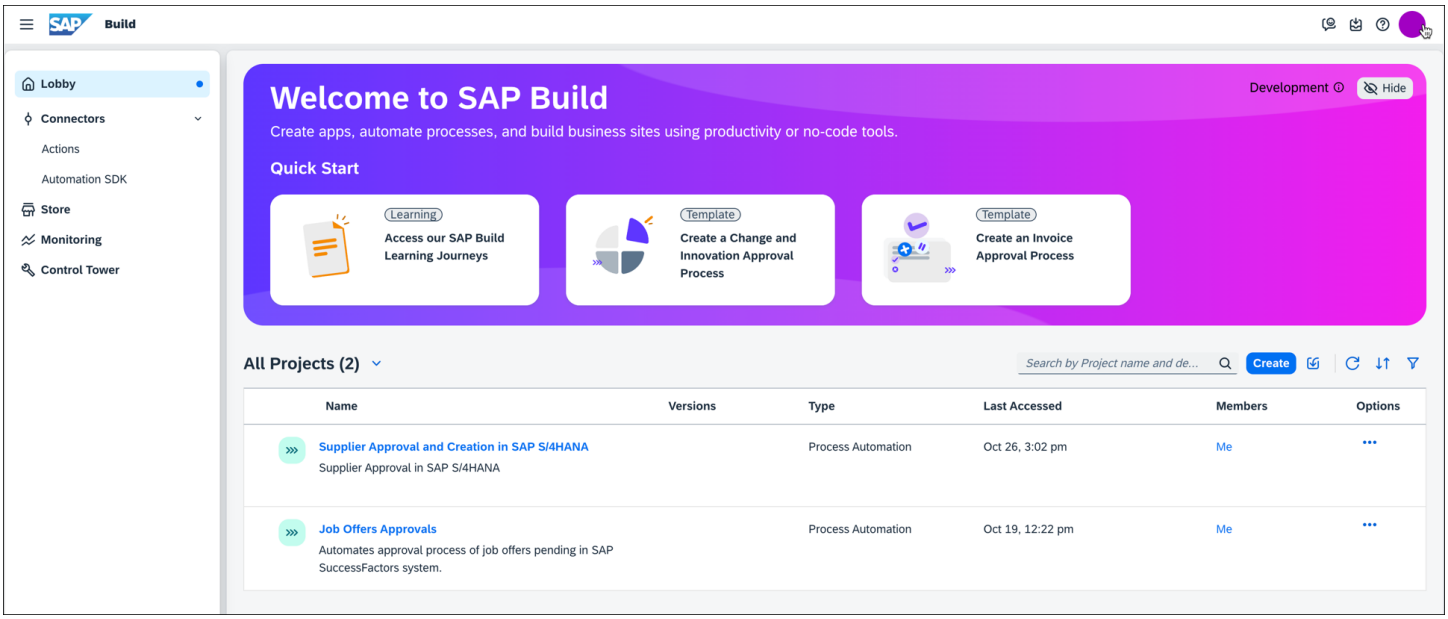
Once subscribed to SAP Build Process Automation, you can create, manage, and deploy projects using the Lobby.

For more information, see Create and Manage Projects.

Create and Manage Projects

After subscribing to SAP Build Process Automation, you create and manage projects in the lobby.

This lobby includes projects you've created, those shared with you, or that you've imported from external sources or the store.



ℹ Note

User principles are case insensitive. For example, “user@mail.com” is treated as the same entity as “USER@mAil.cOM”. Therefore, sharing a Project/Package with them is transparent.

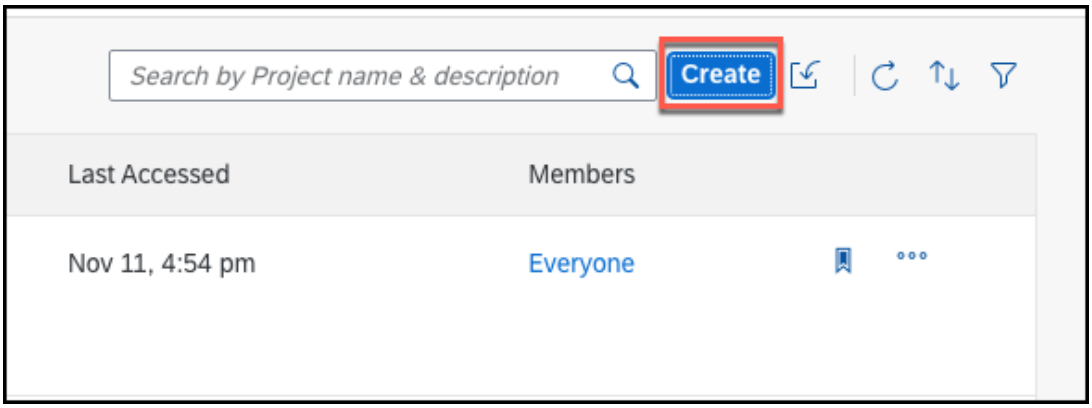
Project Types

With an SAP Build Process Automation subscription, you can create and manage the following project types:

- **Business Process Project** - Create, deploy, automate, and run digital business process by configuring process artifacts. Examples of business process projects include: Investment requests, invoice approvals, and sales order approvals.
- **Actions Project** - Embed external actions and capabilities into your business process projects by uploading an open API specification file in JSON format. Actions projects allow external systems and solutions to communicate with SAP Build Process Automation.

Create a Business Process Project

When creating a business process project, choose **Create** > **Build an Automated Process** > **Business Proces** to start from scratch. This creates an empty project, allowing you to define your own content.



Alternatively, you can take advantage of the **Quick Start** project options. This creates the project container and prompts you to add the relevant steps for the project.

For more information about business process projects and skills, see Business Process Projects

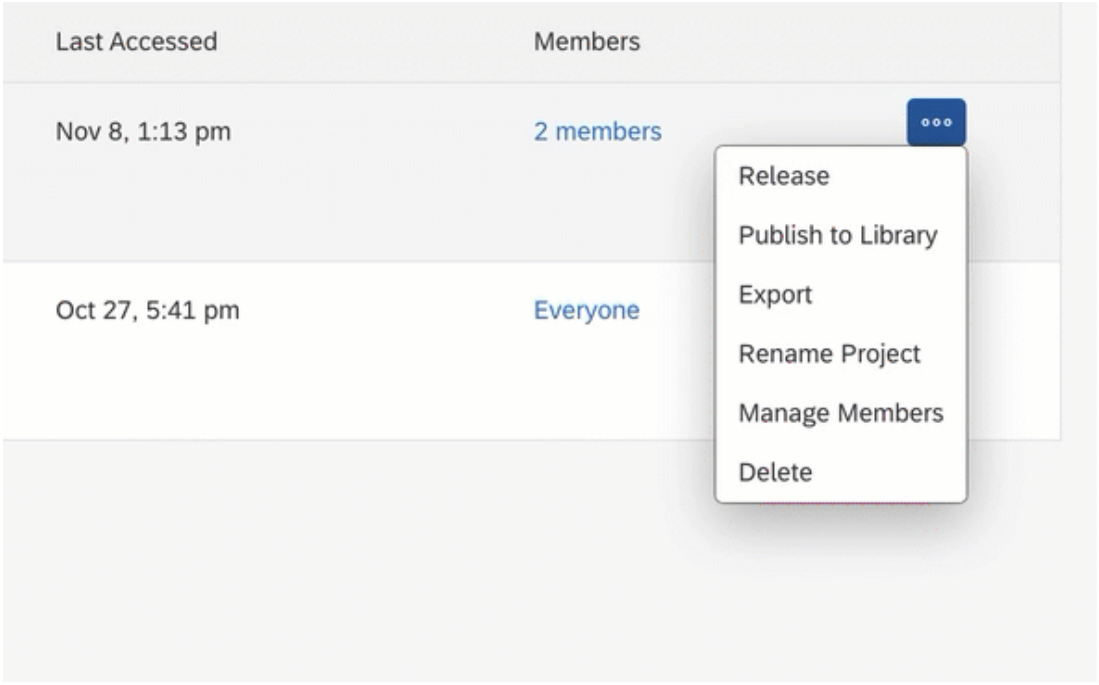
Create Actions Project

To create an actions project, choose **Connectors** > **Actions** > **Create** > **Choose an API Source** and then enter the necessary details.

For more information about Actions project, see Create an Action Project.

Manage Existing Projects

For existing projects, choose **More Options** to see further project management abilities. These abilities depend on the permissions you hold and can include releasing, publishing, renaming, sharing, and deleting your projects.



For more information about your project management options, see Manage Existing Projects.

Business Process Projects

In SAP Build Process Automation, you create a business process project using a combination of one or more process focused skills or building blocks. These skills are known as **artifacts**.

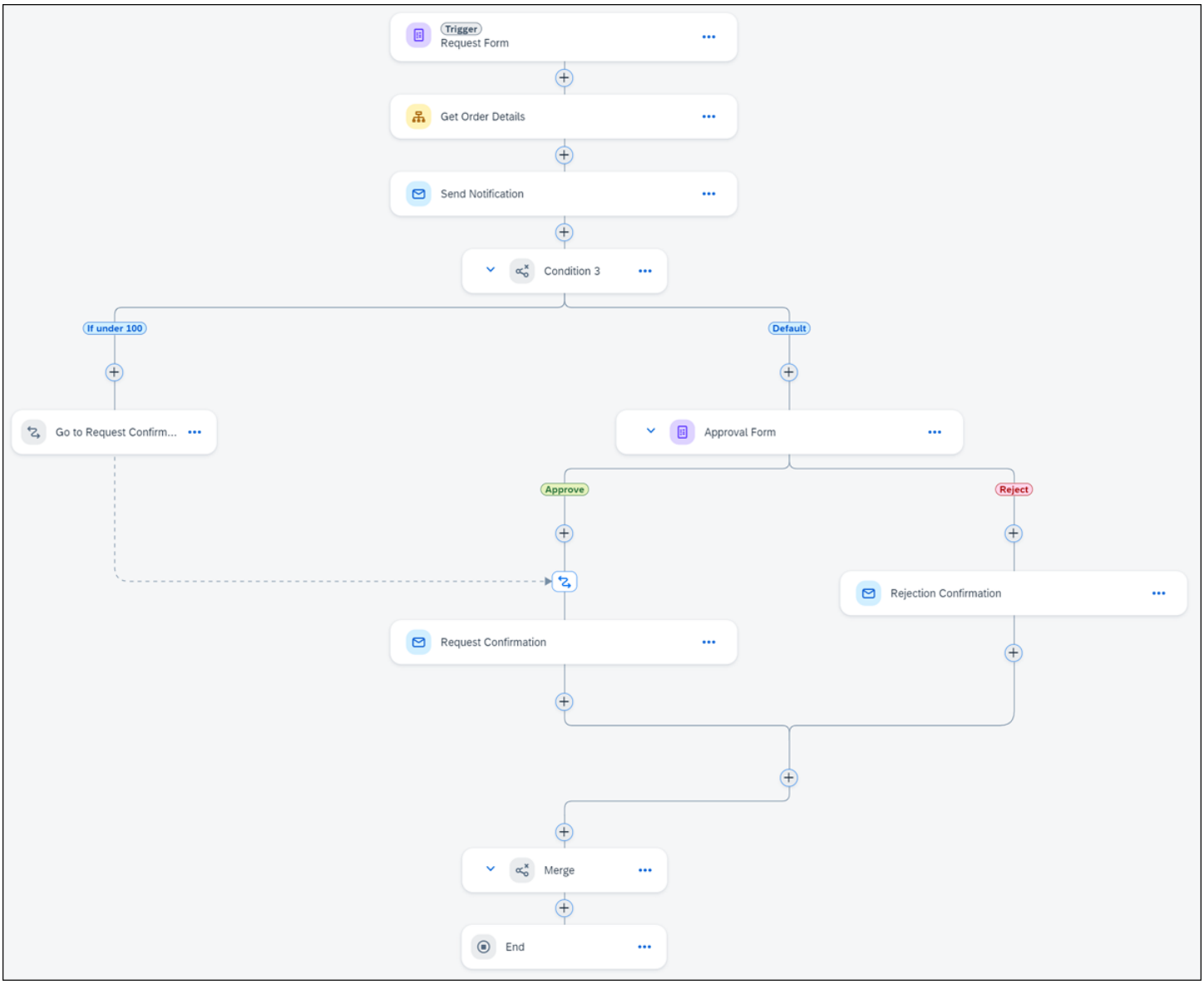
Artifacts help you to configure common steps or tasks needed for a process to run successfully. And the number and order of the artifacts you use depend on the process you're creating. As such, there are no mandatory artifacts needed in a business process project.

To help explain the artifacts available in SAP Build Process Automation, we'll divide them into two categories: Process and Automation

Process Artifacts

These include: Process editor, forms, approval forms, automations, decisions, and visibility scenarios

In this example, process artifacts have been used to create an investment request process:

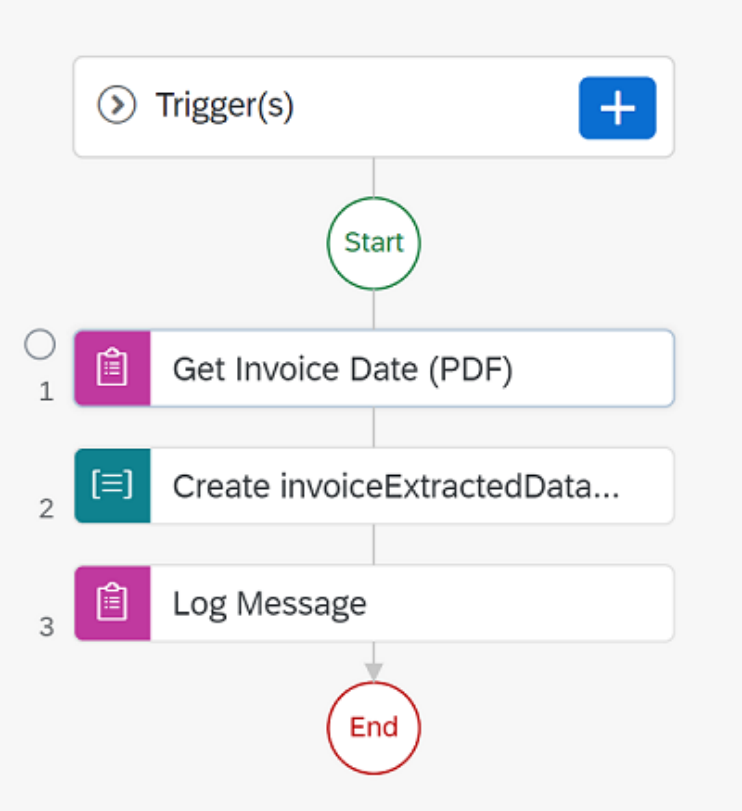


For an overview of available process artifacts, see Process Artifacts.

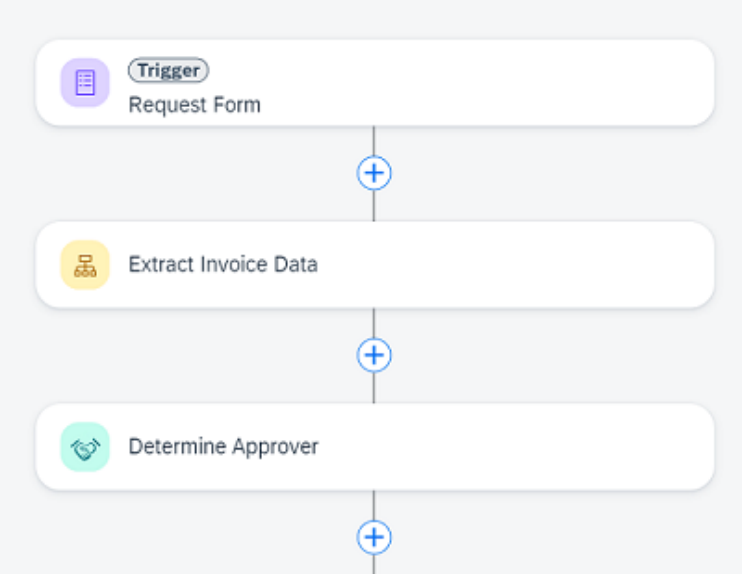
Automation Artifacts

These include: Alerts, applications, project launchers, user tasks, data types, files, and document templates

In this example, automation artifacts have been created to extract data from a PDF:



With this automation then added to a process:



For an overview of available automation artifacts, see Automation Artifacts.

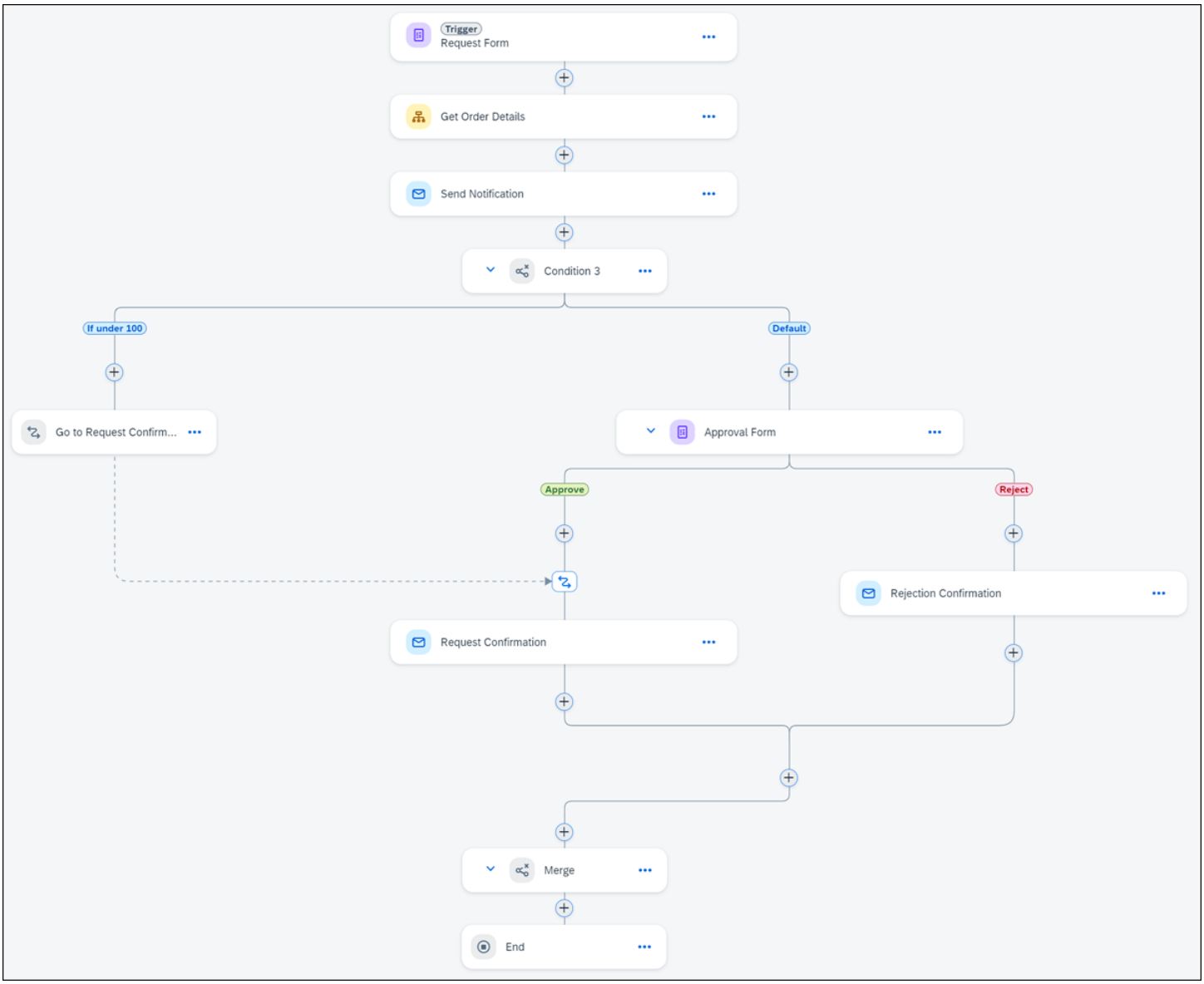
Process Artifacts

Process artifacts allow you to both create a multistage process and to configure how that process runs and is monitored.

Process Editor

This is a visual canvas on which you map out your business process from start to finish. Other process artifacts are then added to this canvas, with process controls and connectors used to decide how information flows when the process is running.

For example, the process editor was used to create this basic investment request process:



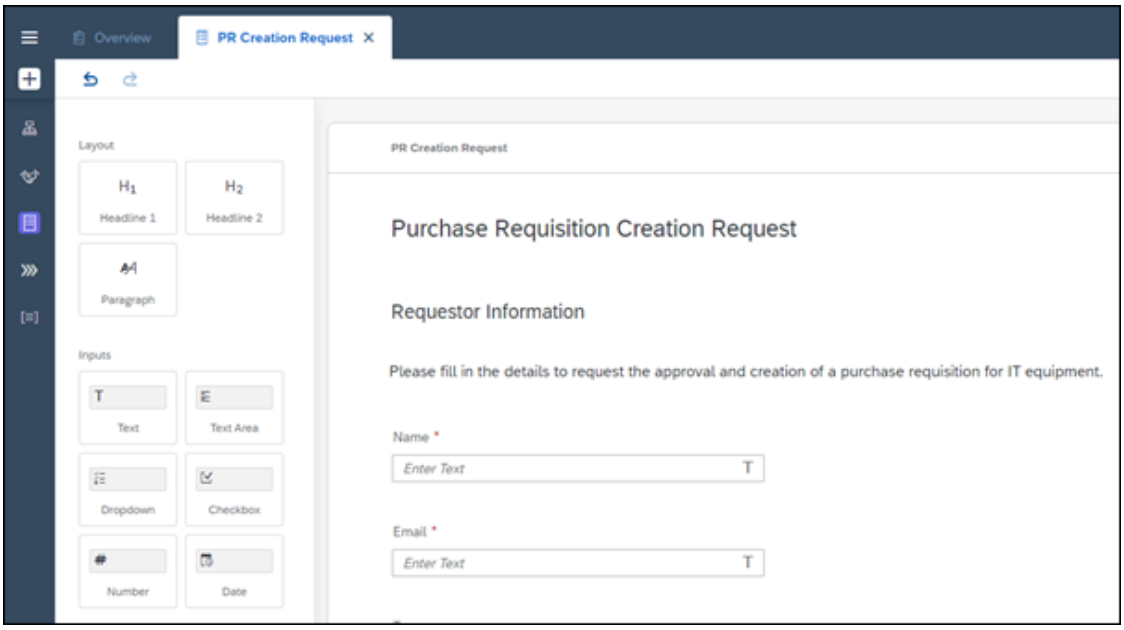
This canvas view is the 'behind the scenes' of the process and isn't shown to a person participating in that process.

For more information about using the process editor, see: Create a Business Process

Forms

Interactive forms are created by adding text elements and input fields to a blank page. Forms can be both starting triggers for a live process or used as optional steps later in the same process.

In the following example, a request form has been created:

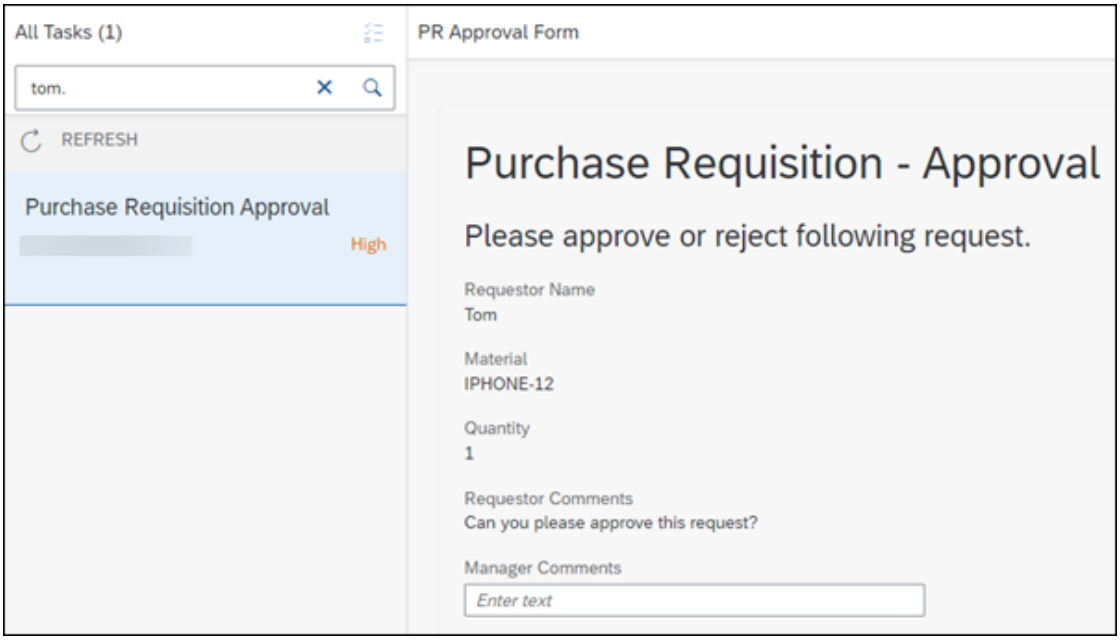


Published forms are often how someone submitting a request interacts with the process. Here they access a URL or a SAP Build Work Zone tile, enter their request details, then click 'Submit'.

For more information about creating and adding forms to your process, see Create a Form

Approval Forms

Based on the information submitted in the request form, approval forms share the request with those responsible for reviewing them. The assigned approvers for the process then decide whether to approve, query, or reject the request. Submitted requests appear as tasks in the approver's inbox, helping them to manage requests in one system.

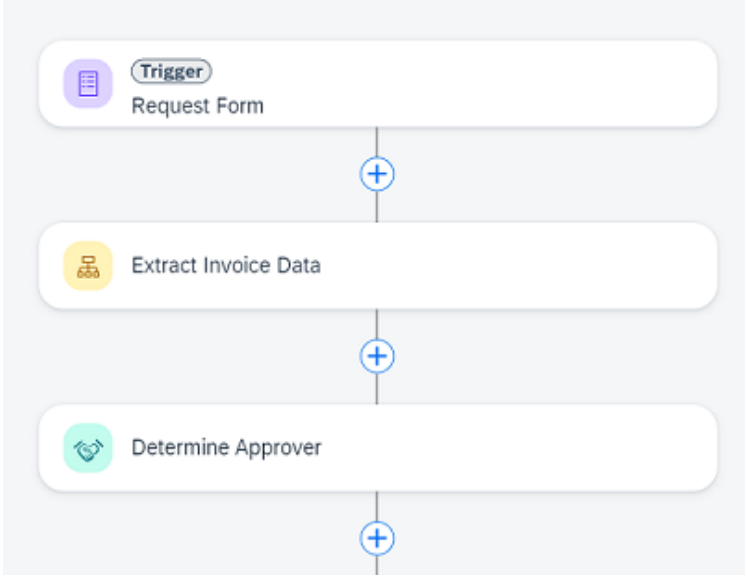


For more information about creating and configuring approval forms, see Create an Approval Form

Automations

The use of intelligent bots to automate manual, repetitive tasks such as retrieving data from a spreadsheet or submitting information to a database. Automations can be both included in a business process, and configured using the process editor, or used as stand-alone projects. That's why you'll find them mentioned here, too.

In the following example, an automation that extracts invoice data has been added to a process:

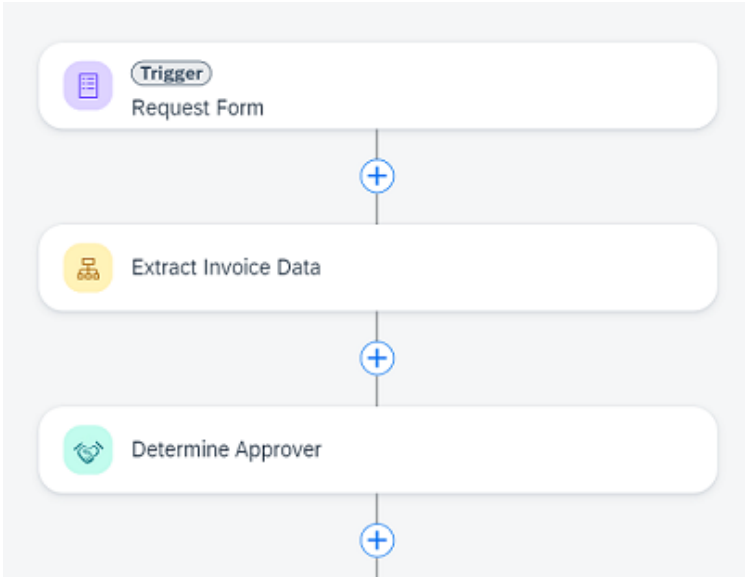


For more information about creating automations, see Create and Design Automations

Decisions

Effective processes respond and react to the information available. The way they behave in SAP Process Automation, and the direction the process moves, is based on adding and configuring process decisions. For example, if a request comes in from someone in Location A, then root that process to Location A's purchasing team.

In this example process, a decision determines who should approve the submitted request:



Using decisions, one process can effectively handle many scenarios at once.

For more information about creating and configuring process decisions, see Create a Decision

Visibility Scenarios

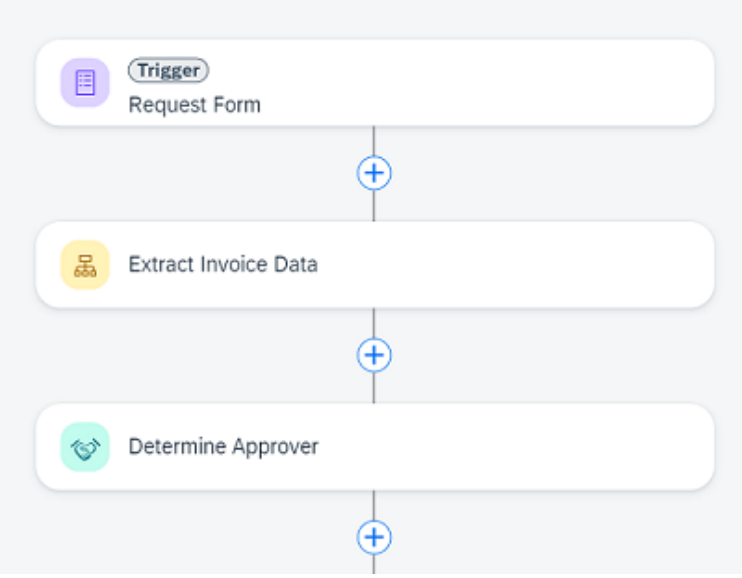
While not added to the process diagram, a visibility scenario allows you to configure and then monitor live versions of the process. By selecting the processes and choosing the type of information displayed, you have greater insights into the efficiency of your process (and react where necessary).

For more information about configuring visibility scenarios for your process, see Configure a Visibility Scenario

Automation Artifacts

Automation artifacts can be used to create and run an automation. These artifacts are created in the project overview area and can then be added to and configured in an existing automation. As such, they can't be used independently in your business process.

In this example, an artifact has been created to extract information from an invoice:

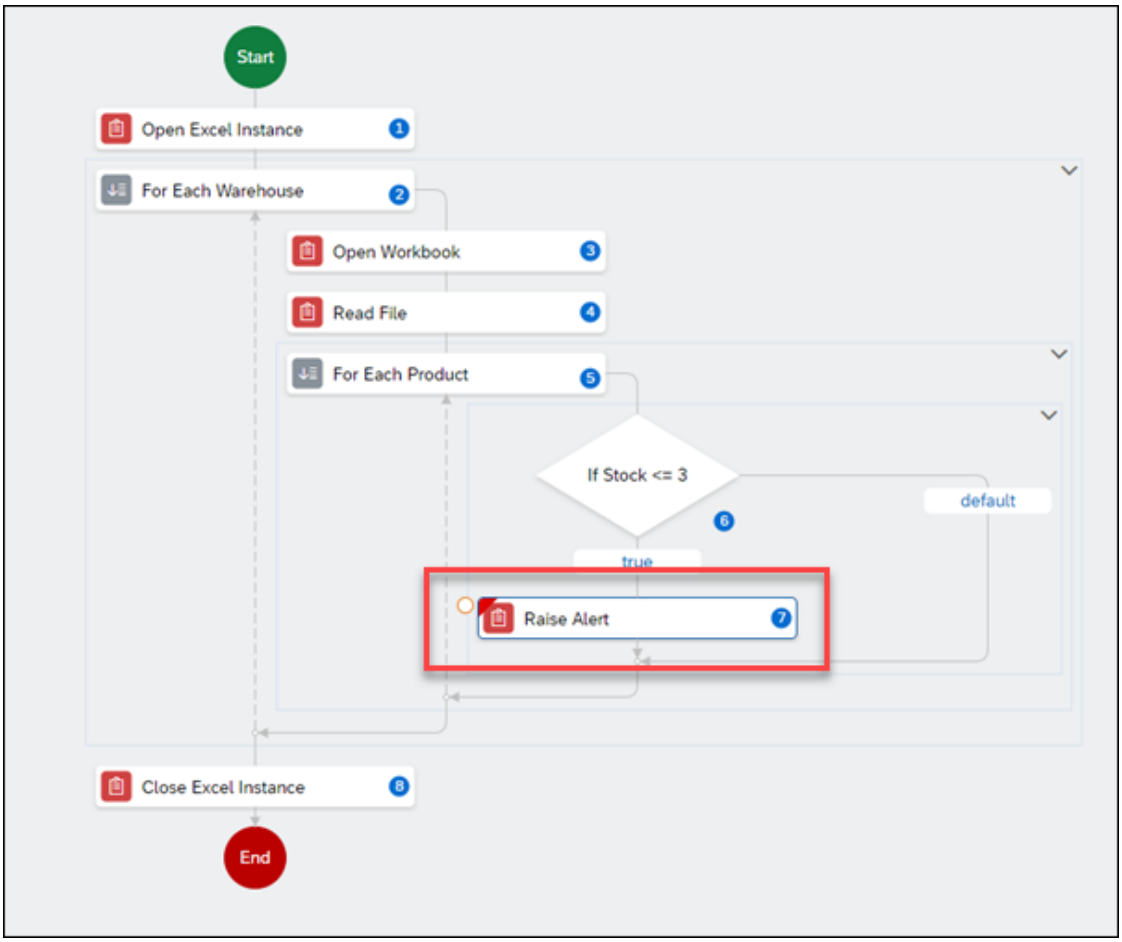


As you can see, an automation can be both a stand-alone project and included in a wider process. In both situations, there are several automation focused artifacts available in SAP Build Process Automation that help to build more complex, intelligent bots.

Alerts

These identify when specified business events occur while an automation is running. For example, when an item is out of stock or a limit has been reached. Alerts are then enhanced using alert handlers. These trigger an email to be sent to the people or system who need to know this information.

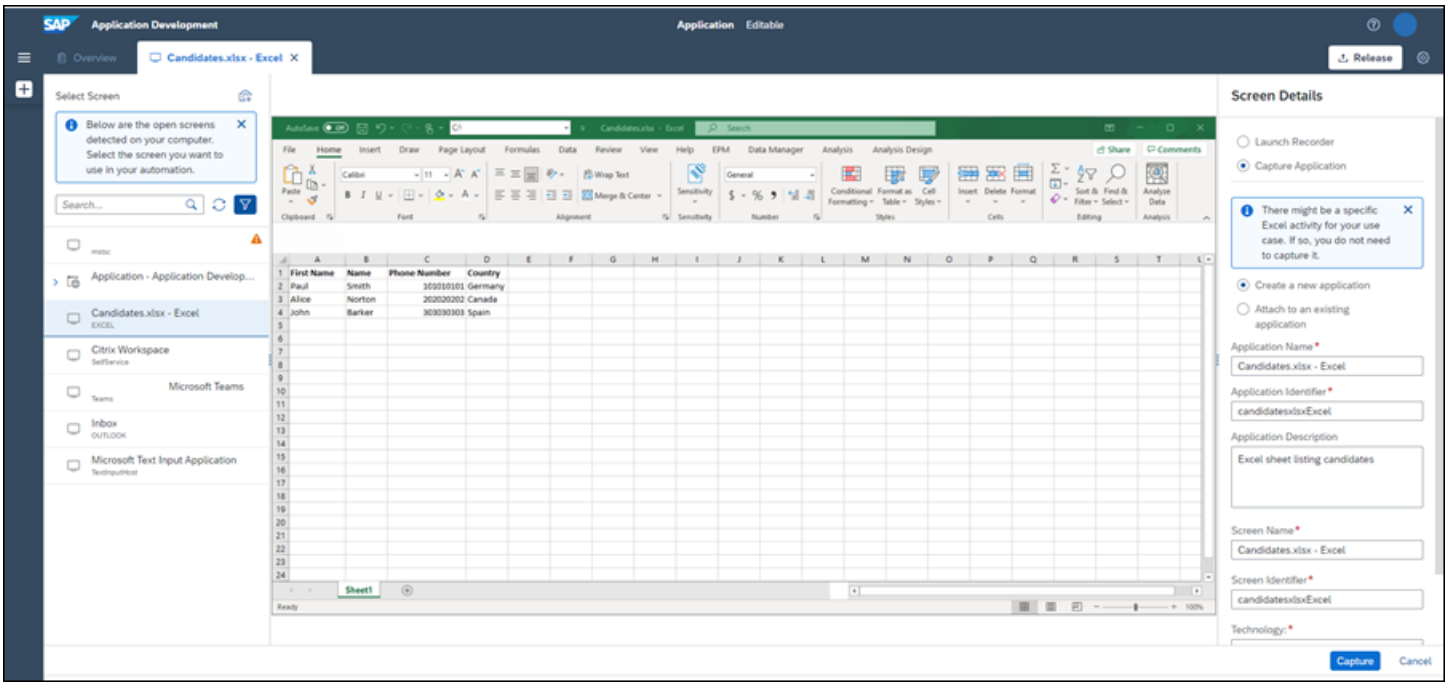
In this example, an alert has been created to notify those responsible when stock level has fallen under 3 units:



For more information about adding alerts to your automation, see Create an Alert

Applications

This artifact allows you to identify applications to control during a live automation, such as a website or software. Known as capturing and declaring applications, the automation can be configured to recognize a screen and interact with the elements displayed.

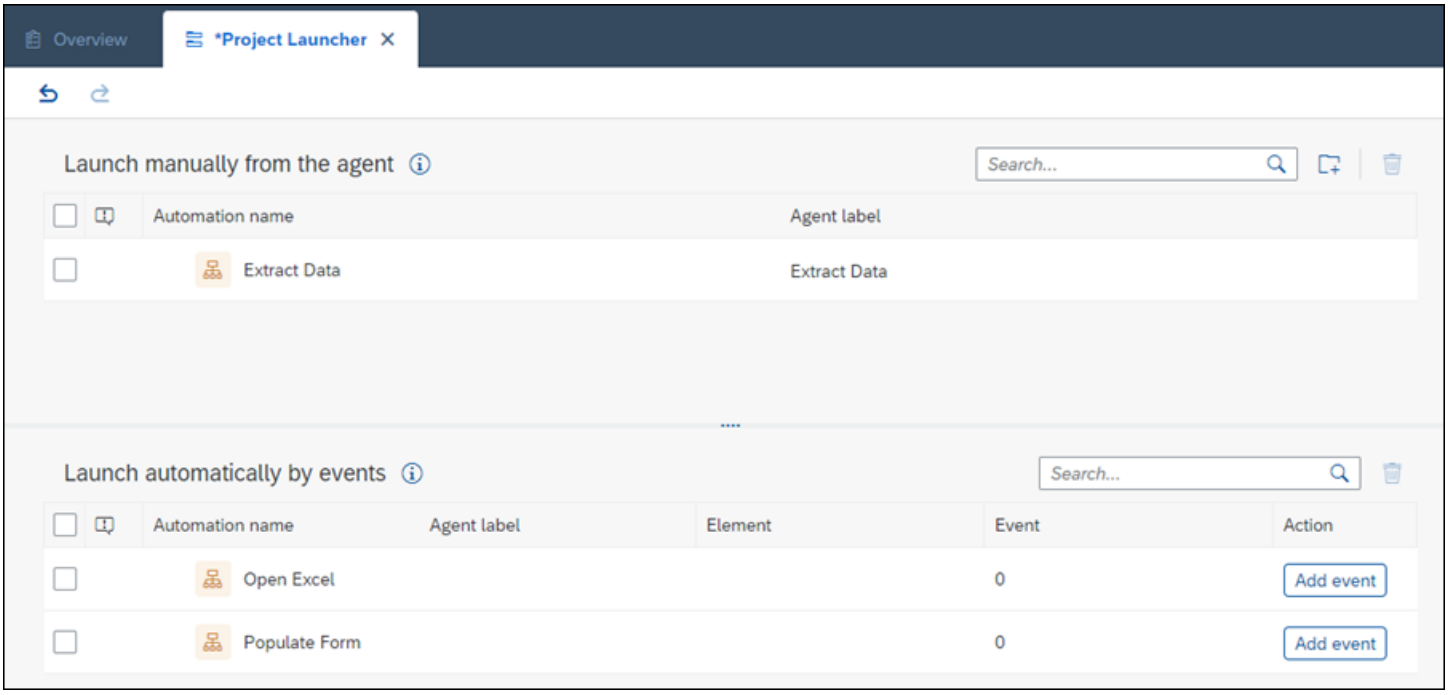


For more information about capturing applications with an automation, see Capture and Declare Applications

Project Launcher

Attended automations (those run under human supervision) can be started from a 'systray' agent or system tray agent on your device. By designing a project launcher, you configure which automations can be started (or launched) from that agent. These can either be run manually or automatically based on meeting certain factors.

In this example, the project launcher has been configured to run three automations from the systray:



For more information about creating a project launcher, see Create a Project Launcher

User Task

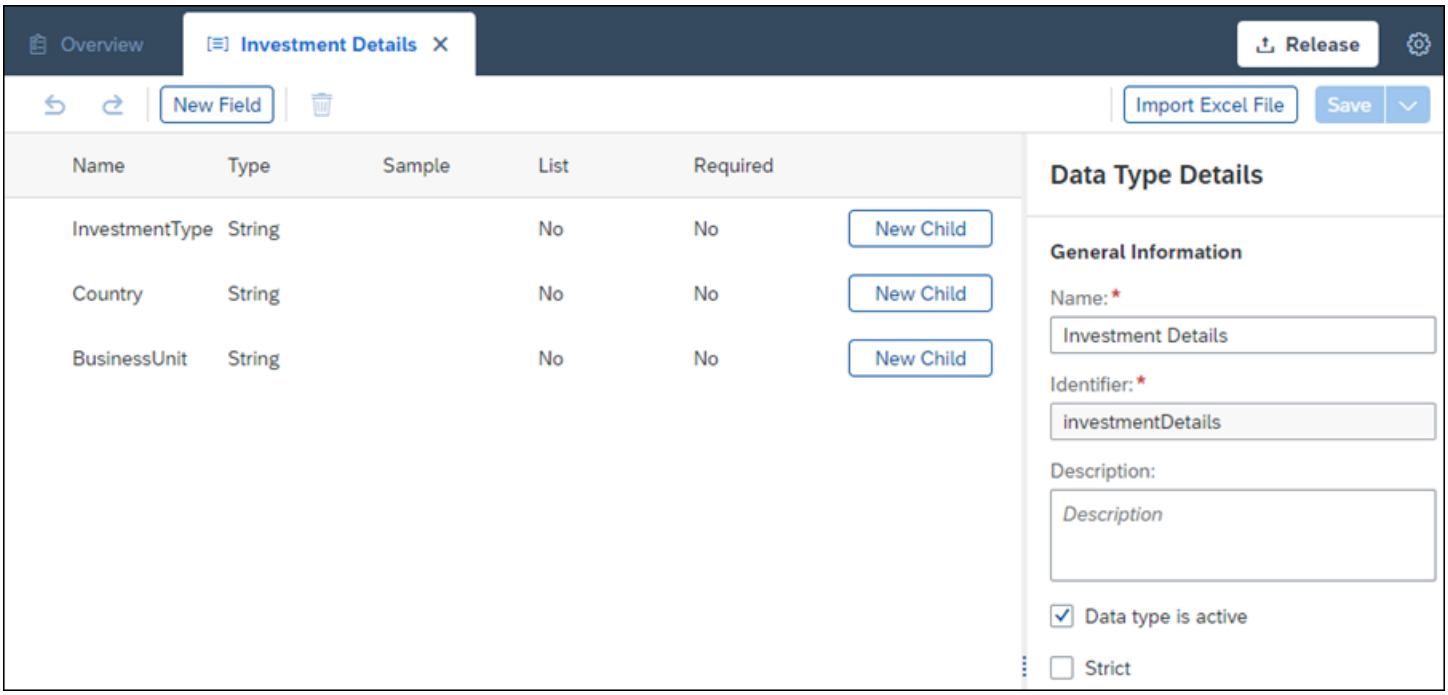
User tasks take the form of messages received in a person's inbox while a process is running. When creating a user task, you define in a message a specific action that needs to be done by the recipient. For example – to enter personal information, attach a file, or choose from a list of options.

For more information about creating user tasks for your automations, see Create a User Task

Data Type

Most automations and business processes require data to be inputted, stored, and shared to successfully run. This data always depends on the specific process or those running it, however. As such, there are different data types for different circumstances. By manually creating data types, you can configure and use the data needed for your specific process.

In this example, three data types have been created to be used in an automation:



For more information about creating and maintaining data types, see Create a Data Type

Files

Files can be both created and edited in SAP Build Process Automation (text, YAML, JSON, and XML) or uploaded and stored within a project. These files can then be integrated and used within an automation, too.

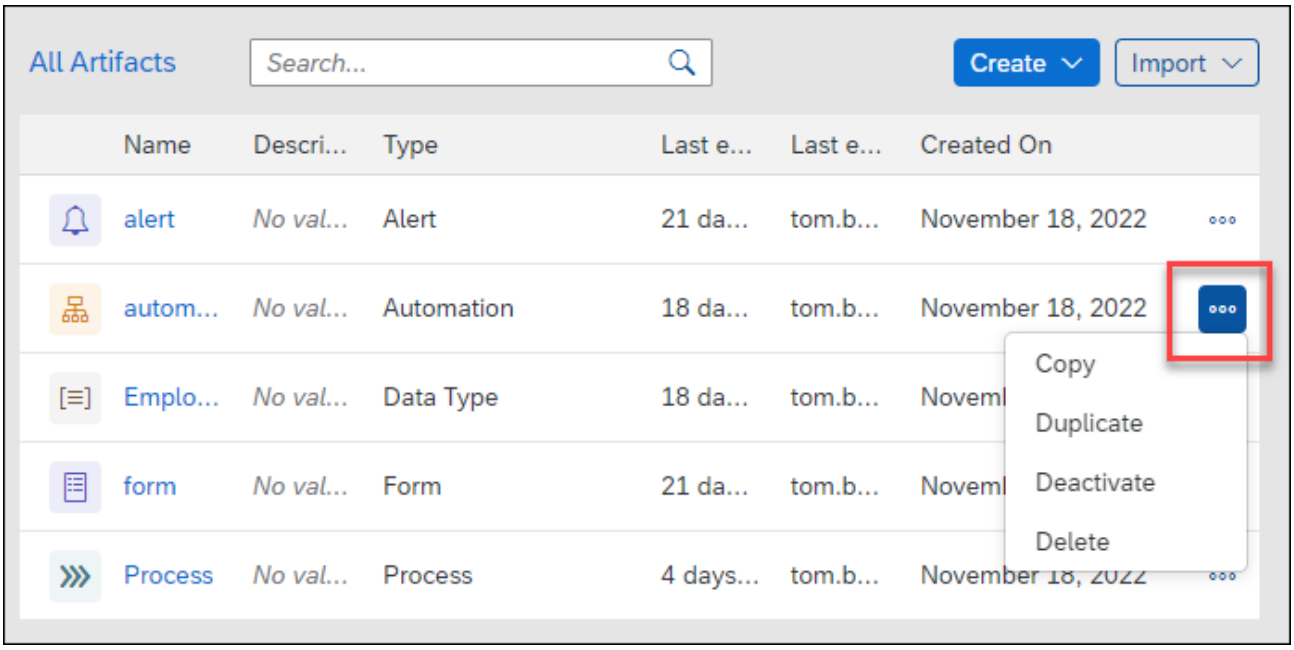
Document Template

Information can be extracted from a document while an automation is running. For the automation to do this, a document template can be created or uploaded to the project. This template is then used to identify the data available and extract it when directed to do so. For example – retrieving customer information from a frequently used invoice layout.

For more information about working with document templates, see: Document Processing and Information Extraction

Manage Existing Artifacts

In SAP Build Process Automation, you create a business process project using a combination of one or more process focused skills or building blocks. These skills are known as **artifacts**. Once created, you can manage your artifacts from your business process project overview page.



Options

When managing your artifacts, you've the following options:

Artifact	Create	Read	Update	Delete	List	Rename	Activate / Deactivate	Copy / Paste / Duplicate	Import Inside
Action Project	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Alert	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	N/A
Application	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Partly (Excel files only)
Data Type	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	N/A
Decision	Yes	Yes	Yes	Yes	Yes	Yes	No	No	Yes
Document Template	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
File	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Form	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	N/A
Approval Form	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	N/A
Process	Yes	Yes	Yes	Yes	Yes	Yes	No	No	N/A
Project Launcher	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Only Copy / Paste	N/A
User Task	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	N/A
Visibility Scenario	Yes	Yes	Yes	Yes	Yes	Yes	No	No	N/A

Using Artifacts Within Other Artifacts

Many artifacts can be used within existing artifacts, such as forms being used within a process.

ℹ Note

If you modify or delete an artifact used within other artifacts, this can affect the validity of these artifacts.

The following table summarizes the available options:

	Action Project	Alert	Application	Automation	Data Type	Decision	Document Template	File	Form	Approval Form	Process	P L
Automation	Yes	Yes	Yes	Yes	Yes	N/A	Yes	Yes	N/A	N/A	No	C
Data Type	N/A	N/A	N/A	N/A	Yes	N/A	Opposite	N/A	N/A	N/A	N/A	N
Decision	N/A	N/A	N/A	N/A	Yes	N/A	N/A	N/A	N/A	N/A	N/A	N
Document Template	N/A	N/A	N/A	N/A	Yes	N/A	N/A	N/A	N/A	N/A	N/A	N
Form	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N
Approval Form	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N
Process	Yes	Yes (via an automation)	Yes (via an automation)	Yes	Yes	Yes	Yes	No	Yes (via an automation)	Yes	Yes	N
Project Launcher	N/A	N/A	Yes	Yes	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N
User Task	N/A	N/A	N/A	N/A	Yes	N/A	N/A	N/A	N/A	N/A	N/A	N
Visibility Scenario	N/A	N/A	N/A	N/A	N/A	Yes	N/A	N/A	Yes	Yes	N/A	N

Create a Business Process

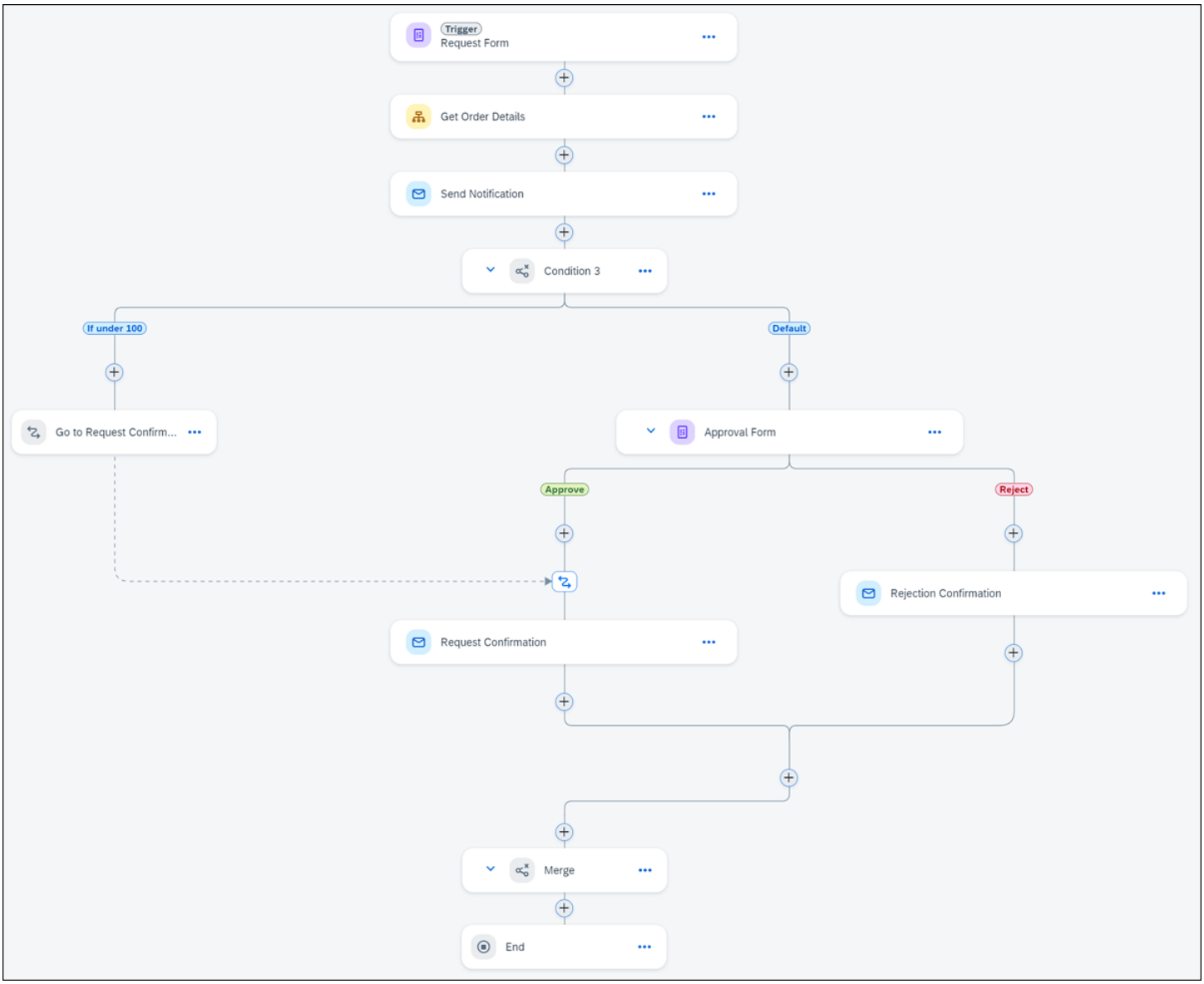
With SAP Build Process Automation, you can visually create a business process version using a combination of artifacts (such as forms and decisions) and process controls (such as branches, conditions, and mail notifications).

Context

A business process is started by defining a trigger, an event that indicates to your SAP Build Process Automation tenant to start a process instance.

Process triggers can be a form, such as a request form, an API call, where an external system starts the process, or an event.

In this example, the process is triggered when a request form is submitted:

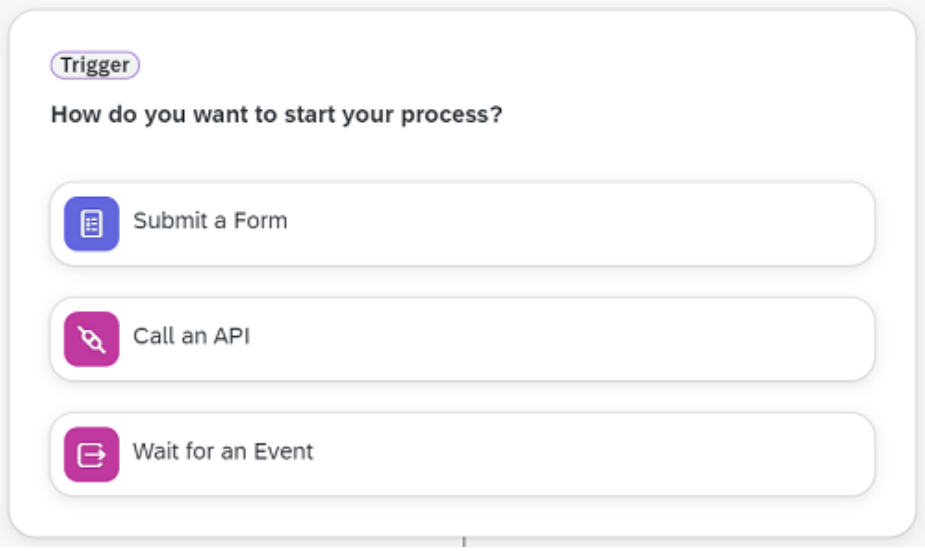


The process then continues with a combination of Process Artifacts (such as forms, approval forms, and decisions) and optional process controls.

Procedure

1. To create your project and process in the **Lobby**, choose **Create** > **Build an Automated Process** > **Business Process**.
2. Enter a name and description for your project, and choose **Create**.
3. Enter a name, identifier, and optionally a description for your process.
4. Choose **Create** again.

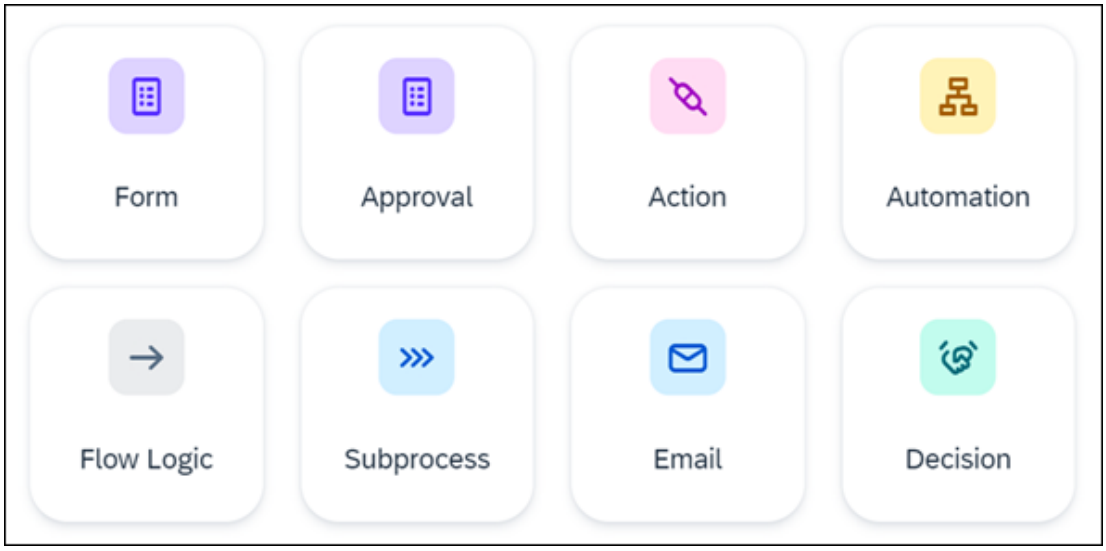
   The process editor loads.
5. Configure your process trigger, choosing between:
   - API trigger - Configure an API Trigger to Start a Process
   - Form trigger - Configure Settings for Forms and Approval Forms
   - Event trigger - Create Event Triggers



6. Click + and select your required process artifacts or process controls. Sometimes, you get to select again whether to create a new artifact or reuse an existing one. Repeat this step as necessary to continue building your process.

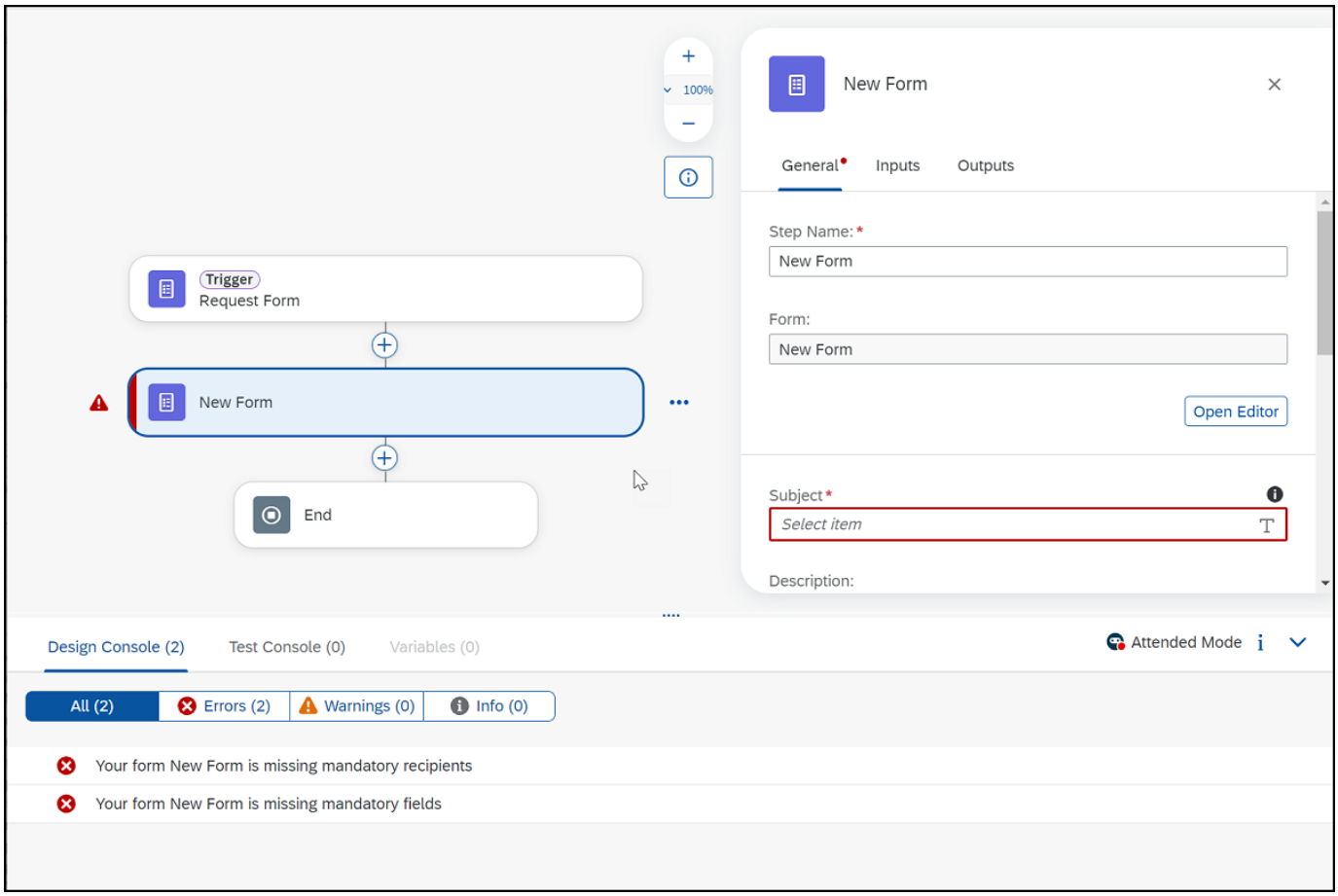
> **Note**
>
> You can change the order of your process steps using drag and drop. The selected step is automatically connected to the process.

The image is interactive. Click any tile to see how to add the respective process artifact.



7. Review and correct all process errors, indicated by a red icon on the impacted artifact.

   In this example, the form artifact is missing a subject. The design console displays the error details.



8. Once the process is completed and all errors are corrected, save your process.

   The project is ready to be released and deployed.

Related Information

Release a Project

Deploy a Project

Configure Process Variables

The outputs of any step that you add to your process can be consumed as inputs for the subsequent steps in a process. In addition, you can create custom variables on a project level that aren't bound to a particular step.

Context

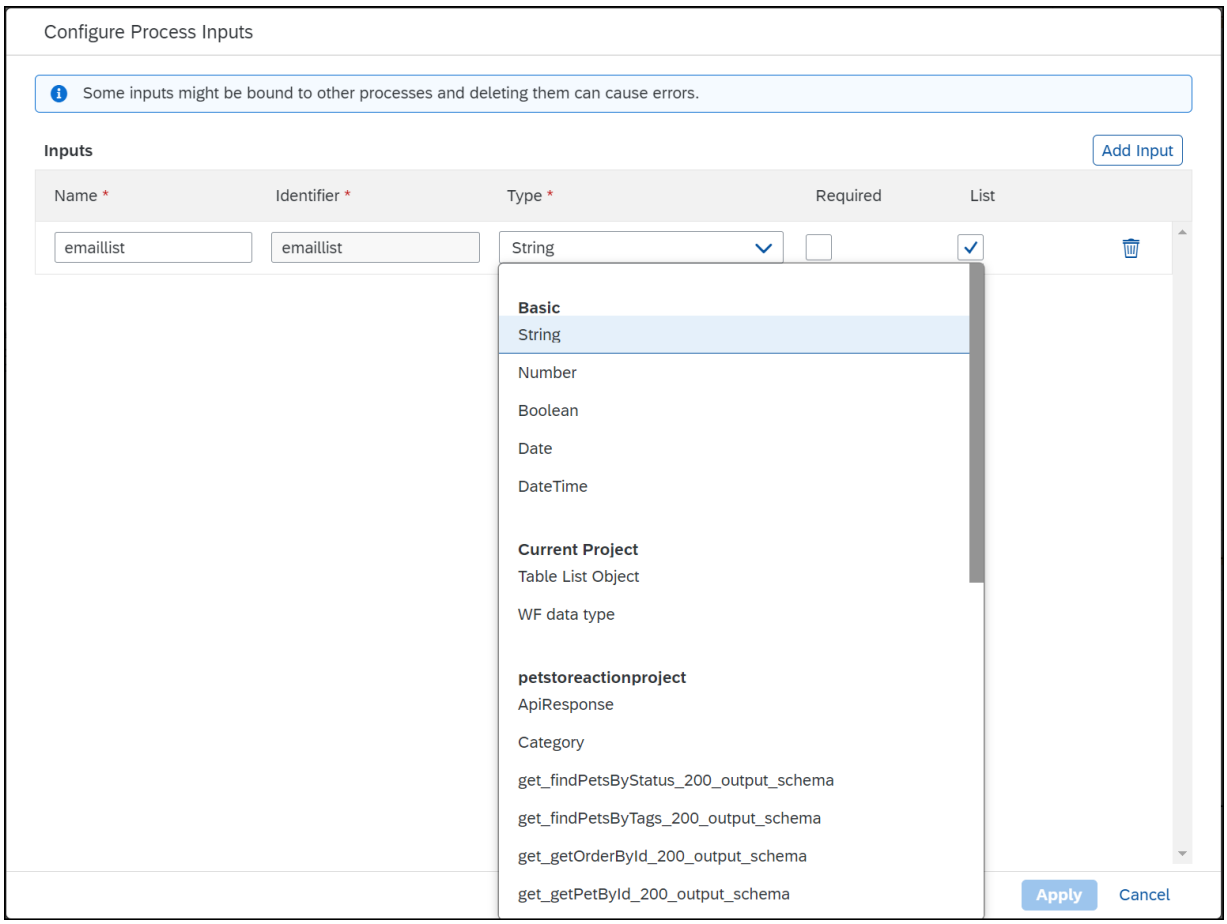
If you add actions or subprocesses to your process, their data types become available as step input to any later step in the process. For data types that contain deep structures, such as business partner information, this saves you much repetitive work.

Using custom variables, you can make any information available at the global level at any stage in the process. For more information on the use cases, see Use Cases for Custom Variables.

Procedure

1. For the input variables, add an action to your process. See Add Actions to a Process.
2. Click anywhere on the canvas and open the side panel.
3. On the **Process Details** side panel, choose **Variables** > **Process Inputs**.
4. To configure process input variables, next to **Process Inputs**, choose **Configure**.
   a. Choose **Add Input**, and enter the following data:

- **Name** and **Identifier**: For example, `businessPartner`.
- **Type**: Select any data type from the list:
  - The basic data types
  - All data types of the current project
  - All date types of any dependent project. These dependencies are, for example, created when adding actions or subprocesses to the project.



- To create a list, select **List**.
- To make this variable a required entry, select **Required**.

5. Next to **Process Outputs**, choose **Configure** and add outputs in the same way as you added inputs.
6. To configure custom variables, choose **Configure** next to **Custom Variables**. Then, choose **Add Variable**, and enter the following data:
   - **Name** and **Identifier**: For example, `businessPartner`.
   - **Type**: Select any basic data type from the list.
   - To create a list, check **List**.
7. Choose **Apply**.
8. Save your changes.

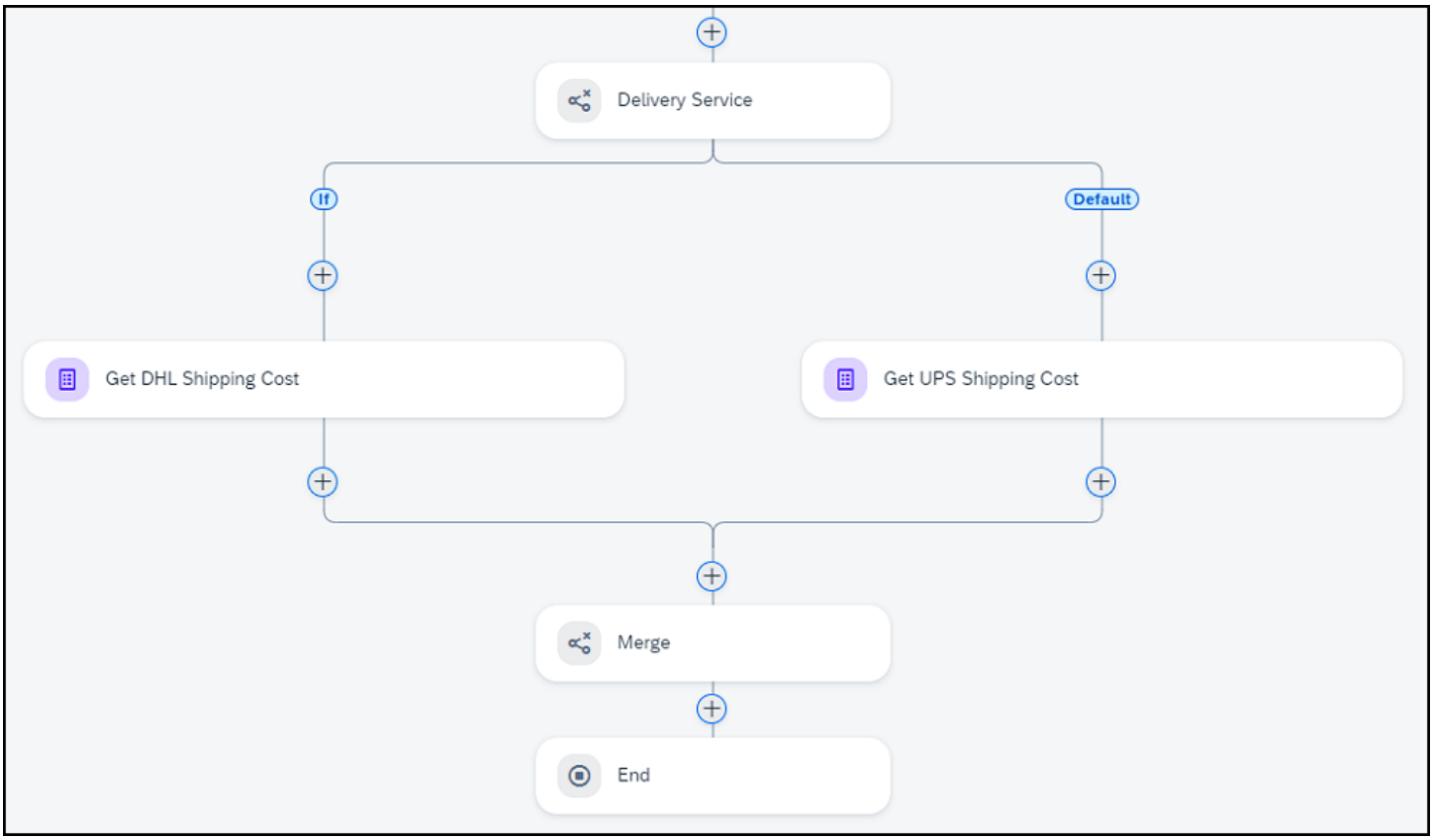
Use Cases for Custom Variables

Using custom variables, you can make any information available at the global level at any stage in the process.

Using custom variables, you can make any information available at the global level at any stage in the process. Custom variables have the following use cases:

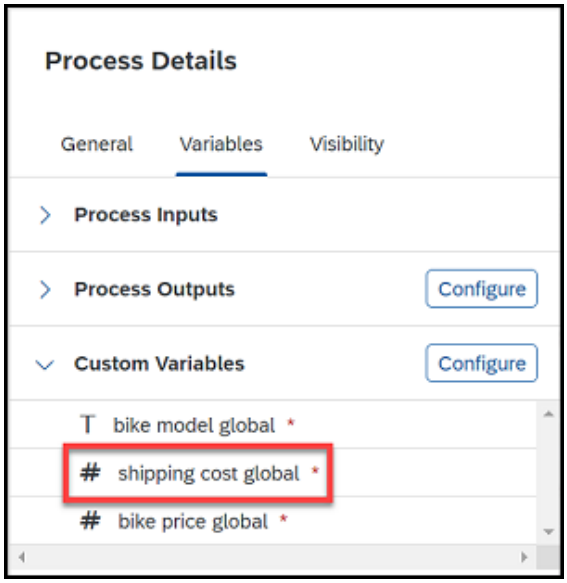
Independency of Branch Outcome

Make the step that follows a condition branching independent of the outcome of the conditional branches. Let's look at an example: We have a condition where the branches have steps for two shipping companies. The **Final Approval** form needs to cover both possible outcomes to map the **Shipping Cost** field. So, we map an independent custom variable instead of the specific process output from the steps in the branches.

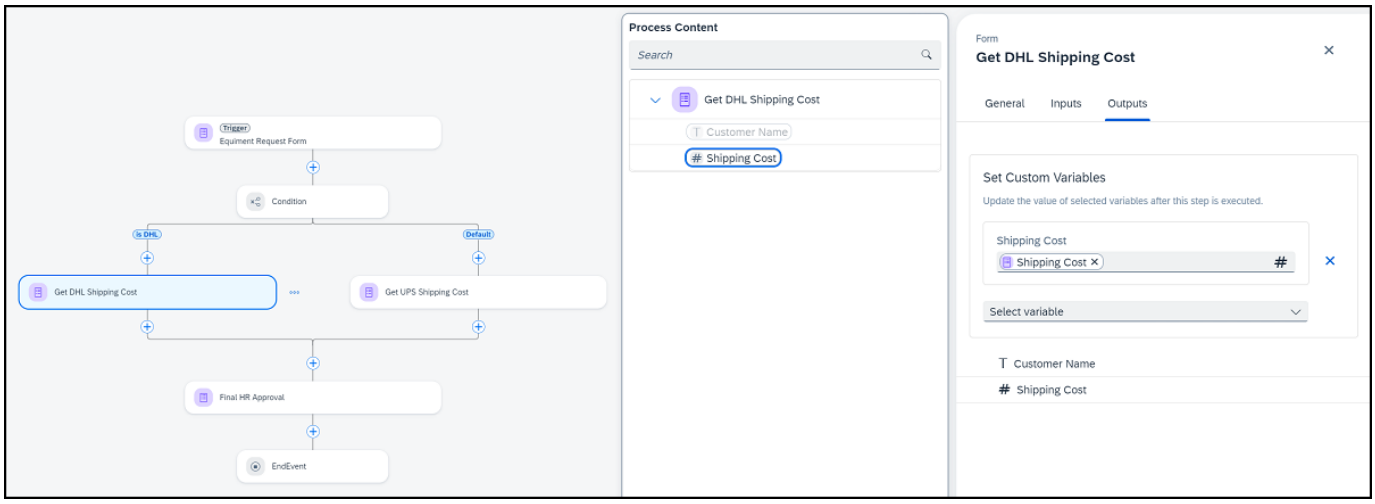


The following shows the order, in which the custom variable is inserted into the process.

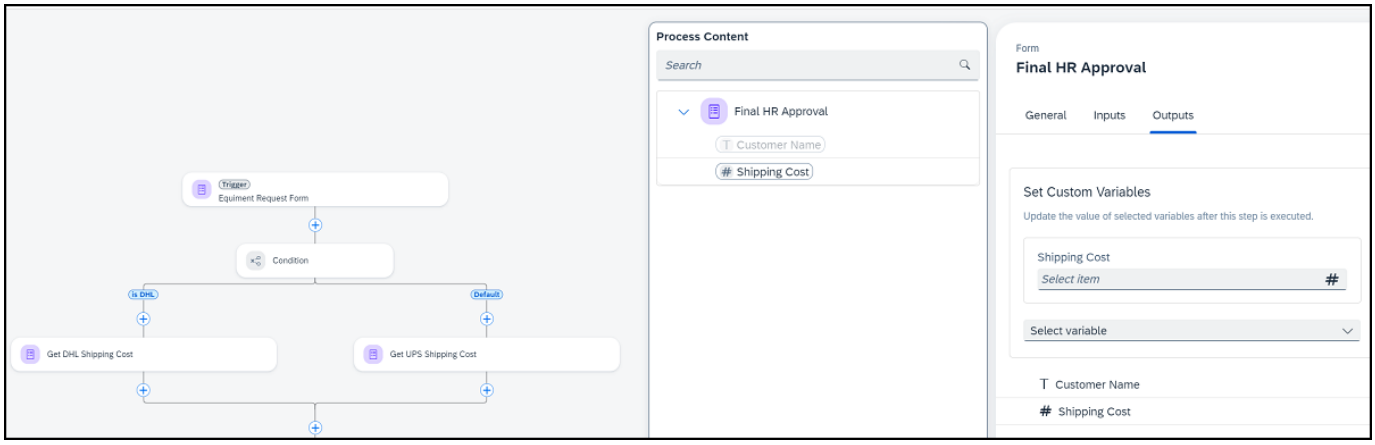
1. We define the **shipping cost global** custom variable for the process.



2. In the two steps in the branches, we use the new custom variable on the **Outputs** tab.



3. In the **Final Approval** form, we can now map the **Shipping Cost** field to the custom variable as well, and make it independent of the condition branches.

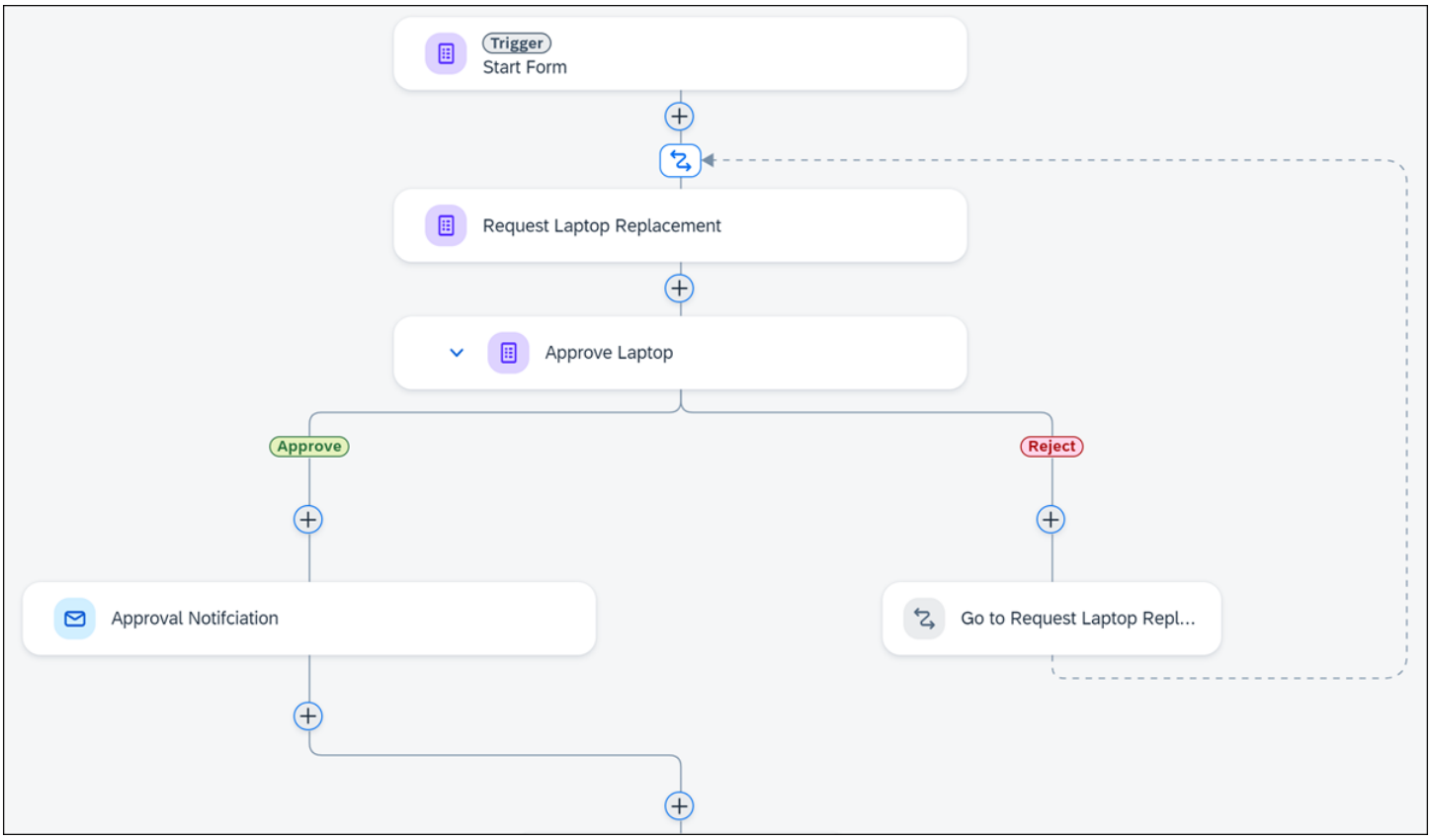


4. At runtime, the final approval form receives an input for the shipping cost, no matter which branch is used.

Rework Request

Send a request back for rework while preserving the already-entered values and the comments added by the parties involved.

Let's assume that a step needs rework before the approver accepts it. To send the process back to the previous step that needs rework, you can define a custom variable that is already available before the approval step outcome is available. In this way, the previous step can anticipate the result of the approval step and use the correct variable.



One Field, Multiple Processes

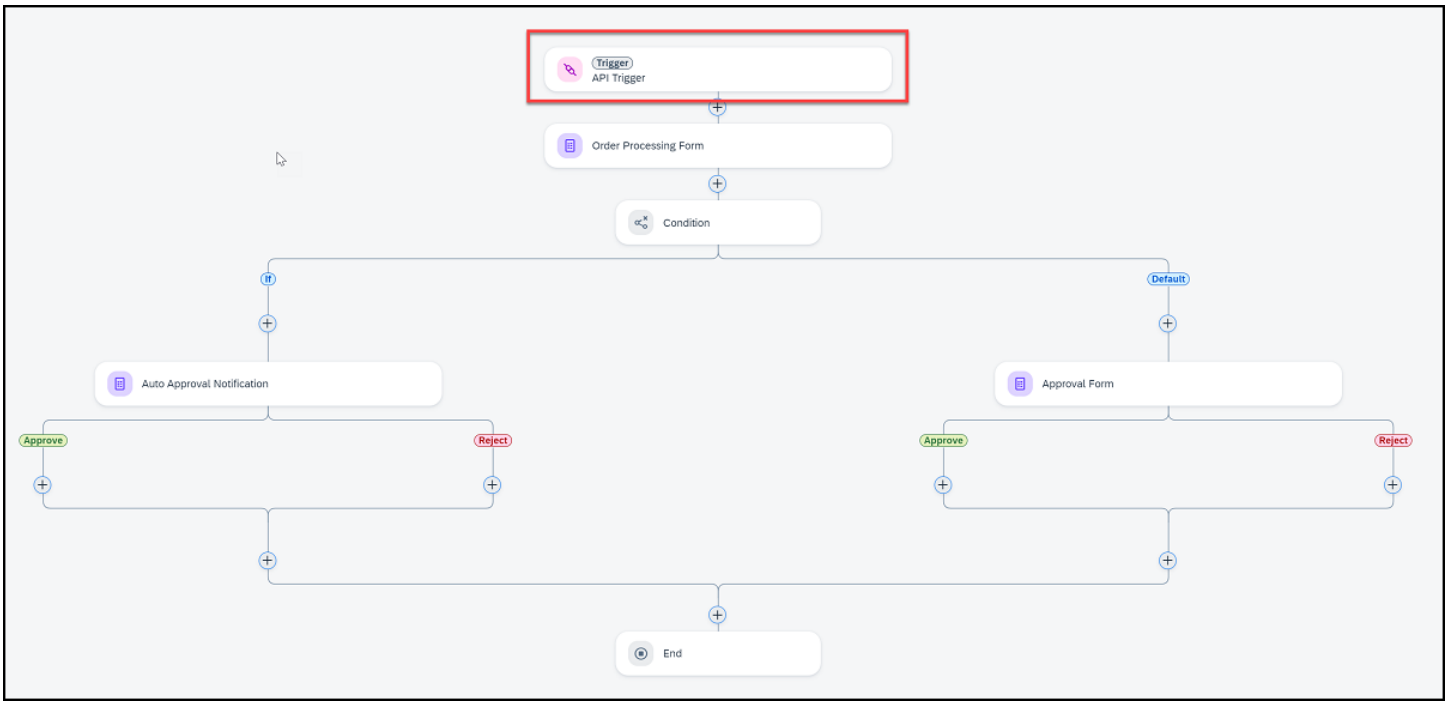
Multiple process steps edit the same field. This field is immediately available and doesn't need to be mapped and carried forward.

Configure an API Trigger to Start a Process

You can start an instance of your process using an API call, with the inputs for the call configured in the process editor. These inputs can then be used as input fields in your process, for example, in a form or approval form.

Context

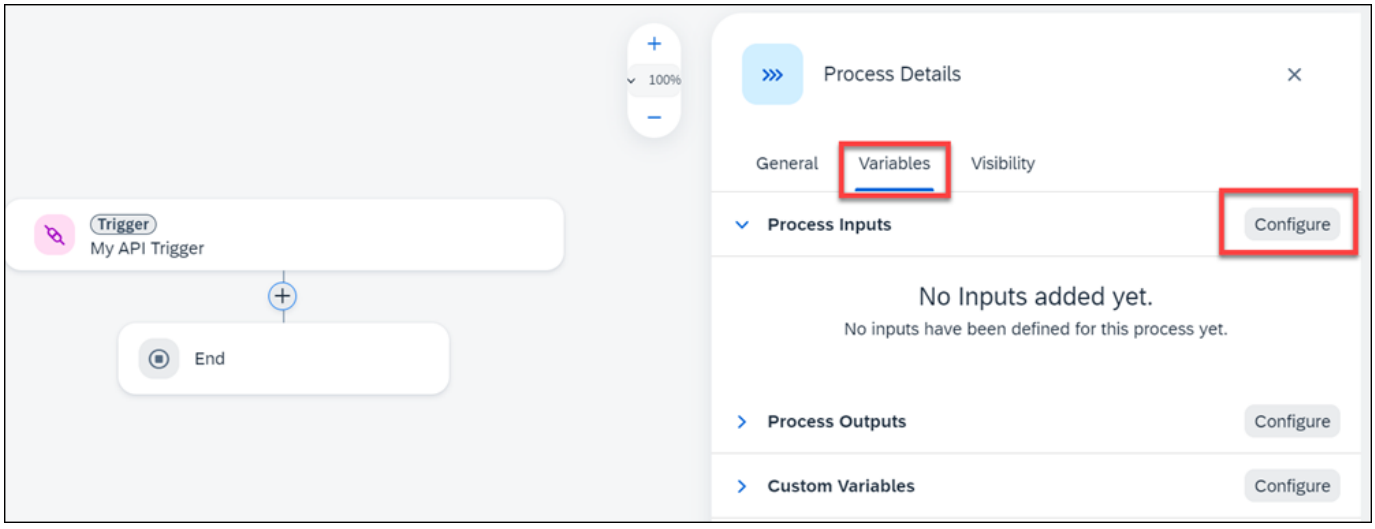
In this example, an API call is used to trigger a sales order process:



You can display such API triggers in the same overview as the automation triggers. For deployed processes, go to Monitor > Manage > Triggers from the lobby.

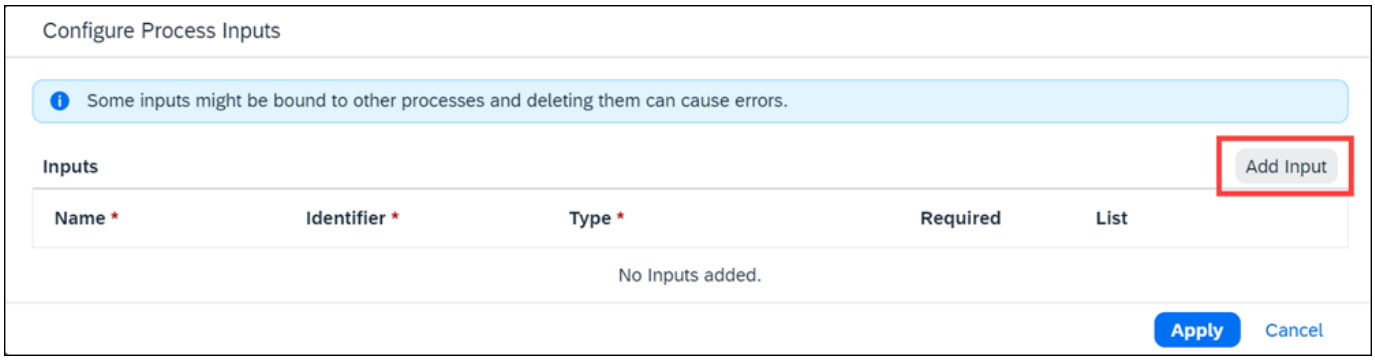
Procedure

1. On the Trigger tile in the process editor, choose Add a Trigger. Then, select Call an API.
2. Enter a name for your trigger and a description. The identifier is generated based on the name.
3. Click anywhere on the canvas to display the Process Details side panel.
4. Choose Variables > Process Inputs > Configure.

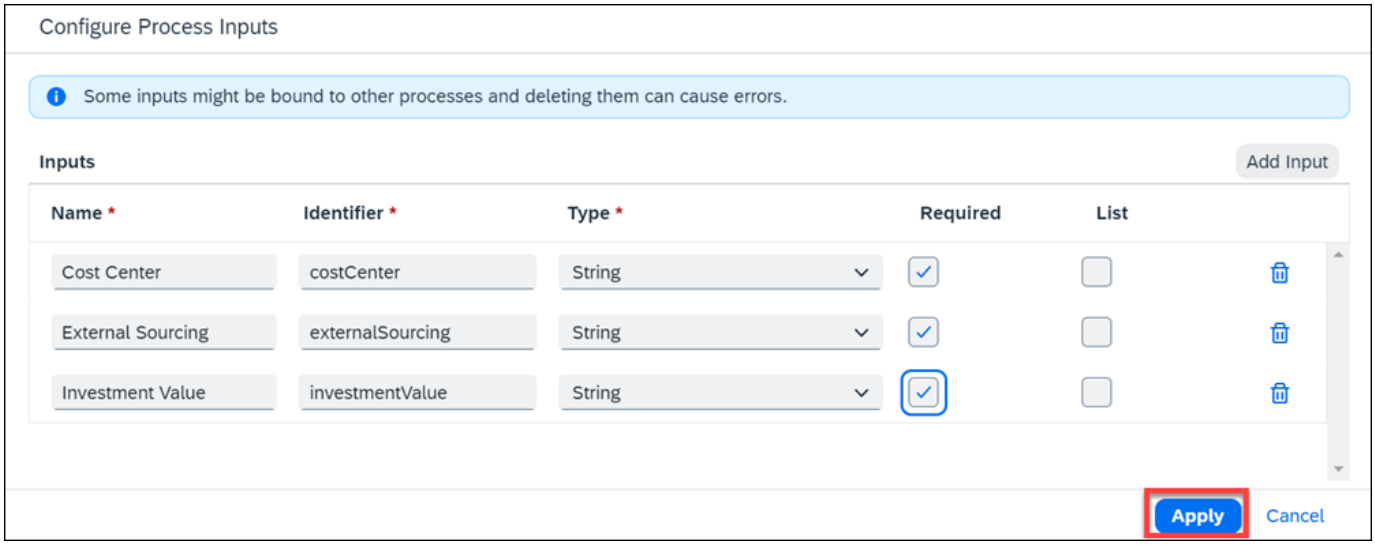


5. Click Add Input and then enter a Name, Identifier, and String, and specify if this is a Required Input.

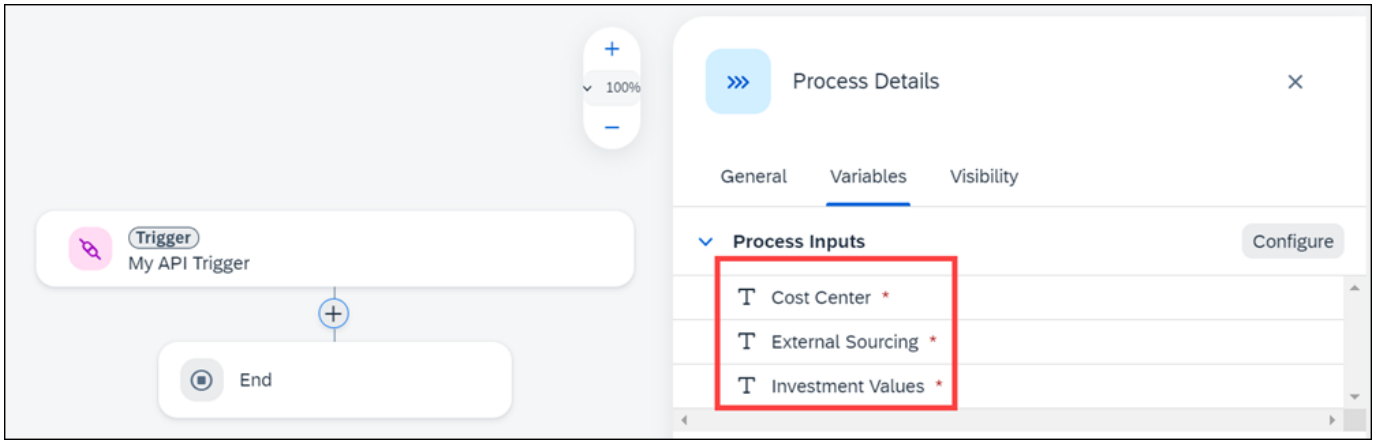
   Repeat this step to add all necessary inputs.



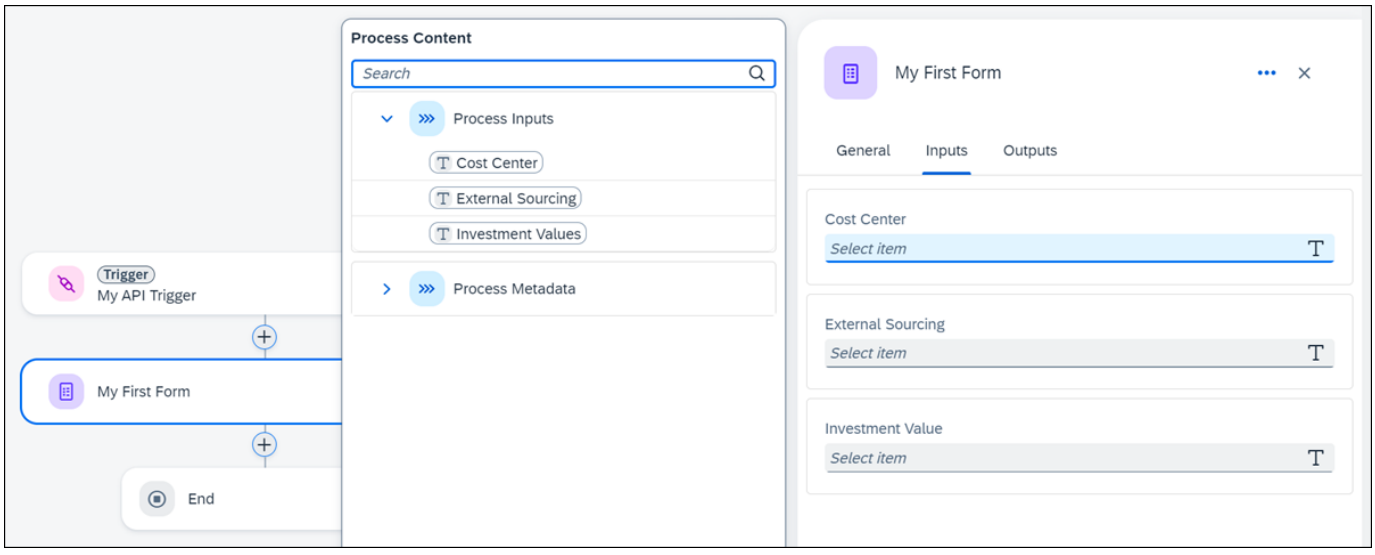
6. Choose Apply.



The inputs are displayed in the process settings side panel.



7. Once a form has been added to your process, you can then map these inputs to the form fields:



8. Save your changes.
9. With the API call set as your process trigger, you can now test the call using Postman. See SAP Community - How to Test Your API Call Trigger Using Postman.

> **Note**
>
> We support the ISO 8601 format for date and time:
>
> YYYY-MM-DD (2023-03-16)
>
> hh:mm:ss (15:33:16)

Create Event Triggers

Prerequisites

- Your SAP S/4HANA Cloud system is available and connected with SAP Build Process Automation, and is able to receive all required notification events. For more information, see Enable the Consumption of SAP S/4HANA Cloud Events.
- You have set up an Event Mesh service instance in your SAP BTP cockpit subaccount. For more information, see Setting Up SAP Event Mesh in BTP Cockpit.
- You have created your queue and your queue subscriptions. For more information, see Manage Queues.
- You have created a webhook subscription to subscribe to an event queue. For more information, see Manage Queue Subscriptions.
    - You have created a service key. For more information, see Create a Service Key for the SAP Build Process Automation Instance.
    - You have fulfilled the **Webhook URL** by using the URL from the instance of the **SAP Build Process Automation plan standard** > **Service Key** > `api endpoints` and adding `internal/be/v1/events`. As a result, your webhook URL should be as follows:

      `https://spa-api-gateway-bpi-eu-prod.cfapps.sap.hana.ondemand.com/internal/be/v1/events`.
    - You have inserted the **Client ID**, the **Client Secret**, and the **Token URL** by adding **oauth/token** to the latter. You can copy and paste these credentials from the service key.

Context

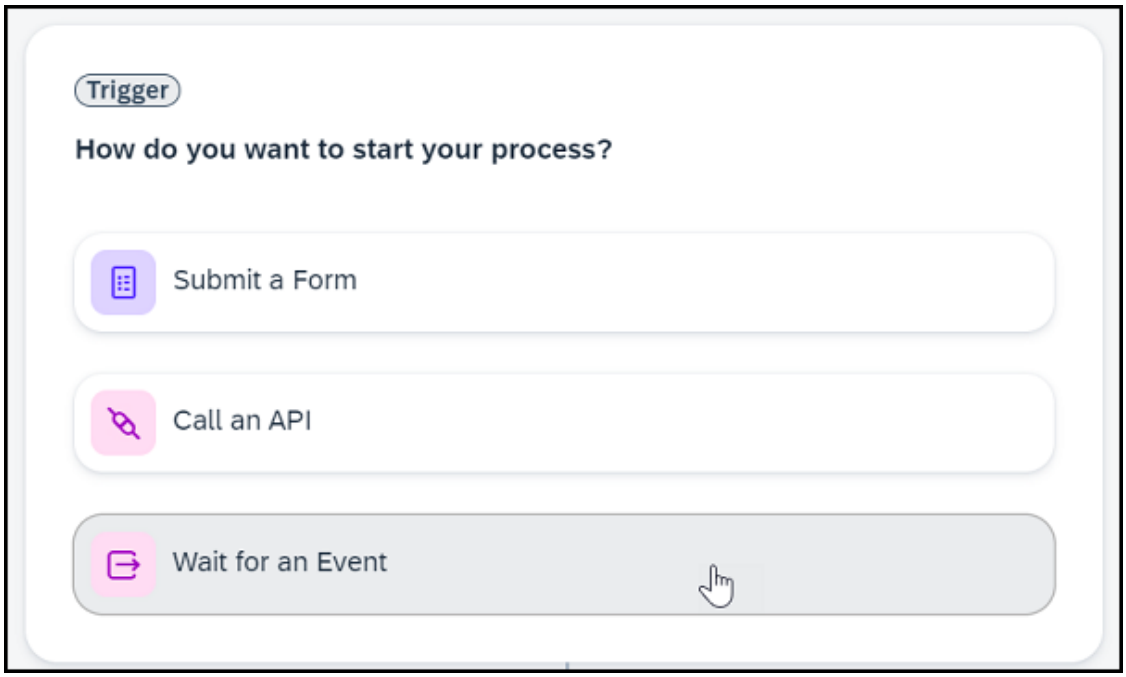
The event triggers onboard and listen to back-end events emitted from an external source system, and reacts to such events by triggering artifacts such as processes and automations.

All public events published on SAP Business Accelerator Hub for SAP S/4HANA Cloud and SAP S/4HANA are supported.

You already have created a project and can now add a process as an artifact that has to be started by an event trigger.

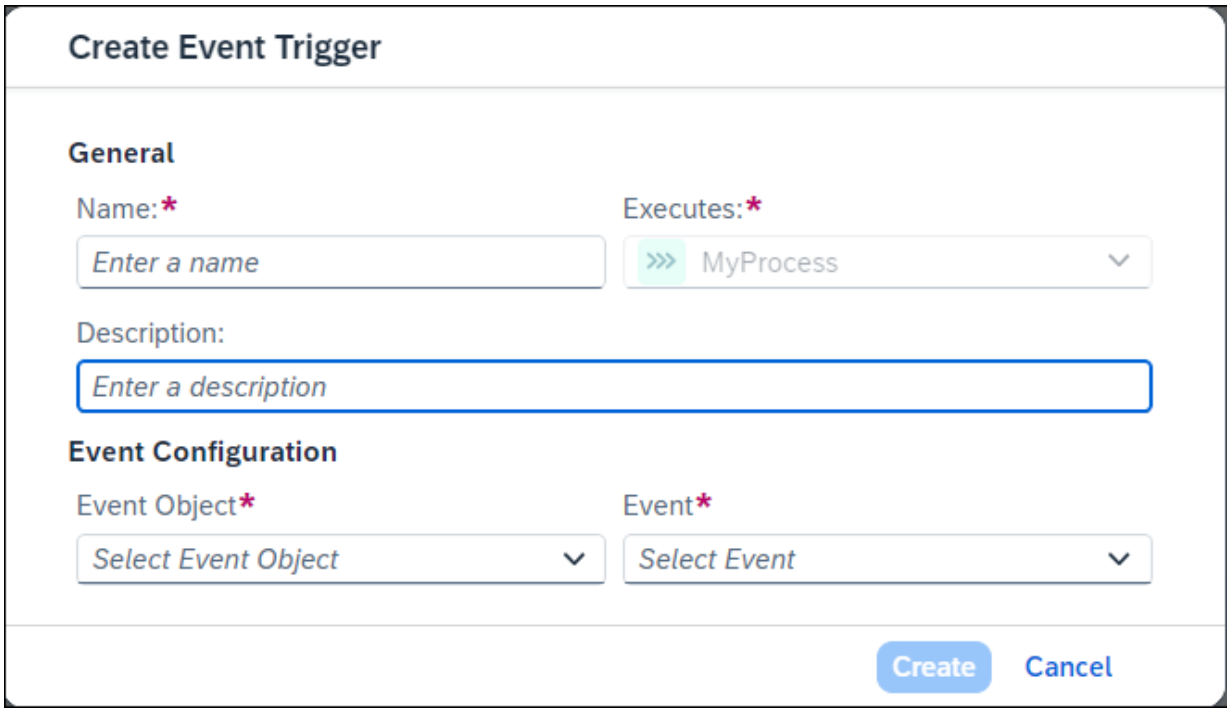
Procedure

1. On the **Trigger** tile in the process editor, choose **Add a Trigger**. Then, select **Wait for an Event**.



2. Enter the following data:

Field	Description
General	
Name	Enter the name of your event trigger. **Note** The event trigger name cannot be longer than 64 characters.
Executes	Set by default to the current process name and is read-only.
Add Description	Enter the description of your event trigger. **Note** The description cannot be longer than 1024 characters.
Event Configuration	
Event Object	Select an event object from the predefined list with consolidated event groups. You can use the fuzzy search to match your choice easier.
Event	Select an event that's available for the event object you have chosen at the previous step.



3. Choose **Create**.

   A success message for creating the event trigger appears.

Related Information

Manage Event Triggers

Manage Event Triggers

You can edit, enable or disable, and delete event triggers using the different options of SAP Build Process Automation.

Related Information

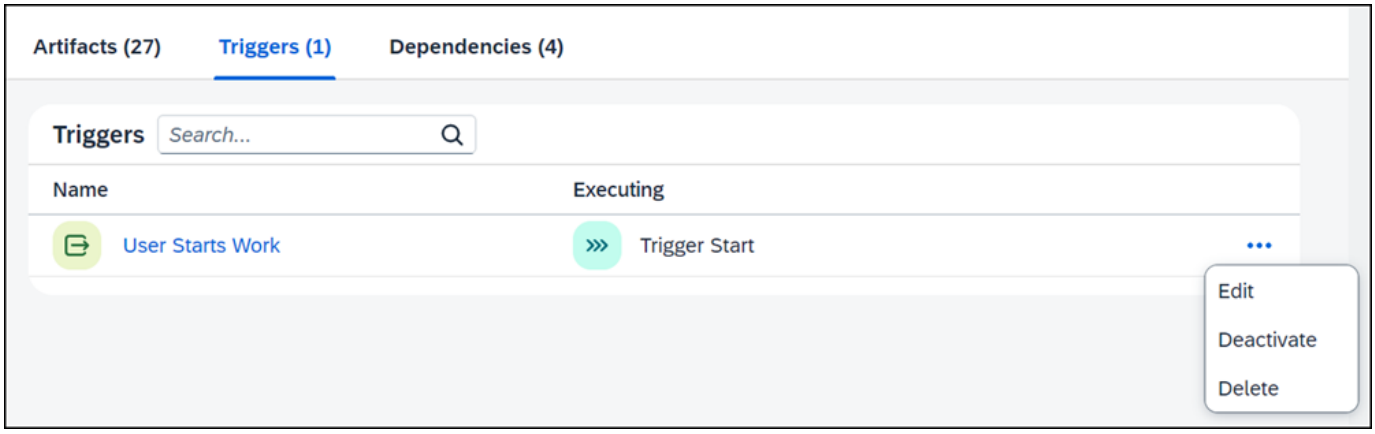
Create Event Triggers

Manage Event Triggers from the Project Overview

You can edit, enable or disable, and delete event triggers using the option of project **Overview**.

Procedure

1. To manage your event trigger, choose **Overview** > **Triggers** > **... (More)**.



2. Select one of the following options:
    - **Edit**: To change the event trigger's **Name** or **Description**
    - **Deactivate**: To not deploy the trigger

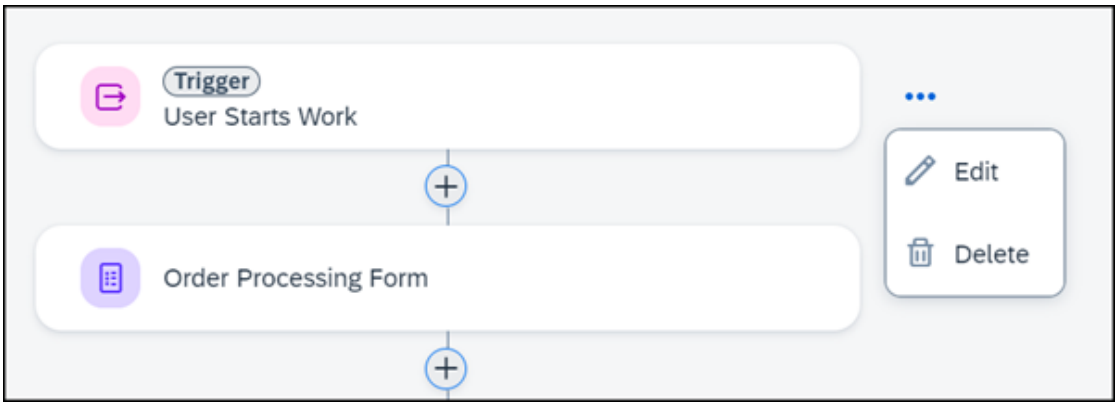
      As a result, the event trigger becomes grayed out.
    - **Activate**: To redeploy the event trigger
    - **Delete**: To delete the event trigger artifact
3. Save your changes.

Manage Event Triggers from the Process Editor

You can edit, enable or disable, and delete event triggers using the option of process editor.

Procedure

1. To manage your event trigger, go to **Overview** and choose **... (More)**.
2. Choose one of the following actions:
    - **Edit**: To change the event trigger's **Name** or **Description**

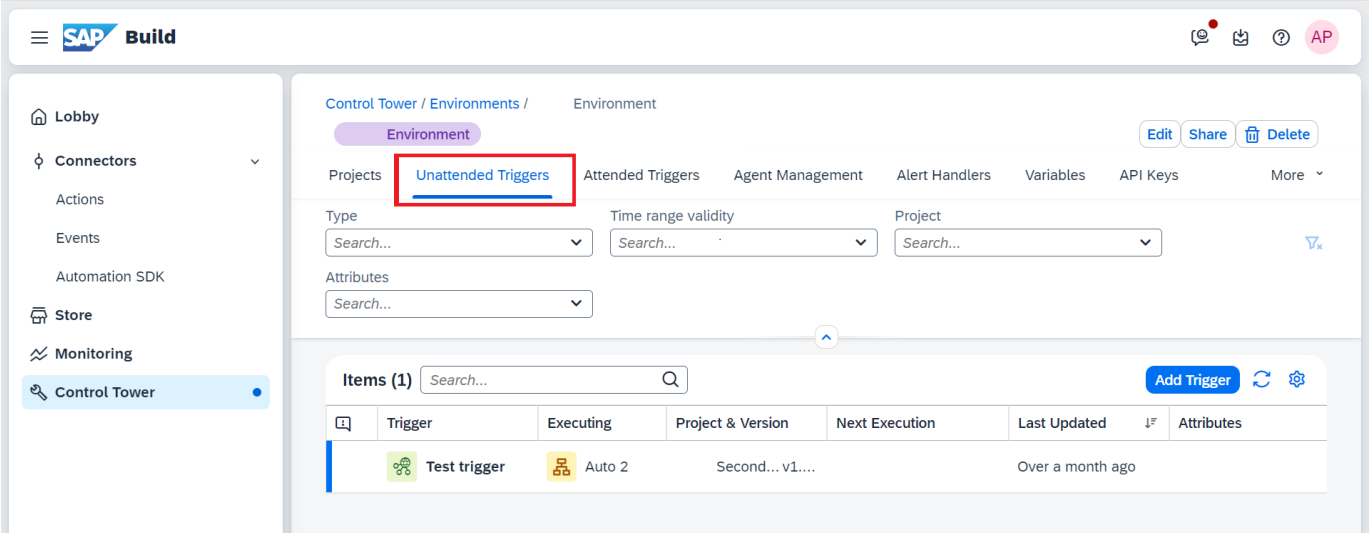
    - **Delete**: To delete the **Event** trigger artifact
3. Save your changes.

Manage Event Triggers from the Triggers View

You can edit, enable or disable, and run event triggers using the option of the **Triggers** view.

Procedure

1. In SAP Build, choose **Control Tower**, then choose the **Environments** tile.
2. Select your environment and choose the **Unattended Triggers** tab.
3. Search for your trigger using the Type, Time range validity, Project, and Attributes filters.



4. Choose ... (More Options) for the selected event trigger and select one of these options.
    - To pause the trigger action, choose Disable.

      As a result, the event trigger becomes grayed out.
    - To execute the event trigger processes, choose Enable.
    - To change the event trigger Name or Description, choose Edit.
    - To execute the event trigger in order to initiate event delivery with the user-provided data immediately, choose Run Now.
        - If you choose Run Now, the Run My Event Trigger pop-up window appears. The Event and Executes fields are predefined with the event trigger data.
        - In the Test with following data field, a structured message header must be provided, including the event data with the respective inputs. Ensure the following properties are correctly configured, as in the example below:

          Sample Code

```
{

"type": "sap.s4.beh.salesorder.v1.SalesOrder.Created.v1",

"specversion": "1.0",

"source": "internal",

"id": "0794ef45-7741-1eea-b7be-ce30f48e9a1d",

"time": "2023-12-04T06:21:52Z",

"datacontenttype": "application/json",

"data": {

"SalesOrder": "3016330",

"EventRaisedDateTime": "2023-08-08T10:16:27.919Z",

"SalesOrderType": "TA",

"SalesOrganization": "1010",

"DistributionChannel": "10",

"OrganizationDivision": "00",
```

```
"SoldToParty": "1000360"

}

}
```

- Ensure the correctness of the message header properties:

Property	Description
"type"	Represents the type of the event associated with the trigger. Update it using the same structure but with the corresponding event object and event.
"specversion"	The default value is “1.0” and must not be altered.
"source"	The default value is “internal” and must not be altered.
"id"	Random identifier with a maximum length of 64 characters. It must be unique. If not unique, then event will not appear as a separate record in Event monitoring UI.
"time"	Date-time in the ISO 8601 format.
"datacontenttype"	Specifies the type of content. The default value is “application/json” and must not be altered.
"data"	Represents the parameters of events utilized in the process definition, including corresponding input values. **Note** You should not provide any properties that are not defined in the process definition. In case of such, the validation will be activated, and then you will not be able to execute the trigger.

- If the event trigger is disabled, the **Run Now** option is grayed out.

5. Save your changes.

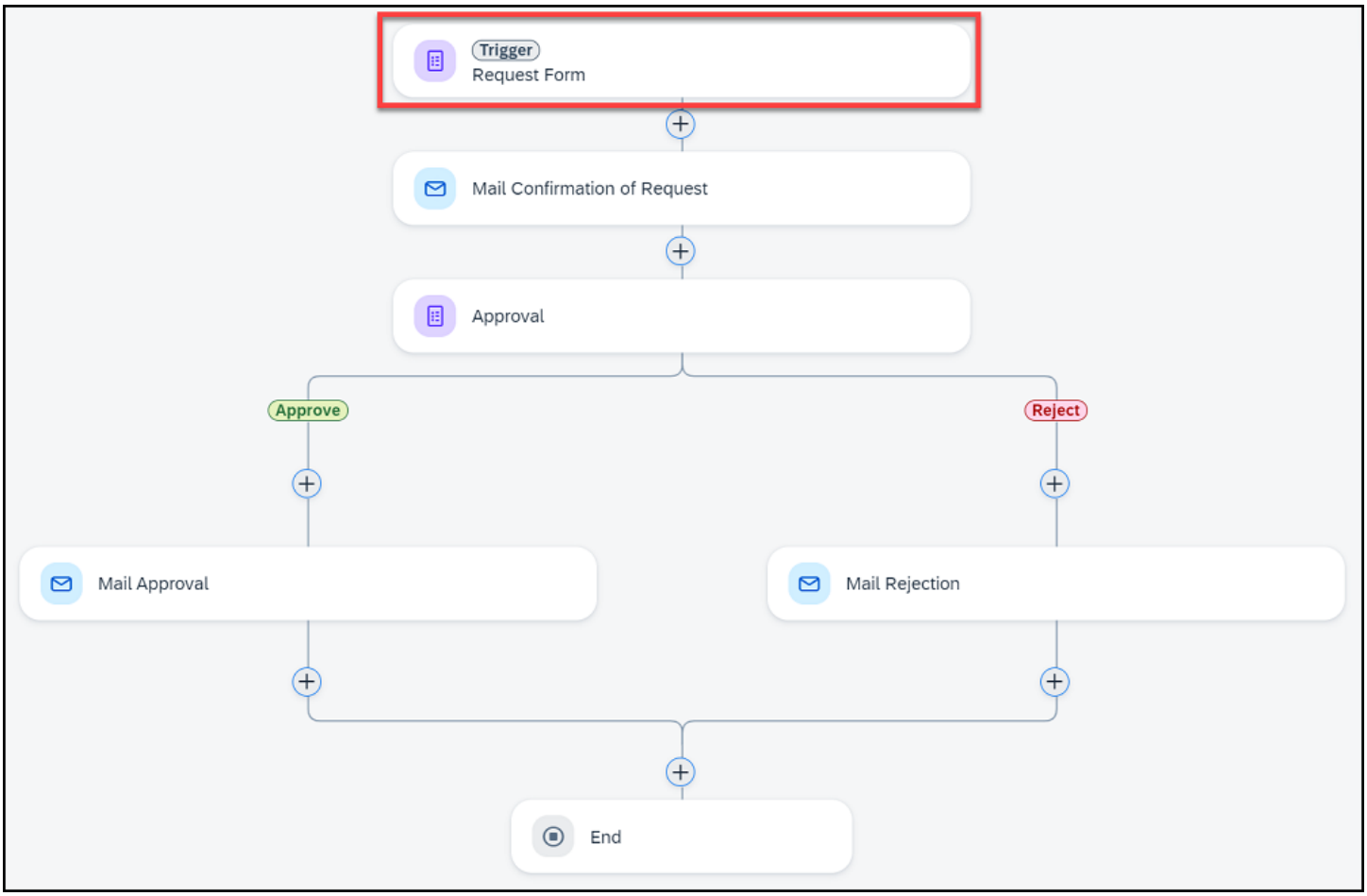
   To monitor the business events, acquired from connected systems, see Acquired Events.

Create a Form

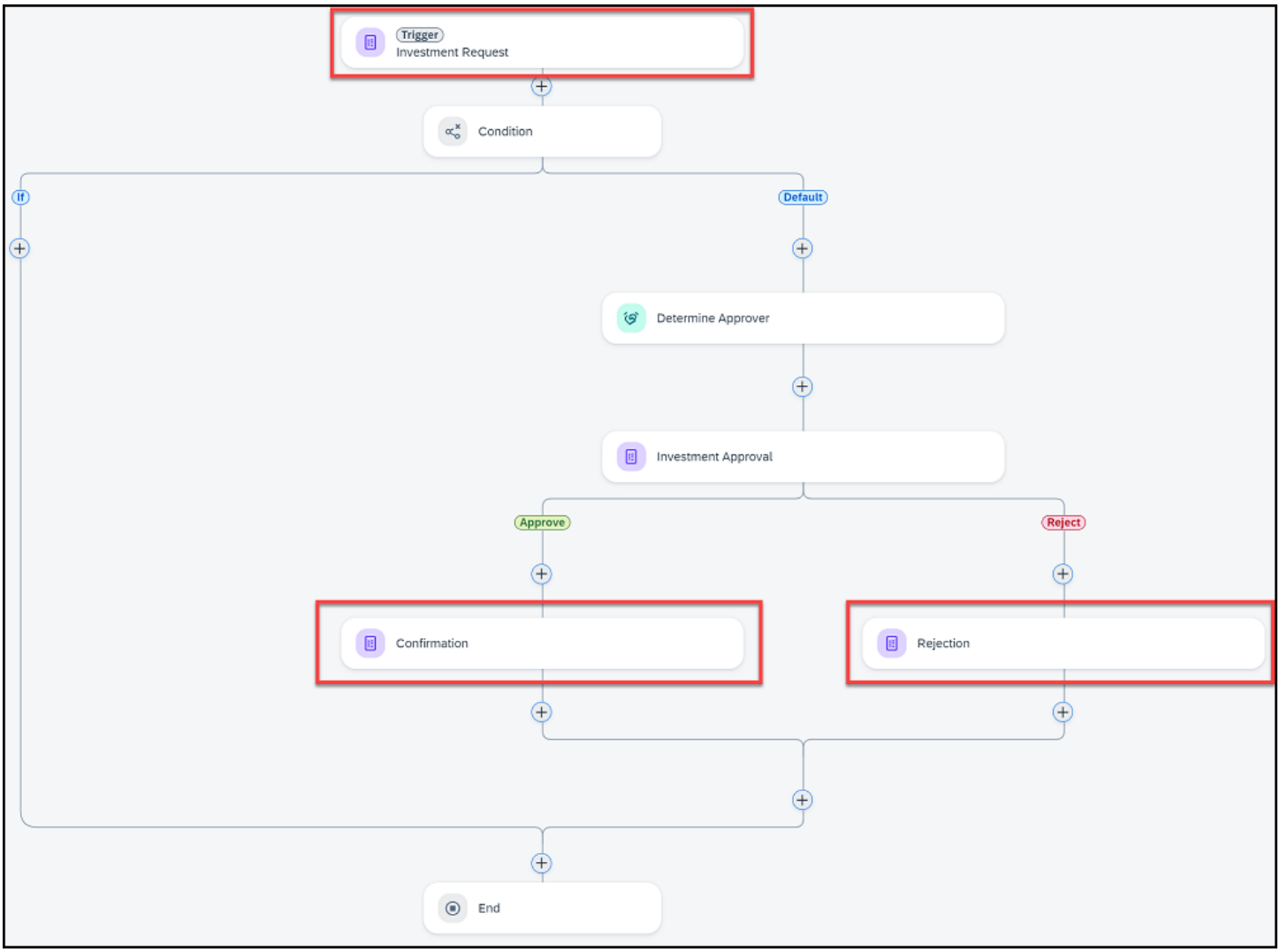
You can create interactive forms in SAP Build Process Automation, allowing you to capture and share information during a running process. Forms can then be a start trigger for a process and added as additional steps in the same process.

Context

In the following simple request process, a request submission form starts the process running. The requestor can access the published request form using a direct link or using a configured file in their SAP BTP.

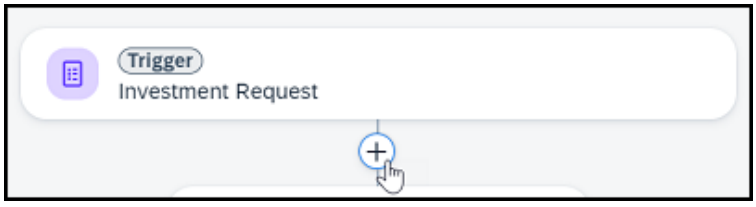


And in the following investment approval request, a request submission form also starts the process running. The requestor can access the published request form using a direct link or using a configured file in their SAP BTP. Then additional confirmation and rejection forms are included in the process, which will be displayed to the recipients as a task in their inbox.



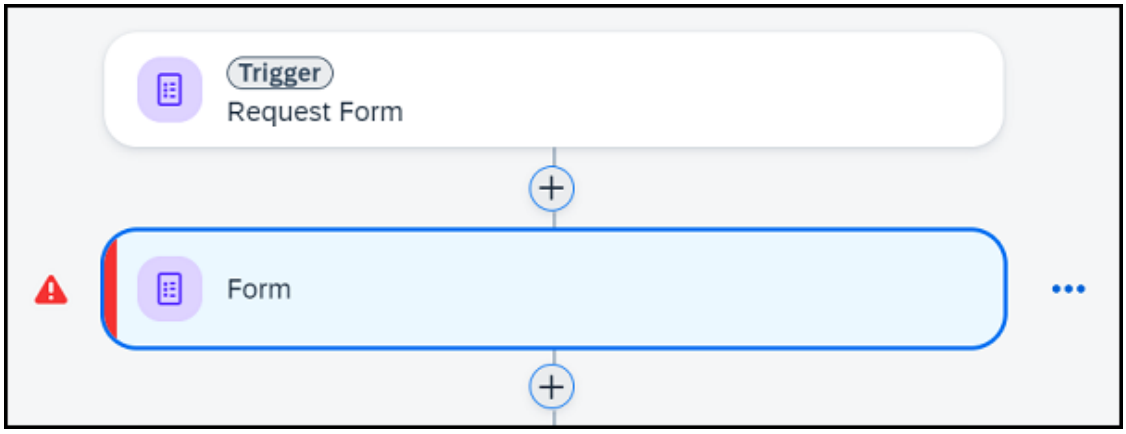
Procedure

1. To create another form in the business process editor, choose + > Form > Blank Form.



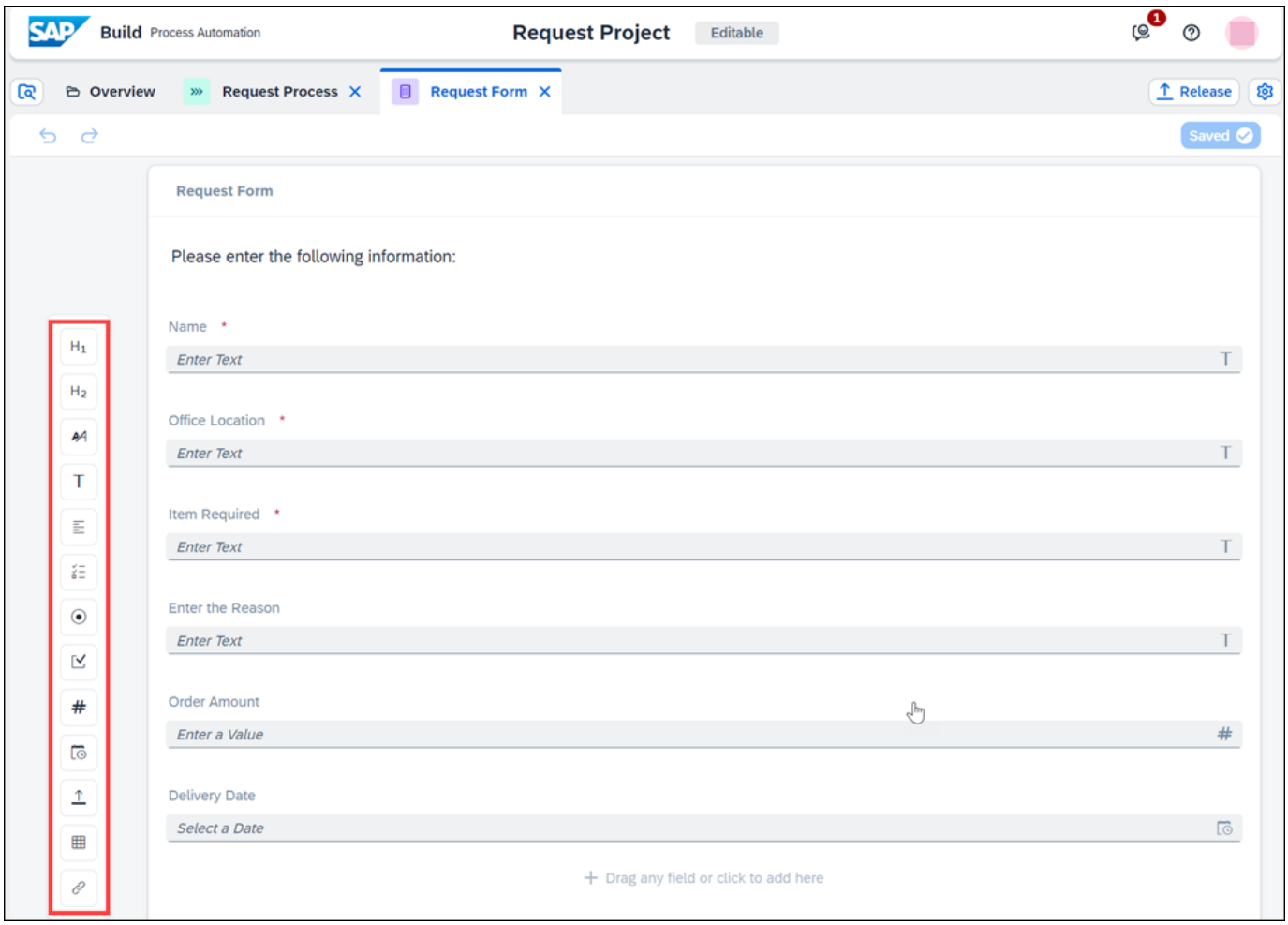
2. Enter a Name, Identifier (used as an internal reference), and an optional Description.
3. Choose Create.

   The form is created and added to the process editor canvas.



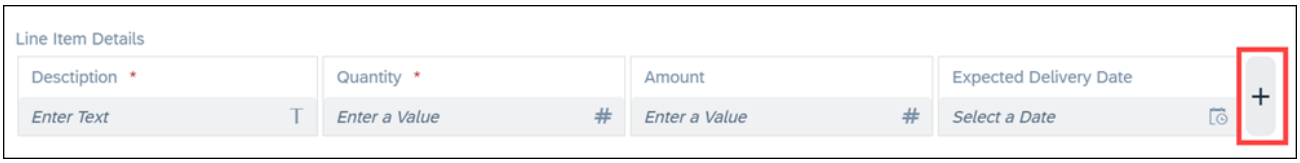
4. Double click and open the Trigger or Form element on the process editor canvas.

5. Drag and drop **Layout** and **Input** fields to the form as required. For an overview of form fields, see Form Input Fields.



6. To add a table, drag and drop the **Table** element to the form and enter a name.

   a. To add a column, choose **+ (Plus)**. For an overview of column types, see Form Tables.

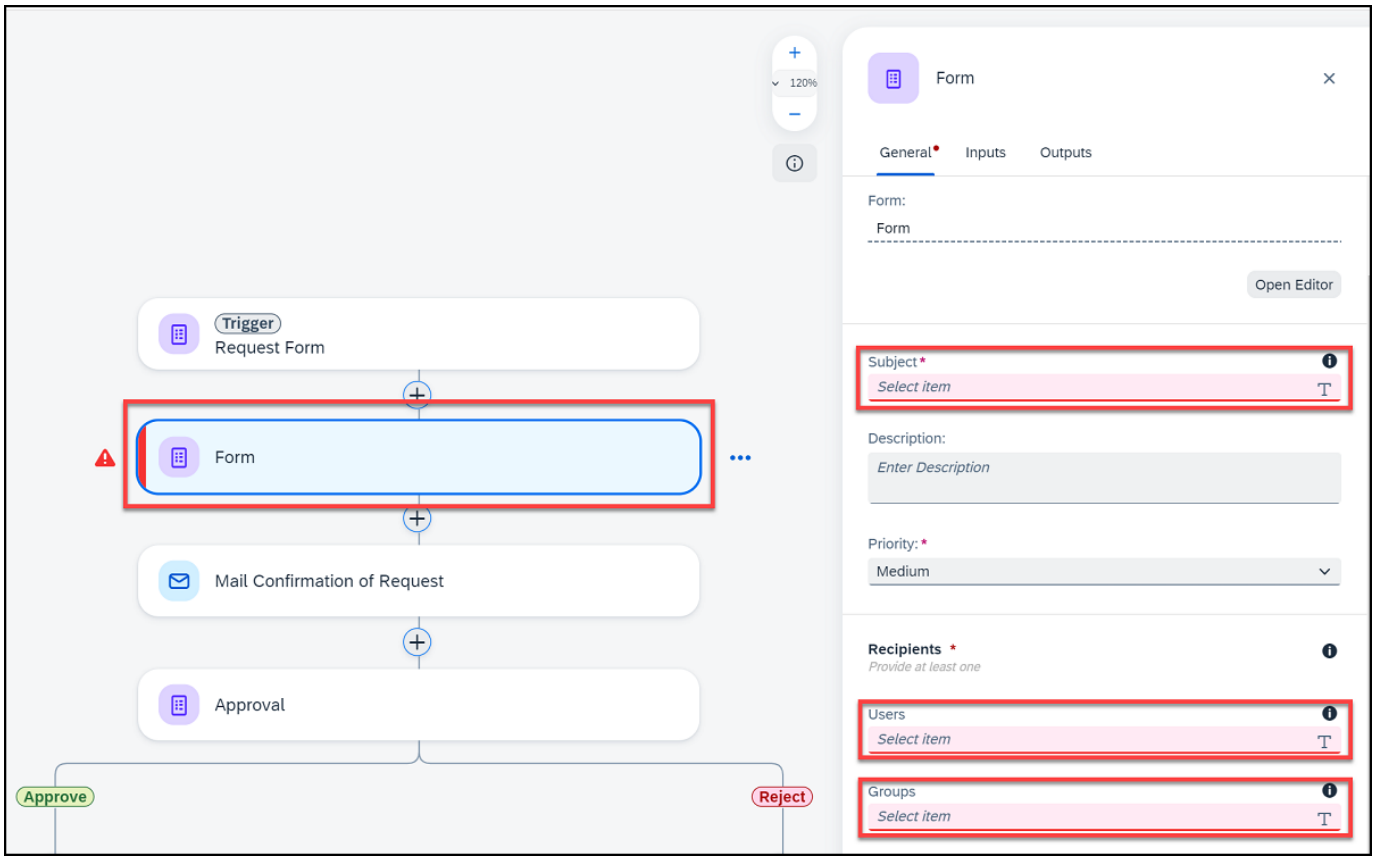


7. **Optional:** Customize the default **Submit** outcome button and use **Next**, **Send**, or a **Custom** text. Choose the cogwheel icon, and select the required text from the **Button Title** dropdown.

   If you select **Custom**, enter your own text. This text is then only available in the language in which you entered it. The predefined labels are translated. If several labels are translated using the same term in the target language, then the application only shows one label.

8. Save your changes.

9. Return to the process editor canvas and click on the form title, opening the form settings side panel.



10. Configure the **Form Settings** as required. See Configure Settings for Forms and Approval Forms.
11. Save your changes.

Form Input Fields

You can add form input fields to both a standard form and an approval form, allowing you to design and manage the information displayed to the process participant.

Available Input Fields

In the form editor, you can directly edit the form name. You can drag and drop the following input fields:

Field	Available Values or Options
Headline 1 Headline 2	Enter text. You can move the element up and down.
Paragraph	Enter text. You can move the element up and down.
Text (maximum 256 characters)	• Minimum character limit • Maximum character limit • Read Only • Required • Description

Field	Available Values or Options
Text Area (maximum 1700 characters)	• Minimum character limit • Maximum character limit • Read Only • Required • Description

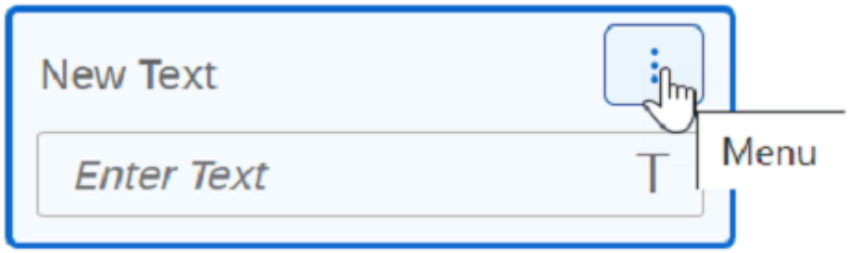
Dropdown	Use one of the following options to define the dropdown entries:
	• Use data from a data set by selecting an action from the library. When a data source is value help enabled, then the dropdown also shows the search icon and a popup for the search entry. See Add and Use Data Sources in Form Input Fields.
	  ◦ Required
	  ◦ Description
	• Enter text to define the options manually.
	  ◦ Required
	  ◦ Read Only
	  ◦ Multiple Selection
	  ◦ Description

Choice	• Enter text for the selection option entries • Vertical Alignment • Read Only • Required • Description
Checkbox	• Read Only • Required • Description
Number	• Fixed decimal places • Minimum value • Maximum value • Read Only • Required • Description
Date	• Based on the user's location • Allow selection of: All Dates, Only Past Dates, and Only Future Dates • Read Only • Required • Description
File Upload / Attachment Requires an active SAP Document Management Service subscription. See Configure SAP Document Management Service for Process Attachments.	• Maximum number of files • Required • Read Only • Description

Field	Available Values or Options
Tables	See Form Tables
Link	Enter a link address of type HTTPS, HTTP, or mailto. Enter a link text to display instead of the URL, a label, and a description that appears in a question mark tool tip. Your input is the default link content. If you use a form that contains a link, the link content can be overwritten by other process content because the link field is available for mapping.

Input Field Settings

You can directly edit input field texts. To move the field up or down or to delete it, use the **Menu** icon.



The **Configuration** section, let's you define more settings. Depending on the field type, you can, for example, mark the field as:

- **Required** - The participant must input data here to progress the active process.
- **Read Only** - The participant can see information entered earlier in the process, but can't edit it.
- **Multiple Selection** - The participant can see select more than one entry.

The **Input Validation** section, let's you define whether and which validation is used for text fields and text areas. When you select an option, the regular expression for that selection is displayed. You can copy it to create a more sophisticated custom expression that you enter as a custom validation. You can directly test the validation you defined using the **Test Text** field. For examples on regular expressions, see, for example, the Mozilla cheat sheet.

The following validation options are available:

- **Letters** - The participant can enter letters only.
- **Numbers** - The participant can enter numbers only.
- **Letters and Numbers** - The participant can enter any character.
- **Alphanumeric** - The participant can use latin letters (a-z, A-Z) and arabic numbers.
- **Custom Validation** - You can enter a regular expression that exactly defines what the participant can enter. For example, you can require three letters, one number....

Add and Use Data Sources in Form Input Fields

You can add data sources to your form input fields, allowing process participants to select from information managed in external systems. This removes the need to manually add fields and information when creating a form.

Prerequisites

Before you can add and use data sources in your forms, make sure that the data sources have been configured in your library. See Configure Data Sources for Form Input Fields.

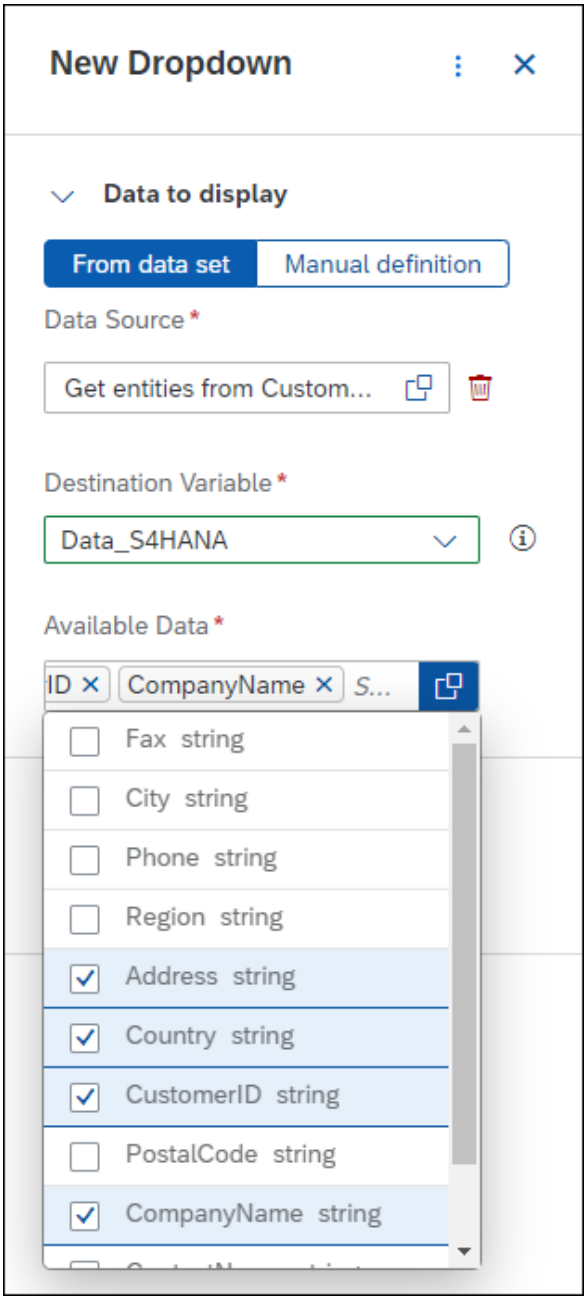
ℹ Note

- Only JSON files are supported and the file size is limited to 5 MB.
- Open API specification files with versions 2.x.x and 3.x.x of JSON type are supported.
- Must include a **GET** request with an array.
- Must not have mandatory input parameters defined.
- To make an action available as a data source, define it's output by setting the **Main Output Array** tag.
- To use the search feature for a data source, the action must have the `$search` parameter defined.

Context

Adding data sources for form input fields allows process participants to select from information shared from external systems. These fields can then be mapped across process steps, ensuring that process data is consistent.

As an example, a dropdown field in the following form is configured to allow process participants to select from customer data shared from SAP S/4HANA. In this case, they can select the customer's country, ID, address, and the company name.



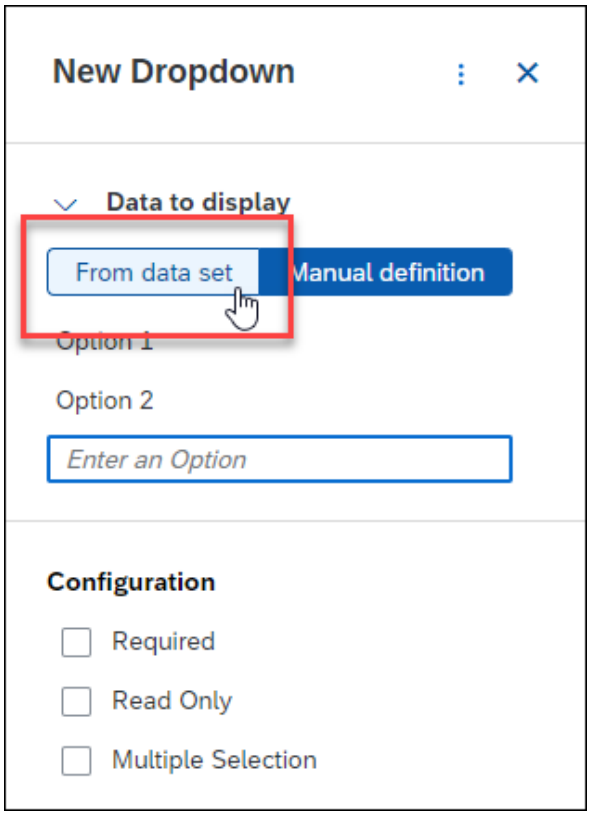
ℹ Note

You can only use dropdown fields.

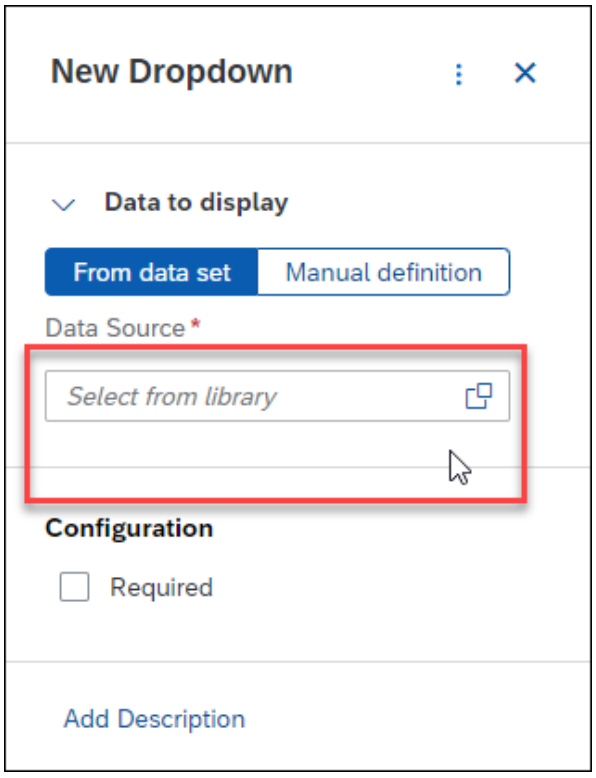
The list shows the labels of the output values that are defined for the action. These labels can be changed in the action editor.

Procedure

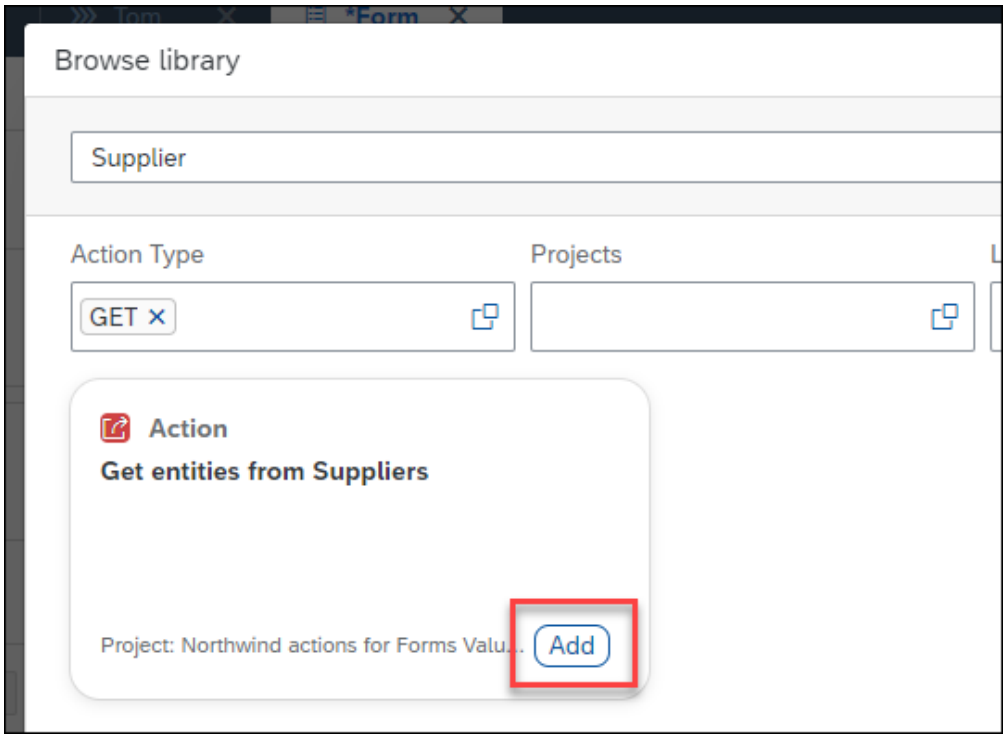
1. In the form editor, drag and drop a **Dropdown** input field to the canvas.
2. In the side pane under **Data to display**, choose **From data set**.



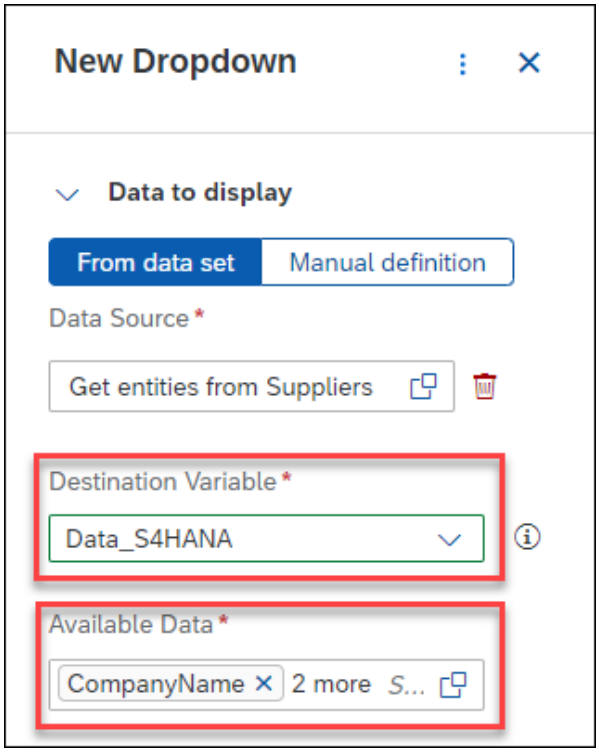
3. Select a library under **Data Source**.



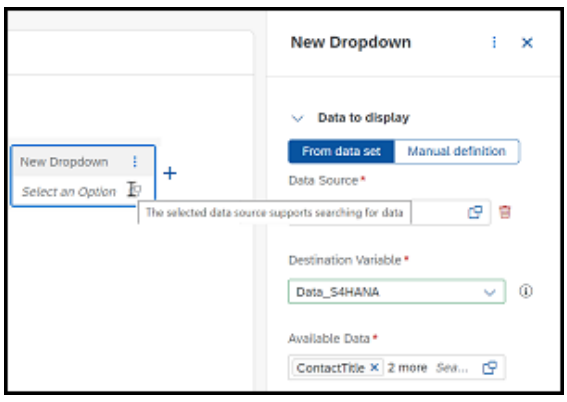
4. Once you've found the **Action** to use, choose **Add**.



5. Select or create a Destination Variable. Destination variables are used to connect the action to an external system.
6. Select the Available Data that you want to display to the process participant in this field.



If the the `$search` parameter is defined, the icon in the dropdown changes:

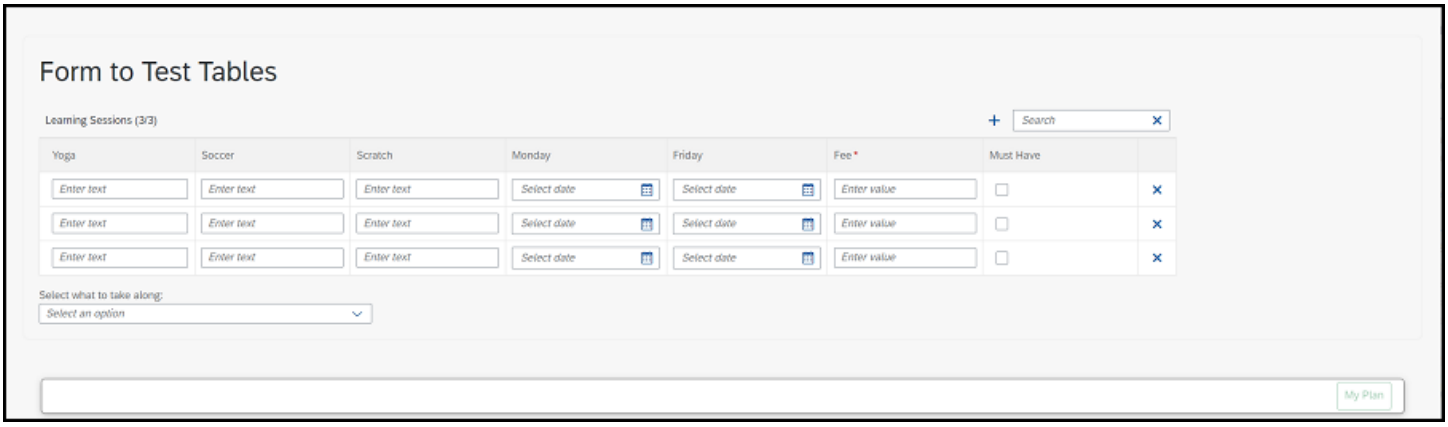


7. Save your changes.

Form Tables

As with form input fields, you can add tables to both a standard form and an approval form, allowing you to design and manage the information displayed to the process participant.

You define the columns of the table, and the process participant adds content with as many rows as needed.



If you create a form based on a form with a table, then you can map those fields as input of the new form using **Bind List**.

ℹ Note

You must bind the table to a data type list of type **Object** and the **Required** constraint, even for single-column tables of type text with string input. Lists with other data types, for example, strings, numbers, or booleans don't work.

The table in the new form is set to read only by default but you can change that setting.

The following general rules apply:

- The number of columns isn't limited.
- The number of rows can't be predefined.
- The **Read Only** setting applies to the entire table and can't be used for individual columns.

The following columns elements are available:

Column	Available Values or Options
Text (maximum 256 characters)	• Minimum character limit • Maximum character limit • Required • Description
Checkbox	• Required • Description
Number	• Fixed decimal places • Minimum value • Maximum value • Required • Description

Column	Available Values or Options
Date	• Based on the user's location • Allow selection of: All Dates, Only Past Dates, and Only Future Dates • Required • Description

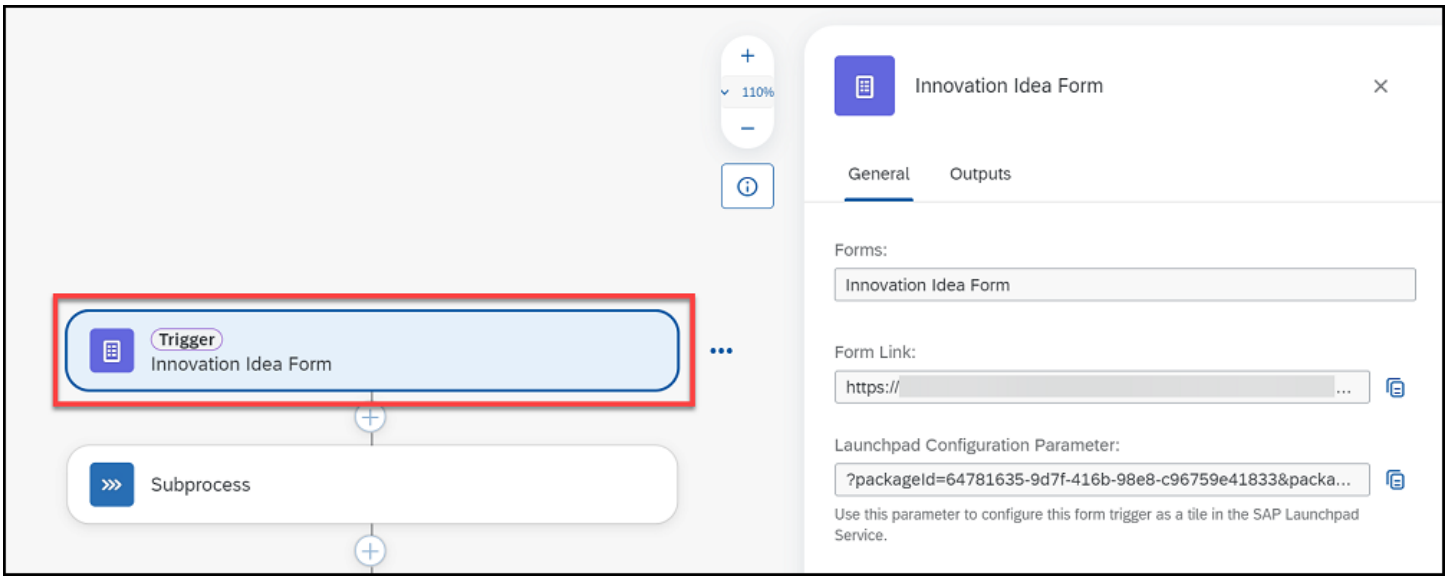
Dropdown	Use one of the following options to define the dropdown entries:
	• Use data from a data set by selecting an action from the library. When a data source is value help-enabled, then the dropdown also shows the search icon and a popup for the search entry. See Add and Use Data Sources in Form Input Fields.
	  ○ Required
	  ○ Description
	• Enter text to define the options manually.
	  ○ Required
	  ○ Read Only
	  ○ Multiple Selection
	  ○ Description
	Note
	If you use a dropdown in the table, you can no longer set the whole table to read only.

Configure Settings for Forms and Approval Forms

You can configure the settings for your forms using the process editor, allowing you to define how the form is displayed to the process participant.

Settings for Start Trigger Forms

You can configure your process start trigger settings by clicking on the **Trigger** tile.



You have the following configuration options:

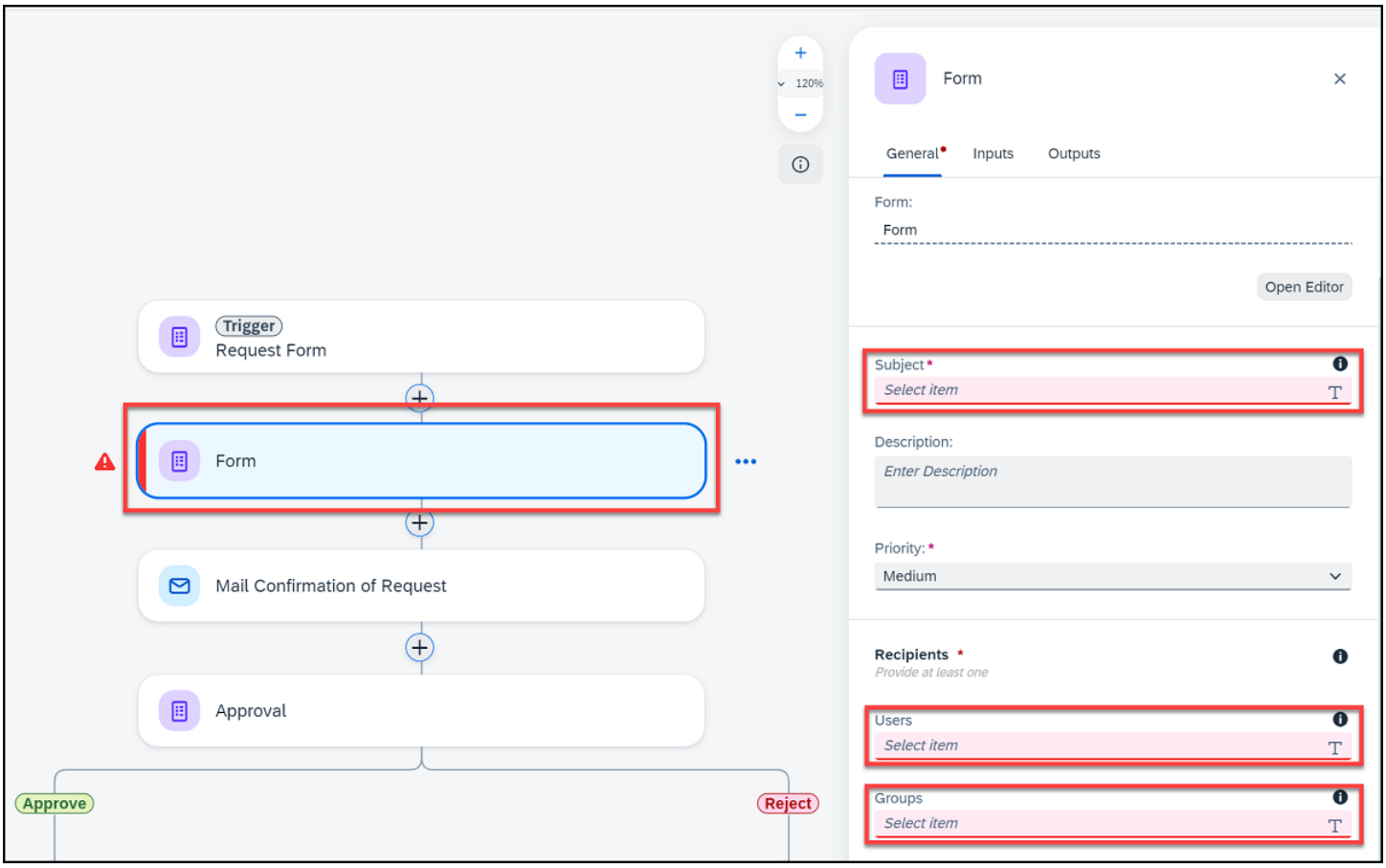
Field	Description

Field	Description
Forms	Select the form you want to use as the trigger for your process. Only forms available in the same business process project can be selected.
Form Link	Available for deployed processes only, this provides a direct link to the published form. The process participant can access this link, enter the relevant information, and then click submit.
Launchpad Configuration Parameter	Available for deployed processes only, this provides the parameters needed to create a tile in SAP Build Work Zone. Clicking the tile opens the form and from there the process participant can enter the relevant information and then click submit.
Outputs	This section provides an overview of the process metadata included in the trigger form. This is a read-only section.

Additional Settings for Forms or Approval Forms

You can configure additional form or approval form settings for forms visible on the process editor canvas only, rather than all forms in the project. These form settings allow you to configure how the form information is displayed as a task in the inbox and who that task is sent to.

To configure the form settings, click on the form element on the process editor:



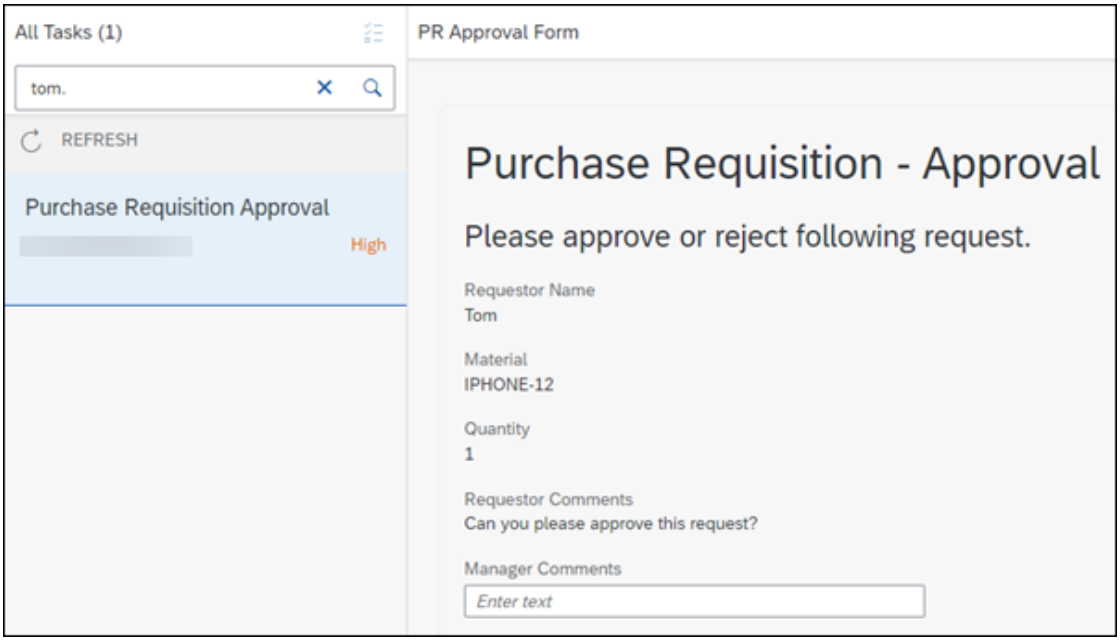
You have the following configuration options:

Field	Description
Step Name	This is the name under which the form displays in the process canvas.
<Name of the form>	This is the name you entered for the form when creating it.
Subject	This is the subject of the task received in the inbox, allowing the participant to identify the item. In the following example, this is displayed as: **Purchase Requisition - Approval**.

Field	Description
Description	This provides the description displayed within the received task, giving the participant more information about the process. In the following example, this is displayed as: **Please approve or reject following request**.
Priority	This defines the priority of the task received in the inbox, giving an indication of when a response is needed. In the following example, this is displayed as **High**.
Recipients	You can assign process metadata and context fields, for example, lists of users or the user who started the process. Alternatively you can enter specific email addresses of users or groups who should receive the task. For more detailed guidelines, see Guidelines for Specifying Recipient Users.
Allow Forwarding	Select to allow the recipient to forward the task to another user.
Due Date	You can set a due date for the task received in the inbox, giving a deadline for the response from the time that the task arrives. Due dates can be set to minutes, hours, days, months, and yearly durations. Choose one of the following due date options: • No Due Date • Static Duration Enter an integer between 1 and 999, and select the unit (minutes, hours, days, months, and years). • Dynamic Duration Select a number input between 0 and 2147483647 from the process content. • Reference Date Select a date input (UTC + 0) from the process content.
Inputs	You can map the process metadata fields here, ensuring that the correct information persists throughout the process.
Outputs	This section provides an overview of the process metadata included in the form. This is a read-only section.

Example of a Task

The following is an example of a task received to the inbox, with the fields taken from the form settings configured using the process editor.



Guidelines for Specifying Recipient Users

When you specify recipient users for a user task, consider the following:

ℹ Note

Carefully consider the impact that the changes described here might have on your overall scenario. Changing certain settings after productive use has started can have a negative impact on scenarios that are incompatible with the change. If applicable, use mechanisms that restrict the impact to the specific scenario.

- Evaluate whether you can use “recipient groups” instead of “recipient users” because there are limits as to how many recipient users may be specified. If you can, you must configure the assignment of users to groups in the identity management-related function of the platform or the identity management back-end systems. This has, for example, the benefit that the assignment of a task to a certain user can be removed using these central identity management functions instead of in the workflow definition and related locations. This usually improves compliance with company and legal requirements. For example, removing an assignment typically becomes effective as soon as authentication tokens expire.
- If you cannot avoid specifying user names using constants or expressions, make sure that you apply the necessary lifecycle actions on the respective events to achieve compliance. For example, use the administrative REST APIs of SAP Build Process Automation, to remove recipient users when they should no longer be assigned to a task because they left their department or the company. Also, ensure that user interfaces that allow configuration of user IDs apply appropriate validation on the task recipients.
- Ensure that the case and spelling of the user ID matches the respective fields of the authentication tokens exactly. It is important that lower or upper case is also considered, because SAP Build Process Automation matches them as is. SAP Build Process Automation must also consider case sensitivity for user names that look like email addresses. There is no metadata that indicates whether user names are actual email addresses or whether case sensitivity is irrelevant. For this purpose, check your identity management system and the related configuration of SAP Business Technology Platform.
- Check the **User ID Source** and its related settings. Consider using “E-Mail” as the configured value, because this might improve consistency of user names in a scenario.
- Ensure that the identifiers, as validated against the user database, are provided. Do not rely, for example, on user names as entered directly by the user.
- Evaluate whether you can disable the creation of "shadow users" In certain constellations, this prevents users from logging in with user names that do not correspond to the canonical identifier, but use a different case.
- Evaluate whether you can configure that user inputs are automatically converted to the expected case. If you can, see the documentation of the SAP Cloud Identity Services feature **Apply Function to Subject Name Identifier** in Convert Subject Name Identifier to Uppercase or Lowercase or the respective configuration of your custom identity management system.

Configure a Start Trigger Form with SAP Build Work Zone

Using this trigger, you can start a business process.

Prerequisites

You have set up SAP Build Work Zone. See Configure SAP Build Work Zone for SAP Build Process Automation and particularly Configure SAP Build Work Zone Content.

Procedure

1. Start the Site Manager:
    a. In your SAP BTP Cockpit subaccount, choose **Services** > **Instances and Subscriptions**.
    b. Search for your SAP Build Work Zone application and choose the **Go to Application** icon next to the app name.
2. Open **Content Manager** > **My Content**.
3. Create a local copy of the **Process Trigger** app by selecting the **Process Trigger** item and choosing **Create a Local Copy**.

4. To configure the local copy, choose **Edit**

    a. Enter a title.

    b. On the **Navigation** tab, enter a parameter named `uri`.

    To do so:

    - In the process editor of SAP Build Process Automation, select the trigger and on the Trigger Settings panel, copy the configuration parameter URL.
    - Paste the configuration parameter URL into the **Default Value** field.

    c. On the **Translation** tab, edit any translated texts as needed.

    d. Save your changes.

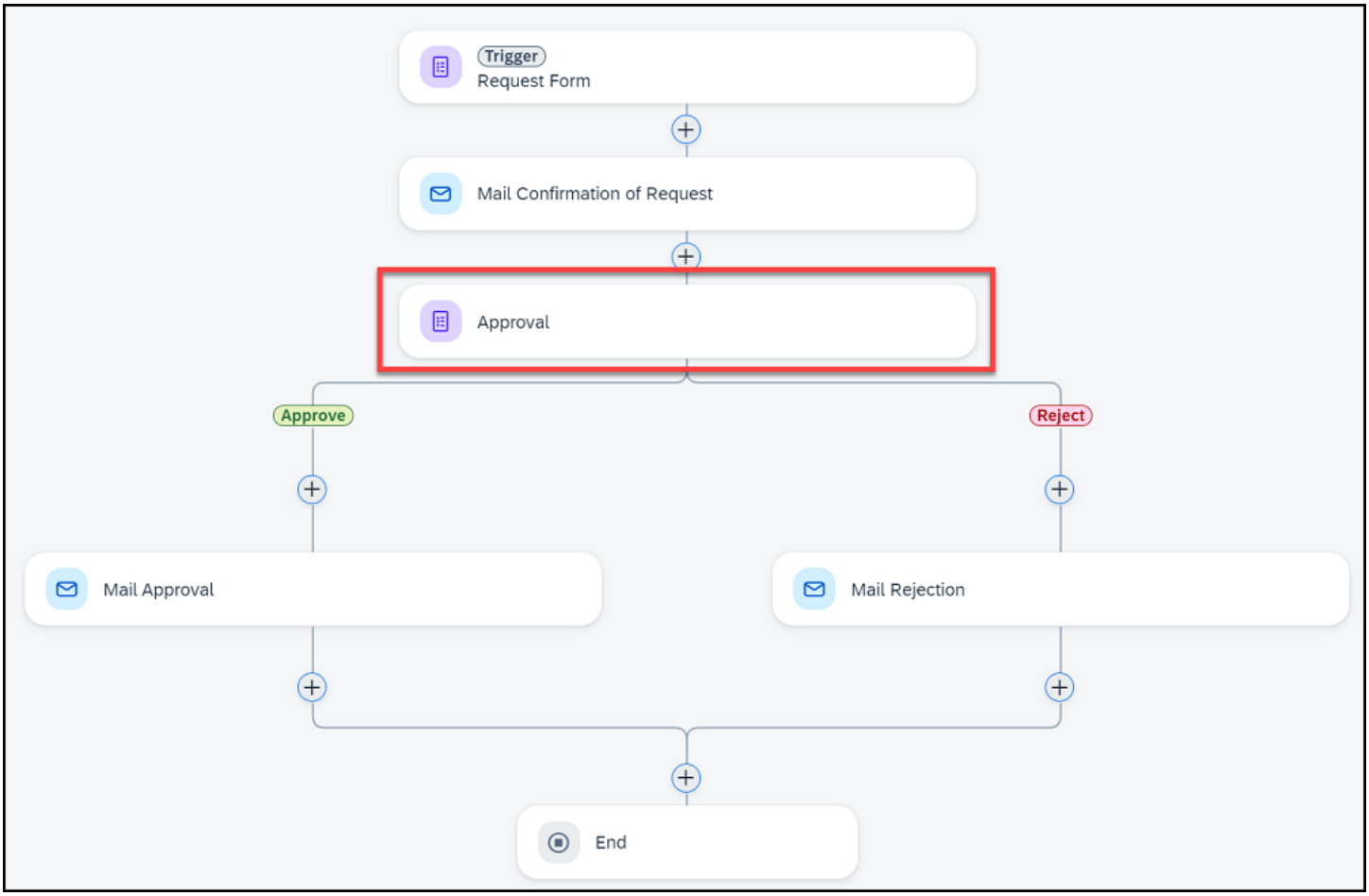
5. Assign the app to your users using standard means of SAP Build Work Zone.

Create an Approval Form

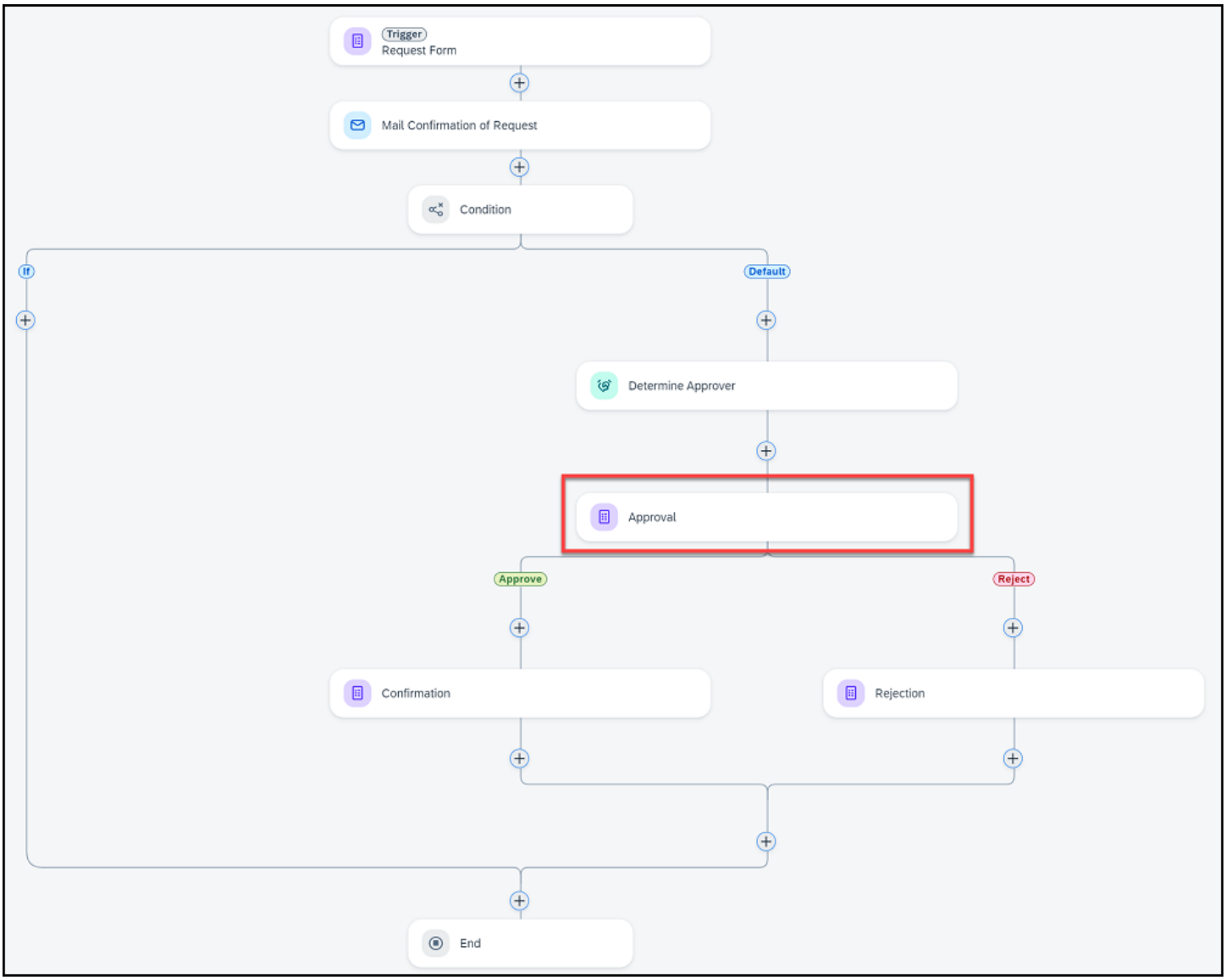
Approvals are an important part of business processes, whether performed manually or automatically approved based on process conditions. With SAP Build Process Automation, you can manage approvals by creating and adding an approval form to a business process.

Context

In the following simple request process, an approval is needed for all submitted requests:

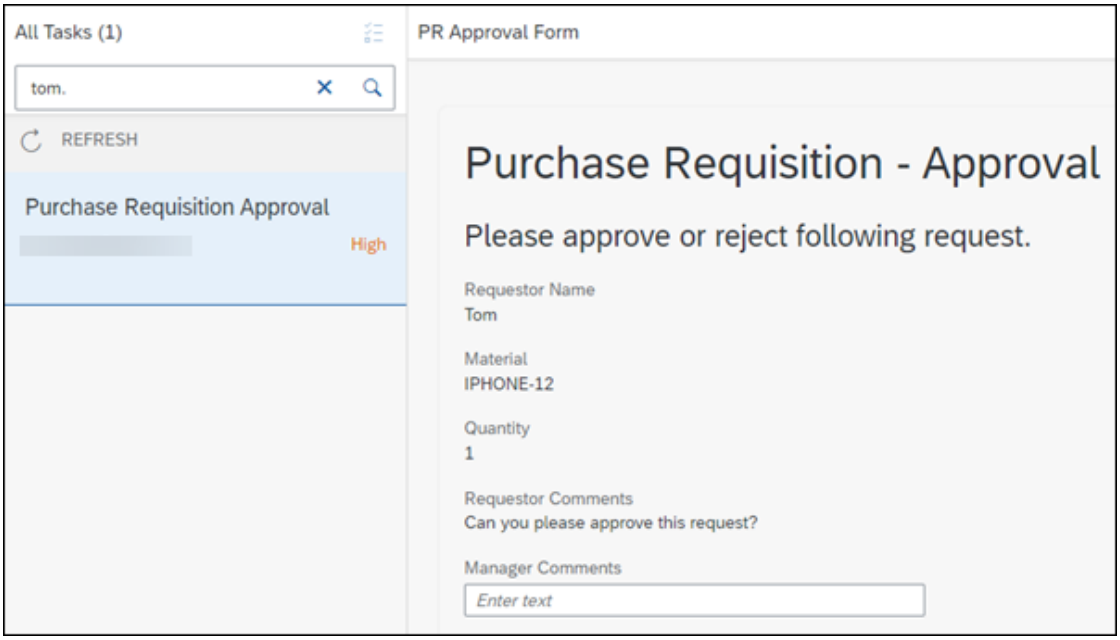


And in the following investment approval request, an approval is only needed when an investment value is higher than a specified amount:



When a process is running, the approver receives a task in their inbox containing the request information. They can then choose to accept, query, or reject the request. Their decision then determines how the process proceeds.

The following is an example of a task received to the inbox, with the fields taken from the form settings configured using the process editor.



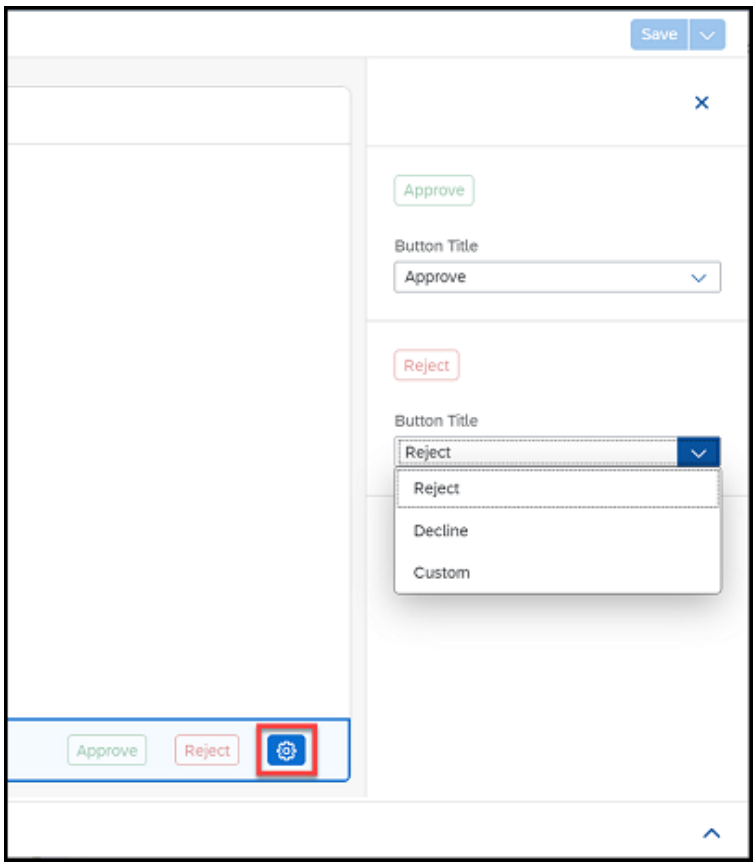
Procedure

1. From the process editor canvas, click **+** and select **Approval - New Approval Form**.
2. Enter a **Name**, **Identifier**, and optionally a **Description**.
3. Choose **Create**.

   The approval form element is now added to your process editor canvas.
4. Open the form editor by double-clicking the approval form element on the canvas.
5. Configure your approval form as required using **Layout** and **Input Fields**.

   For more information about input fields, see Form Input Fields.
6. **Optional:** Customize the default outcome buttons. You can adapt the default **Approve** button to **Accept** or a **Custom** text, and the default **Reject** button to **Decline** or a **Custom** text. Choose the cogwheel icon, and select the required text from the **Button Title** dropdown.

   If you select **Custom**, enter your own text. This text is then only available in the language in which you entered it. The predefined labels are translated. If several labels are translated using the same term in the target language, then the application only shows one label.



7. Save your changes.

   The approval form has now been saved to your process.
8. Navigate back to the process editor.
9. Click on the approval form element on the canvas, opening the approval settings menu.
10. Configure your approval settings as needed.

    For more information about configuring approval settings, see Configure Settings for Forms and Approval Forms.
11. Save your changes.

    The configured approval form is added to your business process.

Import an SAPUI5 Form

Import an SAPUI5 form and use it in your process in SAP Build Process Automation. This allows you to create SAPUI5 apps and use them as task UIs in your process.

Prerequisites

- You've created an SAPUI5 application. For more information about creating an application, see Creating a Custom Task UI.
- You've added the namespace `"sap.bpa.task"` to the `manifest.json` for your application and you've defined the inputs, outputs, and outcomes for the application. For more information, see Technical Information for Adapting the SAPUI5 Application.
- You've adapted the `Component.js` file and the `App.view.xml` file. For more information, see Technical Information for Adapting the SAPUI5 Application.
- You've deployed your application. For more information, see Build and Deploy in the documentation for SAP Business Application Studio.

Context

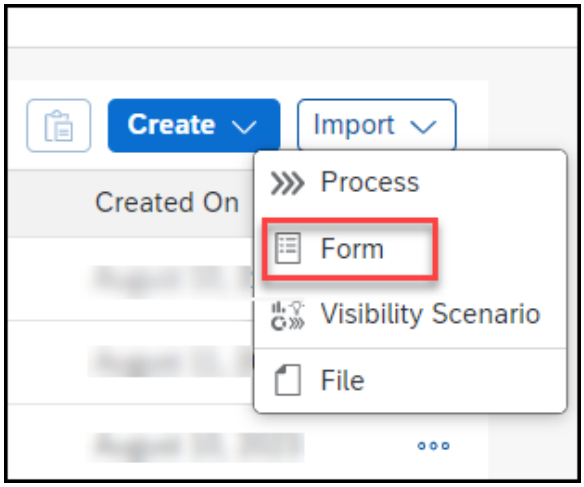
When you import a SAPUI5 form into SAP Build Process Automation, you can use this SAPUI5 app in the process editor in the same way as you use any other form. You can map the inputs and outputs of your form to the rest of the process and you can also use the outputs of the previous task in your form.

Note

You cannot use an SAPUI5 form as a start form.

Procedure

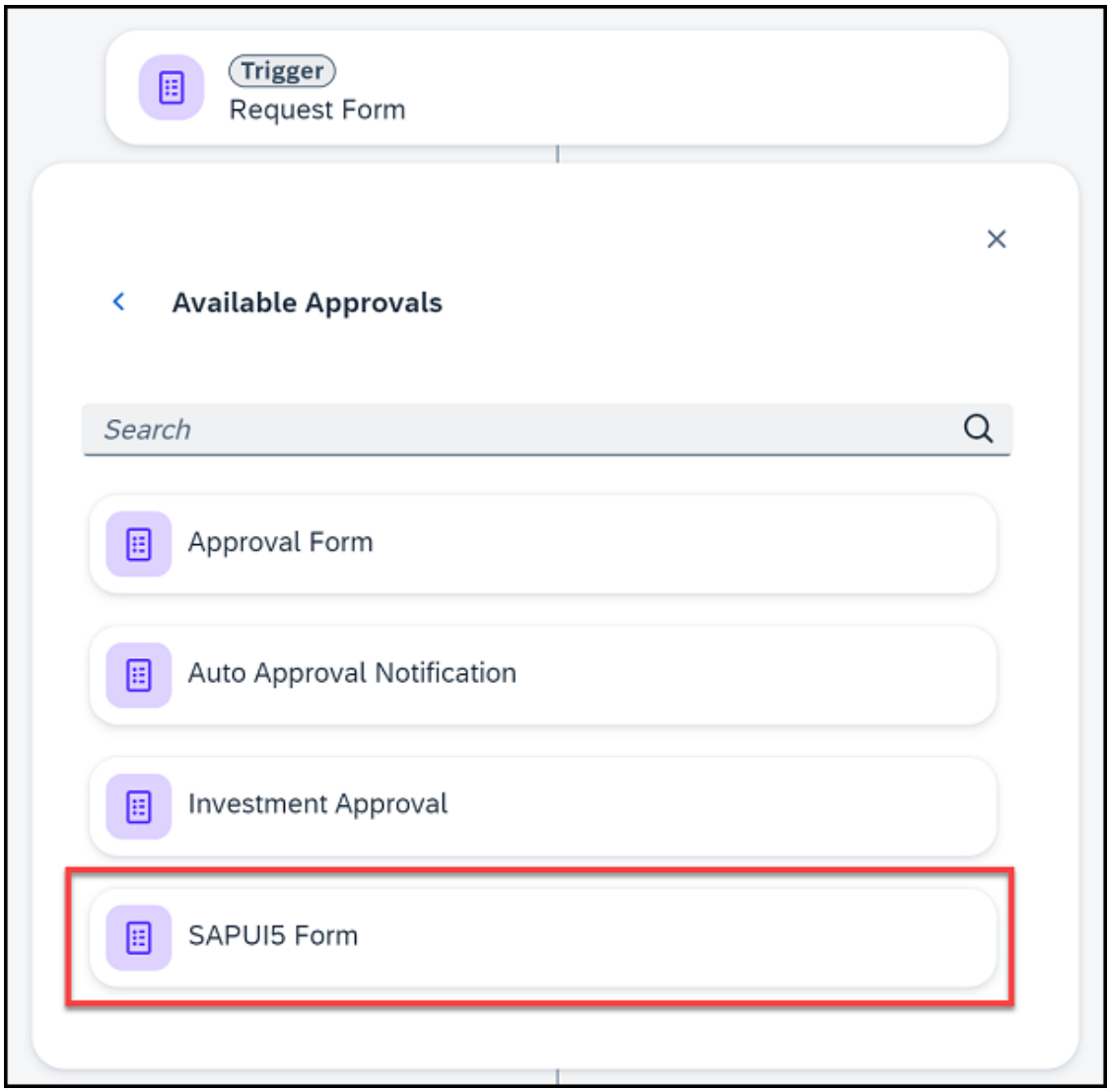
1. In the **Overview** tab of your project, choose **Import** > **Form**.



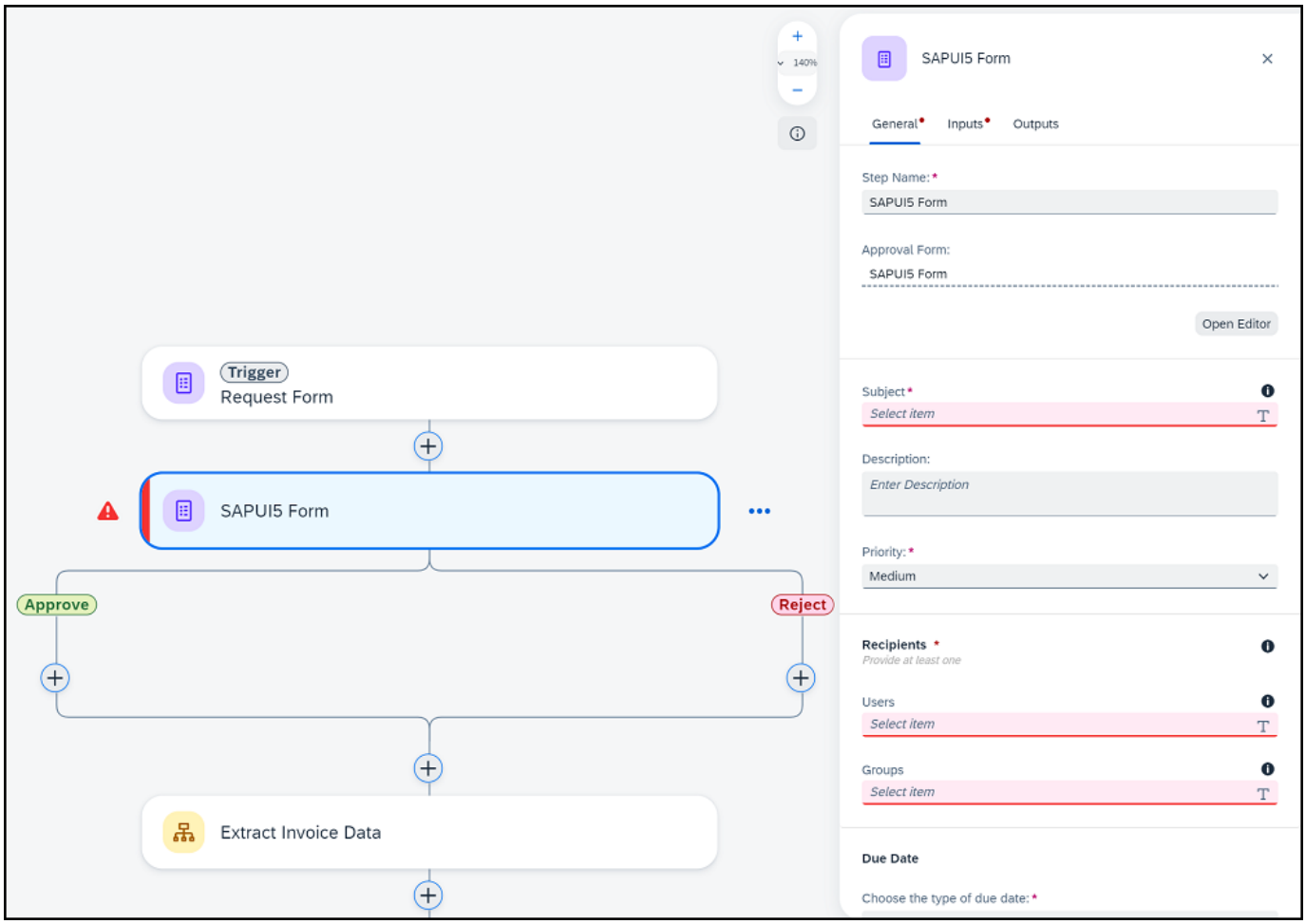
2. In the **Import Form** screen, enter the **Application ID** and the **Application Version** from the `manifest.json`.

   The form is imported and the manifest URL is displayed. You can now add the form to your process.

3. From the process editor canvas, choose **+**, then **Form** or **Approval** (depending on the category of the form) and choose your SAPUI5 form.



4. Choose the form element on the canvas to open the form settings.



5. Configure your form settings as required. For more information, see Configure Settings for Forms and Approval Forms.
6. Save your changes.

   The configured SAPUI5 form is added to your business process.

Technical Information for Adapting the SAPUI5 Application

To use SAPUI5 applications as forms in SAP Build Process Automation, you need to add the `"sap.bpa.task"` namespace to the `manifest.json` file. You also need to adapt the `Component.js` file and the `App.view.xml` file in accordance with the `manifest.json`. The examples shown here are all compatible with each other.

Configuring the manifest.json File

Add the namespace `"sap.bpa.task"` to the `manifest.json` for your application. The following is an example:

Example

```
{

  ...
  "sap.bpa.task": {
    "_version": "1.0.0",

    "outcomes": [
      {
        "id": "approve",
        "label": "Allow"
      },
      {
```

```
          "id": "reject",
          "label": "Deny"
        }
      ],
      "inputs": {
        "$schema": "http://json-schema.org/draft-07/schema",
        "title": "input",
        "type": "object",
        "required": ["newText"],
        "properties": {
          "newText": {
            "type": "string",
            "title": "Textfield",
            "description": "Description for Textfield"
          }
        }
      },
      "outputs": {
        "$schema": "http://json-schema.org/draft-07/schema",
        "title": "output",
        "type": "object",
        "required": ["newText"],
        "properties": {
          "newText": {
            "type": "string",
            "title": "Textfield",
            "description": "Description for Textfield"
          }
        }
      },
      "category": "approval"
    }
  }
```

The namespace must have the following properties:

- The `_version` must be the namespace version.
- The `category` can be either `standard` or `approval`, depending on the type of form you want to use in SAP Build Process Automation. The outcomes depend on the category you choose.
- The outcomes can be an array of objects that have an `id` and a `label`. The `id` can be `submit`, `approve` or `reject`. You can enter any text for the `label`.
    - If the `category` is `standard`, the outcome must be `[{id:"submit", label: "AnyLabel"}]`
    - If the `category` is `approval`, the outcome must be `[{id:"approve", label:"SomeLabel"}, {id:"reject", label: "SomeLabel2"}]`.

      Note

      The outcomes you define in the `manifest.json` determine the actions that the application has to create for **My Inbox**.

- The `inputs` must be a draft-7 JSON schema and define the context object you receive with the Workflow API for Cloud Foundry. For more information, see the Inputs and Outputs supported by SAP Build Process Automation Forms section below.
- The `outputs` must be a draft-7 JSON schema and define the context object you have to send with the Workflow API for Cloud Foundry. For more information, see the Inputs and Outputs supported by SAP Build Process Automation Forms section below.

Configuring the Component.js File

Once you've created your SAPUI5 application, you need to adapt the `Component.js` file.

Replace the definition of the `Component.js` file, which is the second parameter of `UIComponent.extend`, using the following example:

Example

```
{
        metadata: {
          manifest: "json",
        },

        /**
         * The component is initialized by UI5 automatically during the startup of the app and calls the
         * @public
         * @override
         */
        init: function () {
          // call the base component's init function
          UIComponent.prototype.init.apply(this, arguments);

          // enable routing
          this.getRouter().initialize();

          // set the device model
          this.setModel(models.createDeviceModel(), "device");

          this.setTaskModels();
          const rejectOutcomeId = "reject";
          this.getInboxAPI().addAction(
            {
              action: rejectOutcomeId,
              label: "Deny",
              type: "reject",
            },
            function () {
              this.completeTask(false, rejectOutcomeId);
            },
            this
          );
          const approveOutcomeId = "approve";
          this.getInboxAPI().addAction(
            {
              action: approveOutcomeId,
              label: "Allow",
              type: "accept",
            },
            function () {
              this.completeTask(true, approveOutcomeId);
            },
            this
          );
        },

        setTaskModels: async function () {
          // set the task model
          var startupParameters = this.getComponentData().startupParameters;
          this.setModel(startupParameters.taskModel, "task");
```

```
    // set the task context model
    var taskContextModel = new sap.ui.model.json.JSONModel(
      this._getTaskInstancesBaseURL() + "/context"
    );
    this.setModel(taskContextModel, "context");

    // parse Date objects and set in own model
    await taskContextModel.loadData(this._getTaskInstancesBaseURL() + "/context");
  },

  _getTaskInstancesBaseURL: function () {
    return (
      this._getWorkflowRuntimeBaseURL() +
      "/task-instances/" +
      this.getTaskInstanceID()
    );
  },

  _getWorkflowRuntimeBaseURL: function () {
   var ui5CloudService = this.getManifestEntry("/sap.cloud/service").replaceAll(".", "")
   var ui5ApplicationName = this.getManifestEntry("/sap.app/id").replaceAll(".", "");
   var appPath = `${ui5CloudService}.${ui5ApplicationName}`;
   return `/${appPath}/api/public/workflow/rest/v1`
  },

  getTaskInstanceID: function () {
    return this.getModel("task").getData().InstanceID;
  },

  getInboxAPI: function () {
    var startupParameters = this.getComponentData().startupParameters;
    return startupParameters.inboxAPI;
  },

  completeTask: function (approvalStatus, outcomeId) {
    this.getModel("context").setProperty("/approved", approvalStatus);
    this._patchTaskInstance(outcomeId);
  },

  _patchTaskInstance: function (outcomeId) {
    const context = this.getModel("context").getData();
    var data = {
      status: "COMPLETED",
      context: {...context, newText: context.newText || ''},
      decision: outcomeId
    };

    jQuery.ajax({
      url: `${this._getTaskInstancesBaseURL()}`,
      method: "PATCH",
      contentType: "application/json",
      async: true,
      data: JSON.stringify(data),
      headers: {
        "X-CSRF-Token": this._fetchToken(),
      },
    }).done(() => {
      this._refreshTaskList();
```

```
          })
        },

        _fetchToken: function () {
          var fetchedToken;

          jQuery.ajax({
            url: this._getWorkflowRuntimeBaseURL() + "/xsrf-token",
            method: "GET",
            async: false,
            headers: {
              "X-CSRF-Token": "Fetch",
            },
            success(result, xhr, data) {
              fetchedToken = data.getResponseHeader("X-CSRF-Token");
            },
          });
          return fetchedToken;
        },

        _refreshTaskList: function () {
          this.getInboxAPI().updateTask("NA", this.getTaskInstanceID());
        },
      }
```

Configuring the App.view.xml File

If you want to use a text input, replace the App element in your `app.view.xml` file as follows:

Example

```
<App id="app">
        <Input id ="input" value = "{context>/newText}"></Input>
    </App>
```

Inputs and Outputs supported by SAP Build Process Automation Forms

The following schema shows all property definitions a SAP Build Process Automation form can output or handle as input.

Sample Code

```
{
  "$schema": http://json-schema.org/draft-07/schema,
  "title": <"input" or "output">,
  "type": "object",
  "required": [],
  "properties": {
    "newText": {
      "title": "New Text",
      "description": "",
      "type": "string",
    },
    "newTextArea": {
      "title": "New Text Area",
```

```
        "description": "",
        "type": "string",
      },
      "newNumber": {
        "title": "New Number",
        "description": "",
        "type": "number"
      },
      "newDate": {
        "title": "New Date",
        "description": "",
        "type": "string",
        "format": "date"
      },
      "newChoice": {
        "title": "New Choice",
        "description": "",
        "type": "array",
        "uniqueItems": true,
        "items": {
          "type": "string",
          "enum": ["Option 1", "Option 2"]
        }
      },
      "newChoice2": {
        "title": "New Choice",
        "description": "",
        "type": "string",
        "enum": ["Option 1", "Option 2"]
      },
      "newDropdown": {
        "title": "New Dropdown",
        "description": "",
        "type": "string",
        "enum": ["Option 1", "Option 2"]
      },
      "newCheckbox": {
        "title": "New Checkbox",
        "description": "",
        "type": "boolean"
      },
      "newTable": {
        "title": "New Table",
        "description": "",
        "type": "array",
        "items": {
          "$ref": "#/definitions/newTableItems"
        }
      }
    },
    "definitions": {
      "newTableItems": {
        "type": "object",
        "required": [],
        "properties": {
          "newText": {
            "title": "New Text",
            "description": "",
            "type": "string"
```

```
          }
        }
      }
    }
  }
```

Update the `xs-app.json` File

In the `xs-app.json` in your UI module folder (the same folder that also contains the webapp), you must replace the following code snippet:

Sample Code

```
{
    "source": "^/bpmworkflowruntime/(.*)$",
    "target": "/public/workflow/rest/$1",
    "service": "com.sap.spa.processautomation",
    "endpoint": "api",
    "authenticationType": "xsuaa"
},
```

Use it with the following code snippet instead. Make sure that it is the first entry of the routes array.

```
{
    "source": "^/api/(.*)$",
    "target": "$1",
    "service": "com.sap.spa.processautomation",
    "endpoint": "api",
    "csrfProtection": true,
    "authenticationType": "xsuaa"
},
```

Troubleshooting

When you build your application, if you receive an error message stating that your UI5 CLI installation is outdated, update the `@ui5/cli` version in the `package.json` file of the UI module. You can use versions from version 3 onwards, for example "^3.4.0.".

Add Flow Controls

You can add and configure controls to your business process.

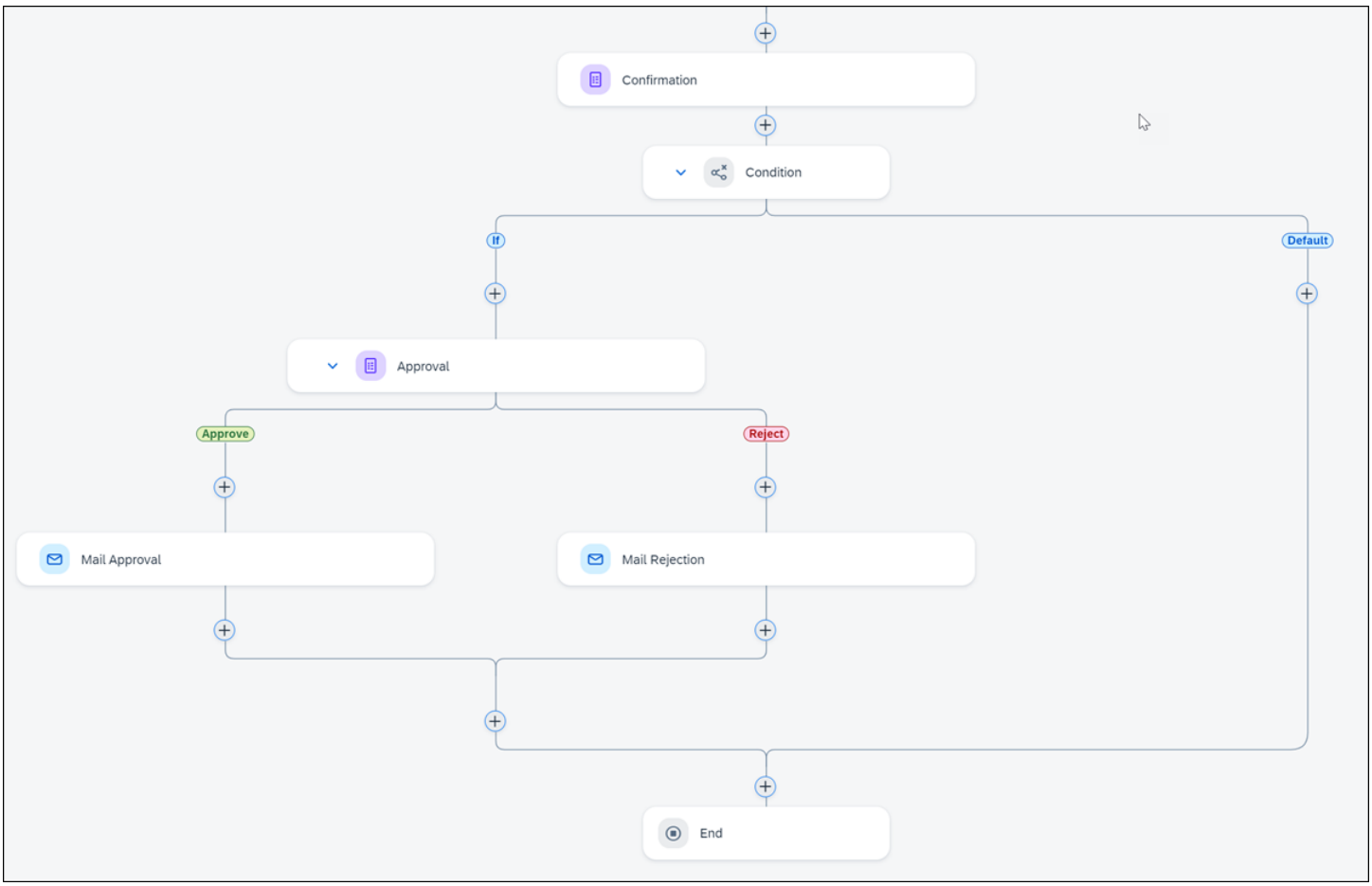
- Add and Configure Process Conditions
- Add a Branch to a Process

Add and Configure Process Conditions

You can add and configure conditions to a business process, enabling you to route a running process based on applying IF or DEFAULT rules to the process context.

Context

In the following example, this process condition checks if the value of a submitted request is over 1000. If yes, then the request is sent for approval. If no, then the request is considered as a default request, automatically approved, and a confirmation mail is sent.

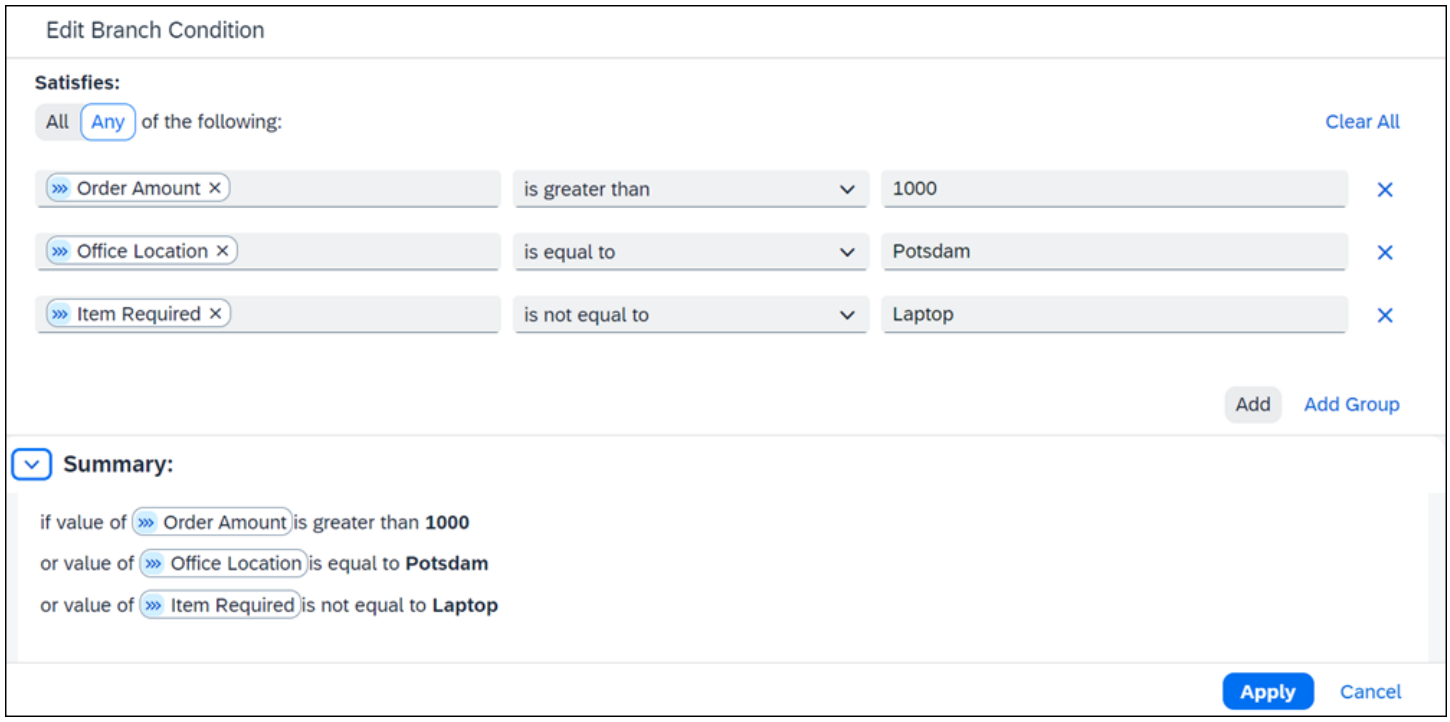


For this example, the condition is configured like this:

You can also create more detailed conditions by using the All or Any feature.

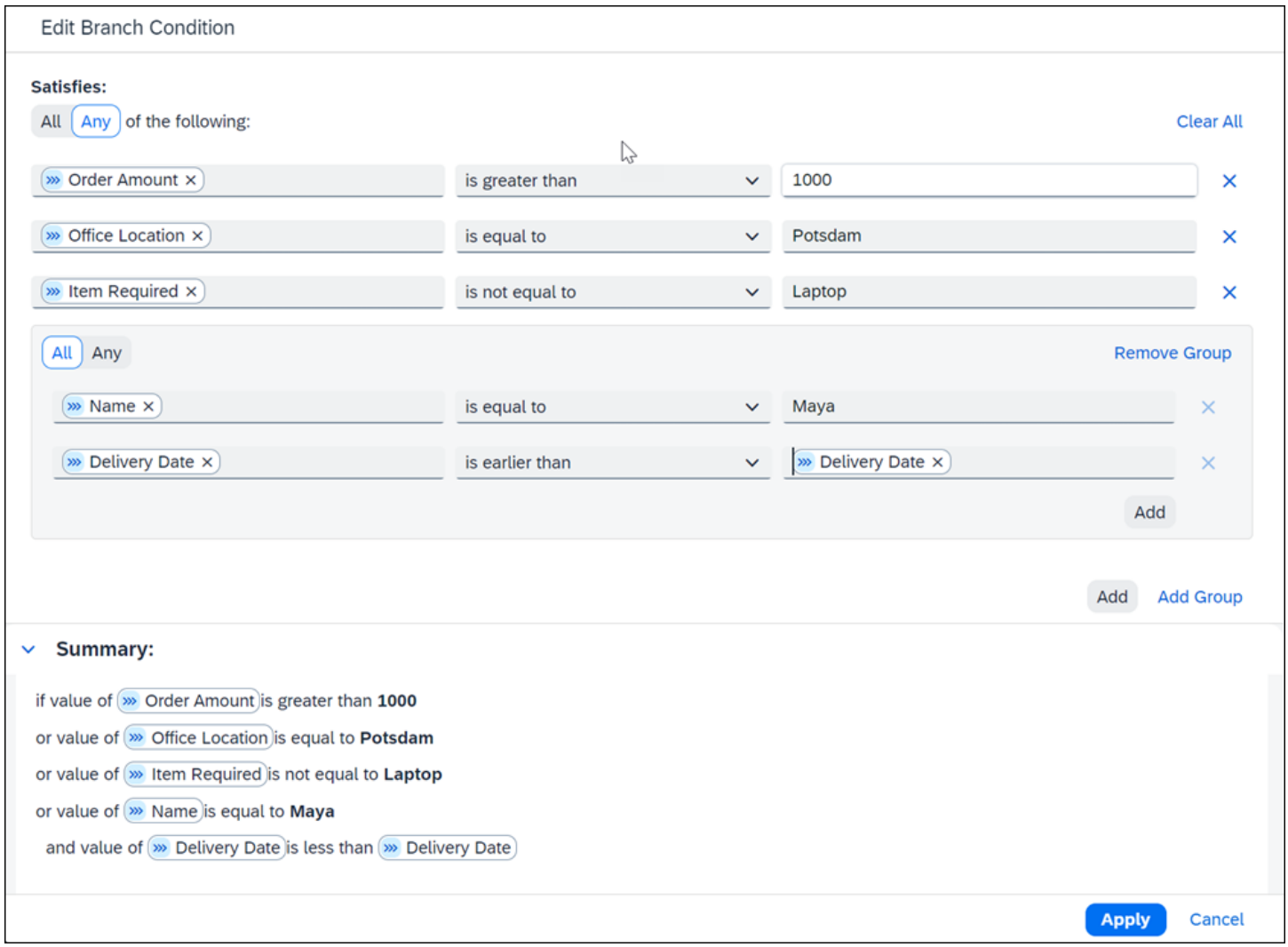
In the following example, the process must include Any of the following three values for the process to be sent for approval:

- Value of the request is greater than 1000.
- Office location is Potsdam.
- The item requested isn't a laptop.



You can also apply both the All and Any rules to groups of rules, such as in the following example:

In this example, the process must include any of the first group conditions AND all of the second group of conditions for the request to be approved.



Procedure

1. From the process editor canvas, choose + (Plus) > Flow Login > Condition.

   The condition is added to the process and the side panel is displayed.

2. **Optional:** Enter a Condition Name and a Branch Name to customize how the condition is displayed in the process.
3. Choose Open Condition Editor.
4. Configure a minimum of one condition to be satisfied.

   For example, the office location must be Potsdam:

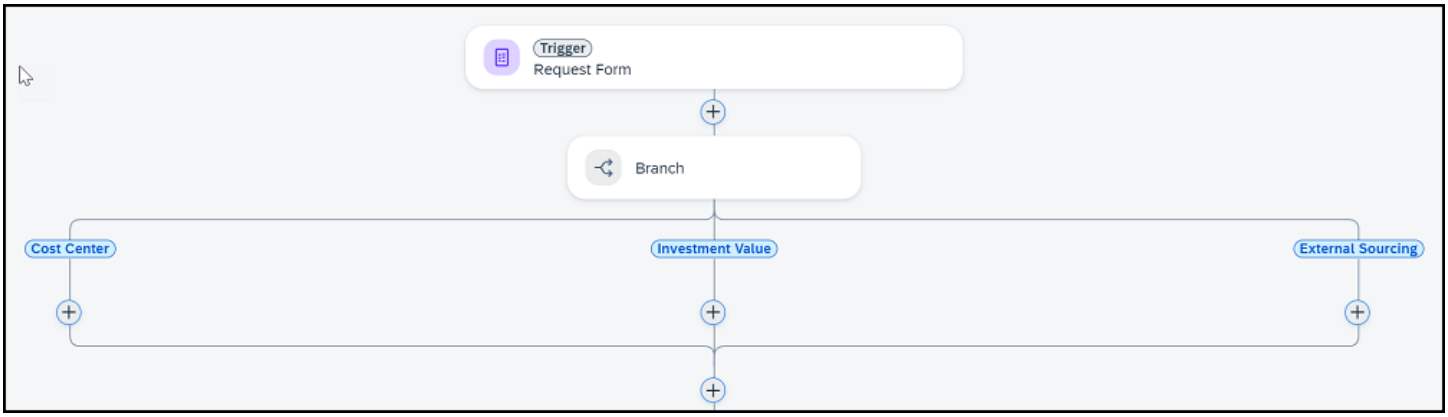
5. Choose Apply.
6. Save your changes.

Add a Branch to a Process

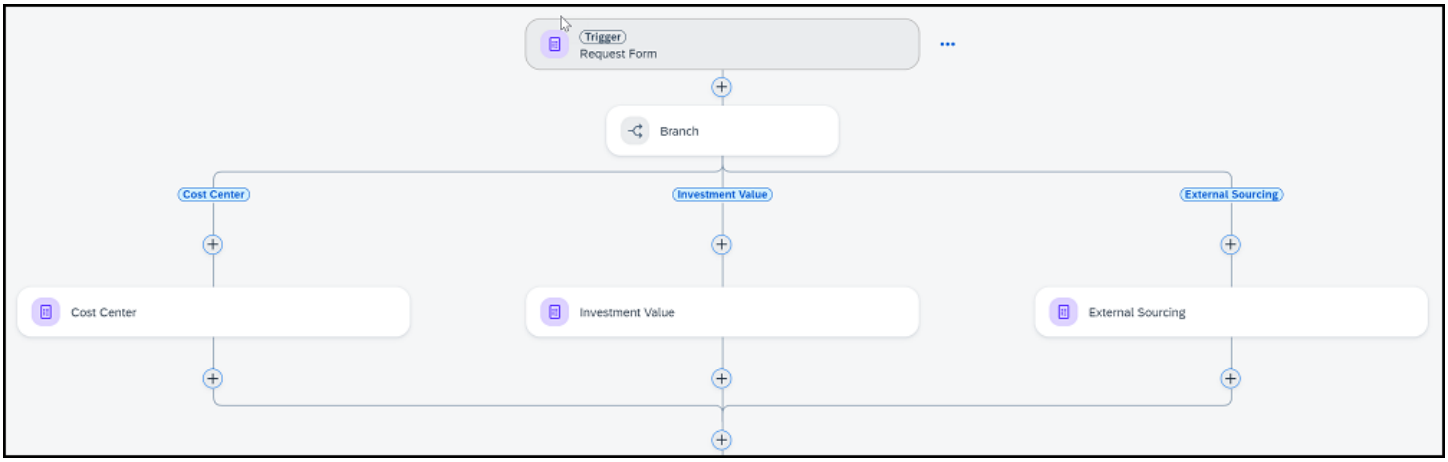
You can add a maximum of 10 parallel branches to your business process, enabling you to run multiple process tasks at the same time. These tasks run concurrently, with the process progressing once all tasks are complete.

Context

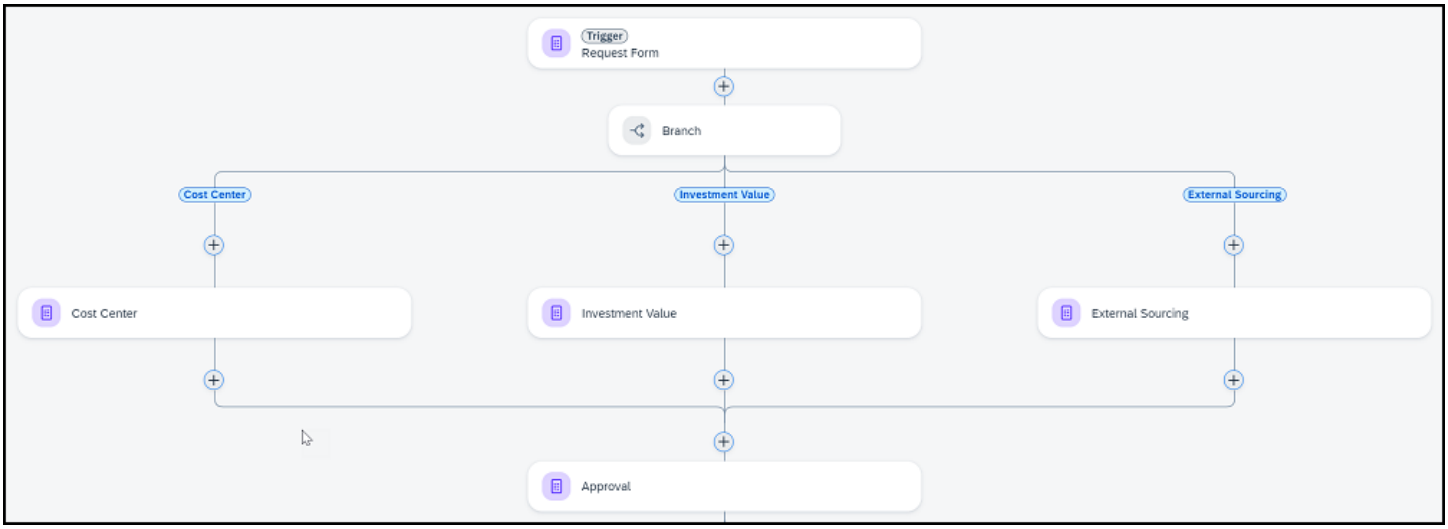
In the following example, three branches have been added to an investment request process and labeled accordingly:



These branches are then enhanced by adding process steps, in this case three additional forms:

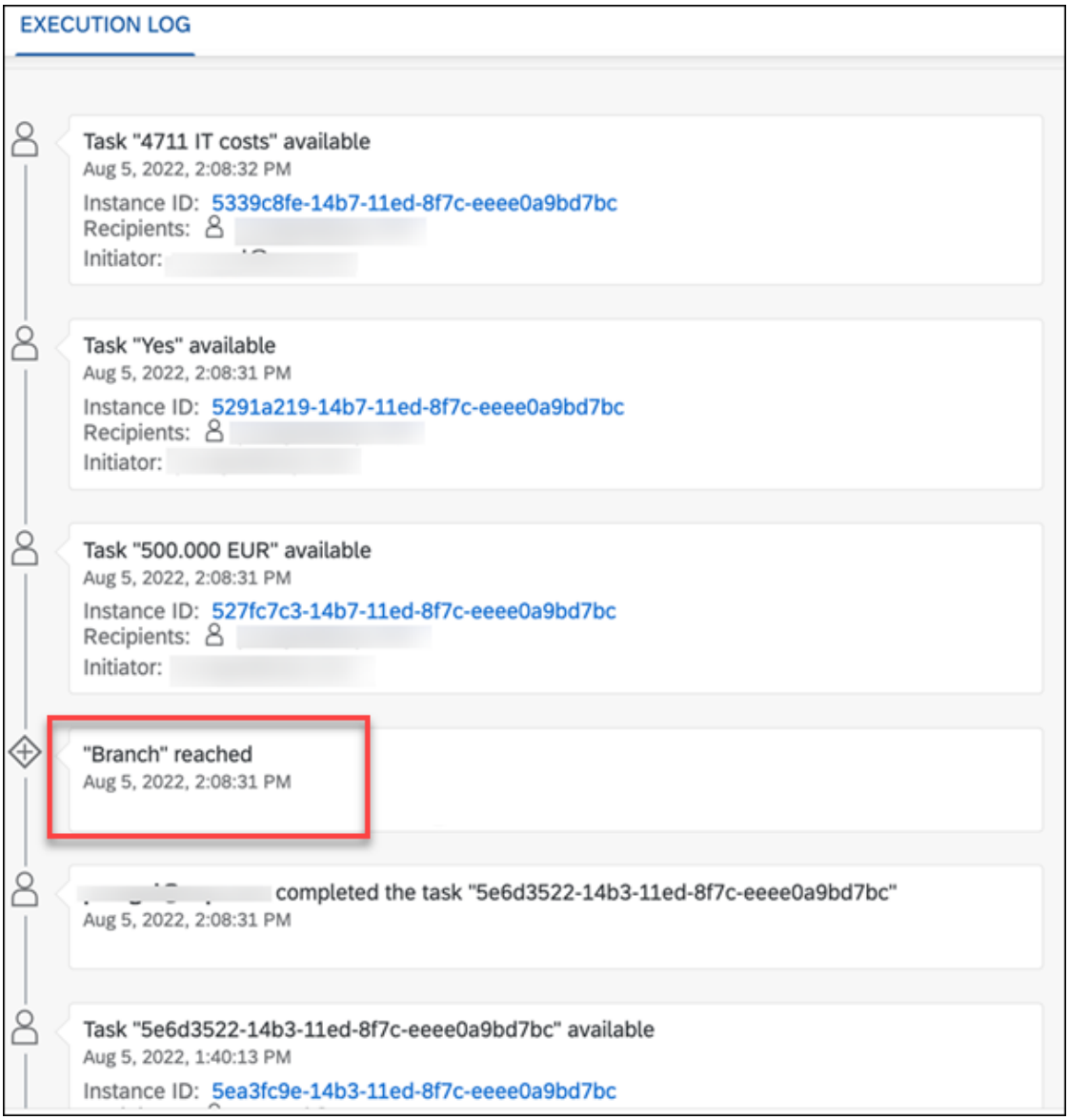


An approval form is then added to the process. This approval is only triggered once all three branches have successfully processed.

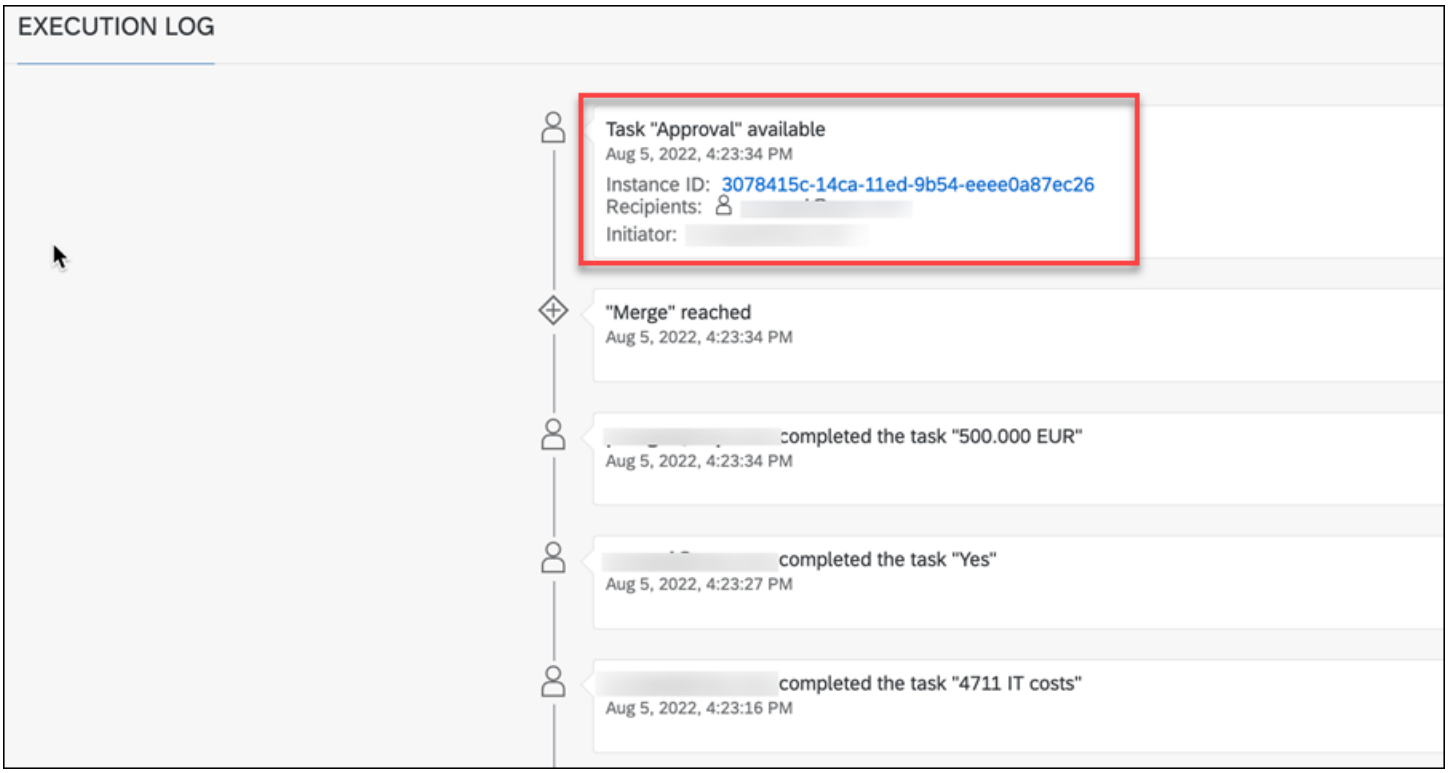


You can monitor the status of branch tasks in a running process using the Execution Log in the Monitoring area.

In this example the process has reached the branch:



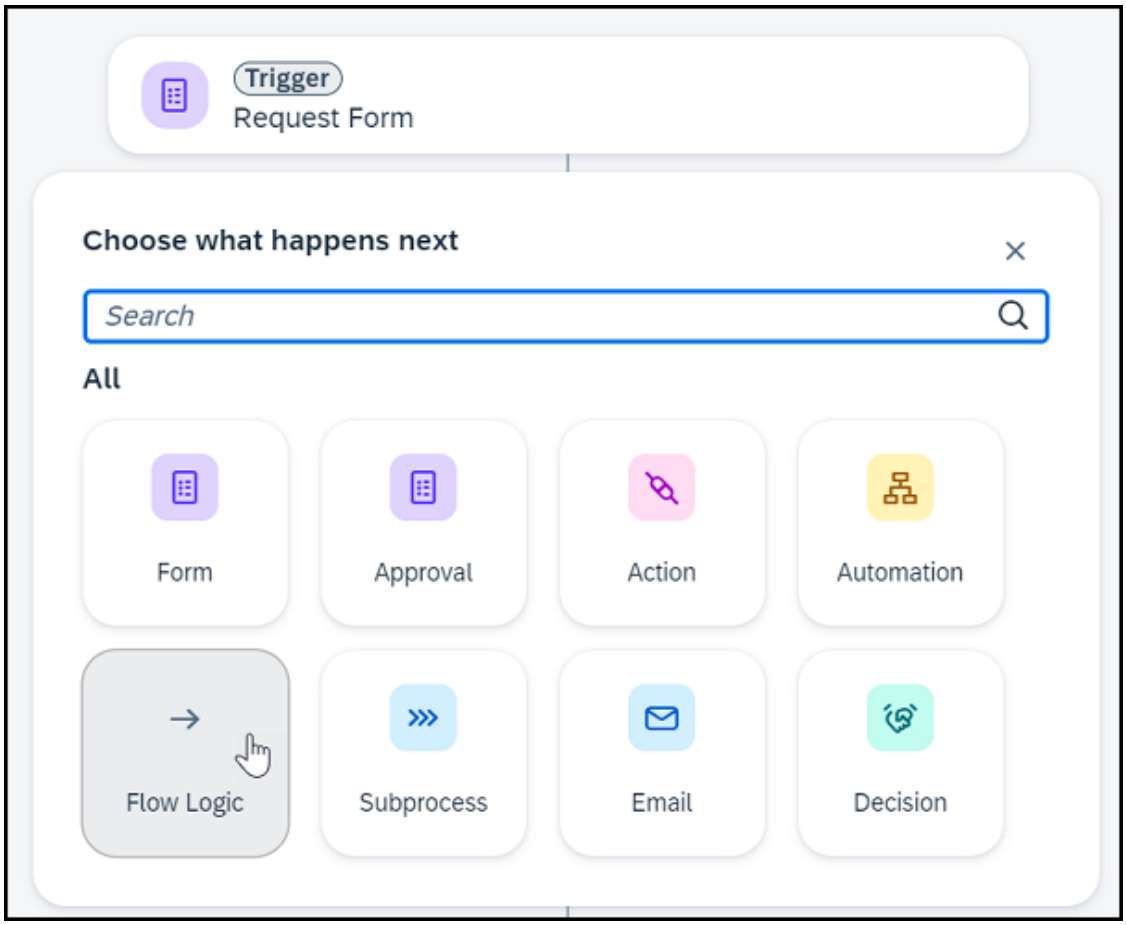
And in this example all branch tasks have successfully processed and the approval has triggered:



Procedure

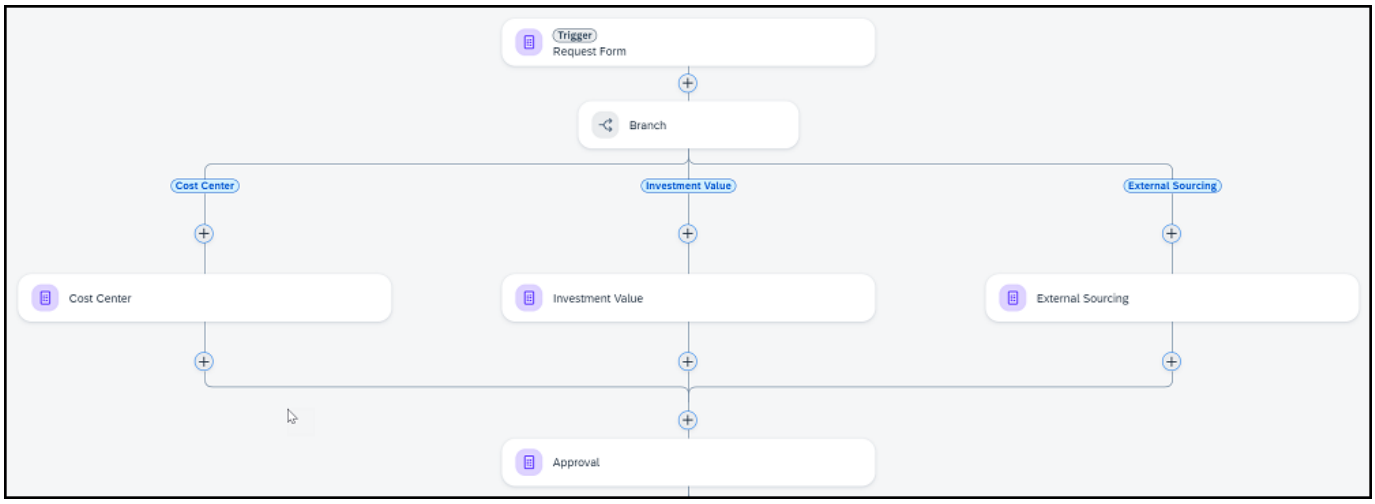
1. From the process editor, click + and select Controls - Branch.

   From the process editor, choose + > Flow Logic > Parallel Branch.

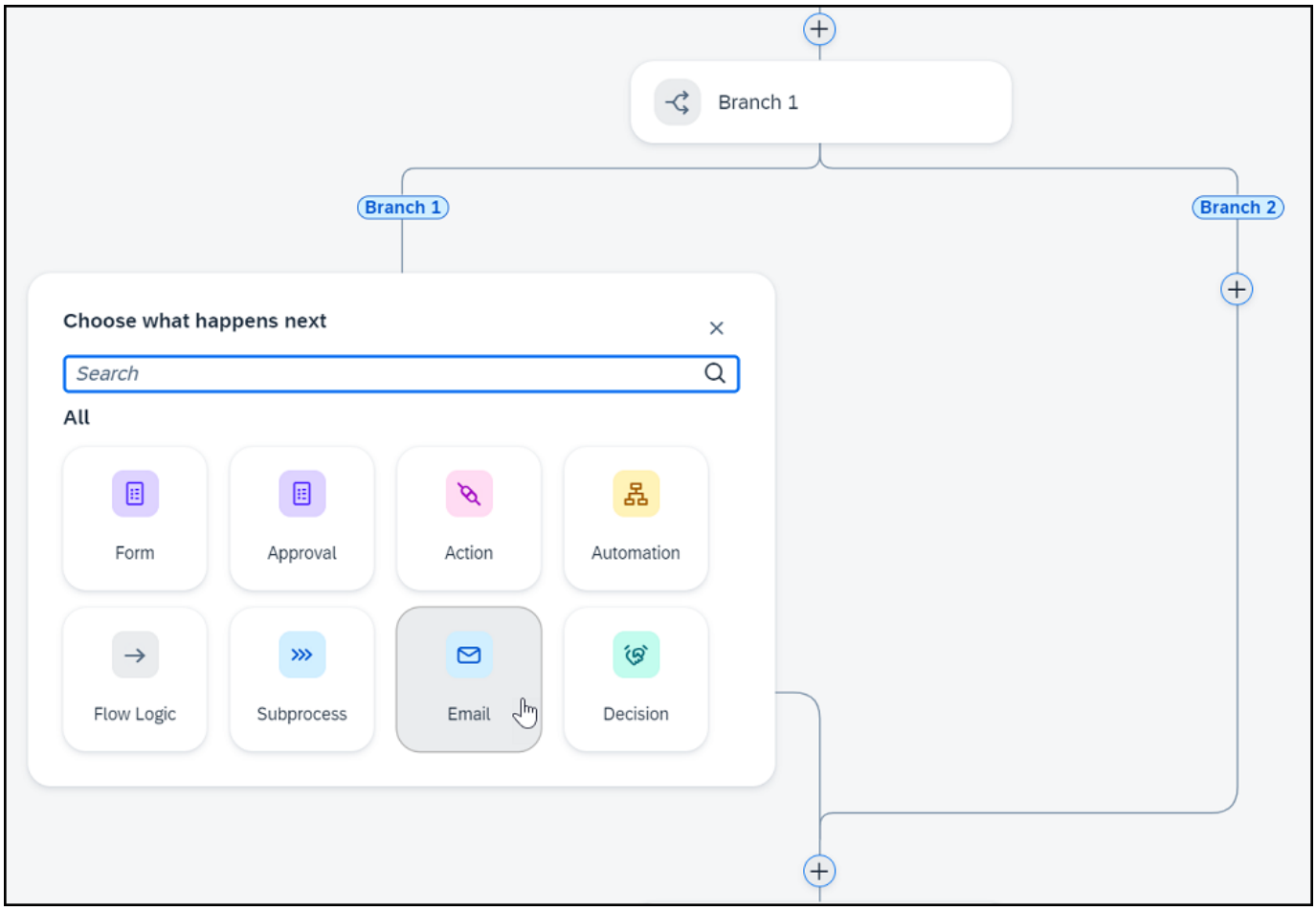


   Two branches are added to the process and the branch settings are displayed.

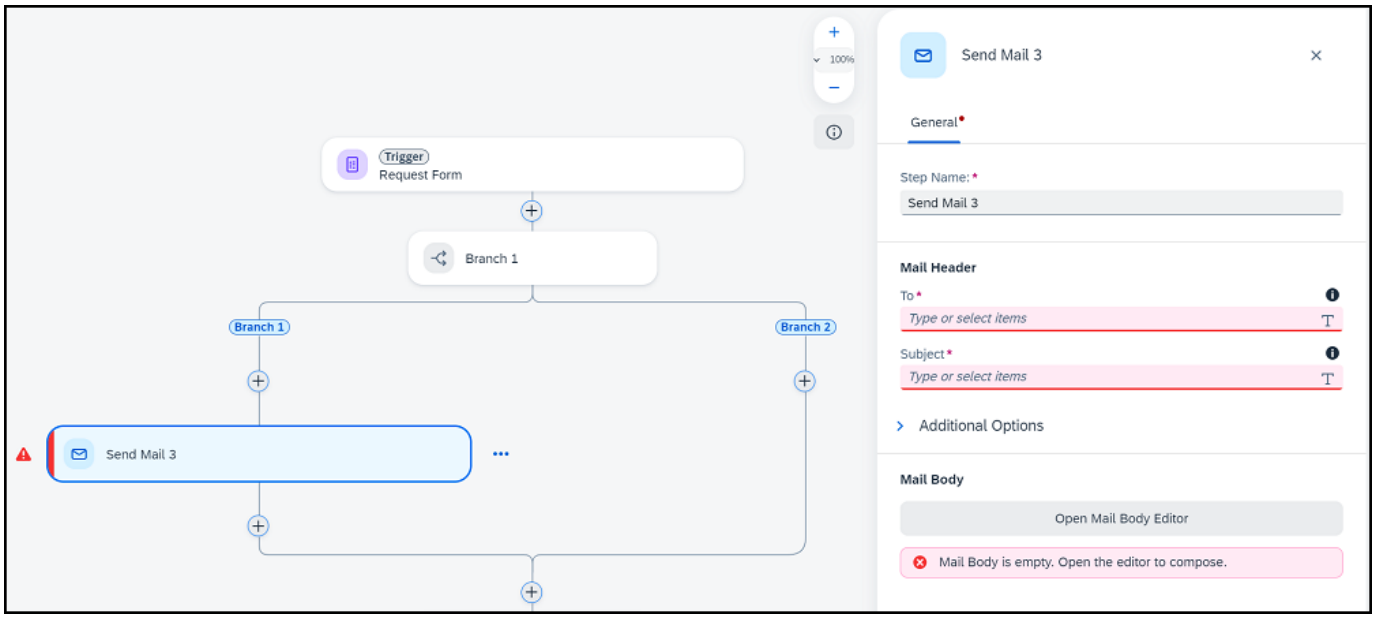
2. Configure the branch accordingly, adding an optional Branch Name and additional branches using Add branch.



3. Add process steps to each branch using + and selecting the required step. In this example, a mail notification is added to the branch.



4. Configure the process step, ensuring that all errors are corrected. In this example, the mail notification is missing the **Mail Header** information.



5. Once all branches are configured, save your changes.

   The branches have been added to the process and will be processed concurrently when the process is running.

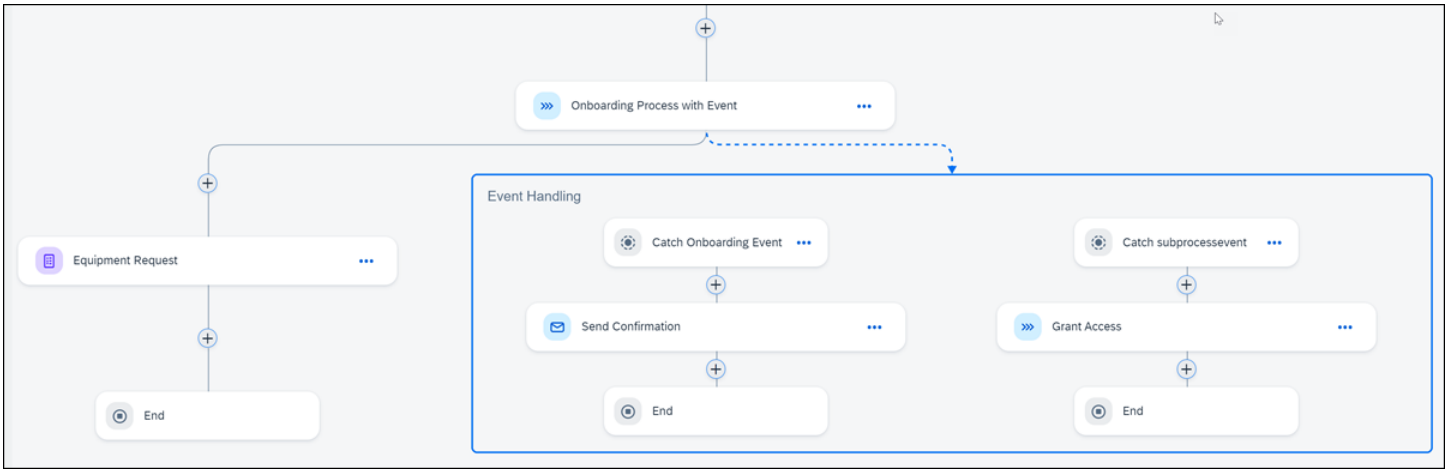
Raise an Escalation Event in Process Editor

You can raise and configure escalation events from anywhere in a business process.

Context

For an escalation event, you only define its name without any action attached. If the process reaches that event, it throws an escalation to its parent process. If you configure event handling for this event in the parent processe, then the escalation is caught and you can send, for example, a notification.

In the following sample, we have a parent process with a subprocess that catches two raised events.



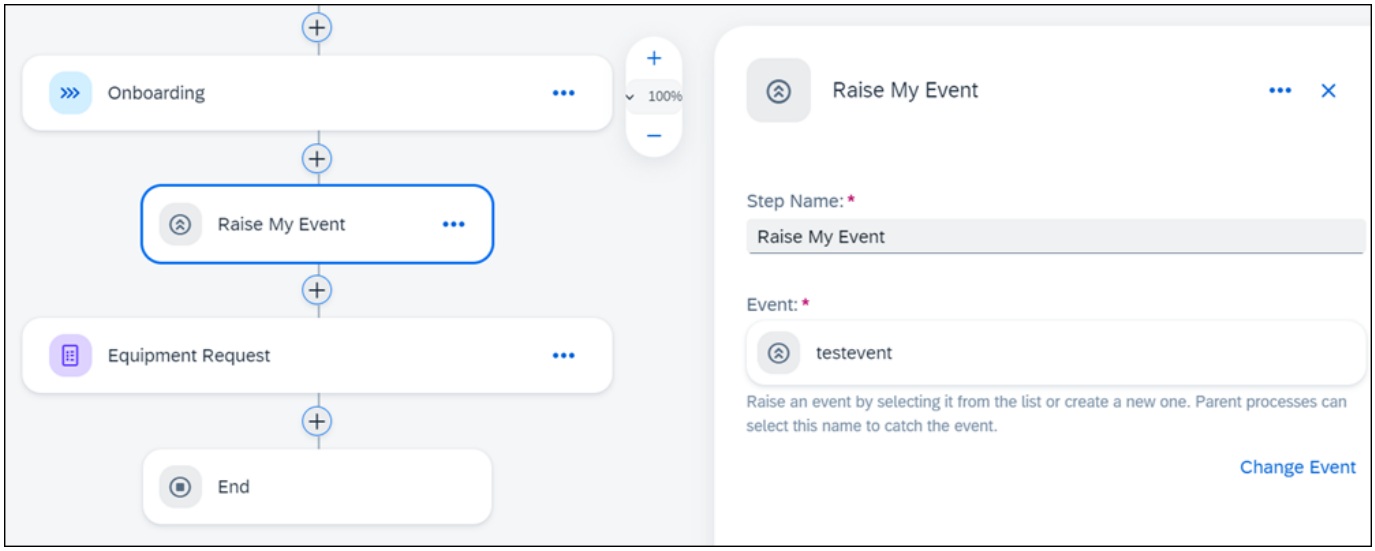
Procedure

1. From the process editor canvas, choose + (Plus) > Flow Controls and Events > Raise Event.
2. On the Raise Event tile, choose Create Event.
3. Enter a name for the event and choose Create.

   The event is added to the process.

4. **Optional:** On the side panel, you can change the event by choosing Change Event.

You can then select an event from the list that opens or create another event from there.



5. Save your changes.

Handle Events Using the Parent Process

You can catch events from a business process or workflow.

Context

The parent process can catch an event. Not catching an event is not considered an error.

The following types of events can be raised by a subprocess or a workflow:

- Escalation event: This event type can be caught by its parent process.
- Timer events: This event type can be caught by its parent process or a user task, for example, an approval.

Handle Escalation Events

Procedure

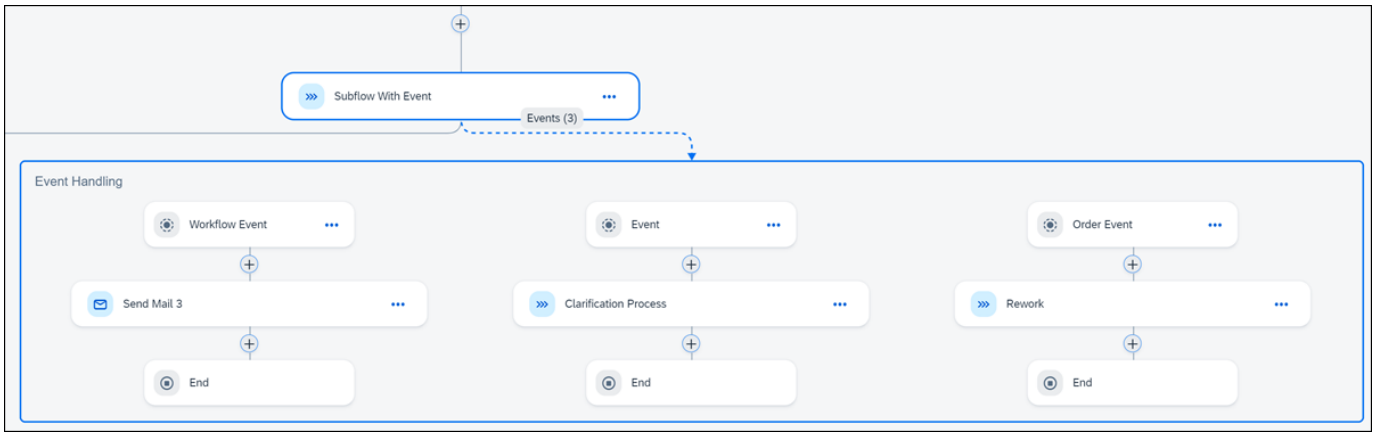
1. From the step menu of the process or workflow, choose *Options*, and then choose **Handle Event**.

   If your process does not contain a raised event yet, choose **Open Subprocess** to add an event to it.

2. On the **Event Handling** tile, choose **Catch Event**.
3. In the catch flow that is created, choose **+ (plus)** to add the step to catch the event.

   You can catch multiple events, and they are then displayed next to each other. The number of handled events is displayed if you click the process or workflow. If you then click the number, the **Event Handling** tile opens.

4. Configure the event on the side panel.
5. Save your changes.



Handle Timer Events

Procedure

1. From the step menu of the process or workflow, choose *Options*, and then choose **Handle Event**.

   If your process does not yet contain a raised , choose **Open Subprocess** to add an event to it.

2. On the **Event Handling** tile, choose **Timer Event**.
3. In the catch flow that is created, choose **+ (plus)** to add the step to catch the event.

   You can catch multiple events and they are then displayed next to each other. The number of handled events is displayed if you click the process or workflow. If you then click the number, the **Event Handling** tile opens.

4. Configure the event on the side panel.

   Select whether the timer is based on the task creation date or a reference date:

   - **Task Creation**: Set the toggle to **On Task Creation** to start the event handling directly or to **After Task Creation** to start the event handling only after the completion of the wait duration that you define.
   - **Reference Date**: Select a date from the process context list, for example, the sales order date.

5. Save your changes.

Handle a Timer Event with a User Task

You can catch timer events from a user task, for example, an approval or a form.

Procedure

1. From the step menu of the user task, choose *Options*, and then choose **Handle Event**.
2. On the **Event Handling** tile, choose **Timer Event**.
3. In the catch flow that is created, choose **+ (plus)** to add the step to catch the event.

   You can catch multiple events that are then displayed next to each other. The number of handled events is displayed if you click the process or workflow. If you then click the number, the **Event Handling** tile opens.

4. Configure the event on the side panel.

   Select whether the timer is based on the task creation date, the due date, or a reference date:

   - **Task Creation**: Set the toggle to **On Task Creation** to start the event handling directly or to **After Task Creation** to start the event handling only after the completion of the wait duration that you define.
   - **Due Date**: Start the event handling when the due date of the user task arrives.

If you select this option for a user task that has no due date defined, you get a notification telling you to update the user task first before you can use this option.

- **Reference Date**: Select a date from the process context list, for example, the sales order date. The event handling is started on that date.

5. Save your changes.

Add Actions to a Process

You can embed external skills and capabilities into your SAP Build Process Automation projects using actions. Action projects can either be created in the lobby, allowing you to upload an Open API specification file, or imported from the store to your library.

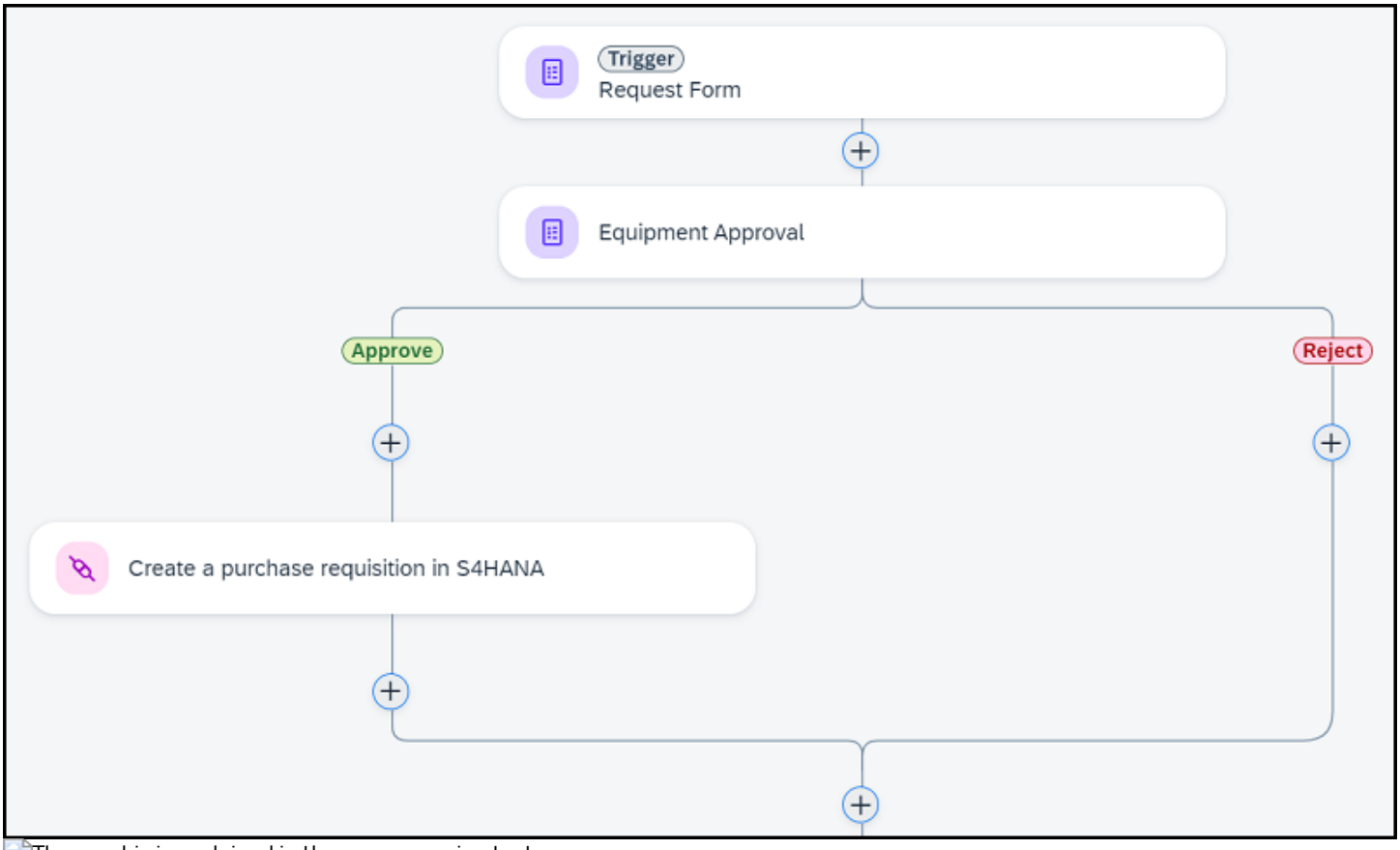
Prerequisites

Before adding an action to your business process, the corresponding action project must be available in your lobby. There are two ways to add an action project to your lobby:

- Create an Action Project
- Use the Store

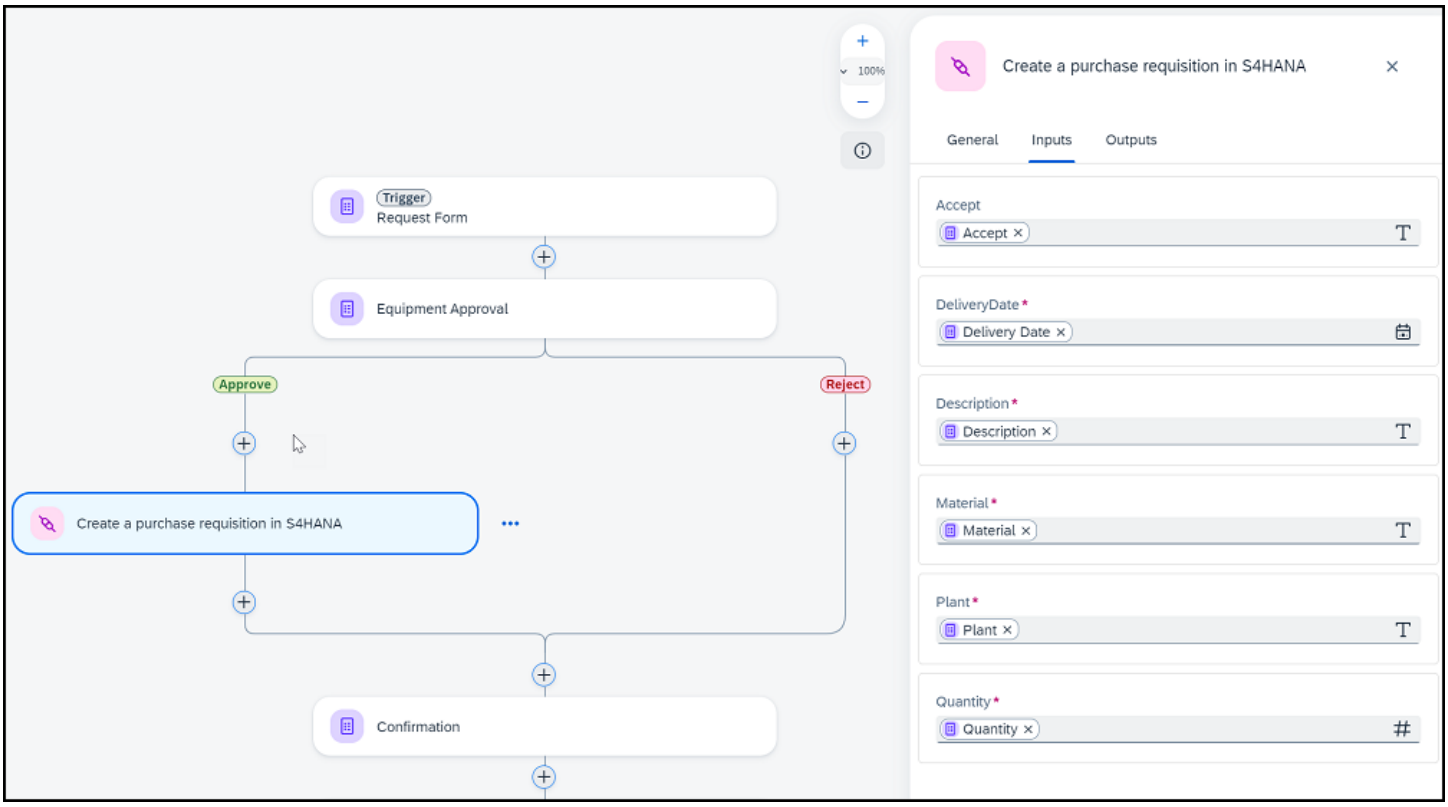
Context

In the following simple request process, a request submission form starts the process running. If the request is approved, an action is used to create a purchase requisition in an external system:



The graphic is explained in the accompanying text.

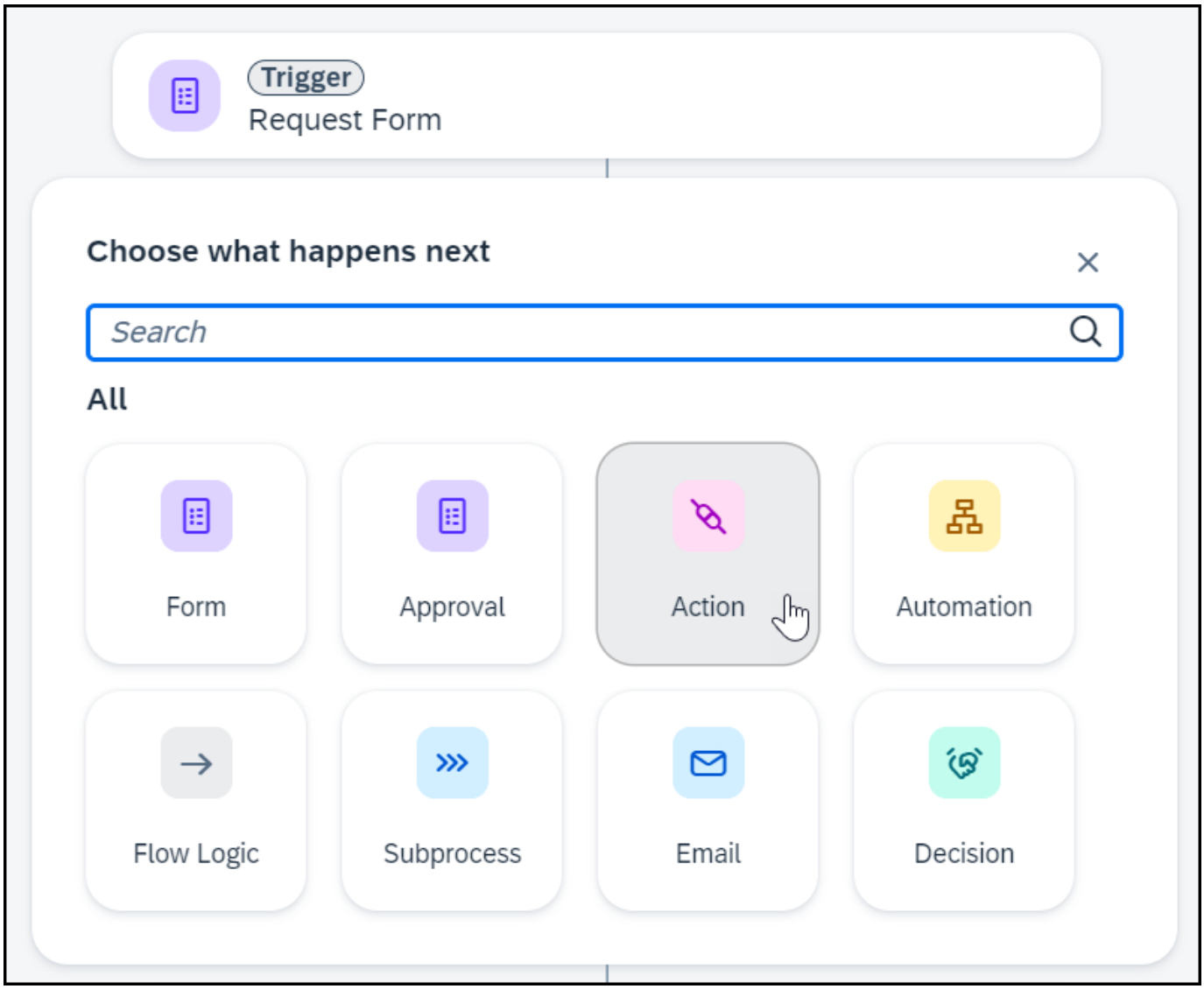
And the inputs of the purchase requisition are mapped from the fields used in the form:



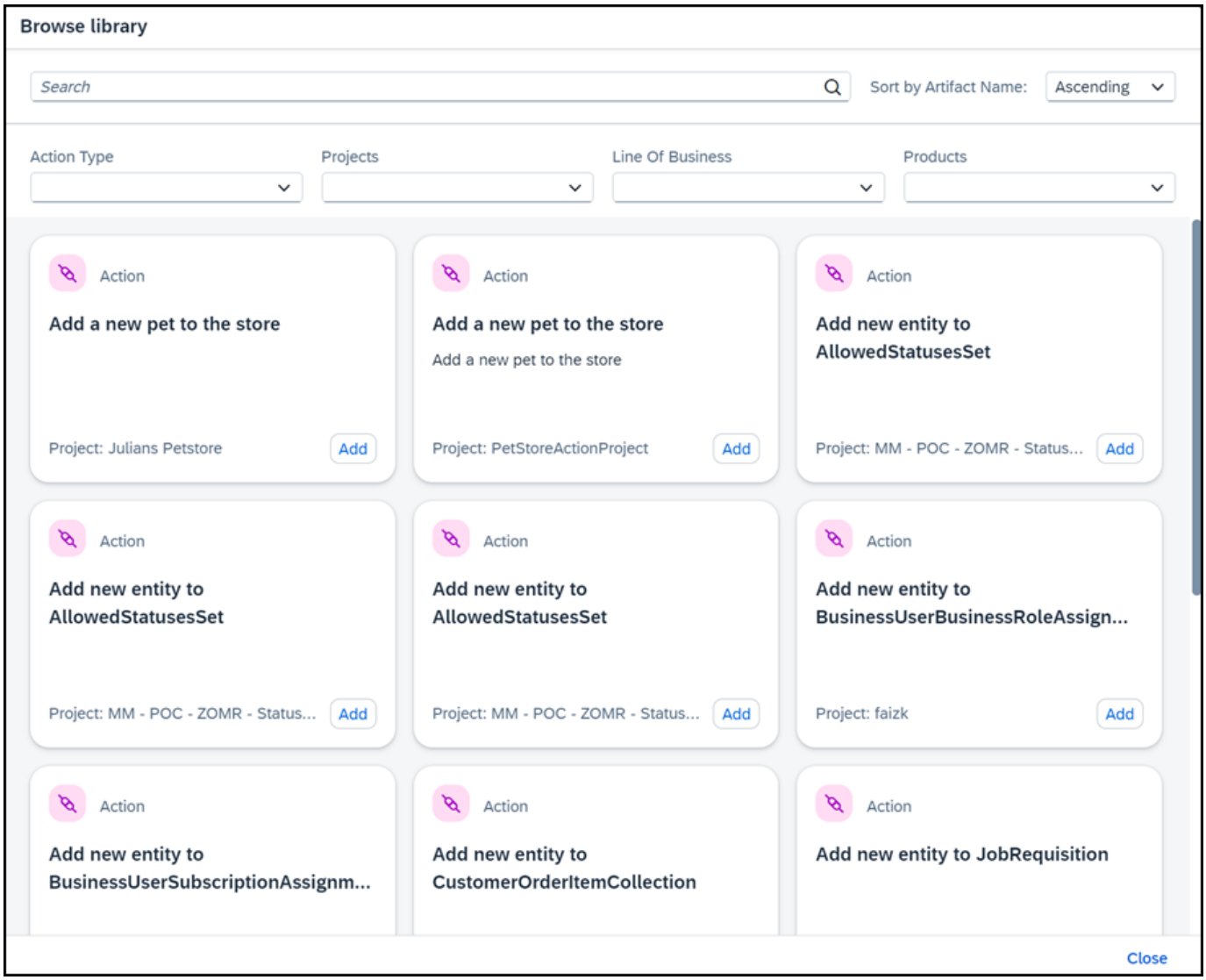
Procedure

1. From the process editor canvas, choose + and select **Actions - Browse Library**.

   From the process editor, choose **+** > **Flow Logic** > **Parallel Branch**.

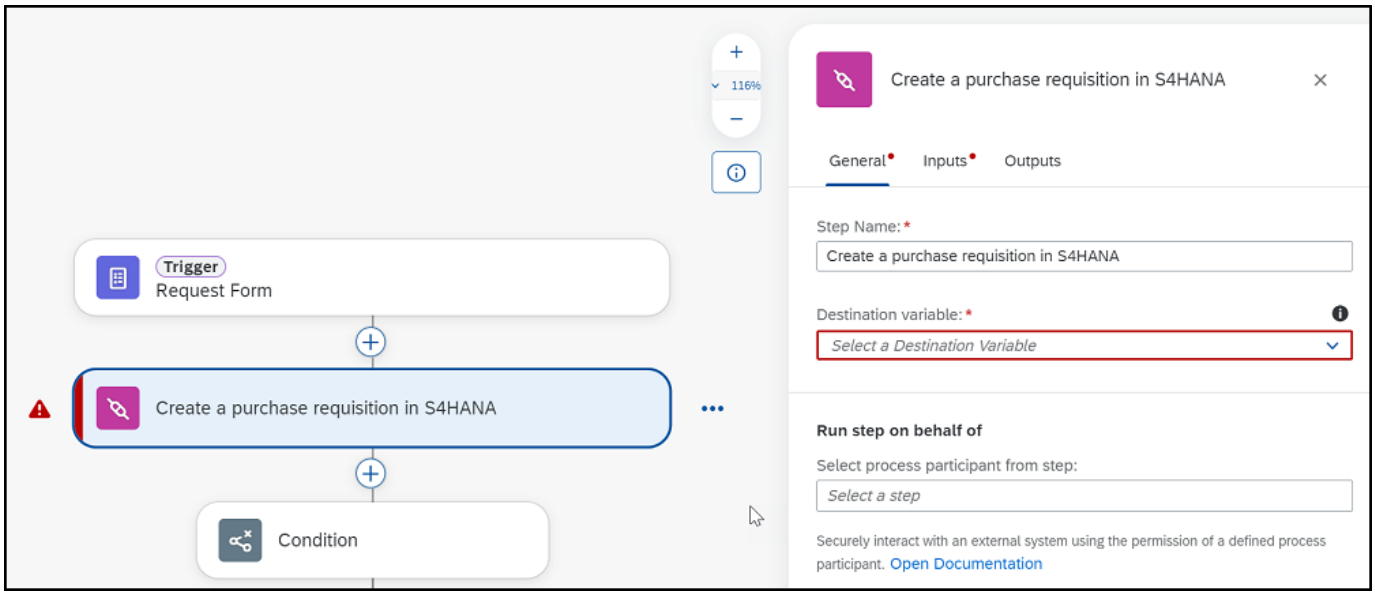


2. Search for the action you want to add to your process, and choose **Add**.



The action is added to your process editor canvas.

3. Click the Action on the canvas, opening the configuration side panel.



4. Configure your action as needed. Outstanding configuration items are highlighted in red and displayed in the design console.

   In this example, the action requires a destination to be configured and for inputs to be mapped.

ℹ Note

When configuring your action, you may need to add and configure other process or automation artifacts before your project can be released and deployed.

5. Save your changes.

   Your action has been added to your process. It can now be consumed when the process is running.

Run Step on Behalf Of

With this feature, you can allow business users who participate in the business process to perform an action on external systems or to execute a subprocess or workflow in the same system.

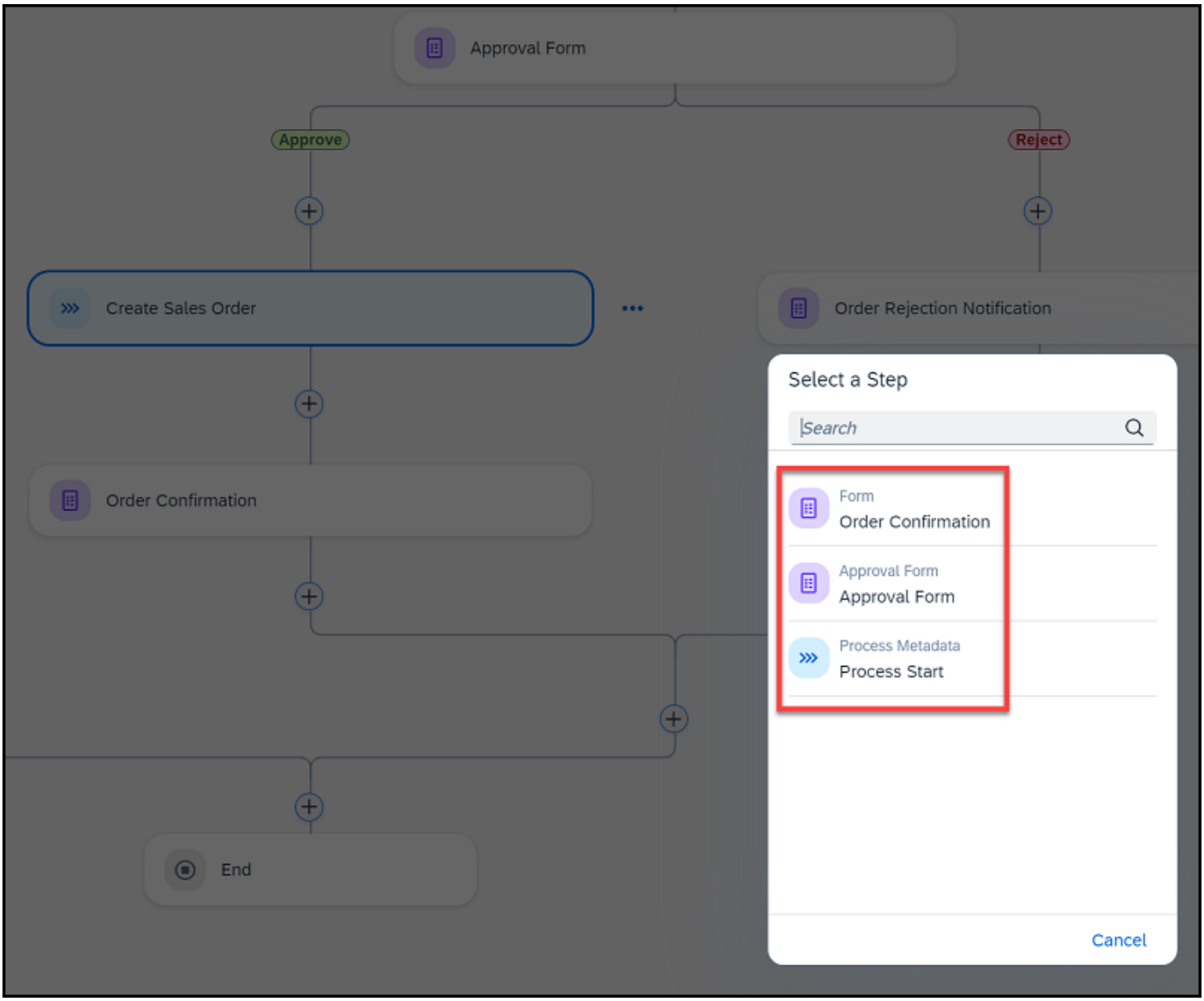
This feature also provides clear information on who triggered the step. This information can help your IT department and is valuable during the audit process.

Prerequisites

- To enable principal propagation, the identity of cloud users gets forwarded to a remote system or service in the Cloud Foundry environment. Principal propagation is possible between two cloud systems or between a cloud and an on-premise system. Set up the connections to the required system using the SAP Destination service. See Configure SAP Build Process Automation Destinations.
- It's important that the step that you are mapping includes the details of a human user who has access to the required system. If the user doesn't have access or if the user is a technical user, the process might fail at runtime.
- The steps eligible to be run on behalf of another user are actions or subprocesses.

Context

You can select the process participant, on whose behalf the step runs, from the process steps. That is, you either map the user who started the process or a user who executed any previous step of the process. That is, you select a step and doing so, indirectly reuse its user in the subprocess. Then, the action or subprocess is executed on behalf of that process participant.



In a procurement process, for example, an employee requests a laptop and gets the manager's approval. As a result, a purchase requisition is created on behalf of the manager who approved the request. Therefore, the purchase requisition is run on behalf of the manager.

Sometimes determining the user of this previous step is tricky:

- We don't recommend using the user of a step inside a condition branch. This is because the outcome of the condition might come from another branch that doesn't include the step you are referring to, in which case, the user that you set isn't available and the process fails.
- By default, selecting a user in **Run step on behalf of** is optional. However, in a sequence of steps like a purchase requisition, it becomes mandatory because the subprocess has an action with **Run step on behalf of** configured to process start.

  Example

  You create a process A that includes an action for which you set **Run step on behalf of** to the user who started the process.

  You create a second process B in which you include process A as a subprocess. Now, this subprocess requires an entry in **Run step on behalf of**.

- It's also possible, that you set up your process well, but then you delete the step that contained the user that you referred to in the action or subprocess. An error message prompts you to enter a user for the action or subprocess again.

Related Information

User Propagation from the Cloud Foundry Environment to SAP S/4HANA Cloud

User Propagation from the Cloud Foundry Environment to SAP SuccessFactors

User Propagation between Cloud Foundry Applications

Add a Subprocess to a Process

You can add subprocesses to your main process in the process editor, allowing you to modularize your business process and reuse subprocesses..

Prerequisites

The subprocess must be in the same or a dependent project of your main process in the process editor.

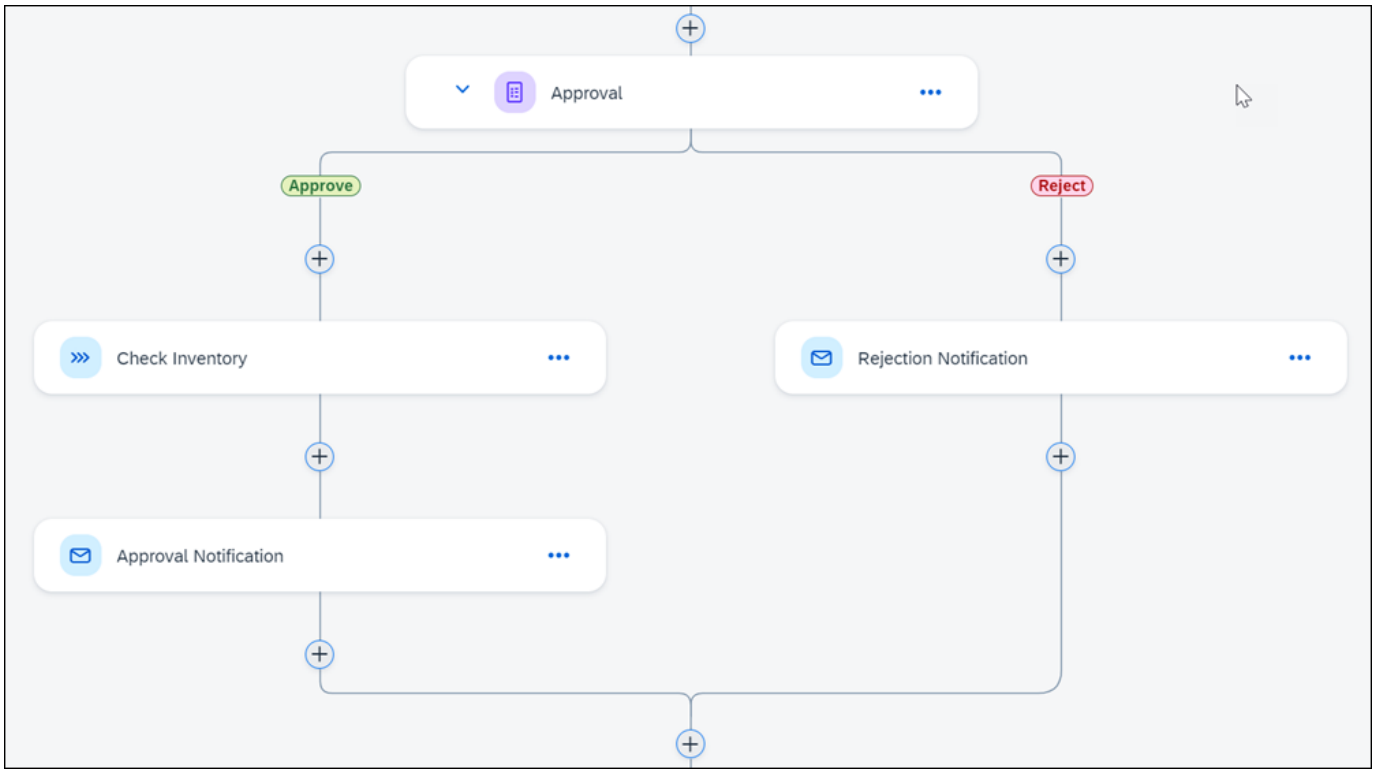
Context

Subprocesses can help you organize your processes. Technically they are simply processes but you can use them to create reuse sequences. Then, you can add them to multiple processes and increase your efficiency and consistency. There are the following types of subprocesses:

- A process that is included, and later run, as part of a main business process.

  Eligible subprocesses need to be available in the same business process project. All processes contained within the same project can be a main process or used as a subprocess, depending on how you use them.

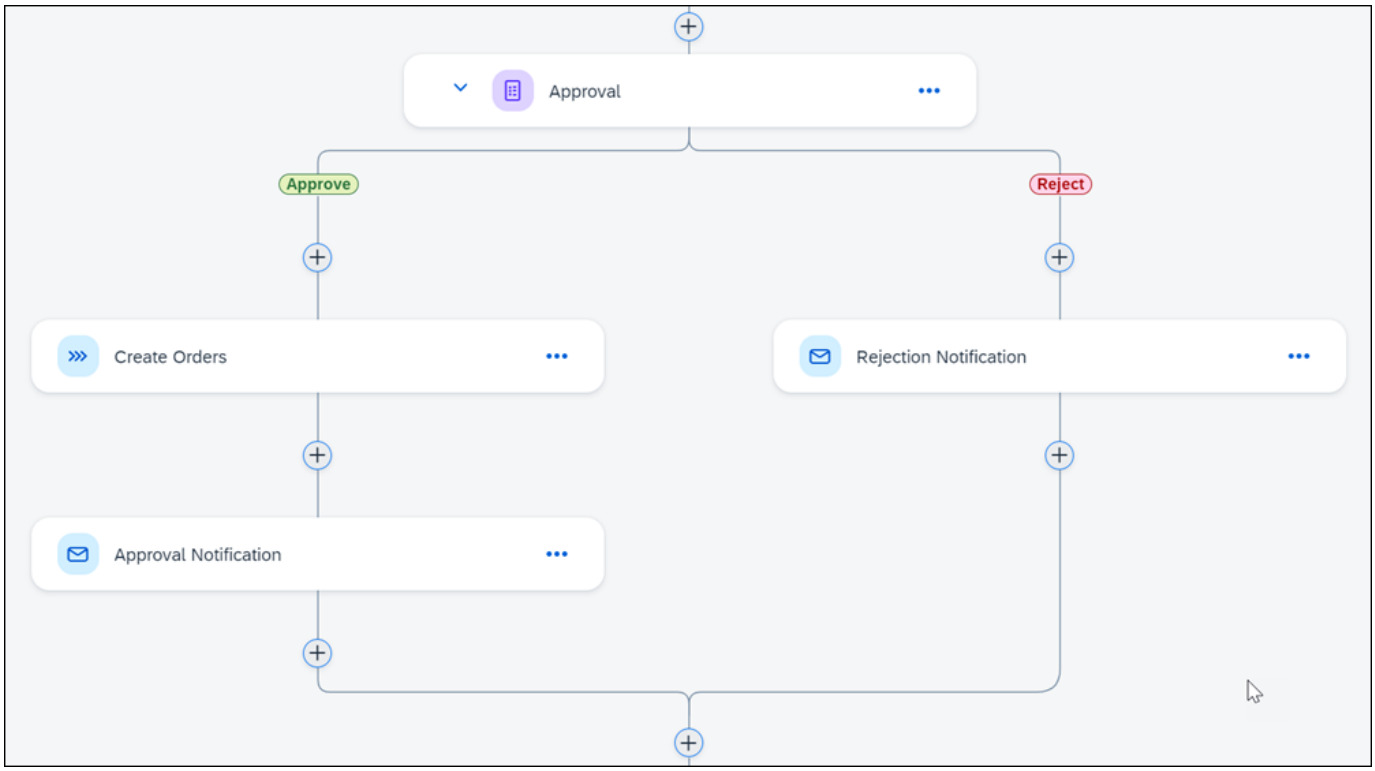
  As an example, a subprocess that checks for inventory levels is included as part of the following purchase order request process:

  The subprocess must run successfully before the approval mail notification is sent to the requestor.

- A workflow that was developed in SAP Business Application Studio or was added as part of live process projects from the store.

  As an example of a workflow started from a process, this request process includes a workflow to create orders for approved requests:



In this case, the workflow runs only for approved requests. This workflow must run successfully before the overall process progresses, which means the approval mail is only sent once the order has been created.

Procedure

1. From the process editor canvas, choose **+ (plus)** > **Subprocess**
2. Select one of the following options:
   - Use a listed existing process. If a listed process is grayed out, it doesn't fulfill all criteria.
   - Create a new subprocess by choosing **Blank Subprocess**. This is simply another process that you create from scratch.
   - Use an existing workflow by choosing **Call a Workflow**.
3. Configure any necessary process or workflow **Inputs** and **Outputs**.
4. Save your changes.

   The process or workflow is added as a subprocess and must run successfully before the main process continues.
5. If you inserted a blank process, a new workflow, or a workflow from the library, choose the respective link in the image below and design that subprocess.

Related Information

Create a Business Process
Call a Workflow
Modeling a Workflow
Create a Workflow Module
Introducing Subprocesses as Referenced Subflows in SAP Build Process Automation

Determine the Workflow Definition ID

Find the workflow definition ID of live process projects.

Procedure

1. In SAP Build, choose **Monitoring**.

2. Under **Manage**, choose **Processes and Workflows**.
3. Select the relevant workflow.

Call a Workflow

You want to add a prebuilt workflow as a subprocess.

Prerequisites

From the process editor canvas, you choose **Call a Workflow** to insert an existing workflow as a subprocess.

Procedure

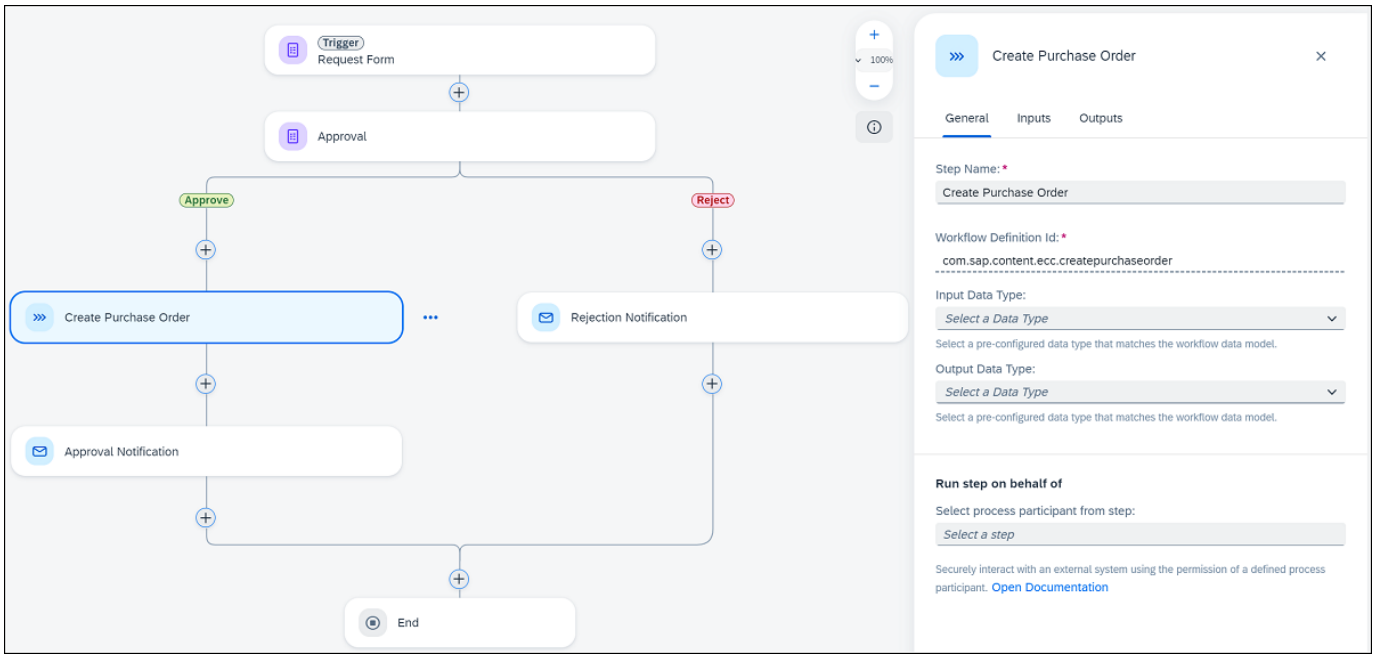
1. On the **Call Workflow** screen, select a workflow and choose **Add**.

   On the **General** tab, the workflow definition ID is filled in automatically and is read only. The workflow name is also prefilled but can be changed.

2. Select an existing **Input Data Type** or click **+ New Data Type** to create one.

   If creating a new data type, you must then configure this via your **Project Overview** area. See Create a Data Type.

3. Select an existing **Output Data Type** or click **+ New Data Type** to create one.



4. Save your changes.

Add Mail Notifications to a Process

You can add mail notifications to your business process using the process editor, allowing you to send preconfigured emails to recipients while a process is running.

Prerequisites

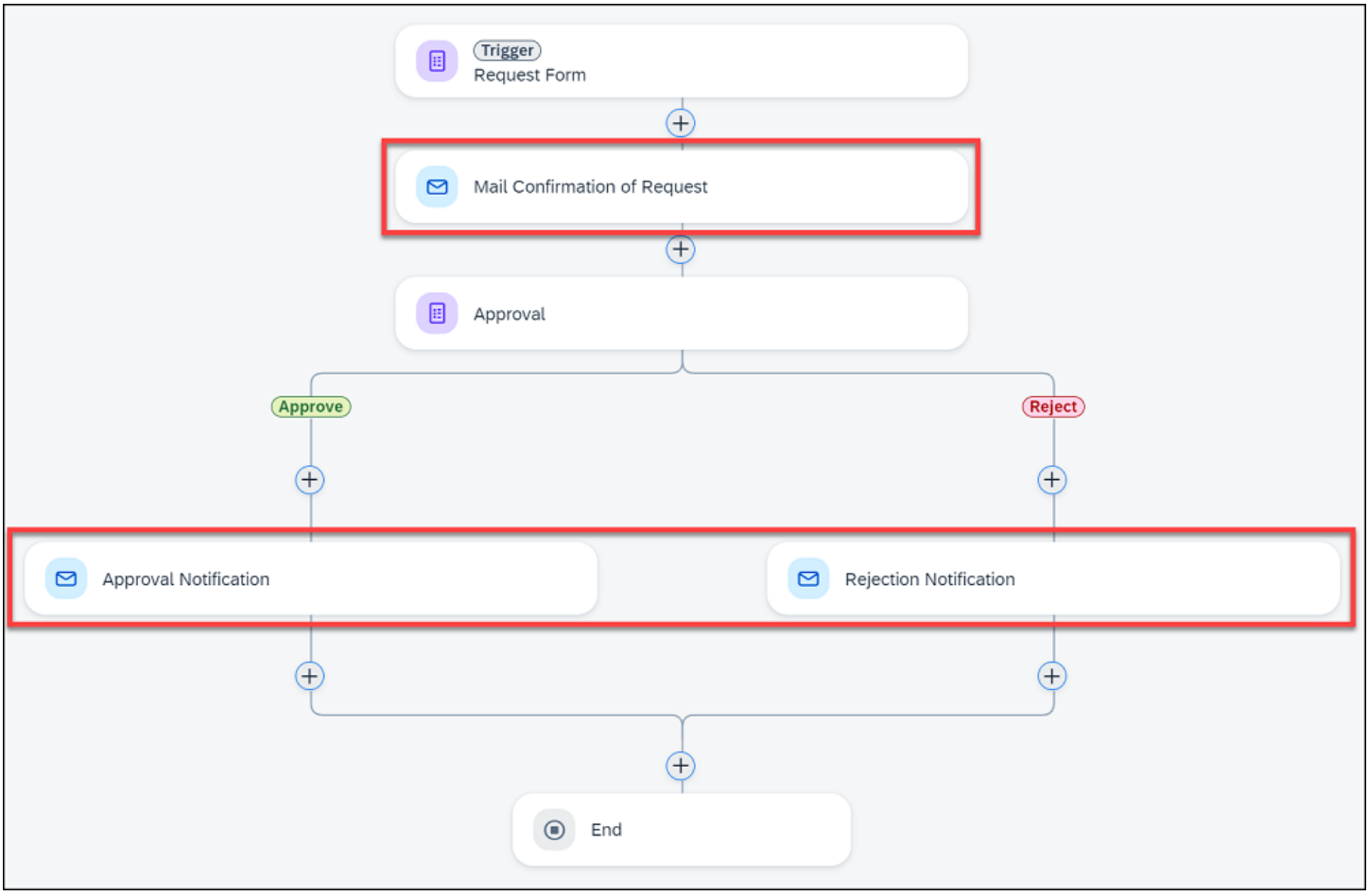
Before adding mail notifications to a process, you must first configure your SMTP mail destinations for SAP Build Process Automation. See Configure an SMTP Mail Destination.

This configuration includes assigning the 'From' address for sent mail, for example: user@example.com. This inbox must be able to receive replies.

Context

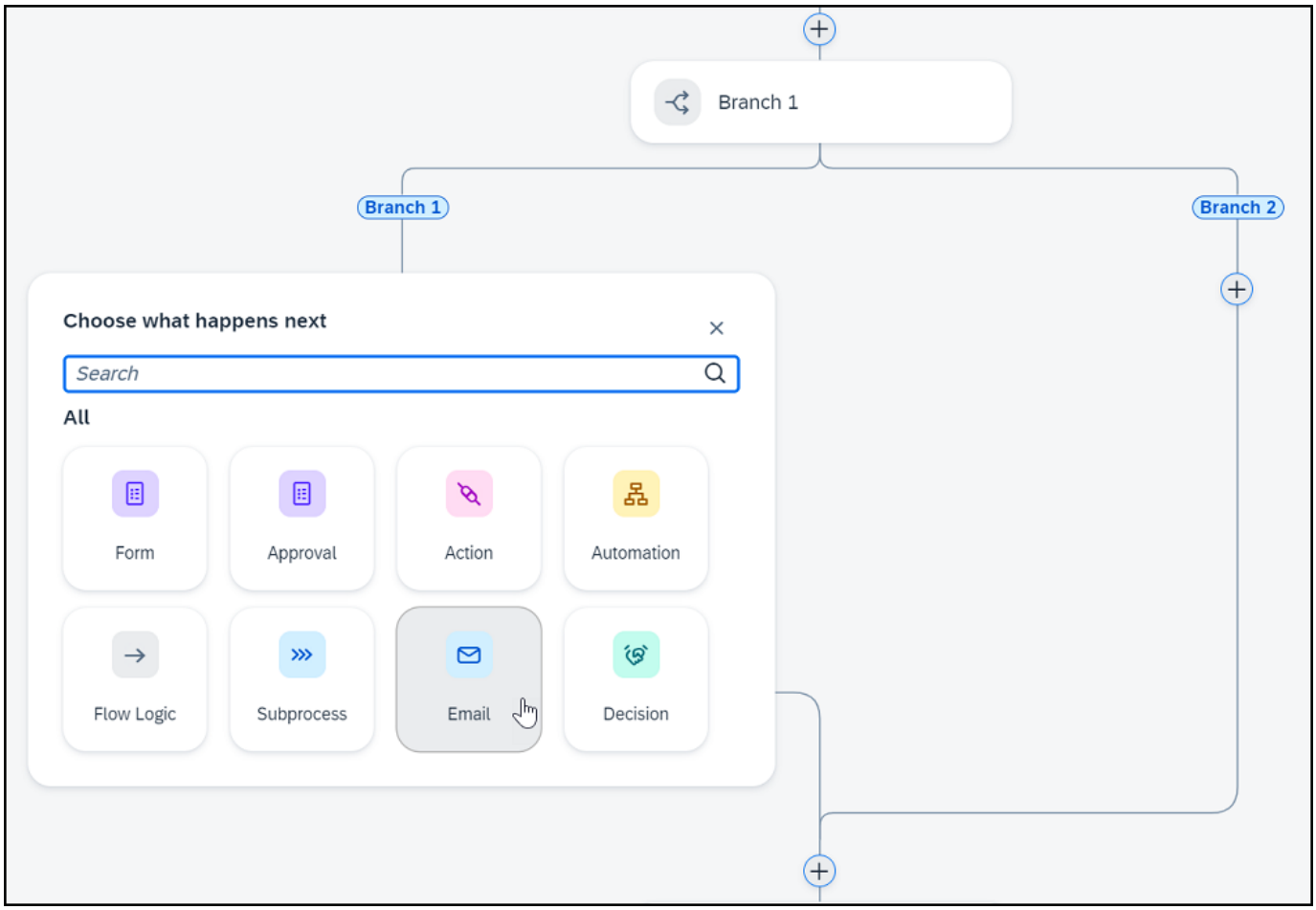
As an example, there are three mail notifications in this simple request process:

- Confirmation that the request was received
- Confirmation that the request was approved
- Confirmation that the request was rejected

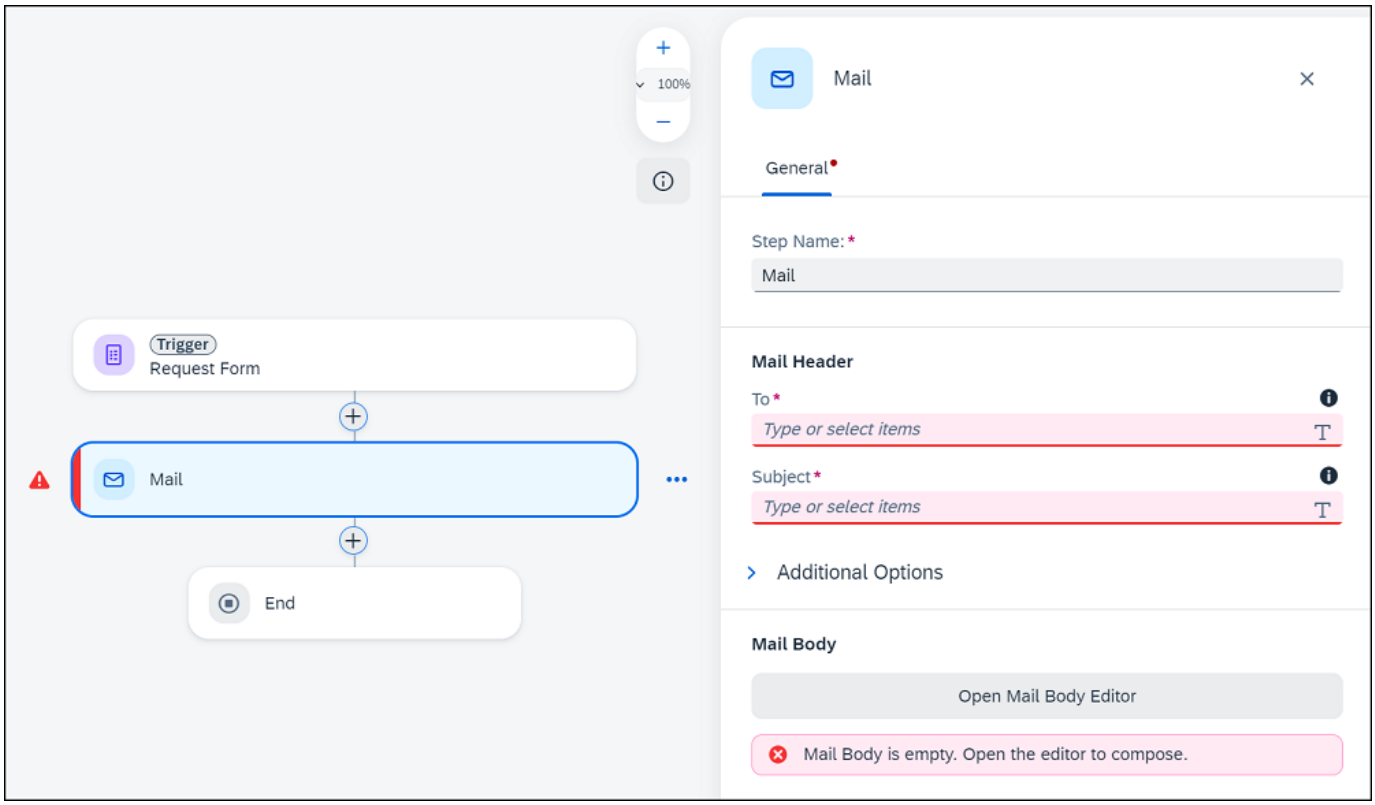


Procedure

1. Either choose + or right-click on the process editor canvas and select **Mail**.



The mail notification is added to the process and the settings are displayed in the side panel:

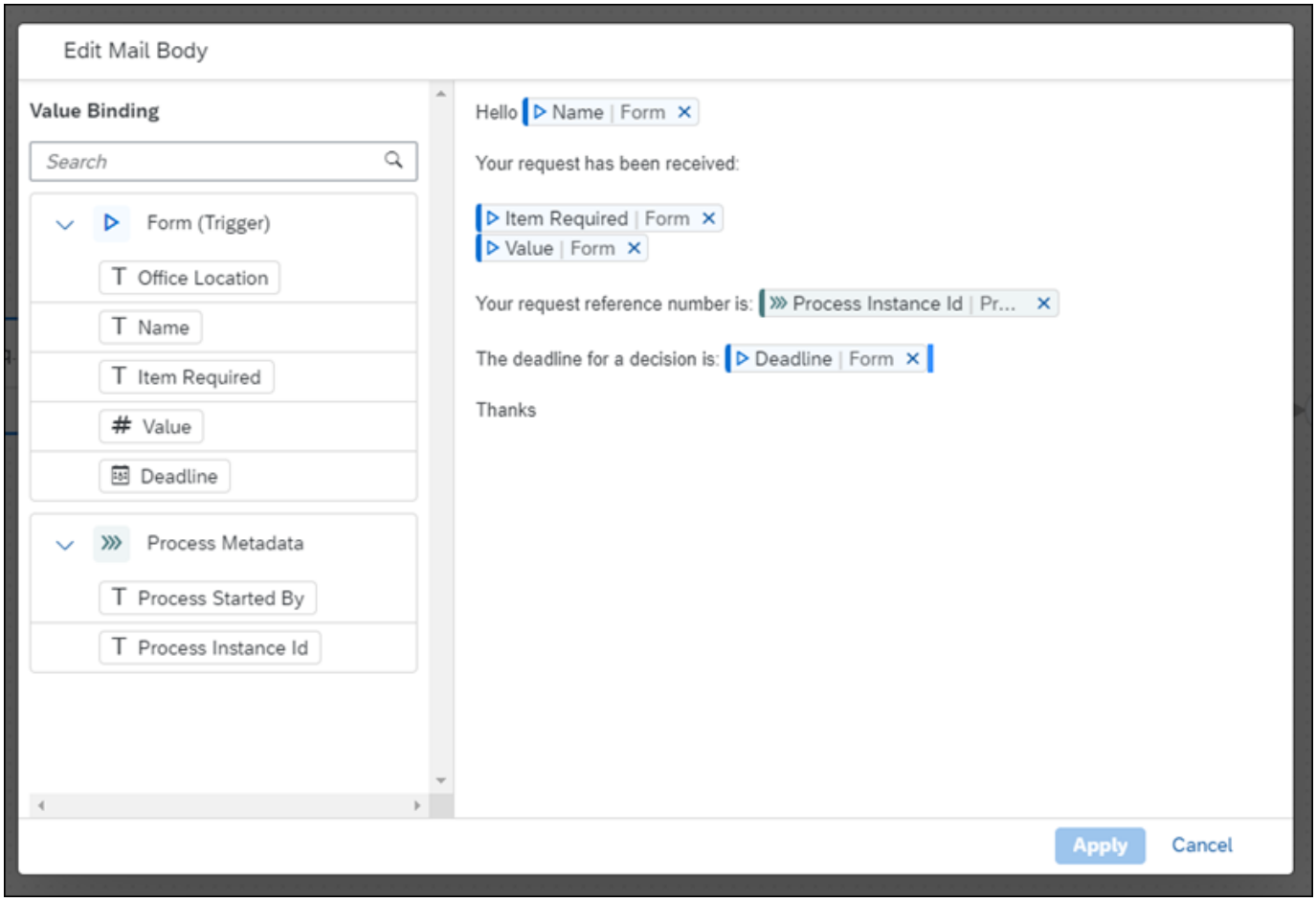


2. Configure the **Mail Header** fields.

Field	Description
To	The recipient of the mail notifications. Either add specific mail addresses or use information taken from process metadata or context. For more detailed guidelines when adding recipients, see Guidelines for Specifying Recipient Users

Field	Description
Subject	The subject of the mail itself. Either add specific text here or use information taken from process metadata or contex.
CC	The copied recipient of the mail notifications. Either add specific mail addresses or use information taken from process metadata or contex. For more detailed guidelines when adding recipients, see Guidelines for Specifying Recipient Users
BCC	The blind copied recipient of the mail notifications. Either add specific mail addresses or use information taken from process metadata or contex. For more detailed guidelines when adding recipients, see Guidelines for Specifying Recipient Users

3. Choose **Open Mail Body Editor** and configure the mail body. The mail body can include the following:
    - Text
    - Process context information (such as the Form fields in the example)
    - Process metadata (such as the 'Process Started By' information)



4. Choose **Apply**.
5. Review and fix any missing mandatory fields that are still marked in red.

Results

The mail notification is added to the process, with mails sent to recipients when a process is running.

Add Wait for Duration to a Process

You can add a wait to a process, controlling how long the live process waits (or pauses) before continuing.

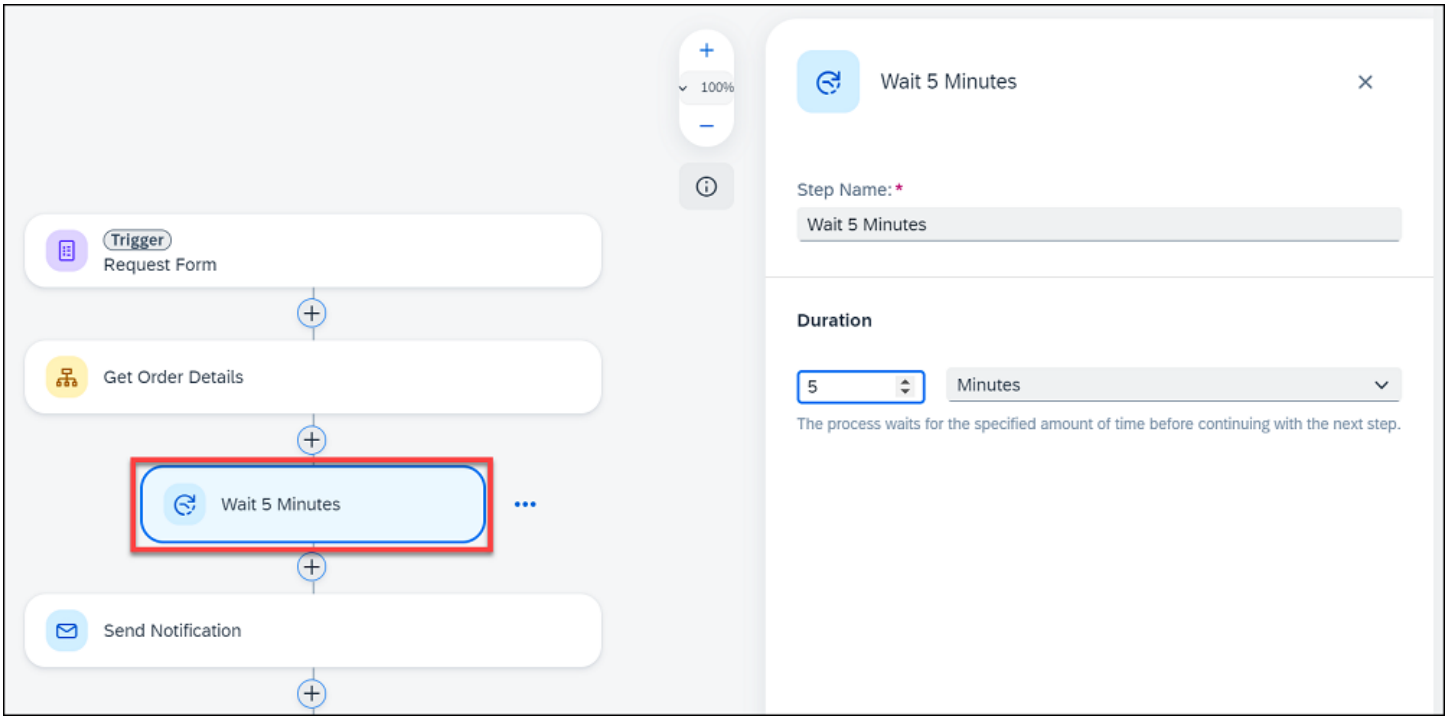
Context

Waits can be added and configured for the following intervals:

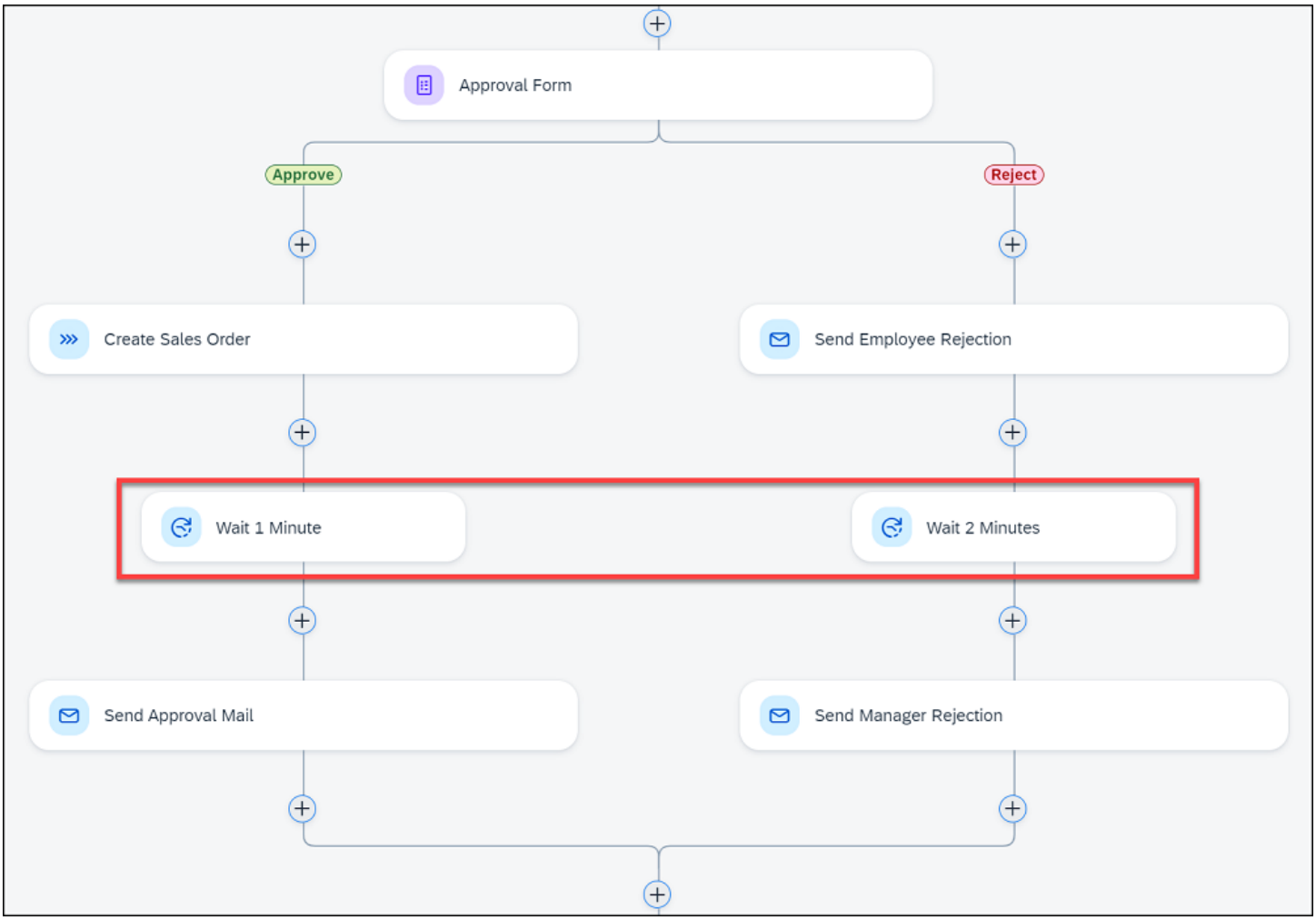
- Minutes
- Hours
- Days
- Months

These intervals then occur from the time the live process first reaches that point. As an example, if the process reaches a 1 hour wait at 09.37, the process continues at 10.37.

In the following example, the process is configured to wait for 5 minutes after an automation has run before the approval is sent:

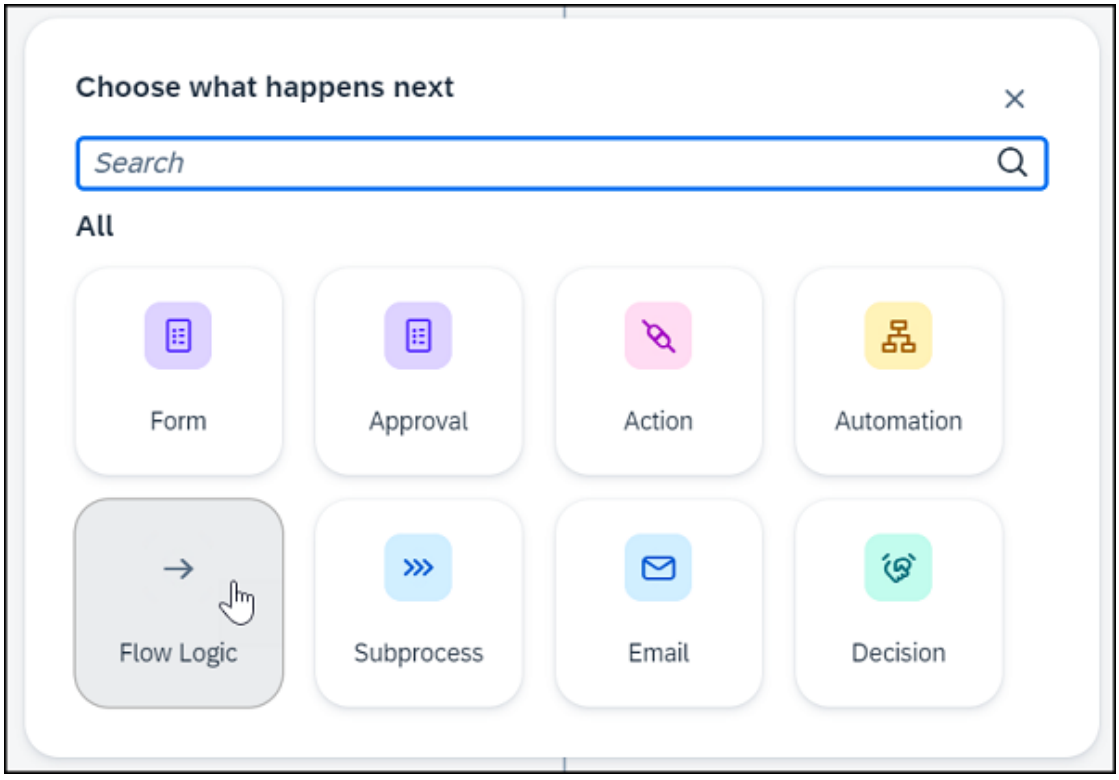


Multiple wait events can be added to the same process, too. In the following example, the process waits for 1 minute after a request has been approved, and 2 minutes after a request has been rejected:



Procedure

1. In the process editor, choose **+ (plus)** > **Flow Logic** > **Wait for Duration**.



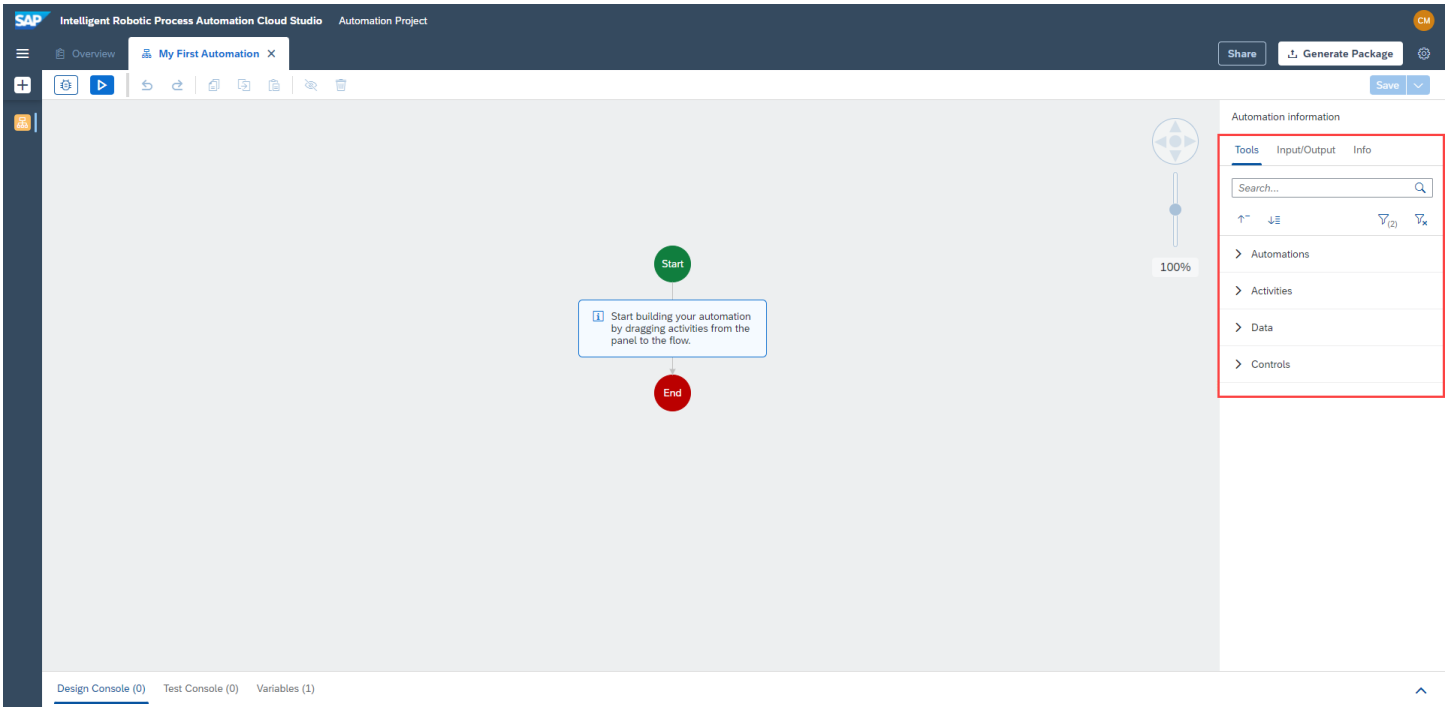
2. Enter a **Step Name**.
3. To define the duration, select a **Unit** of wait (minutes, hours, days, and months) and enter a **Value**.
4. Save your changes.

Create and Design Automations

Automations are composed of a succession of steps you build in SAP Build Process Automation. An automation can orchestrate multiple activities on different applications and screens available on a specific computer.

Automation Tools

To build an automation, you have access to a list of tools from the side panel.



- *Automations*

  You can use a previously created automation from this cloud project, as a tool to include in the flow of another automation.

- *Screens*

  An Application Screen is a capture of any application. For more information, see the section about Capture and Declare Applications.

- *Activities*

  Activities are used to build the workflow of your automation. They come from the SDK packages imported in the Cloud Studio the first time you create an automation. For more information, see the section about Automation Activities Provided by SDK Packages

- *Data Types*

  A data type is a complex data used to describe a data structure. It can be defined as an input or an output and used throughout your automation. For more information, see Data Types.

- *Controls*

  Controls allow you to add tools such as conditions, loops, and scripts to your automation. For more information, see Add a Control to an Automation.

◇	**Condition**	Inserts multiple situations defined by an expression, that determines the following step.
●	**End**	Stops the automation.
↓≡	**For Each**	Inserts a sequence of actions to perform on a list of objects such as users.

	Forever	Repeats the step in a loop until it meets the required conditions.
	Repeat	Repeats the step for a defined number of times.
	Loop End	Stops a step set in a loop.
	Screen switch	Inserts multiple situations defined by the screen the user is working on, that determine the following step.
	Custom script	Inserts a step defined by custom Javascript mode.
	Try	Checks if errors occur in a sequence of actions and defines a behavior depending on the error type.
	Stop automation in error	Stops the automation in an error state.

Create a Decision

Create decisions to add them to your process.

Prerequisites

You have created the data types required to configure the input and output of the decision. See Create a Data Type.

Context

A decision consists of a collection of rules, which are used to digitize and automate the business decisions within a business process. After you create a decision, you can define your business logic by adding rules to it. There are two types of rules:

- Decision table: A decision table is a collection of input and output rule expressions in a tabular representation. Multiple rule conditions and results can be configured using a decision table.
- Text Rule: A text rule is the collection of rule expressions in a simple if-then format.

Decision Diagram is a graphical representation of the decision logic from the input to the output. Using a decision diagram, you can view the inputs/output of a decision and the rules of the decision. You can also click on each component of the decision diagram, for example, a text rule or a decision table, to view the configured details on the right panel.

Vocabulary includes input/output parameters and decision variables. This vocabulary can be used to configure the conditions and results when modelling a rule.

Reusable Rules: Reusable rules, as the name implies, are versatile rules which can be incorporated into other rules. These rules exhibit the following features:

- Both text rules and decision tables can serve as reusable rules.
- The result returned by the reusable rule, which can only be executed within another rule, can be consumed in a rule expression.
- When configuring conditions and results in the wizard, reusable rules will be displayed under **Vocabulary**, allowing you to incorporate them into rule expressions.
- A text rule with execute operations cannot be used as reusable rule.

To designate a rule as a reusable rule while modelling a decision, simply mark the rule as **Reusable Rule** in the **Create Rule** screen of the wizard. The rule will then be listed under **Reusable Rules** in the **Rules** tab.

Procedure

1. In the **Overview** tab of your business process project, choose **Create** > **Decision**
2. Provide a **Name**, **Identifier**, and optionally a **Description**.
3. Choose **Create**.
4. To configure the inputs and output of the decision, do the following:
    a. Choose the decision from the **Decision Diagram** and then choose the **Inputs/Output** tab to configure the inputs and the output
    b. Under **Input Parameters**, choose **Add Input Parameter** and configure the inputs by providing a **Name**, **Description** and choosing the **Type** of the input parameters.

    You can either select any of the **Current Project** data types that you created earlier or one of the following **Basic** data types:

    - Boolean
    - Date
    - Number
    - String
    - Date/Time

    Select **List** if the input parameter is a collection of a specific data type.

    ### ℹ Note

    - Once you select **List**, this list input parameter will be displayed under **Vocabulary** when you configure your conditions and results in the wizard.
    - However, you will not be able to use these list parameters to configure conditions in condition headers and these will appear greyed out in the **Configure Conditions** screen. See Model a Rule in a Decision Table.
    - Once a rule is added, you can use these list parameters in the cells of the decision table while modelling rule expressions.

    ### ✤ Example

    - If you have the following input in a tabular form with the IDs of employees and their cities as list input parameters, you can use 'BLR' EXISTSIN (Employees, CityCode) to determine whether it contains the input 'BLR'. When you configure your inputs, you can use 'Employees' and select the **List** checkbox.

    Employees

Employee_ID	CityCode	Age
L321	BLR	30
L322	MUM	40

    - If 'Age' is a list input parameter, you can use AVG(Employees, Age) to get the average of the age of all the employees.

    c. Similarly, under **Output Parameter**, choose **Add Output Parameter** and configure the output for the decision by providing a **Name**, **Description** and choosing the **Type** of the output parameter.

    Type can be either any of the **Current Project** data types that you created earlier or one of the **Basic** data types.

    ### → Tip

Select **List** if you want to model the decision table with **All Match** hit policy. This allows you to fetch all instances that meet the specified conditions and return them as an array of multiple values in the result.

ℹ Note

- If you want to change this decision table with **First Match** hit policy later, deselect **List**. Also, edit the rule to confirm that the hit policy is displayed as **First Match**. The rule engine then fetches the first occurrence that matches the condition and returns it as result.
- When you select a basic data type for your output parameter in the **Type** dropdown, you will be unable to select the **List** checkbox.

Add Rules to a Decision

Add decision tables or text rules to a decision based on your business scenario.

Prerequisites

You're familiar with the operators and functions used to model rule expressions. See

- Operators
- Functions

Procedure

1. Choose the **Rules** tab and then choose **Add Rule**.
2. To add a decision table, see Model a Rule in a Decision Table.
3. To add a text rule, see Model a Text Rule .
4. To create a duplicate of a rule, see Duplicate a Rule

ℹ Note

Once you add multiple rules, they will be executed in the sequence listed in the **Rules** tab under **Execution Order**. You can modify this order by moving the rules up or down.

Also, you can perform the following actions in the **Rules** tab:

- Delete any rule by selecting the rule and choosing **Delete** ()
- Drag and drop a rule to change the order.
- Move a rule up or down to change the order by choosing **Move Up** () and **Move Down** ()
- Use the **More** () option beside each rule, to do the following:
  - Add a rule
  - Duplicate a rule. See Duplicate a Rule.
  - Delete a rule
  - Move a rule up
  - Move a rule down

Add Variables to a Decision(Optional)

Add variables to a decision to store information of one rule execution and reuse it in another rule.

Context

In the Variables tab, you can add intermediate variables to your decision, that can be used to temporarily store the output of one rule execution and reused as an input for another rule execution.

ℹ Note

The scope of these variables are only within the particular decision to which these are being added and are not available elsewhere in the process.

Procedure

1. Choose the Variables tab and then choose Add Variable. The Decision Variable screen opens up.
2. Provide a Name, Description and choose a data type for the variable from the Type dropdown.

   In the Type dropdown, you can either select any of the Current Project data types that you created earlier or one of the following Basic data types:

   - Boolean
   - Date
   - Number
   - String
   - Date/Time
3. Choose Save. The variable gets added successfully and you can see it in the Variables tab.

   You can also edit, duplicate or delete a variable by using the More (⋮) option available beside each variable.
4. You can use these decision variables when you configure your conditions or results while modelling a rule. These variables will be displayed in the rule wizard as part of the vocabulary for a rule condition as well as a result.

Model a Text Rule

A text rule is the collection of rule expressions in a simple if-then format. You can model a text rule by storing the return value in a result data type.

Context

You can model a text rule in two different ways:

- **Text rule with a result vocabulary**: A text rule that stores the return value in a result vocabulary.
- **Text rule for execute operations**: When you do not assign a result vocabulary to a text rule, you can perform execute operations on existing input/output parameters or decision variables. You can update or append values to these. You can also add multiple execute operations in a single text rule.

  For example, you can use FOREACHITEM to perform loop operation on List type parameters.

  ### ℹ Note

  When referring to execute operations, we support the following functions as part it:

  - APPEND
  - UPDATE
  - FOREACHITEM

See also:

- Model a Text Rule with Result Vocabulary
- Model a Text Rule for Execute Operations

Model a Text Rule with Result Vocabulary

A text rule with result vocabulary is a text rule that stores the return value in a result vocabulary.

Procedure

1. To open the **Create Rule** wizard, go to the **Rules** tab and choose **Add Rule**.
2. Under **Rule Details**, choose the **Rule Type** as **Text Rule**, provide the **Rule Name** and **Rule Description**.

   > **Note**
   >
   > If you want to add this rule as a reusable rule, mark it as **Reusable Rule (Optional)** using the switch. Reusable rules, as the name implies, are versatile rules which can be incorporated into other rules. For more information about resuable rules, see Create a Decision.
   >
   > If required, you can again convert this into an orchestrated rule by clicking on ('**More**') and choosing **Edit**.

3. Choose **Next Step**.

   The **Configure Results** screen opens.

4. Choose a **Vocabulary** or its fields. You can also change the **Result Vocabulary** in the dropdown.

   Alternatively, you can also search for the vocabulary or its fields using the search box.

   If you have created an input or output parameter or a decision variable that refers to a data type with fields of another data type, it is displayed as a nested attribute under **Vocabulary** .

   > **Note**
   >
   > You can delete one or more **Result Attributes** by selecting the attributes.
   >
   > You can also choose or to move the entries up or down.

5. Choose **Next Step**.

   In the **Review** screen, review your text rule details and choose **Create**.

   You have created the rule and can now configure it.

6. Provide the rule condition in the **If** field of your text rule.

   > **Note**
   >
   > For more information about the syntax of operators and functions, see:
   >
   > - Operators
   > - Functions

7. Provide the values or expressions for the result vocabulary in the **Then** field.

   If you have defined vocabulary that include fields with the **Enumeration** type, and have specified a set of values for those fields, you can incorporate these values by using (value help) .

8. **Optional:** To add multiple if-then conditions, choose **Add Else If**.
9. **Optional:** If the previous if-then conditions are false, choose **Add Else** to provide the values or expressions for the result data object.

10. Choose **Save**.

> **Note**
>
> Check the **Design Console** for errors after saving the decision.

Model a Text Rule for Execute Operations

Procedure

1. Choose **Add Rule**.

   The **Create Rule** wizard opens.

2. Under **Rule Details**, choose the **Rule Type** as **Text Rule**, provide the **Rule Name** and **Rule Description**.

3. Choose **Next Step**.

   The **Configure Results** screen opens.

4. Choose **No Result (use execute operations)** from the **Result Vocabulary** dropdown and then choose **Next Step**.

5. In the **Review** screen, choose **Create**.

   You have successfully created the rule. You can now configure the rule.

6. In the **If** field, provide the rule condition.

7. Press CTRL + SPACE in the **Then** field of your text rule to view the list of operations and loop functions that can be performed on any vocabulary of the project.

8. Choose the **Append** or **Update** operation or loop function to be performed on the vocabulary, and then provide the required input for the operation or function according to the syntax.

   If you want to configure loop functions on list type parameters, do the following:

   a. Choose **Loop Functions**.

   b. In the **Configure Loop Functions** screen, choose `FOREACHITEM` from the **Function** dropdown.

   c. From the **Vocabulary** dropdown, choose an entry of type **Table** or a rule that returns a vocabulary of type **Table**.

   d. From the **Current Item** dropdown, select the current row of the list for which you want to use the loop function.

   > **Note**
   >
   > To select the **Current Item**, ensure that you have created a decision variable that matches the **Table** type specified in the **Vocabulary**.

   e. (Optional) In the **Where** field, provide a filter condition for retrieving the values from the **Vocabulary**.

   f. Press CTRL + SPACE in the textbox to configure the execute operations to be executed in a loop and choose **Apply**.

   You can view the loop function syntax in the **Function Label** field.

9. (Optional) Choose **Add Else If** to add multiple if-then conditions.

Model a Rule in a Decision Table

Model the rule conditions in a tabular format as a decision table.

Context

A decision table is a collection of input and output rule expressions in a tabular representation. Multiple rule conditions and results can be configured using a decision table.

In some use cases, you may need to match multiple conditions in a decision table. You can specify how the rule engine should fetch the results by changing the hit policy. There are two types of hit policies:

- **First match**: The rule engine fetches the first occurrence that matches the condition and returns it as the result.
- **All match**: The rule engine fetches all the occurrences that matches the condition and returns them as the result.

Procedure

1. Choose **Add Rule**.

   The **Create Rule** wizard opens.

2. Under **Rule Details**, choose **Decision Table** as the **Rule Type**, and provide a **Rule Name** and **Rule Description**.

   Select the **Hit Policy** from the dropdown.

   > **Note**
   >
   > If you want to add this rule as a Reusable Rule, simply mark it as **Reusable Rule (Optional)** using the switch. If required, you can again convert this into an orchestrated rule by clicking on 'More' (⋮) and choosing **Edit**. For more information on Resuable Rules, see Create a Decision.
   >
   > .

3. Choose **Next Step**.

   The **Configure Conditions** screen opens.

4. Choose a vocabulary or its fields to configure the conditions. Alternatively, you can directly write expressions in the text box for **Condition** to use these as condition headers. Press CTRL + SPACE for autosuggestions.

   The **Label** is automatically filled in based on the chosen vocabulary, but you can modify it if needed.

   Select the operators that you need from the **Operator (Optional)** dropdown.

   If you have created an input/output parameter or a decision variable referring to a data type which has fields of another data type, this appears under **Vocabulary** as a nested attribute when you configure the conditions.

   > **Note**
   >
   > Under **Condition Details**, you can delete one or more entries, if necessary, and also move the conditions up or down using the arrows

5. Choose **Next Step**.

   The **Configure Results** screen opens.

6. Under **Vocabulary**, choose a vocabulary or its fields to configure the results. You can also change the result vocabulary in the **Result Vocabulary** dropdown. Alternatively, you can also search for the vocabulary or its fields using the search box.

   If you have created an input/output parameter or a decision variable referring to a data type which has fields of another data type, this appears under **Vocabulary** as a nested attribute when you configure the results.

   The **Label** is automatically filled in based on the chosen vocabulary, but you can modify it if needed.

   > **Note**
   >
   > Under **Result Details**, you can delete one or more **Result Attributes** by selecting them, and also move the entries up or down using the arrows.

7. Choose **Next Step**.

   The **Review** screen opens and you can review your Decision Table details.

8. Choose **Create**.

   You have successfully created the rule.

9. Provide the fixed values, rule conditions, or expressions in the columns of the decision table row. To add more conditions, choose Add Row > Insert First.

   Note

   - To add a row below a row, select the row and then choose Add Row > Insert After. By default, the row is added to the top of the table.
   - If you do not have input values for a given cell in a row, leave the cell empty.

     Example

     If you do not have any values for Equipment Type in the first row, you can leave the cell empty as shown:

If				Then		
countryofCompany	company of the Employee	jobTitle of the Employee	Employee is isFulltimeEmployee	Currency	Equipment Type	Equipment Price
is equal to 'USA'	is like is 'SAP'	is like is '%'	is equal to true	'EUR'		35.96
is equal to 'DEU'	is like is 'SAP'	is like is '%'	is equal to true	'EUR'	'Notebook'	76.98

   If you have defined vocabulary that include fields with the 'Enumeration' type, and have specified a set of values for those fields, you can incorporate these values by using the value help icon .

10. To delete a condition from the decision table, select the row, then choose Delete Row.
11. To edit rows in the decision table, select the required row, and choose the required option from one of the following:
    - Copy Row
    - Cut Row
    - Paste Row
12. Choose Save.

    Note

    Check the Design Console for errors after saving the decision.

13. To import or export a decision table as a spreadsheet, see Import or Export a Decision Table.

Import or Export a Decision Table

You can import or export a decision table rule as an MS Excel spreadsheet. You can export the content of the decision table, work on it offline and then import it again.

Procedure

To export a decision table to your local folder, do the following:

1. From the Rules tab, click on the rule and then choose Export.

   The decision table is downloaded in the form of an MS Excel spreadsheet.

To import a decision table from your local folder, do the following:

2. From the Rules tab, click on the rule and then choose Import.

   The Import Decision Table pop-up screen opens.

3. Browse for the Excel spreadsheet and choose Open. Then choose Import on the pop-up screen.

You get a message confirming that the decision table has been successfully imported.

4. Choose **Save**.

Edit a Rule

You can edit an existing rule using the **Edit Rule** wizard.

Prerequisites

You must first create a rule.

Procedure

1. To open the **Edit Rule** wizard, choose your rule from the decision diagram or from the **Rules** tab and then choose *(Re-Configure the Rule)* .
2. Edit the rule according to your business requirements.

   Note

   You cannot change the **Type of Rule** from a decision table to a text rule and vice versa once created.

3. Review the rule and then choose **Finish**.

Duplicate a Rule

Instead of creating a new rule, you can create a duplicate of an existing rule and add your changes. You can use these duplicate copies of rules in scenarios that require multiple rules with minor changes in business logic.

Context

To create a duplicate of a rule, do the following:

Procedure

1. In the **Rules** tab, select the rule for which you want to create a duplicate copy and choose **More (...)** > **Duplicate**.
2. Provide the **Name** and **Description**. The rule to be copied from is autopopulated in the **Duplicate From** dropdown list.
3. Choose **OK**.

   Note

   The rules will be executed from top to bottom in the same order as it appears on the screen under the **Rules** tab. See Create a Decision

Operators

Use the following operators to model the rule expressions or conditions.

The operators are listed in the autosuggest list only if they are relevant for the rule expression.

Mathematical Operators

Mathematical operators let you perform binary operations.

Operator Name	Operator Syntax	Example
dd	+	product.amount + product.taxvalue
subtract	-	product.total - product.discount
multiply	*	product.amount * product.taxrate
divide	/	product.totalamount / COUNT(products)

Logical Operators

Logical operators let you perform logical operations. These operations return a boolean value.

Operator Name	Operator Syntax	Examples
and	AND	customer.name = 'John' AND customer.country ='INDIA'
or	OR	customer.country = 'US' OR customer.country ='INDIA'
not	NOT	NOT('John')

Comparison Operators

Comparison operators compare fixed values, data objects, or rule expressions and return **true** or **false**.

Operator Name	Operator Syntax	Examples
is equal to	=	order.productD = product.ID
is not equal to	!=	order.productD != product.ID
is greater than	>	wallet.balance > order.total
is equal to or greater than	>=	wallet.balance >= order.total
is less than	<	wallet.balance < order.total
is equal to or less than	<=	wallet.balance <= order.total

Range Operators

Range operators check for a particular value in a range of continuous values and return **true** if it is within the range. If not, they return **false**. You can choose to include or exclude the upper and lower values of the range.

Operator Name	Operator Syntax	Examples
in	IN	customer.score IN (1..10)
not in	NOTIN	customer.score NOTIN (1..10]

You can use round brackets - '(' or ')' - to exclude the limit values and square brackets - '[' or ']' - to include the limit values.

Example

Example	Description
customer.score IN [1..10]	customer.score can be any number between 1 and 10, including 1 and 10.
customer.score IN [1..10)	customer.score can be any number between 1 and 10, excluding 10.

Example	Description
customer.score IN (1..10]	customer.score can be any number between 1 and 10, excluding 1.
customer.score IN (1..10)	customer.score can be any number between 1 and 10, excluding 1 and 10.

Array Operators

Array operators check for a particular value in an array of values and return **true** if the value is present in the array. If not, they return **false**.

Operator Name	Operator Syntax	Examples
exists in	EXISTSIN	customer.country EXISTSIN ['IND','GER','AUS']
does not exist in	NOTEXISTSIN	customer.country NOTEXISTSIN ['IND','GER','AUS']

Functional Operators

Functional operators perform operations on strings or expressions that return a string value.

Operator Name	Operator Syntax	Examples
matches	MATCHES	customer.name MATCHES 'Jo.*' customer.name MATCHES '*Dr\.*|*Mr\.*' **Note** Dot star (.*) represents any character.
does not match	NOTMATCHES	customer.name NOTMATCHES 'Jo.' **Note** Period (.) represents a character.
contains string	CONTAINS	customer.name CONTAINS 'Dr.'
does not contain string	NOTCONTAINS	customer.name NOTCONTAINS 'Dr.'
starts with	STARTSWITH	customer.name STARTSWITH 'J'
does not start with	NOTSTARTSWITH	customer.name NOTSTARTSWITH 'J''
ends with	ENDSWITH	customer.name ENDSWITH 'J'
does not end with	NOTENDSWITH	customer.name NOTENDSWITH 'J'

Functions

Use the following functions to model the rule expressions or conditions.

Functions are listed in the autosuggest list only if they are relevant for the rule expression.

Time and Duration Functions

Time and duration functions let you perform operations on a date, time, and timestamp values. You can select the date and timestamp values from the date and time picker in the **Fixed Value** section. You can also enter the date or timestamp in the **Fixed Value** field using the

syntax given in this section.

Function Name	Function Syntax	Example
is in next	ISINNEXT(<test value>, <quantity value>)	ISINNEXT('2019-01-10','2 WK')
is not in next	ISNOTINNEXT(<test value>, <quantity value>)	ISNOTINNEXT('2019-01-10T22:00:00Z','2 HR')
is in last	ISINLAST(<test value>, <quantity value>)	ISINLAST(process.start time, '20 MIN')
Today	TODAY()	Order.Shipdate = TODAY()
Tomorrow	TOMORROW()	Order.Readydate = TOMORROW()
Yesterday	YESTERDAY()	Order.Readydate = YESTERDAY()
is not in last	ISNOTINLAST(<test value>, <quantity value>)	ISNOTINLAST(employee.birth year, '18 ANN')
add seconds	ADDSECONDS(<test value>, <quantity value>)	ADDSECONDS(Employee.LoginTime,15) ADDSECONDS('2019-01-10T22:00:00Z', 20)
add minutes	ADDMINUTES(<test value>, <quantity value>)	ADDMINUTES(Order.Time, 10)
add hours	ADDHOURS(<test value>, <quantity value>)	ADDHOURS(Employee.LoginTime, 2)
add days	ADDDAYS(<test value>, <quantity value>)	ADDDAYS(License.EndDate, 30) ADDDAYS('2019-01-30', 30)
add weeks	ADDWEEKS(<test value>, <quantity value>)	ADDWEEKS('2019-01-20', 14)
add months	ADDMONTHS(<test value>, <quantity value>)	ADDMONTHS(License.ExpiryDate, 4) ADDMONTHS('2019-01-30',1)
add quarters	ADDQUARTERS(<test value>, <quantity value>)	ADDQUARTERS('2019-01-30', 3) This function return a date three quarters after 2019-01-30, that is, 2019-10-30
add years	ADDYEARS(<test value>, <quantity value>)	ADDYEARS(Employee.YearOfBirth,2) This function returns a date, two years after the input date. ADDYEARS('2019-01-30T18:00:00Z', 4) This function returns a timestamp value of '2023-01-30T18:00:00Z'.
subtract seconds	SUBTRACTSECONDS(<test value>, <quantity value>)	SUBTRACTSECONDS(Employee.LoginTime, 10) SUBTRACTSECONDS('2019-01-20T22:00:00Z', 90)
subtract minutes	SUBTRACTMINUTES(<test value>, <quantity value>)	SUBTRACTMINUTES('2019-01-30T22:00:00Z', 10) SUBTRACTMINUTES(Order.ShipTime, 20)
subtract hours	SUBTRACTHOURS(<test value>, <quantity value>)	SUBTRACTHOURS(Plant.ShutDownTime,4)
subtract days	SUBTRACTDAYS(<test value>, <quantity value>)	SUBTRACTDAYS(Plant.OpenDate, 10)
subtract weeks	SUBTRACTWEEKS(<test value>, <quantity value>)	SUBTRACTWEEKS('2020-10-10',7)
subtract months	SUBTRACTMONTHS(<test value>, <quantity value>)	SUBTRACTMONTHS(Customer.SignUpDate,1)

Function Name	Function Syntax	Example
subtract quarters	SUBTRACTQUARTERS(<test value>, <quantity value>)	SUBTRACTQUARTERS('2020-10-10', 2) This function returns a date, 2020-04-10 which is 2 quarters before 2020-10-10.
subtract years	SUBTRACTYEARS(<test value>, <quantity value>)	SUBTRACTYEARS(Vehicle.MaufacturingDate, 2) This function returns a date which is 2 years before the Vehicle.MaufacturingDate
seconds between	SECONDSBETWEEN(<test value>, <quantity value>)	SECONDSBETWEEN(Employee.LogoutTime, Employee.LoginTime) SECONDSBETWEEN('2019-01-10T22:00:00Z','10-01-2019T12:40:00Z') SECONDSBETWEEN('2019-01-10','2019-01-08')
minutes between	MINUTESBETWEEN(<test value>, <quantity value>)	MINUTESBETWEEN(Employee.LogoutTime, Employee.LoginTime) MINUTESBETWEEN('2019-01-10T22:00:00Z','10-01-2019T12:40:00Z') MINUTESBETWEEN('2019-01-10','2019-01-08')
hours between	HOURSBETWEEN(<testValue>, <quantityValue>)	HOURSBETWEEN(Employee.LogoutTime, Employee.LoginTime) HOURSBETWEEN('2019-01-10T22:00:00Z','2019-01-09T12:40:00Z') HOURSBETWEEN('2019-01-10','2019-01-08')
days between	DAYSBETWEEN(<test value>, <quantity value>)	DAYSBETWEEN(TODAY(), Emplyee.JoiningDate)
weeks between	WEEKSBETWEEN(<testValue>, <quantity value>)	WEEKSBETWEEN(TODAY(), Employee.JoiningDate)
months between	MONTHSBETWEEN(<test value>, <quantity value>)	MONTHSBETWEEN(TODAY(),Employee.DateofBirth) MONTHSBETWEEN('2019-01-10','2018-01-01')
quarters between	QUARTERSBETWEEN(<test value>, <quantity value>)	QUARTERSBETWEEN(TODAY(),Employee.DateofBirth
years between	YEARSBETWEEN(<test value>, <quantity value>)	YEARSBETWEEN(TODAY(),Employee.DateofBirth)
Second	SECOND(<test value>)	SECOND('2019-01-10T12:11:22.333Z') The function returns the seconds part of the timestamp, that is, 22.333
Minute	MINUTE(<test value>)	MINUTE('2019-01-10T12:11:22.333Z') The function returns the minutes part of the timestamp, that is, 11.
Hour	HOUR(<test value>)	HOUR('2019-01-10T12:11:22.333Z') The function returns the hour part of the timestamp, that is, 12.

Function Name	Function Syntax	Example
Month	MONTH(<test value>)	MONTH('2019-01-10T22:11:22.333Z') The function returns the month part of the timestamp, that is, 01. MONTH('2019-10-15') The function returns the month part of the date, that is, 10.
Year	YEAR(<test value>)	YEAR('2019-01-10T22:11:22.333Z') The function returns the year part of the timestamp, that is, 2019. YEAR('2019-10-15') The function returns the year part of the date, that is, 2019.
Day of week	DAYOFWEEK(<test value>) Returns the day of the week with Monday having a value of 1. Value ranges between 1 and 7.	DAYOFWEEK('2019-01-10T22:11:22.333Z') The function returns a value of 4.
Day of month	DAYOFMONTH(<test value>) Returns the day of the month. Value ranges between 1 and 31.	DAYOFMONTH('2019-01-10T22:11:22.333Z') The function returns a value of 10.
Day of year	DAYOFYEAR(<test value>) Returns the day of the year. Value ranges between 1 and 366	DAYOFYEAR('2019-01-10T22:11:22.333Z') The function returns a value of 10.

Select Functions

Select functions let you retrieve select values from a table. You can configure the select functions, using the autosuggest list, in the **Select Functions** configuration window.

Function Name	Function Syntax	Example
top	TOP(table, number)	TOP (customer,5)
select	SELECT(table, col1...coln)	SELECT(customer, Rating, Region)
first	FIRST(table)	FIRST(customer)

Aggregate Functions

Aggregate functions let you perform calculations on a set of values and return a single result value or a set of result values.

Function Name	Function Syntax	Example
average	AVERAGE(table, column, indexcolumn1..indexcolumnN)	AVG (customer, orderAmount)
sum	SUM(table, column, indexcolumn1..indexcolumnN	SUM (customer, OrderAmount)
count	COUNT(table, indexcolumn1..indexcolumnN)	COUNT(customers)
count distinct	COUNTDISTINCT(table, column)	COUNTDISTINCT(customers, region)
distinct	DISTINCT(table, column)	DISTINCT(customers)

Function Name	Function Syntax	Example
minimum	MIN(table, column, indexcolumn1..indexcolumnN)	MIN(customer, orderAmount)
maximum	MAX(table, column, indexcolumn1..indexcolumnN)	MAX(customer, orderAmount)

Advanced Functions

Advanced functions can be used to perform advanced operations on string, mathematical, amount and geographical values.

Function Name	Function Syntax	Examples
concatenate	CONCAT(<string1>,<string2>,<string3>..... <stringN>)	CONCAT('Firstname','LastName') CONCAT(Contact_S.ContactName , 'abc', 'def')
round	ROUND(<number>,<value>)	ROUND(customer.credit_rating,2)
power	POWER(<number>,<power>)	POWER(2,3)
sin	SIN(<number>)	SIN(2)
cos	COS(<number>)	COS(2)
is within	ISWITHIN(<coordinates of the point>, <coordinates of the polygon>) **Note** Coordinates are in GeoJSON format.	ISWITHIN(customer.location , '{"type": "Polygon","coordinates": [[100.0, 0.0],[101.0, 0.0],[101.0, 1.0],[100.0, 1.0],[100.0, 0.0]]}')
is not within	ISNOTWITHIN(<coordinates of the point>, <coordinates of the polygon>) **Note** Coordinates are in GeoJSON format.	ISNOTWITHIN(customer.location , '{"type": "Polygon","coordinates": [[100.0, 0.0],[101.0, 0.0],[101.0, 1.0],[100.0, 1.0],[100.0, 0.0]]}')
is null	ISNULL(<Data object of type Element>)	ISNULL(Customer.ContactName)
is not null	ISNOTNULL(<Data object of type Element>)	ISNOTNULL(Customer.ContactName)
is initial	ISINITIAL(<Data object of type Element>)	ISINITIAL(customer.name)
is not initial	ISNOTINITIAL(<Data object of type Element>)	ISNOTINITIAL(customer.name)

Select and Aggregate Functions in Decision Tables in Spreadsheet

If you are modeling a decision table in Microsoft Excel, enter Select and Aggregate functions using the following syntax:

Function Name	Function Syntax	Example
sort ascending	SORTASC(table,column)	Average(Filter(SortAsc(Customer, Name), Age > 30, player_height)
sort descending	SORTDESC(table,column))	
filter	filter(table, condition)	

Filter is the same as the Where field in aggregate and select function configuration windows.

Functions for Text Rule Execute Operations

The following execute operations and loop functions can be used while modeling a text rule without a result data object..

Execute Operations

Function	Syntax	Input	Description	Example
Update	UPDATE(<Target>, <Source>)	Target Entity	The target data object or attribute of type Structure or Element, that should be updated as per the value of the Source.	UPDATE(Customer.Employee Name, Employee.Employee Name)
		Source Entity or Source Expression.	The value of the Target Entity is updated to the value of the Source Entity or the value returned by the Source Expression. The Target and Source should be of the same data object type. **Note** You can also use resuable rules as the source expression by selecting them from the autosuggest list.	The data object, Customer.Employee Name is updated as per the value of Employee.Employee Name.
Append	APPEND(<Target>, <Source>)	Target Entity	The target data object of type Table to which the Source has to be appended.	APPEND(FlightTable, Flight) The data object Flight is appended to the data object FlightTable.
		Source Entity or Source Expression	A data object that should be appended to the Target data object. The source data object should be of type Structure or Table or a rule that returns a data object of type Structure or Table.	

Loop Functions

Loop Function	Syntax	Input	Description	Example
For each item	FOREACHITEM(<Vocabulary>, < currentitem>)	Vocabulary which is a data object of type Table or a rule that returns a data object of type Table. The current item is a decision variable pointing to the same type as the table	You can also provide additional filter conditions for performing the execute operations	FOREACHITEM(FlightTable, FlightItem)
		Single or multiple execute operations to be executed in a loop		

Create a Visibility Scenario

Visibility scenario allows you to track the performance of end-to-end processes by providing visibility and intelligence to processes within the business process project.

Prerequisites

You have created a business process project. For more information on how to create a business process project, see Business Process Projects.

Context

Visibility scenario allows you to track the performance of end-to-end processes by providing visibility and intelligence to processes within the business process project.

Example

You can create a visibility scenario for a capital expenditure request approval process to ensure that the requests are approved on time. You can further enhance the visibility scenario by configuring various aspects such as phases, correlation conditions, state, status, and actions to know when a request is in a critical state and to take corrective actions to resolve the status. You can also configure process performance indicators like value of capital expenditure requests where approvals are overdue, approved capital expenditure requests by business unit, etc. When you activate the visibility scenario, you can use the Process Workspace to track the approval requests, take corrective actions on the critical requests and understand the performance of the process using the configured process performance indicators.

Procedure

1. In the **Overview** tab of your business process project, choose **Create** > **Visibility Scenario**.
2. Provide a **Name**, **Identifier**, and optionally a **Description**.
3. Choose **Create**.

   This action creates and opens the visibility scenario. The visibility scenario is automatically locked to prevent other users and applications from modifying it.
4. In the **General** tab, you can edit the following details:

Name	Description
Scenario Name	Scenario name represents a human-readable name for a visibility scenario.
Scenario Description	Brief description of the visibility scenario.
Instances Label	Instances label represents a group of instances in the **Process Workspace**. For example: Orders, Deliveries.
Instance Label	Instance label represents an instance in the **Process Workspace**. For example: Order, Delivery. **Instance Label** value can contain static text, a dynamic value that refers to the attribute ID of the visibility scenario, or a combination of both static text and dynamic value. **Example** Leave Request for {EmployeeID} In this example, {EmployeeID} is the reference to the attribute EmployeeID in the visibility scenario. The value is resolved dynamically based on the attribute value.

5. You can further enhance your visibility scenario by adding processes and configuring elements such as state, status, calculated attributes, actions, phases, and performance indicators. This allows for real-time visibility of running processes, providing insights into both current and past performance. For more information, see Configure a Visibility Scenario.
6. Choose **Save** to save your changes, which will also automatically unlock the scenario. If you wish to continue to edit your scenario, choose **Save** > **Save and keep lock** button.

Note

- If the process in the visibility scenario is modified or deleted, choose the *Reimport* icon to reflect the changes in the visibility scenario. In case the process is not available in the project, select the process and choose *Delete*.
- If you have modified the visibility scenario and want to revert to the last saved settings before saving the recent changes, choose **Reset** button. **Reset** button is enabled only if the process in the visibility scenario is either modified or deleted.

Note

Upon successful deployment of the business process project, processing is automatically scheduled for the visibility scenario. You can use the dashboard icon for the respective visibility scenario within the **Monitoring** > **Visibility Scenarios** tile to track the processes in real-time, filter and search instances, analyze performance indicators, and view the detailed information about an instance. For more information, see Gain Visibility Using Process Workspace.

Related Information

Configure a Visibility Scenario

Configure a Visibility Scenario

Configure a visibility scenario to gain real-time visibility on processes by providing actionable insights on running processes that helps you to understand the current and the past process performance.

Prerequisites

You have created a visibility scenario within a business process project. For more information, see Create a Visibility Scenario.

Context

You can add one or more processes and further enhance your visibility scenario by configuring state, status, calculated attributes, actions, phases, and performance indicators to gain real-time visibility on running processes that helps you to understand the current and the past performance.

This image is interactive. Hover over each area for a description. Click highlighted areas for more information.

Please note that image maps are not interactive in PDF output.

Configure Processes

You can add one or more processes to create a series of logically related activities or tasks that are performed together in a visibility scenario.

Context

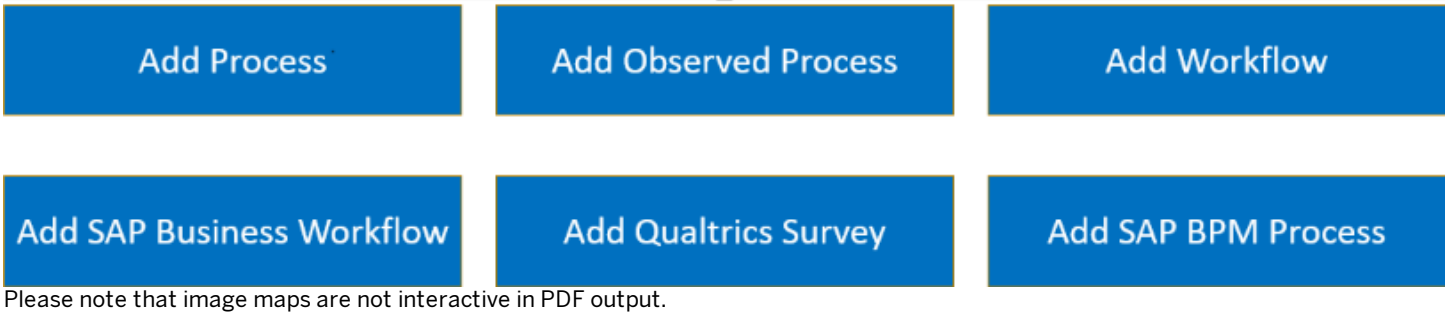
A visibility scenario allows users to track the performance of end-to-end processes. You can add the processes involved along with events and context information.

Configure the following entities for each process of a visibility scenario:

- **Events**: Events are actions originating from either a process fragment or a business object type, which are part of the process.
- **Context**: Context attributes are a subset of the process context. You need to define these attributes and push it along with the events. Examples of the context attributes are order type, division, company code, order value, delivery quantity, promised delivery date, actual delivery date, and so on.

To add a process, choose the **Processes** tab and select *Add Process* icon, and select one of the following options from the drop down list.

This image is interactive. Hover over each area for a description. Click highlighted areas for more information.



Please note that image maps are not interactive in PDF output.

Note

- To edit the name of the process, choose the required process in the **Processes** tab and choose *Edit Process* icon.
- To delete a process, choose the required process in the **Processes** tab and choose *Delete Process* icon. The auto-updated artifacts such as events, context, state, status, and performance indicators are deleted.

Add Process

You can add one or more process/subprocess available in your business process project to your visibility scenario.

Procedure

1. Choose **Add Process**.
2. Select the required process from the list.
3. Choose **Save**.

Note

Subprocesses are technically simply processes that can be used to create reuse sequences. Eligible subprocesses need to be available in the same business process project. All processes contained within the same project can be a main process or used as a subprocess, depending on how you use them.

Add Observed Process

Observed process can be used to represent the observable events and steps of your process. You can use this procedure to add an observed process to the visibility scenario.

Context

The process visibility capability supports out-of-the-box visibility for immediate time to value. The observed process provides visibility on processes running elsewhere, or on your operational data where there is no notion of a process. You need to push the events and context belonging to the observed process.

For more information on pushing events in Live Process Project, see Push Events.

For more information on pushing events in Business Process Project, see Push Events to a Visibility Scenario.

Procedure

1. In the **Add Observed Process** dialog box, provide a **Process ID** that represents the process definition or the object type of a process participant.
2. Provide **Process Name** and choose **OK**.

   The process is created under the **Processes** tab.
3. To add events to the newly created process, perform the following steps in the **Events** section:

   a. Choose *Create New Event* icon.

   b. In the **Add Event** dialog box, provide the following details:

Name	Description
Event Type	This field refers to the `eventType` attribute of the event. For more information, see Supported Event Formats.
Event Name	This field represents a human-readable name for the event that is unique across the visibility scenario.

   You can choose to configure a **Step** if it is associated with an event. To configure a step, provide the following details for the step:

Name	Description
Name	This field represents a human-readable name for the step.
ID	This field refers to the ID of the step that is associated with an event.
Type	This field refers to the type of a step in the process. **Example** User Task, Intermediate Message Event.
Step Status	This field depicts the status of the step based on the event defined. Possible values for step status are **Completed**, **In Progress**, **Abruptly Ended**, and **Failed**. **Example** For an Order Created event, the step status could be set as **In Progress** and for the Order Completed event, the step status could be set as **Completed**.

   Note

   Ensure that the combination of **Event Type** and **Step ID** is unique within the process.

   c. Choose **OK**.

   Note

- You can add multiple events by performing the above steps.
- You can edit an event by choosing *Edit* icon in the **Events** section.
- To delete an event, choose *Delete* icon.
- You can edit a step by choosing *Edit* icon in the **Step** section.
- Removing a step removes all the events grouped under that step.
- All the events that do not contain step-related information are grouped under **Process Events**.

4. To add an icon to a step, choose *Edit* icon for the required step and select the desired **Icon** and **Color**. The icon will then be visible in the path view for the respective step in **Process Workspace**.

ℹ Note

To configure an icon for a step, both **Icon** and **Color** must be configured.

5. To add context for a process, perform the following steps in the **Context** section:

   a. Choose *Create New Context* icon.

   b. In the **Add Context** dialog box, provide the following details:

Name	Description
Name	A human-readable name which is unique across the visibility scenario.
ID	ID of the attribute

Path	This field provides the path of the context attribute in the event. For example, path for Candidate ID attribute is CANDIDATE_ID if the event is of the following format:
	Sample Code
	`{`
	`"processDefinitionId": "onboard",`
	`"processInstanceId": 3567,`
	`"eventType": "CREATED",`
	`"timestamp": "2018-08-25T02:07:42Z",`
	`"context": {`
	`"CANDIDATE_ID": 3361`
	`}`
	`}`

Data Type	The following data types are supported: **String**, **Double**, **Integer**, **Timestamp**, **Date**, **Time**, and **Boolean**. For more information, refer to the **Context Attributes** section in Attributes.

   c. Choose **OK**.

ℹ Note

- You can add multiple context attributes by performing the above steps.
- You can edit the context by choosing *Edit* icon under the **Context** section.
- To delete a context, choose *Delete* icon.

6. **Save** the visibility scenario.

Related Information

Configure Processes

Push Events to a Visibility Scenario

For an observed process that spans across systems, the applications must push events. These events are consumed by process visibility capability to provide insights on your running processes.

Prerequisites

You have the service instance created for SAP Build Process Automation. For more information, see Subscribe to SAP Build Process Automation (Standard Plan).

Procedure

1. Fetch the OAuth token. For more information, see Using SAP Build Process Automation APIs.
2. Invoke the following REST endpoint to push events:

   ```
   POST {api}/public/visibility/runtime/rest/v1/data-acquisition/data
   ```

   To know more on how to determine the service configuration parameters to access APIs, see Determine Service Configuration Parameters.

Related Information

Supported Event Formats
Sample Event Format

Supported Event Formats

The process visibility capability provides visibility and intelligent insights for processes running in one system or across systems. To achieve this, you must ensure that the data available. Specifically, the data must contain events and context attributes.

Event Attributes

The following table provides the supported event attributes:

Attribute Name	Description	Data Type	Mandatory
processDefinitionId	Determines the process definition or the object type of a process participant.	String	Yes
processInstanceId	Determines the unique execution of an object type or process that the event belongs to.	String/Integer	Yes
eventType	Identifies the type of event that has occurred. The event types can be started, completed, updated, and so on.	String	Yes
activityDefinitionId	Determines the activity or a step to which the given event belongs to.	String	No

Attribute Name	Description	Data Type	Mandatory
activityInstanceId	Used to determine the unique execution of an activity or step.	String	No
timestamp	Determines the time at which an event occurred. The timestamp format must be yyyy-mm-ddThh:mm:ssZ. **Example** 2018-08-25T02:07:42Z	Date Time timestamp must be in the ISO 8601 format.	Yes
eventId	Determines the ID of the event.	String	No
activityType	Determines the type of the activity/step. This attribute is relevant only when the notion of steps comes from the runtime event.	String	No
activityName	Defines the name of the activity/step. This attribute is relevant only when the notion of steps comes from the runtime event.	String	No
activityStatus	Determines the status of the activity/step when the event is observed. The permissible values are IN_PROGRESS, COMPLETED, FAILED, and ABRUPTLY_ENDED. This attribute is relevant only when the notion of steps comes from the runtime event.	String	No
eventName	Defines the name of the event.	String	No
processVersionId	Defines the version of the process for which the given event belongs to.	String	No
userDisplayName	Displays the user name.	String	No
userId	Determines the user responsible for triggering the event.	String	No
context	Contextual information that is required to be available for the process instance. It is recommended to send the required context with the first event of a process or whenever the context is available or updated.	Object	No
application	This attribute is only relevant when pushing events from Qualtrics. The value needs to be passed as "qualtrics".	String	No

Context Attributes

Each process participant in a scenario can have a context associated with it. The context enriches an instance and helps in correlating events across process participants to form a scenario instance. If the same context attributes are sent with one or more events, context attributes are updated to the value sent with the latest occurring event.

Related Information

Sample Event Format

Sample Event Format

This section provides the sample event format to push the events that are consumed by the process visibility capability.

The following sample JSON snippet shows how the events and context can be pushed:

Note

- Events should not contain characters not supported by the JSON format.
- The context attributes support the following data types: Integer, Double, Datetime, Boolean, and String.

Sample Code

```
[{
    "processDefinitionId": "e064d8be81092a99908e11ebb37c482ae32e3548",
    "processInstanceId": "c0229ba89e7211eb8a2e00000068b352",
    "activityDefinitionId": "e0aaaccd8702b77d908e11ebb8e9482ae32e3548",
    "activityInstanceId": "c190f6b69e7211eb8d8d00000068b352",
    "stepId": "e0aaaccd8702b77d908e11ebb8e9482ae32e3548",
    "stepInstanceId": "c190f6b69e7211eb8d8d00000068b352",
    "eventId": "1eb9e72c1c1014754ee100000068b352",
    "eventType": "CREATED",
    "timestamp": "2021-04-16T05:15:27Z",
    "context": {
       "CANDIDATE_ID":3361,  //Integer//
       "JOB_REQUISITION_ID":901,  //Integer//
       "FIRST_NAME":"John",  //String//
       "REGION":"United Kingdom",  //String//
       "JOB_OFFER_DATE":"2018-08-25T02:07:42Z",  //TimeStamp//
       "CURRENT_TITLE":"Engineer",  //String//
       "EMAIL_ID":"johndoe@sap.com",
       "JOB_APP_STATUS":"Offer",  //String//
       "LAST_NAME":"Doe",  //String//
       "JOB_APPLICATION_ID":2002  //Integer//
   }
}]
```

The following sample JSON snippet shows how an event without the context can be pushed:

Sample Code

```
[{
    "processDefinitionId": "onboard",
    "processInstanceId": 3567,
    "eventType": "CREATED",
    "timestamp": "2018-08-25T02:07:42Z"
}]
```

The following sample JSON snippet shows how an event can be pushed from Qualtrics:

Sample Code

```
[{
      "processDefinitionId":"SV_0ML5d7iQX6Q3s5n", //Survey ID//
      "processInstanceId":"R_25ZQyzapnr4u9bk", //Survey Response ID//
      "eventType":"Survey Submitted",
      "timestamp":"2018-08-25T02:07:42Z",
      "application":"qualtrics",
      "QID1":"Very satisfied",
      "QID2":"Yes",
      "QID3":8
}]
```

For more information about the supported event formats, see Supported Event Formats.

Add Workflow

You can add one or more workflows to your visibility scenario to gain visibility on them.

Prerequisites

You have chosen **Add Workflow** option under **Processes** tab to add a workflow as a process in your visibility scenario.

Procedure

1. In the **Add Workflow** dialog box, choose the required workflow from the list of deployed workflows.

   ### Note

   If you have two or more workflows as processes in your visibility scenario, you must correlate these processes by defining correlation conditions. For more information, see Configure Correlation Conditions.

2. **Save** the visibility scenario.

   On adding a workflow as a process to the visibility scenario:

   - Events are added for start event, end event, intermediate message events, and user task. For more information, see Events.

     ### Example

Step Type	Step Name	Event Name
Start	Onboarding	Onboarding Started
End	Onboarding	Onboarding Completed
User Task	Confirm Equipment	Confirm Equipment Created
		Confirm Equipment Completed
Intermediate Message Event	Waiting for PO Approval	Waiting for PO Approval Triggered
		Waiting for PO Approval Reached
Referenced Subflow	Approval	Approval Created/Approval Completed

     ### Note

     The start and the end events are processed as process events but are displayed as step events.

   - The context is updated for the selected workflow. The business key and workflow attributes configured for the workflow are added as context attributes. For more information about configuring workflow attributes, see Configure Workflow Attributes

for Process Visibility Capability. For more information about business keys, see Define Workflows.

> **Note**
>
> If there is no context attributes updated for the workflow in the visibility scenario, you need to configure the custom workflow attributes while modeling the workflow, and reimport the workflow to your visibility scenario. For more information on configuring the workflow attributes, see Configure Workflow Attributes for Process Visibility Capability.

- If attributes of type Double, Float, Timestamp, and Date need to be passed from the workflow capability, they can still be configured as workflow attributes of type String and the data type of these attributes can be changed in the editor.

  > **Example**
  >
  > Consider defining workflow attributes Capex, Bill Date of type String in workflow attributes. When a workflow model is imported, attributes are imported as String attributes. However, you can change the data type of Capex attribute to Double and Bill Date to Timestamp. If the values are passed, the attributes are processed. For more information, see Attributes.

  > **Note**
  >
  > We support ISO 8601 format for **Timestamp** and **Date** data types. If values passed cannot be converted, they are not processed at runtime.

- The state of the scenario is configured to include all of the end events of all the workflows.

  > **Note**
  >
  > You can remove one or more end events if they don't correspond to the end of the scenario.

- The substatuses **Workflow Failed** and **Workflow Suspended** are added under the **Critical** status. These substatuses are evaluated if any of the workflows used in the visibility scenario fail or are suspended.
- The following performance indicators are added:
  - Suspended Instances
  - Canceled Instances
  - Failed Instances
  - Step Cycle Time by Step
  - Open Instances by Step

> **Note**
>
> If the workflow used in a visibility scenario is modified, the changes can be reflected in the scenario model by choosing *Reimport* icon. After reimporting the workflow:
>
> - Events and context get updated for the selected workflow.
> - The state of the scenario is configured to include all the end events of all the workflows.
> - The substatuses **Workflow Failed** and **Workflow Suspended** are added under the **Critical** status if they are not present.
> - Changes made to the scenario such as adding a calculated attribute or a phase are retained, and would need to be reconfigured if necessary.

Related Information

Configure Processes

Add SAP Business Workflow

You can add SAP Business Workflow to your visibility scenario to gain visibility on the business workflow.

Prerequisites

- To enable visibility on SAP Business Workflow, you must configure the SAP S/4HANA On Premise system. For more information, see Configuring the SAP Build Process Automation Integration.
- You have a destination configured on SAP BTP cockpit with the following details.
  - URL: If the connection to the back-end system happens via the cloud connector, the URL is configured as following:

System	URL
SAP S/4HANA On Premise	`http://<virtual host: virtual port>`

  - Proxy Type: The value of this attribute determines how the connection needs to be made to the back-end system.

    For **OnPremise**, the connection to the back-end system happens via the cloud connector. For example, SAP S/4HANA on-premise system.
  - Location ID: Applies to **OnPremise** proxy type only and it specifies the location of the cloud connector.
  - Authentication Type: Basic
  - User ID & password: Credentials for the basic authentication.
  - Additional parameters:

Additional Properties	Details
bpmprocessvisibility.applicationType	The type of application used. Expected `applicationType` is BusinessWorkflow.
bpmprocessvisibility.systemName	Name of the source system to be displayed with the process definition in the Configure Visibility Scenarios tile.
bpmprocessvisibility.systemId	Back-end system ID.
bpmprocessvisibility.client	Back-end system client number.
(Optional) bpmprocessvisibility.appendClientIdToURL	The value is set to true if the client ID that is specified as part of the parameter `bpmprocessvisibility.client` needs to be appended to the base URL for API calls.

- You have chosen Add SAP Business Workflow to add business workflows as processes in your visibility scenario.

Procedure

1. In the Add SAP Business Workflow dialog box, choose the system from the list of systems that are configured as part of the destinations defined in SAP BTP cockpit.
2. Choose the required workflow from the list of active business workflows.
3. Save and Activate the visibility scenario.

   On adding a business workflow as a process to the visibility scenario:

   - Events are added for start event, end event, and for all the steps defined as part of the workflow.

     ### Example

Step Type	Step Name	Process Name	Event Name

Step Type	Step Name	Process Name	Event Name
Start	Onboarding	Leave Request Application	Leave Request Application Started
End	Onboarding	Leave Request Application	Leave Request Application Completed
Dialog, Background, Subflow	Approve Request		Approve Request Started
			Approve Request Completed

- The context is updated for the selected workflow. Context attributes available for the business workflow are added as context attributes.
- The following performance indicators are added:
  - Open Instance by Status
  - Completed Instances by Status
  - Cycle Time
  - Step Cycle Time by Step
  - Open Instances by Step
  - Cancelled Instances

Note

- The state of the scenario is configured to include all of the end events of all the workflows added to the scenario.
- You can remove one or more end events if they don't correspond to the end of the scenario.
- If the workflow used in a visibility scenario is modified, the changes can be reflected in the scenario model by choosing *Reimport*.
- After reimporting the workflow, changes made to the scenario such as adding a calculated attribute or a phase are retained, and would need to be reconfigured if necessary.

Add Qualtrics Survey

You can add the Qualtrics survey as a participant to the scenario to view and analyze the experience data, which helps in optimizing an end-to-end process along with the operational insights.

Prerequisites

- You have survey definitions in **Qualtrics**.
- You have defined a destination in your subaccount. For more information, see Managing Destinations.

  Enter the following details:

  - **URL**: `https://yourdatacenterid.qualtrics.com/API/v3`

    Example

    https://sapsandbox.eu.qualtrics.com/API/v3

    Note

You can find your datacenter ID by logging into your account and looking at the URL in the browser address bar. Your datacenter ID is the part before qualtrics.com.

- **Authentication Type**: OAuth Client credentials.
- **OAuth Client credentials**: Credentials to access the Qualtrics APIs to fetch surveys.

  For more information about creating an OAuth client to fetch the credentials, see Understand Qualtrics API Integration Using OAuth.
- **Additional parameters**:

Additional Properties	Details
bpmprocessvisibility.applicationType	The type of application used. Expected `applicationType` is **qualtrics**. This property is required to identify the destination as a Qualtrics destination.
bpmprocessvisibility.scope(Optional)	**Note** This parameter should be provided only if the OAuth client is created with scopes. The OAuth client used must have **read:surveys** or **manage:all** scope and one of them have to be passed as value to the additional parameter.

Procedure

1. In the **Add Qualtrics Survey** dialog box, choose a survey from the list of available surveys.

   On adding a survey as a process to the visibility scenario:

   a. An event is added to represent the submission of survey responses.

   b. The context for the survey is updated. Multiple choice survey questions with single reply and graphic sliders are introduced as context attributes. Embedded data added to the survey are also added as context attributes. For more information on the different question types used in the survey, see Question Types.

   Note

   Events can be pushed from Qualtrics to process visibility capability in bulk or as single events.

   - To push events in bulk, you can configure the integration flow available in SAP Business Accelerator Hub.
   - Single events can be pushed by configuring actions in Qualtrics. For more information, see Creating and Managing Actions.

2. **Save** the visibility scenario.

Add SAP BPM Process

You can add a process from SAP Business Process Management to your visibility scenario to gain visibility on the SAP BPM process.

Prerequisites

- You have configured SAP Business Process Management. For more information, see Configuring Business Process Management.
- To enable visibility on SAP BPM process, you need to perform configuration steps on SAP Business Process Management with SAP NetWeaver 7.5 SPS21. For more information, see SAP Note 3012514.
- To enable the communication with SAP BPM system, you have the destination configured on SAP BTP cockpit with the following details:

Property	Description
Name	Describes the name of the destination.
Type	HTTP
URL	The connection to the back-end system happens via the cloud connector. The URL is configured as following: System: SAP S/4HANA Cloud on-premise, ECC systems URL: http://<virtual host: virtual port>
ProxyType	The value of this attribute determines how the connection needs to be made to the back end system. For **On Premise** system, the connection to the back end system happens via the cloud connector. For example, S/4HANA Cloud on-premise, ECC systems.
Authentication	Basic
Location ID	Specifies the location of the cloud connector.

Additional Properties	Description
bpmprocessvisibility.applicationType	Type of application used from the configured source system. Expected **applicationType** is BPM.
bpmprocessvisibility.systemName	Name of the source system to be displayed with the process definition in the UI.
bpmprocessvisibility.systemId	Back end System ID.

- You have installed and configured SAP Cloud Connector to provide connectivity between SAP BTP applications and on-premise systems. For more information, see Connecting Cloud Applications to On-Premise Systems.
- You have enabled the `asyncpush.switch` property in SAP NetWeaver Administrator. For more information, see Enabling the Consumption of SAP Process Visibility Capability in SAP Business Process Management.
- You have chosen **Add SAP BPM Process** to add an SAP BPM process to your visibility scenario.

Procedure

1. In the **Add SAP BPM Process** dialog box, choose the system from the list of systems that are configured as part of the destinations defined in SAP BTP cockpit.
2. Choose the required SAP BPM process from the list of active processes.
3. **Save** and **Activate** the visibility scenario.

   On adding an SAP BPM process to the visibility scenario:

   - Events are added for start event, end event, and for all the steps defined as part of the process.

Example

Step Type	Step Name	Process Name	Event Name
Start	Onboarding	Leave Request Application	Leave Request Application Started

Step Type	Step Name	Process Name	Event Name
End	Onboarding	Leave Request Application	Leave Request Application Completed
Dialog, Background, Subflow	Approve Request		Approve Request Started
			Approve Request Completed

- The context is updated for the selected process. Context attributes available for the process are added as context attributes.
- The following performance indicators are added:
  - Open Instance by Status
  - Completed Instances by Status
  - Cycle Time
  - Step Cycle Time by Step
  - Open Instances by Step
  - Cancelled Instances
  - Number of Suspended Instances

Note

- Upon the initial activation of a visibility scenario, configure the required settings for the SAP Business Process Management system in SAP NetWeaver Administrator to send the events and context. For more information, see Configure the SAP BPM System.
- The state of the scenario is configured to include all of the end events of all the processes added to the scenario.
- You can remove one or more end events if they don't correspond to the end of the scenario.
- If the process used in a visibility scenario is modified, the changes can be reflected in the scenario model by choosing *Reimport*.
- After reimporting the process, changes made to the scenario such as adding a calculated attribute or a phase are retained, and would need to be reconfigured if necessary.

Configure Correlation Conditions

Correlation conditions define how processes are related to each other in a visibility scenario. Use this procedure to add correlation conditions to the visibility scenario.

Context

A visibility scenario can consist of heterogeneous processes from different process providers or it can contain events from various objects in SAP S/4HANA. Correlation conditions are required if you have two or more processes in your visibility scenario.

You can correlate the processes by defining correlation conditions. Each correlation condition is defined as a combination of a source and a target process, and the context in which they relate to each other. The relation between the source context and target context is defined by the operator of the condition.

The following correlation types are supported:

- **1:1 Correlation**: Defines a 1:1 relationship between the processes of a visibility scenario. For example, consider an employee onboarding process in SAP Success Factors that is triggered for a new hire. This process triggers an extension workflow on SAP BTP for equipment provisioning. A 1:1 correlation exists between the processes as each new hire procures an equipment.
- **N:1 Correlation**: Defines an N:1 relationship between the processes of a visibility scenario. For example, consider a scenario having an order and a delivery process where multiple orders are part of the same delivery. There exists an N:1 correlation between the processes as multiple orders can be associated with a single delivery.

Procedure

1. Choose the **Correlation** tab.
2. Choose *Add Correlation Condition*.
3. In the **Add Correlation Condition** dialog box, choose the context of the process that you want to use as the **Source Context**.
4. Choose the required operator that you want to use for correlation from the **Operator** dropdown list.

   The supported operators based on the correlation between the processes are listed in the following table:

Operator	Example

=	**1:1 correlation**
	In an employee onboarding scenario, the onboarded user ID can be used to define the correlation condition.
	`SFSF.UserID = OnboardingExtensionWF.UserID`
	N:1 correlation
	In a scenario having an order and a delivery process that has multiple orders as part of the same delivery, the order process includes the details of the delivery.
	`Order.DeliveryID = Delivery.ID`

IN	**N:1 correlation** The delivery process includes the details of all orders that will be delivered with each delivery. `Order.OrderID IN Delivery.OrderIDs` In this example, the `OrderIDs` is the context attribute that contains an array of IDs. For example, `Order.OrderId = 15`, and the `Delivery.OrderIds = [15,16,18]`.

5. Choose the context of the process that you want to use as the **Target Context**.
6. Choose **OK**.

Note

- You can add multiple correlation conditions by performing the above steps.
- You can edit a correlation condition by choosing *Edit*.
- To delete a correlation condition, choose *Delete*.

7. Choose **Save**.

Note

Ensure that you follow these guidelines when defining correlation conditions:

- Each process must be a part of at least one correlation condition.

- A lead participant is the process participant whose event starts the scenario, and you derive the lead participant from the defined correlation conditions. The lead participant is not defined as a target participant in any of the correlation conditions. Currently, you can define one lead participant.
- Avoid cyclic dependencies between the correlation conditions.
- A process can be a target process in only one correlation condition.

Configure Phases

Phases are a sequence of process steps that are part of your visibility scenario, and they are necessary if the business user needs to track and analyze a specific stage or part of the visibility scenario.

Context

Phases provide a business-friendly aggregated view on a big process, by grouping several process activities and events. A phase must have at least one start and one end event.

Procedure

1. Choose the **Phases** tab.
2. Choose *Add Phase*.
3. In the **Add Phase** dialog box, provide a **Name** and **ID** for the phase.
4. Define the start of a phase by choosing the required start events from the **Start Events** drop down list.
5. Define the end of a phase by choosing the required end events from the **End Events** drop down list.
6. **Optional:** Navigation link enables business users to navigate to different applications and find more information on a particular phase. To configure the navigation link corresponding to a phase, toggle the **Configure Navigation Link** button and provide the application URL.

   The application URL can be configured in the following ways:

   - For a **Business Process Project**, the application URL can be configured in the following ways:
     - **Environment Variable** can have the environment variable of the type **Destination**. Choose the required environment variable from the drop-down menu.

       The **Destination** created on SAP BTP cockpit can either have the complete application URL or can have the host to connect to, and the relative URL as part of the **Path** attribute.

       Note

       - Ensure that you have created an environment variable of type **Destination** in your business process project. To configure the environment variable, see Configure Project Properties and Configure Environment Variables.
       - On deploying the business process project, you can choose the required destination from the drop-down menu for the corresponding environment variable.

     - **Path** can have the complete application URL.
   - For a **Live Process Project**, the application URL can be configured in the following ways:
     - **Path** can have the complete application URL.
     - **Destination** configured on SAP BTP cockpit can have the complete application URL.
     - **Destination** configured on SAP BTP cockpit can have the host to connect to, and the relative URL is part of the **Path** attribute.

The destination you provide is the destination specified in the consumer subaccount and you can add an additional custom property **`bpmprocessvisibility.navigationLink = true`** to the destination when configuring destinations in the SAP BTP cockpit and this destination appears in the **Navigation Link** section of the **Destination** field **dropdown** list.

To configure destinations, use the standard SAP BTP mechanisms in the SAP BTP cockpit. For more information, see Managing Destinations.

Example

```
https://my87878.s4hana.ondemand.com/ui#BusinessPartner-manage&//C_BusinessPartner(BusinessPar
```

In the above example,

- `https://my87878.s4hana.ondemand.com` is the base URL that can be configured as URL in the destination created on your subaccount.
- `/ui#BusinessPartner-manage&//C_BusinessPartner(BusinessPartner={BusinessPartnerID},D` is the relative URL that can be configured as **Path**. Attributes references can be provided in the context within curly braces. The value is resolved dynamically based on the attribute value. {BusinessPartnerID} refers to the context attribute in the scenario. The value is resolved dynamically based on the attribute value.

Note

Path can have references to defined attributes in the scenario. These are resolved dynamically based on the processed value of the referenced attribute. To reference an attribute for dynamic resolution, type { in the path field and choose the required attribute from the list of proposed attributes. In the earlier example, {BusinessPartnerID} is the attribute used in the URL.

7. **Optional:** You can choose to hide a phase by defining **Conditions for Exclusion**. Phases are displayed in the instance details view based on the evaluation of the phase conditions.

Note

By default, all phases are available for all the instances.

8. Choose **Save**.

Note

- You can add multiple phases by following the above steps.
- To edit the name or ID of a phase, choose the phase in the **Phases** tab and choose *Edit Phase*.
- To delete a phase, choose the required phase under the **Phases** tab and choose *Delete Phase*.
- To change the order in which the phases are displayed in the **Process Workspace**, choose *Order Phases*.

Configure States

You use this procedure to configure the various possible states of a scenario instance.

Context

You can configure the following states for a scenario instance:

- **Open**: A scenario instance is in open state if the configured completed events or abruptly ended events have not been observed.
- **Completed**: A scenario instance is in completed state if the events that signify the end of the process are observed. For example, order completed and shipment completed.
- **Abruptly Ended**: A scenario instance is in an abruptly ended state if the events that signify that the process has ended have been observed before meeting the required criteria. For example, order canceled and request rejected.

Note

A scenario instance is in an open state until a completed or an abrupt end event is observed. The first incoming event that is processed to form the scenario instance is automatically considered as the start event and the scenario instance is considered to be in the open state. If both the completed and abrupt end events are observed, the scenario instance state is based on whichever event occurred first.

Procedure

1. Choose the **State** tab.
2. To modify the default name of the open state, edit the **Name** field in the **Open** section.
3. To configure events that change an instance to a completed state, navigate to the **Completed** section and select the events from the **End Events** dropdown list.
4. To configure events that change an instance to an abruptly ended state, navigate to the **Abruptly Ended** section and select the events from the **Abrupt End Events** dropdown list.

   ### Note

   - You can configure multiple completed and abrupt end events.
   - If you want to modify the default name of a completed state or an abruptly end state, modify the **Name** field in the **Completed** section or in the **Abruptly Ended** section respectively.

5. Choose **Save**.

Configure Status

You can configure visibility scenario target and sub-statuses to define the status of a scenario instance.

Context

The scenario target defines the expected time by when a scenario instance must be completed. The status depicts how a scenario instance is progressing based on the scenario target or the defined custom sub-statuses. For more information about default statuses and sub-statuses, see Status.

Procedure

1. Choose the **Status** tab.
2. To configure the target for a scenario, perform the following steps in the **Target** section:

   a. Select the target type from the **Target Type** dropdown list:

Target Type	Description
Constant	Target is a fixed time for the scenario.
Attribute	Target is based on the selected context attribute.
None	No target is defined for the scenario.

   b. Depending on the **Target Type** that you chose, perform the following steps:

   - For **Constant**, provide a **Target Value**.
   - For **Attribute**, choose the required **Attribute** from the dropdown list. The **Attributes** that are displayed in the dropdown are of data type timestamp and date.

     ### Note

     We support ISO 8601 format for data type **Timestamp** and **Date**.

c. Provide the **Threshold** percentage for the target.

> **Note**
>
> Threshold violation occurs for the instance when the elapsed time exceeds the threshold target cycle time.

3. To configure a sub-status for a status, perform the following steps:

   a. Choose *Add Sub-Status.*

   b. In the **Add Sub-Status** dialog box, provide a **Name** and **ID** for the sub-status.

   c. Choose **OK**.

   The sub-status details appear on the right.

   d. Choose one of the following from the **Expression Type** dropdown list:

Expression Type	Description
Latest Event	If any of the given events is the latest occurred event for an instance, then the sub-status is set based on the latest event occurred. If you have chosen this expression type, choose the required events from the **Event** dropdown.
Event A occurred and not Event B	The sub-status is set based on the occurrence of an event A, when the event B has not occurred. If you have chosen this expression type, choose the required events as **Event A** and **Event B** from the respective dropdown menus.
Event A occurred and not Event B in a Timeframe	The sub-status is set based on the occurrence of an event A, when the event B has not occurred within the configured timeframe. If you have chosen this expression type, choose the required events as **Event A** and **Event B** from the respective dropdown menus and set the **Duration** to define the timeframe. **Note** In case the **Event B** occurs later than the specified timeframe, you can select the check box to re-evaluate and reset.

<table>
<tr><th>Expression Type</th><th>Description</th></tr>
<tr><td>Context Attribute</td><td>The sub-status is evaluated based on the condition defined on the context attribute. The context attribute can be of data type Integer, Boolean, or String.

If you have chosen this expression type, choose the required Attributeand Operator from the respective dropdown lists and provide a Value for the context attribute.

The supported operators based on the data type of the selected attribute are listed in the following table.

<table>
<tr><th>Datatype of the Attribute</th><th>Supported Operators</th></tr>
<tr><td>String</td><td>▪ Equal To
▪ Not Equal To
▪ Empty
▪ Not Empty</td></tr>
<tr><td>Integer</td><td>▪ Equal To
▪ Not Equal To
▪ Less Than
▪ Less than or equal to
▪ Greater than
▪ Greater than or equal to</td></tr>
<tr><td>Boolean</td><td>Equal To</td></tr>
</table>
</td></tr>
</table>

Note

- To change the Name and ID of a sub-status, choose the sub-status from the list and choose *Edit*.
- To delete a sub-status, choose the sub-status from the list and choose *Delete Sub-Status*.
- To modify the properties of a sub-status that you have added, choose the sub-status from the list and modify the required property in the details section.
- To change the order of the sub-statuses, choose *Order Sub-Status*. You can use the navigation icons to reorder the sub-status.

4. Choose Save.

Status

Status depicts the instance status based on how the instance is progressing. The progress of an instance is determined by targets, phases, and custom events.

The following statuses are defined for a visibility scenario:

Critical: An open instance is in critical status when the elapsed time is more than the target cycle time, or a certain condition is met for the instance. Status of the instance is critical when one of the sub-statuses defined under it evaluates to true.

At Risk: An open instance is at risk when the elapsed time is more than the threshold target cycle time or a certain condition is met for the instance. Status of the instance is at risk when one of the sub-statuses defined under it evaluates to true.

On Track: An open instance is in on track status if it is neither critical nor at risk.

Completed without Violation: Status of an instance when it is completed within the target cycle time.

Completed with Violation: Status of an instance when it is completed and the elapsed time has exceeded the target cycle time.

Target

Target is the expected time by when a scenario instance must be completed. The target can be of the type constant or dynamic based on a context attribute of data type timestamp or date. For constant target type, the target value provided is considered to be a fixed time. For dynamic target type, the target value is the selected context attribute value.

ℹ Note

A target configured based on an attribute must have timestamp or date values in ISO 8601 format.

Sub-Status

Sub-status provides insights on why an instance is in a certain status. By default, the following sub-statuses are defined for a scenario status:

Status	Default Sub-Status	Description
Critical	Overdue	An instance is critical when the elapsed time is more than the target cycle time. The overdue sub-status appears only if the target is defined.
At Risk	Threshold Violation	An instance is at risk when the elapsed time is more than the threshold target cycle time. Threshold violation appears only if the target is defined.
On Track	On Time	An instance is on track when the elapsed time is less than the target cycle time.
Completed with Violations	Completed with Time Violations	An instance acquires the status completed with violations, when it is completed and the elapsed time has exceeded the target cycle time. Completed with time violations appears only if the target is defined.
Completed without Violations	Completed	An instance acquires the status completed without violations, when it is completed within the target cycle time.

Configure Attributes

Attributes provide meaningful information about an instance. They help to identify an instance from a group of instances. The line-of-business users use them to search for an instance, aggregate instances, or filter instances.

Context

You can configure calculated attributes and entities that are required to configure the performance indicators. For detailed information about the different types of attributes available, see Attributes.

Procedure

1. Choose the **Attributes** tab.

ℹ Note

You have the following set of attributes that are already defined in the left panel:

- **Default**
- **Context**
- **Step**

2. To configure default attributes in your visibility scenario, see Configure Default Attributes.
3. To configure context attributes in your visibility scenario, see Configure Context Attributes.
4. To add calculated attributes to your visibility scenario, see Add Calculated Attributes.
5. To add entities to your visibility scenario, see Add Entities.
6. Choose **Save**.

Related Information

Attributes

Configure Default Attributes

Default attributes are created automatically for each visibility scenario and are used to define performance indicators.

Context

Default attributes can be further configured to define the way they are displayed on the **Process Workspace**. For the detailed information on the available default attributes for a visibility scenario, see Attributes.

Procedure

1. Choose the required default attribute present under the **Default** section in the left pane.
2. You can choose to define the way attributes are displayed in the **Process Workspace** by selecting one of the following options in the **Importance** drop-down list:

Name	Display Area
Key	Displays the attribute in the **Instances View** overview page as a column in the instances table. **Note** The context attributes defined as **Key** appear as the primary set of filters in the **Process Workpace**.
Significant	Displays the attribute in the **Instances View** details page as a field in the information tab.
Internal	This attribute isn't displayed in **Scenario Instances View** or the **Instance Details View**, but can be used within performance indicators.

3. Optional: Navigation link enables business users to navigate to different applications and find more information on a particular attribute. To configure the navigation link corresponding to an attribute, toggle the **Configure Navigation Link** button and provide the application URL.

   The application URL can be configured in the following ways:

   - For **Business Process Project**, the application URL can be configured in the following ways:
     - **Environment Variable** can have the environment variable of the type **Destination**. Choose the required environment variable from the drop-down menu.

The **Destination** created on SAP BTP cockpit can be configured in the following ways:

- **Destination** can have the complete application URL.
- **Destination** can have the host to connect to, and the relative URL as part of the **Path** attribute.

Note

- Ensure that you have created an environment variable of type **Destination** in your business process project. To configure the environment variable, see Configure Project Properties and Configure Environment Variables.
- On deploying the business process project, you can choose the required destination from the drop-down menu for the corresponding environment variable.

- **Path** can have the complete application URL.

- For **Live Process Project**, the application URL can be configured in the following ways:
  - **Path** can have the complete application URL.
  - **Destination** configured on SAP BTP cockpit can have the complete application URL.
  - **Destination** configured on SAP BTP cockpit can have the host to connect to, and the relative URL as part of the **Path** attribute.

The destination you provide is the destination specified in the consumer subaccount and you can add an additional custom property **`bpmprocessvisibility.navigationLink = true`** to the destination when configuring destinations in the SAP BTP cockpit and this destination appears in the **Navigation Link** section of the **Destination** field **dropdown** list.

To configure destinations, use the standard SAP BTP mechanisms in the SAP BTP cockpit. For more information, see Managing Destinations.

Example

```
https://my87878.s4hana.ondemand.com/ui#BusinessPartner-manage&//C_BusinessPartner(BusinessPartner='
```

In the above example,

- `https://my87878.s4hana.ondemand.com` is the base URL that can be configured as URL in the destination created on your subaccount.
- `/ui#BusinessPartner-manage&//C_BusinessPartner(BusinessPartner={BusinessPartnerID},DraftUI` is the relative URL that can be configured as **Path**. Attributes references can be provided in the context within curly braces. The value is resolved dynamically based on the attribute value. {BusinessPartnerID} refers to the context attribute in the scenario. The value is resolved dynamically based on the attribute value.

Note

- Navigation links can be configured for any attributes but they add value only for the attributes having the importance **Key** and **Significant**.
- Path can have references to defined attributes in the scenario. These are resolved dynamically based on the processed value of the referenced attribute. To reference an attribute for dynamic resolution, type { in the path field and choose the required attribute from the list of proposed attributes. In the earlier example, {BusinessPartnerID} is the attribute used in the URL.

4. Optional: To change the order of the attributes representation in the **Process Workspace**, choose *Order Attibutes* and select any of the following options:
   - **Instances View** to arrange the **Key** attributes in the order they need to be displayed as columns in the scenario instances view.

- Instance Details View to arrange the **Key** and **Significant** attributes in the order they need to be displayed in the scenario instance details view.

You can use the navigation icons to reorder the attributes.

5. Choose Save.

Related Information

Attributes

Configure Context Attributes

Context attributes are defined as part of the process context and are pushed together with the events.

Context

You can configure the context attributes that are pushed along with events to your visibility scenario.

Procedure

1. Choose the required context attribute present under the Context section in the left pane.
2. Optional: You can edit some of the General Properties of the context attributes.
   - You can choose one of the following data types for the context attributes: String, Double, Integer, Timestamp, Date, Time, Duration.
   - If you have chosen Double or Integer data type, you can select one of the following Unit types:

Unit Type	Description
Constant	The unit value is static.
Attribute	The unit value can be an attribute chosen from the drop-down. This value is dynamically resolved at runtime.
None	No unit is defined for the measure attribute.

3. To define where the attributes are displayed in the Process Workspace, choose one of the following in the Importance drop-down list:

Name	Display Area
Key	Displays the attribute in the Instances View overview page as a column in the instances table. **Note** The context attributes defined as Key appear as the primary set of filters in the Process Workpace.
Significant	Displays the attribute in the Instances View details page as a field in the information tab.
Internal	By default, the attribute is not displayed in the Scenario Instances View or the Instance Details View. However, you can choose the required attribute in the View Settings dialog box present under the Settings option to display it in the Scenario Instances View.

4. By default, the Archivable check box is selected. You can uncheck to exclude the attribute from being archived. For more information, see Download Archived Data.

5. Optional: Navigation link enables business users to navigate to different applications and find more information on a particular attribute. To configure the navigation link corresponding to an attribute, toggle the **Configure Navigation Link** button and provide the application URL.

   The application URL can be configured in the following ways:

   - For **Business Process Project**, the application URL can be configured in the following ways:
     - **Environment Variable** can have the environment variable of the type **Destination**. Choose the required environment variable from the drop-down menu.

       The **Destination** created on SAP BTP cockpit can be configured in the following ways:

       - **Destination** can have the complete application URL.
       - **Destination** can have the host to connect to, and the relative URL as part of the **Path** attribute.

       Note

       - Ensure that you have created an environment variable of type **Destination** in your business process project. To configure the environment variable, see Configure Project Properties and Configure Environment Variables.
       - On deploying the business process project, you can choose the required destination from the drop-down menu for the corresponding environment variable.

     - **Path** can have the complete application URL.
   - For **Live Process Project**, the application URL can be configured in the following ways:
     - **Path** can have the complete application URL.
     - **Destination** configured on SAP BTP cockpit can have the complete application URL.
     - **Destination** configured on SAP BTP cockpit can have the host to connect to, and the relative URL as part of the **Path** attribute.

   The destination you provide is the destination specified in the consumer subaccount and you can add an additional custom property **`bpmprocessvisibility.navigationLink = true`** to the destination when configuring destinations in the SAP BTP cockpit and this destination appears in the **Navigation Link** section of the **Destination** field **dropdown** list.

   To configure destinations, use the standard SAP BTP mechanisms in the SAP BTP cockpit. For more information, see Managing Destinations.

   Example

   ```
   https://my87878.s4hana.ondemand.com/ui#BusinessPartner-manage&//C_BusinessPartner(BusinessPartner='
   ```

   In the above example,

   - `https://my87878.s4hana.ondemand.com` is the base URL that can be configured as URL in the destination created on your subaccount.
   - `/ui#BusinessPartner-manage&//C_BusinessPartner(BusinessPartner={BusinessPartnerID},DraftUl` is the relative URL that can be configured as **Path**. Attributes references can be provided in the context within curly braces. The value is resolved dynamically based on the attribute value. {BusinessPartnerID} refers to the context attribute in the scenario. The value is resolved dynamically based on the attribute value.

   Note

   - Navigation links can be configured for any attributes but they add value only for the attributes having the importance **Key** and **Significant**.
   - Path can have references to defined attributes in the scenario. These are resolved dynamically based on the processed value of the referenced attribute. To reference an attribute for dynamic resolution, type `{` in the path field and choose

the required attribute from the list of proposed attributes. In the earlier example, {BusinessPartnerID} is the attribute used in the URL.

6. Optional: To change the order of the attributes representation in the **Process Workspace**, choose *Order Attibutes* and select any of the following options:
    - **Instances View** to arrange the **Key** attributes in the order they need to be displayed as columns in the scenario instances view.
    - **Instance Details View** to arrange the **Key** and **Significant** attributes in the order they need to be displayed in the scenario instance details view.

    You can use the navigation icons to reorder the attributes.
7. Choose **Save**.

Related Information

Attributes

Add Calculated Attributes

You can configure calculated attributes to define performance indicators for a visibility scenario.

Context

Calculated attributes are the attributes whose value is calculated based on the expression type defined. Calculated attributes are used within performance indicators to analyse the performance of the visibility scenario.

Procedure

1. Choose the **Attributes** tab.
2. Choose *Add Calculated Attribute*.
3. In the **Add Calculated Attibute** dialog box, provide the following:
    a. Provide a **Name** and **ID** for the calculated attribute.
    b. Choose an **Expression Type** from the drop-down list.
    c. Configure the required properties based on the selected **Expression Type** and choose **OK**.

    The calculated attribute is created under the **Calculated** group.
4. Based on the **Expression Type** that you choose, you can optionally edit some of the **General Properties** of the calculated attributes.

    For more information on the supported expression types, see Attributes.

    - If the **Data Type** drop-down list is editable, then choose one of the following data types: **String**, **Double**, **Integer**, **Timestamp**, **Date**, **Time**, **Duration**.
    - If the **Aggregation Role** drop-down list is editable, then you can choose one of the following:

Name	Description
Measure	Use this option to configure a measurable attribute, which is by default aggregated when selected in a query.
Dimension	Use this option to configure non-measurable calculated attributes.

    - If you have chosen **Measure** as the **Aggregation Role**, then you can choose one of the following **Aggregation Type** from the drop-down list:

Name	Description
Sum	Value of the attribute is the sum of each instance attribute value upon aggregation.
Min	Value of the attribute is the minimum attribute value among the instances upon aggregation.
Max	Value of the attribute is the maximum attribute value among the instances upon aggregation.
Avg	Value of the attribute is the maximum attribute value of the instances upon aggregation.

- For non-duration type measures, you can choose one of the following Unit type:

Unit Type	Description
Constant	The unit value is static.
Attribute	The unit value can be an attribute chosen from the drop-down. This value is dynamically resolved at runtime.
None	No unit is defined for the measure attribute.

- For the duration type measures, you can choose the required unit. Possible values for the duration data type are **Days**, **Hrs**, **Min**, and **Sec**. It results in the duration being computed based on the unit specified.

5. To define where the attributes are displayed in the Process Workspace, choose one of the following in the Importance drop-down list:

Name	Display Area
Key	Displays the attribute in the Instances View overview page as a column in the instances table. **Note** The context attributes defined as Key appear as the primary set of filters in the Process Workpace.
Significant	Displays the attribute in the Instances View details page as a field in the information tab.
Internal	By default, the attribute is not displayed in the Scenario Instances View or the Instance Details View. However, you can choose the required attribute in the View Settings dialog box present under the Settings option to display it in the Scenario Instances View.

6. Optional: Navigation link enables business users to navigate to different applications and find more information on a particular attribute. To configure the navigation link corresponding to an attribute, toggle the Configure Navigation Link button and provide the application URL.

   The application URL can be configured in the following ways:

   - For Business Process Project, the application URL can be configured in the following ways:

     - Environment Variable can have the environment variable of the type Destination. Choose the required environment variable from the drop-down menu.

       The Destination created on SAP BTP cockpit can either have the complete application URL or can have the host to connect to, and the relative URL as part of the Path attribute.

       Note

- Ensure that you have created an environment variable of type **Destination** in your business process project. To configure the environment variable, see Configure Project Properties and Configure Environment Variables.
- On deploying the business process project, you can choose the required destination from the drop-down menu for the corresponding environment variable.

- **Path** can have the complete application URL.

- For **Live Process Project**, the application URL can be configured in the following ways:
  - **Path** can have the complete application URL.
  - **Destination** configured on SAP BTP cockpit can have the complete application URL.
  - **Destination** configured on SAP BTP cockpit can have the host to connect to, and the relative URL as part of the **Path** attribute.

The destination you provide is the destination specified in the consumer subaccount and you can add an additional custom property **`bpmprocessvisibility.navigationLink = true`** to the destination when configuring destinations in the SAP BTP cockpit and this destination appears in the **Navigation Link** section of the **Destination** field **dropdown** list.

To configure destinations, use the standard SAP BTP mechanisms in the SAP BTP cockpit. For more information, see Managing Destinations.

Example

```
https://my87878.s4hana.ondemand.com/ui#BusinessPartner-manage&//C_BusinessPartner(BusinessPartner='
```

In the above example,

- `https://my87878.s4hana.ondemand.com` is the base URL that can be configured as URL in the destination created on your subaccount.
- `/ui#BusinessPartner-manage&//C_BusinessPartner(BusinessPartner={BusinessPartnerID},DraftUI` is the relative URL that can be configured as **Path**. Attributes references can be provided in the context within curly braces. The value is resolved dynamically based on the attribute value. {BusinessPartnerID} refers to the context attribute in the scenario. The value is resolved dynamically based on the attribute value.

Note

- Navigation links can be configured for any attributes but they add value only for the attributes having the importance **Key** and **Significant**.
- Path can have references to defined attributes in the scenario. These are resolved dynamically based on the processed value of the referenced attribute. To reference an attribute for dynamic resolution, type { in the path field and choose the required attribute from the list of proposed attributes. In the earlier example, {BusinessPartnerID} is the attribute used in the URL.

7. Optional: To change the order of the attributes representation in the **Process Workspace**, choose *Order Attibutes* and select any of the following options:
   - **Instances View** to arrange the **Key** attributes in the order they need to be displayed as columns in the scenario instances view.
   - **Instance Details View** to arrange the **Key** and **Significant** attributes in the order they need to be displayed in the scenario instance details view.

   You can use the navigation icons to reorder the attributes.
8. Choose **Save**.

Note

- To modify the **Name** and **ID** of an attribute, choose the attribute from the list and choose *Edit Attribute*.
- To delete an attribute, choose the attribute from the list and choose *Delete*.
- To modify properties of an attribute, choose the attribute from the list and modify the required property in the details section.

Related Information

Attributes

Add Entities

Context attributes and calculated attributes can be grouped together to form an entity. Entities help business users to define performance indicators at the level of an entity.

Context

An end-to-end process can have context attributes related to different steps within the process. For a process that starts with an order being placed, a bill gets generated for the order and an invoice is created. The process captures the context attributes across all the steps like ordering, billing, invoicing, and so on. In this process, each order contains a bill. A bill can be associated with multiple orders. To define a performance indicator like Overdue Bills, the billing related attributes can be grouped together to form entities. Performance indicators can be defined at the level of an entity.

Entities can be further related to one another and you can define the navigation from one entity to another entity based on their relationship. For example, Overdue Bills is a performance indicator which is defined with level Bills. This enables you to view all billing related attributes when you click on the Overdue Bills card in the **Process Workspace**.

Procedure

1. Choose the **Attributes** tab.
2. Choose *Add Entity*.
3. In the **Add Entity** dialog box, provide a name and an ID for the entity.

   The entity is created with the specified name.
4. Choose the <entity> definition from the left panel and configure the **General Properties** for the entity:
   - **Attributes**: Choose the required attributes to be grouped under this entity from the available list of context and calculated attributes of the scenario from the dropdown list.
   - **Keys**: **Keys** represents attributes that uniquely identifies an element within the entity. Choose one or more key attributes from the dropdown list. For example, OrderID uniquely defines an order within the Order entity.
   - **Navigate To**: Choose an entity from the dropdown list. Using **Navigate To** you can navigate from one entity (Parent entity) to another entity (Child entity) based on the relationship defined in the entity definition.

     Note

     - The child entity should have the key attributes of the parent entity added as its own attributes.
     - Choosing **Instances** from the dropdown list enables you to navigate to the scenario instances view which displays the instances that are being tracked.

   - **Internal Attributes**: Choose the required attributes from the dropdown list that should not appear in the scenario instances view of **Process Workspace**.
5. Choose **Save**.

Note

- You can add multiple entities by performing the above steps.
- To edit the Name or ID of an entity, choose the <entity> definition from the list and choose *Edit*.
- To delete an entity, choose the specific entity definition under the **Attributes** tab and choose *Delete*.

Related Information

Attributes

Attributes

Attributes provide meaningful information about an instance. They help to identify a specific instance in a group of instances. Also, line-of-business users use them to search for an instance, aggregate instances, or filter instances.

Context Attributes

Context attributes are a subset of the process context. You must define these attributes as part of the process context and push them together with the events. Some of the context attributes are order type, division, company code, order value, delivery quantity, promised delivery date, actual delivery date, and so on.

You can define the data type of context attribute to be String, Double, Integer, Timestamp, Date, Time, and Boolean. Even if the process participant pushes these context attributes as String, as long as they are defined to reflect the correct data, the values can be converted, and these attributes can be processed.

Example

Context attributes Capex is defined to be of type Double, Bill Date is defined to be of type Timestamp, Request Time is defined to be of type Time, and IsOrdered is defined to be of type Boolean. The incoming event has the values "345.67" for Capex, "2019-09-09T03:34:45.123Z" for Bill Date, and "true" or "false" for IsOrdered.

These values are processed and Capex has the value "345.67", Request Time has the value "03:24:45",and Bill Date has the value "2019-09-09T03:34:45.123Z". However, if invalid values are passed, for example, "test" for Capex, "2019-09-0903:34:456" for Bill Date, and "test" for IsOrdered, the values cannot be processed.

A Few More Examples Are Listed in the Following Table.

Data Type	Supported Format	Example
Timestamp	ISO 8601	2019-09-09T03:34:45.123Z
Date	ISO 8601	2019-09-09
Time	ISO 8601	03:34:45
Boolean	NA	true or false

Default Attributes

Default attributes are created automatically for each visibility scenario they are required to create meaningful performance indicators.

The following default attributes are available for a visibility scenario:

ID	Name	Description
SC_State	State	Show the state of the instance based on start, end, and abrupt end events of a process. For more information, see Configure States.

ID	Name	Description
SC_Status	Status	Shows the status of the instance based on how an instance is progressing. For more information, see Status. **Note** The default values of a status can be changed in the **Status** tab.
SC_SubStatus	Sub-Status	Provides insights on why an instance is in a certain status. For more information, see Status. **Note** Configure sub-statuses in the **Status** tab.
SC_Start_Time	Start Time	Defines the start timestamp of a scenario instance. This is the timestamp of the first event observed for the scenario instance.
SC_End_Time	End Time	Defines the end timestamp of a scenario instance. If there are multiple end events, it's the timestamp of the first end event observed.
SC_Elapsed_Time	Elapsed Time	The duration between the start time and the end time. If the end time isn't observed, then it's the duration between the start time and the current time.
SC_Number_Of_Instances	Number of Instances	Determines the total number of instances for a given filter criteria or breakdown. For every scenario instance created, the value of this attribute is computed to 1. When this attribute is used as a measure in defining performance indicators, it's computed as the total number of instances for a given filter criteria or breakdown.
SC_Active_Phases	Active Phases	Determines the active phases of the open instance. If an instance has multiple active phases, then the value of this attribute is the phase name of the active phases, separated by a comma.
SC_Completed_Phases	Completed Phases	Determines the completed phases for an instance. If an instance has multiple completed phases, then the value of this attribute is the phase name of the completed phases, separated by a comma.
SC_Active_Steps	Active Steps	Returns one or more steps currently in progress or in failed state for a scenario instance. If an instance has multiple active steps, then this attribute value is the step name of all the active steps separated by comma.
SC_Completed_Steps	Completed Steps	Returns one or more steps that are completed for a scenario instance. If an instance has multiple completed steps, then this attribute value is the step name of the completed steps separated by comma.

ID	Name	Description
SC_Target_Time	Target Time	Defines the target time by when a scenario instance should be completed. This is derived from the start time and target defined for a constant target. For a dynamic target, the value of this attribute is the same as the defined target attribute. **Note** This attribute is added to the scenario when the target is defined for the scenario.
SC_Threshold_Time	Threshold Time	Defines the time after which an open instance is determined to be at risk. This is derived from the start time, target, and defined threshold. **Note** This attribute is added to the scenario, when the target is defined for the scenario.

Calculated Attributes

Calculated Attributes are attributes whose value is calculated based on the expression defined. They are used within performance indicators to analyze the performance of a visibility scenario.

The expressions defined to calculate the attributes can be of the following types:

Event-Based Expressions

These expressions are based on whether or not particular events are observed for an instance.

The following are the supported event-based expressions:

Expression Type	Description	Example	Datatype
Duration between events	The time elapsed between the end event and the start event. Status depicts the instance status based on how an instance is progressing. Progress of the instance is determined by targets, phases, custom events, and predictions. **Note** By default, if the data type of the attribute is an integer, then the duration is displayed in seconds. If the data type of the attributes is duration, then the duration is calculated based on the unit.	Billing Cycle Time: In an order-to-cash (O2C) scenario, you use this expression to determine the time taken for an order to be delivered.	Integer or Duration

Expression Type	Description	Example	Datatype
Has event occurred	Whether or not the given event has occurred for an instance. **Note** The value is evaluated to 1 if the event has occurred and 0 if the event hasn't occurred.	Occurrence of Invoice Creation event: In an O2C scenario, you use this expression to determine whether the invoice has been created.	Integer
Latest Event	Whether any of the given events is the latest occurred event for an instance. **Note** The value is evaluated to 1 if any of the events defined in the list is the latest observed event and 0 otherwise.	Item with validation error: In an O2C scenario, orders can be blocked without further processing. You use this expression to determine the count of such orders.	Integer
Number of times an event has occurred	Total number of times a specific event has occurred for an instance.	Number of times an equipment request is reworked: When an equipment request is rejected, the request must be reworked by the requestor. You use this expression to determine the number of times the request is being reworked.	Integer
Time of an Event	Timestamp of an event.	Approval Date	Timestamp
Event A before Event B	Whether the given source event (Event A) has occurred before the given target event (Event B). **Note** The value is evaluated to 1 if event A has occurred before event B at least once for a given instance, and 0 if this pattern of events isn't observed.	Invoice was created before the creation of purchase order.	Integer
Event A occurred and not Event B	Whether the given source event (EventA) has occurred but the given target event (EventB) hasn't occurred for an instance. **Note** The value is evaluated to 1 if the condition is satisfied and 0 if the condition isn't satisfied.	Billable document is created but items aren't invoiced: In a Consume-to-Cash (C2C) scenario, you use this expression to determine if the billable documents are created but not invoiced for an instance.	Integer

Expression Type	Description	Example	Datatype
Event A occurred and not Event B in a Timeframe	Whether the given source event (EventA) has occurred but the given target event (EventB) has not occurred for an instance within the specified time. The time is calculated based on the time when Event A occurs. **Note** • The value is evaluated to 1 if event B has not occurred within the configured timeframe since the occurrance of event A. Else, the value is evaluated to 0. • If event A or event B is observed again, the attribute value is re-evaluated based on the latest occurrence of the event.	Approval task is created but not completed in three days.	Integer
Starts With Event	Whether any of the given events is the start event for an instance. **Note** The value is evaluated to 1 if any of the configured events evaluates to being the start event of the instance. If this condition is not met, the value is 0.	Invoices are scanned for processing only after the vendor creates an invoice. In such a case, scanning of an invoice should never start the process. You can use this expression to detect these anomalies or compliance issues.	Integer

Attribute-Based Expressions

Attributes that are based on expressions are based on the existing context attributes or default attributes.

The following are the supported attribute-based expressions:

Expression Type	Description	Example	Datatype
Constant	A constant value with aggregation of max, min, sum, or average. A constant string.	Currency as a constant attribute.	Double, Integer, or String

Expression Type	Description	Example	Datatype
Duration between attributes	The difference between two timestamps or dates. **Note** By default, the duration is calculated in seconds, if the datatype of the attribute is integer. The duration is calculated based on the unit if the data type of the attributes is duration.	Time taken to create a billable document.	Integer or Duration
Percentage of Attribute A to Attribute B	The percentage of occurrence of attribute A in the attribute B.	Consider attribute A is the approval cycle time and attribute B is the cycle time. By configuring this expression type, you can view the percentage of approval cycle time taken when compared with the entire cycle time. If the attribute A (approval cycle time) value is 40 and attribute B (cycle time) value is 100, then the PPI evaluates to 40 percentage.	Decimal
Day of Attribute	This expression is used to extract day from a given timestamp or date based attribute. You can use this expression as a dimension in a performance indicator to view a measure value by day.	If you want to view the number of completed instances by day, you create an attribute with Day of Attribute expression on SC_End_Time. This attribute evaluates to the first instant of the day when the instance is completed. While configuring a performance indicator, create a measure as SC_Number_Of_Instances and dimension as the created attribute. You can add a filter to this attribute. For example, add a filter with the day of attribute where the operator is **last x days** and the value is the number of days, then you can view the performance indicator for the last **x** number of days.	Timestamp or Date
Week of Attribute	This expression is used to extract week from a given timestamp or date based attribute. You can use this expression as a dimension in a performance indicator to view a measure value by week.	If you want to view the number of completed instances by week, you create an attribute with Week of Attribute expression on SC_End_Time. This attribute evaluates to the first instant of the week when the instance is completed. While configuring a performance indicator, create a measure as SC_Number_Of_Instances and dimension as the created attribute.	Timestamp or Date
Month of Attribute	This expression is used to extract month from a given timestamp or date based attribute. You can use this expression as a dimension in a performance indicator to view a measure value by month.	If you want to view the number of completed instances by month, you create an attribute with Month of Attribute expression on SC_End_Time. This attribute evaluates to the first instant of the month when the instance is completed. While configuring a performance indicator, create a measure as SC_Number_Of_Instances and dimension as the created attribute. For example, the value corresponding to the month of july 2019 is **2019-07-01T00:00:00**.	Timestamp or Date

Expression Type	Description	Example	Datatype
Quarter of Attribute	This expression is used to extract quarter from a given timestamp or date based attribute. You can use this expression as a dimension in a performance indicator to view a measure value by quarter.	If you want to view the number of completed instances by quarter, you create an attribute with Quarter of Attribute expression on SC_End_Time. This attribute evaluates to the first instant of the quarter when the instance is completed. While configuring a performance indicator, create a measure as SC_Number_Of_Instances and dimension as the created attribute.	Timestamp or Date
Year of Attribute	This expression is used to extract year from a given timestamp or date based attribute. You can use this expression as a dimension in a performance indicator to view a measure value by year.	If you want to view the number of completed instances by year, you create an attribute with Year of Attribute expression on SC_End_Time. This attribute evaluates to the first instant of the year when the instance is completed. While configuring a performance indicator, create a measure as SC_Number_Of_Instances and dimension as the created attribute.	Integer

Process-Based Expressions

These expressions are based on the processes that are part of a visibility scenario.

Expression Type	Description	Example	Datatype
Process Information	This expression is used to retrieve the **Process Instance ID** for a process.	To view the details of a workflow instance associated with a specific scenario instance in the **Monitor Workflows** tile, you require the workflow instance ID. You can use this expression to retrieve the **Process Instance ID** for a process. Using this ID, you can navigate to the **Monitor Workflows** tile to view the details of the workflow instance. **Example** `<url>/cp.portal/site#bpmworkflowmonitor-DisplayInstances?sap-ui-app-id-hint=com.sap.bpm.monitorworkflow&/workflowInstances/{worklfow_instance_id}` Where, `<url>` is the base URL and the `{workflow_instance_id}` is the attribute.	String

Step-Based Expressions

These expressions are based on steps that are part of a visibility scenario.

Expression Type	Description	Type	Datatype

Expression Type	Description	Type	Datatype
Step Information	This expression type is used to retrieve step-based information based on the type chosen. You can choose one of the following types: • Step Instance ID • Step Processor	Step Instance ID: This expression is used to retrieve the step instance ID of a step in a process. **Example** If you want to define an action to increase the priority of a step that is overdue, you require the step instance ID. You can use this expression to retrieve the **Step Instance ID** of a step in a process. This step instance ID can be passed to the workflow as payload while configuring actions. The workflow to be triggered when the action is invoked can further call the workflow API which uses the step instance ID (task instance ID) to increase the priority of the task. For more information, see Workflow API.	String
		Step Processor: This expression is used to retrieve the processor information for a step. You can choose a step only for a process in SAP Build Process Automation/SAP BTP Workflow. The attribute configured using this expression is displayed in the Process Workspace. The step processor value will be empty if the user task is not claimed.	String

Step Attributes

Steps are part of the process intended to perform a specific task. In an end to end process, steps can either be automated or might require manual intervention to complete it.

The following step attributes are available for a visibility scenario:

ID	Name	Description
SC_Step	Step	Name of the Step.

<table>
<tr><th>ID</th><th>Name</th><th>Description</th></tr>
<tr><td>SC_Step_Type</td><td>Step Type</td><td>Determines the type of a step in the process. In the case where the scenario is having a workflow as a process, step type is computed based on the activity type.

Refer to the following table:

<table><tr><th>Activity Type</th><th>Step Type</th></tr><tr><td>User Task</td><td>User Task</td></tr><tr><td>Intermediate Message Event</td><td>Intermediate Message Event</td></tr></table></td></tr>
<tr><td>SC_Step_Cycle_Time</td><td>Step Cycle Time</td><td>Time taken to complete the step</td></tr>
<tr><td>SC_Number_Of_Steps</td><td>Number of Steps</td><td>Determines the total number of steps for a given filter criteria or breakdown.</td></tr>
<tr><td>SC_Step_Outcome</td><td>Step Outcome</td><td>Exposes the business outcome for the processes of type SAP BTP Workflow, and SAP Build Process Automation added to the visibility scenario. For example, Approved, Rejected, and so on.

The business outcome is exposed only for tasks based on forms. The decision configured is exposed as a business outcome</td></tr>
</table>

Note

Steps are evaluated for scenarios having processes of type Workflow from SAP Workflow service, SAP Business Workflow, and Process Template.

Phase Attributes

Phases are a sequence of process steps that are part of your visibility scenario.

The following phase attributes are available for a visibility scenario:

ID	Name	Description
SC_Phase	Phase	Name of the Phase

<table>
<tr><th>ID</th><th>Name</th><th>Description</th></tr>
<tr><td>SC_Phase_Status</td><td>Phase Status</td><td>Determines the status of a phase. Possible values are In Progress, Skipped, Completed, Abruptly Ended, and Not Started.

Refer to the following table:

<table>
<tr><th>Phase Event Conditions</th><th>Phase Status</th></tr>
<tr><td>When the start and end events of the phase are observed</td><td>Completed</td></tr>
<tr><td>When the start of the phase is observed but the end event of the phase is not observed</td><td>In Progress</td></tr>
<tr><td>When the start event of the phase is observed but the end event of the phase is not observed and the scenaio instance is in completed state</td><td>Abruptly Ended</td></tr>
<tr><td>When the start and end events of the phase have not occurred and the scenario instance is in completed state</td><td>Skipped</td></tr>
<tr><td>When the start and end events of the phase have not occurred</td><td>Not Started</td></tr>
</table>
</td></tr>
<tr><td>SC_Phase_Cycle_Time</td><td>Phase Cycle Time</td><td>Time taken to complete the phase</td></tr>
</table>

Configure Actions

Actions help business users to act on situations in a process. Actions are defined at the visibility scenario level, and are therefore available for all instances of the visibility scenario provided the configured conditions are satisfied.

Context

You can define the following type of actions:

- **Navigational** actions are defined when you want to navigate to an application and perform the action.
- **Trigger Workflow** actions are defined when you want to initiate a **Workflow** to respond to certain situations.

  ### Example

  The workflow can handle the escalation flow by either sending an email or creating a task for a person to work on.

Procedure

1. Choose the **Actions** tab.

2. Choose *Add Action*.

3. In the **Add Action** dialog box, provide the following details for an action and choose **OK**.

Name	Description
Name	This field represents a human-readable name for an action.
ID	This field refers to the ID of the action.
Type	This field refers to the type of an action. You can choose one of the following type of actions: **Action** – **Description** Navigational – Navigational action enables the you to navigate to different applications and take specific actions. Trigger Workflow – Trigger Workflow action enables you to initiate a workflow to respond to certain situations.
Sentiment	This field defines its own visual appearance that matches its semantics. You can configure one of the following types of sentiment for your actions: **Type** – **Description** – **Example** Neutral – Neutral action – Claim Positive – Positive action – Approve Negative – Negative action – Reject

ℹ Note

The action configured is shown in the details view of a scenario instance in the **Process Workspace** tile with the name that is configured.

4. You can define conditions for the actions created. Navigate to the **Conditions** section.

   a. Choose *Add Condition.*

   b. In the **Add Condition** dialog box, choose the required **Attribute** from the dropdown list. Conditions can be defined based on all the attributes with the data type **String** and **Integer**.

   c. In the **Operator** dropdown list, choose the required operator.

   d. Choose the required **Value** from the drop-down list.

   e. Choose **OK**.

5. To configure **Navigation Link** for the navigational action, see Navigational Action.

6. To configure **Workflow Details** for a trigger workflow action, see Trigger Workflow Action.

ℹ Note

- The actions are visible in a **Process Workspace** only when the conditions are satisfied.
- You can define multiple conditions for an action and the action is evaluated if the conditions are satisfied.
- If multiple conditions are defined on the same attribute, the action is evaluated if either of the condition is satisfied.
- To modify the **Name** and **ID** of an action, choose the action from the list and choose *Edit Action*.

- To change the order of actions representation in the **Process Workspace**, choose *Order Actions*. You can use the navigation icons to reorder the actions.

7. Choose **Save**.

Related Information

Navigational Action
Trigger Workflow Action

Navigational Action

Configure navigational action to enable the business users to navigate to different applications and take specific actions.

Procedure

1. Configure the application URL for the action:
   - For **Business Process Project**, the application URL can be configured in the following ways:
     - **Environment Variable** can have the environment variable of the type **Destination**. Choose the required environment variable from the drop-down menu.

       The **Destination** configured on SAP BTP cockpit can have the complete application URL or can have the host to connect to, and the relative URL as part of the **Path** attribute.

       Note

       - Ensure that you have created an environment variable of type **Destination** in your business process project. To configure the environment variable, see Configure Project Properties and Configure Environment Variables.
       - On deploying the business process project, you can choose the required destination from the drop-down menu for the corresponding environment variable.

     - **Path** can have the complete application URL.
   - For **Live Process Project**, the application URL can be configured in any of the following options:
     - **Path** can have the complete application URL.
     - **Destination** configured on SAP BTP cockpit can have the complete application URL.
     - **Destination** configured on SAP BTP cockpit can have the host to connect to, and the relative URL is part of the **Path** attribute.

   The destination you provide is the target specified in the consumer subaccount. You must add an additional custom property **`bpmprocessvisibility.navigationLink = true`** while configuring destination in SAP BTP cockpit.

   Use the standard SAP BTP mechanisms in the SAP BTP cockpit to configure destinations. For more information, see Managing Destinations.

   Example

   You can navigate to the SAP S/4HANA Cloud application and take a specific action on the business partner object.

   ```
   https://my87878.s4hana.ondemand.com/ui#BusinessPartner-manage&//C_BusinessPartner(BusinessPartner='
   ```

   In the given example,

   - `https://my87878.s4hana.ondemand.com` is the base URL that can be configured as URL in the destination created on your subaccount.

- `/ui#BusinessPartner-manage&//C_BusinessPartner(BusinessPartner={BusinessPartnerID},DraftUI` is the relative URL that can be configured as **Path** in the **Configure Visibility Scenarios** tile.
- `{BusinessPartnerID}` is the reference to the attribute BusinessPartnerID in the scenario used in the URL. The value is resolved dynamically based on the attribute value.

2. Toggle the **In-Place Navigation** button to navigate to the configured application URL in the same window of your browser. This action opens the configured application URL in the same window of your browser.

> **Note**
>
> If the navigation is configured to an SAP Build Work Zone, standard edition application, the `sap-xapp-state` parameter and its value must be omitted from the URL.

Trigger Workflow Action

You can trigger actions in Process Workspace to handle escalations by configuring the action **Trigger Workflow**.

Prerequisites

You have created an action of the type **Trigger Workflow**.

Context

Trigger Workflow actions are defined when you want to initiate a **Workflow** to respond to certain situations. Business users can trigger actions in Process Workspace to handle escalations. For example, the workflow can handle the escalation flow by either sending an email or creating a task for a person to work on.

Trigger Workflow Action for Business Process Project

Procedure

1. Choose one of the following type of trigger workflow action:
   - **User**: User triggered actions are triggered by business users in **Process Workspace** to handle escalations. For example, trigger an action for process owners to analyze on why the overall cycle time of the process exceeds the target time in few cases.
   - **System**: System triggered actions are triggered at the time of scenario processing if the configured conditions are satisfied. For example, notifying process owners when scenario instances are classified with status **At Risk**.
2. Choose the required **Environment Variable** from the drop-down menu. The environment variable can be of the type **Destination**.

   Ensure that you have created an environment variable of type **Destination** in your business process project. To configure the environment variable, see Configure Project Properties and Configure Environment Variables.

   To configure a destination, use the standard SAP BTP mechanism in the SAP BTP cockpit and provide the following details. For more information, see Managing Destinations.

Parameter	Description
URL	Fetch the **url** from the service key created for SAP Build Process Automation service instance and append it with `public/workflow/rest`
Proxy Type	Internet

Parameter	Description
Authentication	Select the authentication type you need for the connection. - For user triggered actions - `OAuth2JWTBearer` or `OAuth2UserTokenExchange` - For system triggered actions - `OAuth2ClientCredentials`
Client ID, Client Secret, Token Service URL	You can fetch these parameters from the service key created for the SAP Build Process Automation service instance.
Additional Properties	Add an additional property. - For user triggered actions: `bpmprocessvisibility.triggerWorkflow = user` - For system triggered actions: `bpmprocessvisibility.triggerWorkflow = system`

ℹ Note

On deploying the business process project, you can choose the required destination from the drop-down menu for the corresponding environment variable.

3. **Workflow Definition ID:** The definition ID of the workflow that needs be triggered.
4. **Optional:** **Start Context**: A payload to start the workflow.
    - For user trigerred action:

Sample Code

```
{
  "capexRequestID":"{capexRequestID}",
  "Requestor":"{Requestor}",
  "Subject":"Analyze reason for {capexRequestID} exceeding the target KPI of 30 days"
}
```

    - For system trigerred action:

Sample Code

```
{
  "ProcessOwner":"{Email_ID}",
  "Subject":"Scenario instance {instance_id} is running at risk of delayed delivery"
}
```

Attribute references can be provided in the context within curly braces. The value is resolved dynamically based on the attribute value. In the above example, `capexRequestID` and `Requestor` are context attributes in the scenario.

ℹ Note

For user trigerred action, the user triggering the action in **Process Workspace** must have the `ProcessAutomationParticipant` role collection assigned. For more information, see Authorization Configuration.

Trigger Workflow Action for Live Process Project

Procedure

1. Choose one of the following type of trigger workflow action:
   - **User**: User triggered actions are triggered by business users in **Process Workspace** to handle escalations. For example, trigger an action for process owners to analyze on why the overall cycle time of the process exceeds the target time in few cases.
   - **System**: System triggered actions are triggered at the time of scenario processing if the configured conditions are satisfied. For example, notifying process owners when scenario instances are classified with status **At Risk**.
2. Choose the required **Destination** form the drop-down list.

   On choosing the destination, credentials of the workflow capability within the SAP Workflow Management service are fetched that are used to initiate the workflow.

   To configure a new destination, use the standard SAP BTP mechanism in the SAP BTP cockpit and provide the following details. For more information, see Managing Destinations.

Parameter	Description
URL	Fetch the **url** from the service key created for SAP Build Process Automation service instance and append it with `public/workflow/rest`
Proxy Type	Internet
Authentication	Select the authentication type you need for the connection. - For user triggered actions - `OAuth2JWTBearer` or `OAuth2UserTokenExchange` - For system triggered actions - `OAuth2ClientCredentials`
Client ID, Client Secret, Token Service URL	You can fetch these parameters from the service key created for the SAP Build Process Automation service instance.
Additional Properties	Add an additional property. - For user triggered actions: `bpmprocessvisibility.triggerWorkflow = user` - For system triggered actions: `bpmprocessvisibility.triggerWorkflow = system`

3. **Workflow Definition ID:** The definition ID of the workflow that needs be triggered.
4. **Optional: Start Context**: A payload to start the workflow.
   - For user trigerred action:

     ### Sample Code

     ```
     {
       "capexRequestID":"{capexRequestID}",
       "Requestor":"{Requestor}",
       "Subject":"Analyze reason for {capexRequestID} exceeding the target KPI of 30 days"
     }
     ```

   - For system trigerred action:

     ### Sample Code

```
{
  "ProcessOwner":"{Email_ID}",
  "Subject":"Scenario instance {instance_id} is running at risk of delayed delivery"
}
```

Attribute references can be provided in the context within curly braces. The value is resolved dynamically based on the attribute value. In the above example, `capexRequestID` and `Requestor` are context attributes in the scenario.

Note

For user trigerred action, the user triggering the action in Process Workspace must have the `ProcessAutomationParticipant` role collection assigned. For more information, see Authorization Configuration.

Configure Performance Indicators

You can define performance indicators that need to be a part of the overview page of the Process Workspace tile.

Context

Performance indicators represent aggregated information of a measure, grouped by dimension, and applied filters. A group of performance indicators gives the business users a holistic view of the process and enables them to understand process performance at a glance.

Each performance indicator is represented as a card in the overview page of the Process Workspace. They are based on attributes that are defined in the scenario model. For more information about sample performance indicators, see Sample Performance Indicators.

Procedure

1. Choose the Performance Indicators tab in the visibility scenario editor.
2. Choose *Add Performance Indicator*.
3. In the Add Performance Indicators dialog box, provide a Title, Sub-Title, and ID for the performance indicator.
4. In the General section, select your preferred way for displaying performance indicators in the Process Workspace from the Representation dropdown list.

Representation Name	Description
Header	Displays the title, subtitle, and the aggregated value of the selected measure in the header area.
List	Displays the title, subtitle, and the aggregated value of the selected measure in the header area and displays the selected measure grouped by the selected dimension in the chart area.
Bar Chart	Displays the title, subtitle, and the aggregated value of the selected measure in the header area and displays the Bar chart in the chart area where the selected measure is grouped by the selected dimension.
Donut Chart	Displays the title, subtitle, and the aggregated value of the selected measure in the header area and displays the Donut chart in the chart area where the selected measure is grouped by the selected dimension.
Column Chart	Displays thetTitle, subtitle, and the aggregated value of the selected measure in the header area and displays the Column chart in the chart area where the selected measure is grouped by the selected dimension.

Representation Name	Description
Line Chart	Displays the title, subtitle, and the aggregated value of the selected measure in the header area and displays the chart represented in the form of a line in the chart area where the selected measure is grouped by the selected dimension.
Table	Displays the title and subtitle in the header area and displays a list of records where columns are the selected attributes.
Links	Displays the title and subtitle in the header area and displays the hyperlink configured under the Links section.

5. Choose one of the following options from the Level dropdown list.
    - Instance: If the measure and dimension used while creating the performance indicator are based on instance level attributes (Default, Context or Calculated attributes).
    - Step: If the measure and dimension used while creating the performance indicator are based on step level attributes.
    - **<Entity>**: If the measure and dimension used while creating the performance indicator are based on a specific entity level attributes. You can see the entities listed in the dropdown. For more information, see Add Entities.
    - Phase: If the measure and dimension used while creating the performance indicator are based on phase level attributes.
6. In the Data section, choose one of the measure attributes from the Measure dropdown list, and choose the required Dimension from the dropdown list.
7. In the Filters section, perform the following steps to add a filter to your performance indicator.
    a. Choose *Add Filter* .
    b. In the Add Filter dialog box, choose the required Attribute from the dropdown list.
    c. In the Operator dropdown, choose the required operator.

    The supported operators are based on the data type of the selected attribute.
    - For attribute of String data type, the following operators are available:
        - Equal To
        - Not Equal To
    - For context attribute of String data type, the following operators are available:
        - Equal To
        - Not Equal To
        - Empty
        - Not Empty
    - For attribute of Integer, Double, and Duration data types, the following operators are available:
        - Equal To
        - Not Equal To
        - Less Than
        - Less than or equal to
        - Greater than
        - Greater than or equal to
    - For attributes of Timestamp and Date data types, the following operators are available:

Operator	Description
From	Measure value is aggregated on the instances where the selected attribute value is between the From value and the current date.

Operator	Description
To	Measure value is aggregated on the instances where the selected attribute value is between the current date and the To value.
Last x days	Measure value is aggregated on the instances where the selected attribute value is in the Last X Days.
Next x days	Measure value is aggregated on the instances where the selected attribute value is in the Next X Days.
Date Range	Measure value is aggregated on the instances where the selected attribute value is in the provided Date Range.
Before	Measure value is aggregated on the instances where the selected attribute value is before the current date.
Before or On	Measure value is aggregated on the instances where the selected attribute value is before or on the current date.
After	Measure value is aggregated on the instances where the selected attribute value is after the current date.
On or After	Measure value is aggregated on the instances where the selected attribute value is on or after the current date.
Equal To	Measure value is aggregated on the instances where the selected attribute value is Equal To the current date. **ℹ Note** Equal To operator is supported only for the attributes with the **Date** data type. For **Timestamp** based attributes, configure a calculated attribute using the expression type **Day of Attribute**.

- For attribute of Boolean data type the following operator is available:
  - Equal To

d. Choose a Value from the dorpdown list. This value is based on the attribute chosen for the filter.

e. Choose OK.

→ Recommendation

Provide the filter description in Title or Sub-Title to make filter criteria understandable in the Process Workspace.

→ Tip

- To modify the Title and Sub-Title of a performance indicator, choose the performance indicator from the list and choose *Edit Performance Indicator*.
- To delete a performance indicator, choose the performance indicator from the list and choose *Delete Performance Indicator*.
- To modify the properties of a performance indicator, choose the performance indicator from the list and modify the required property in the details section.
- To change the order of performance indicators representation in the Process Workspace, choose *Order Performance Indicators*. You can use the navigation icons to reorder the performance indicators.

8. Choose Save.

Sample Performance Indicators

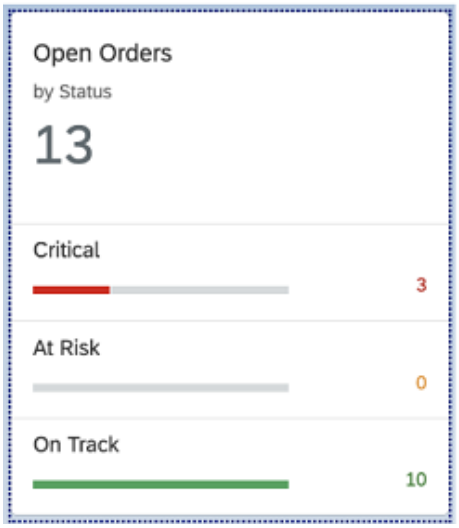
This topic provides a list of examples to help you understand the configuration of performance indicators and how this is represented in the Process Workspace.

Open Orders by Status

This example shows how to display a measure (Number of Instances) grouped by a dimension (Status) in a bar chart. The performance indicator configuration is as follows:

- Title - Open Orders
- Sub-Title - by Status
- Representation - Bar Chart
- Measure - Number of Instances (SC_Number_Of_Instances)
- Dimension - Status (SC_Status)
- Filters- State (SC_State) is equal to Open

The pictorial representation of the example is as follows:



Open Orders by Phase

This example shows how to display a measure (Number of Instances) grouped by a dimension (Active Phases) in a list.

- Title - Open Orders
- Sub-Title - by Phase
- Representation - List
- Measure - Number of Instances (SC_Number_Of_Instances)
- Dimension - Active Phases (SC_Active_Phases)
- Filters- State (SC_State) is equal to Open

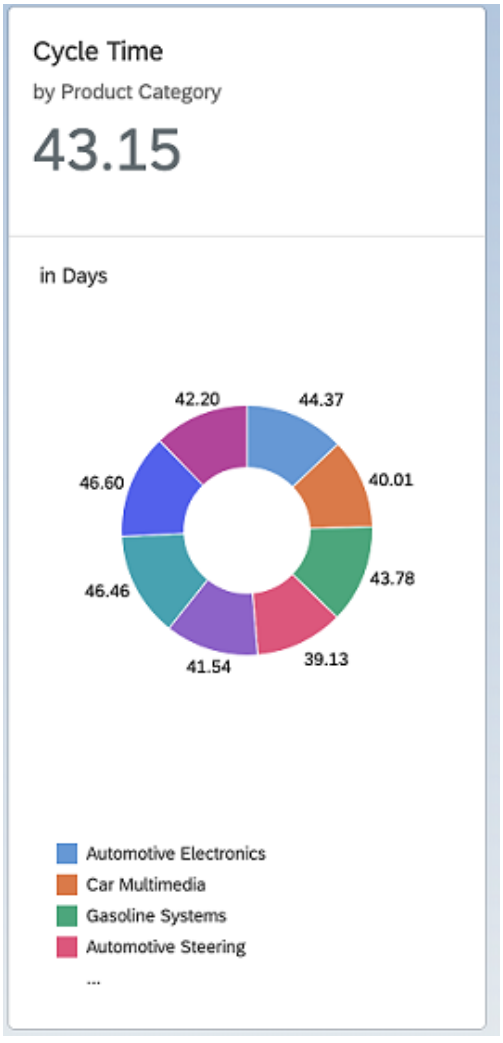
The pictorial representation of the example is as follows:

Cycle Time by Product Category

This example showcases how to display a measure (Cycle Time) grouped by a dimension (Product Category) in a donut chart.

- **Title** - Cycle Time
- **Sub-Title** - by Product Category
- **Representation** - Donut Chart
- **Measure** - Cycle Time (SC_Elapsed_Time)
- **Dimension** - Product Category
- **Filters**- State (SC_State) is equal to Completed

The pictorial representation of the example is as follows:

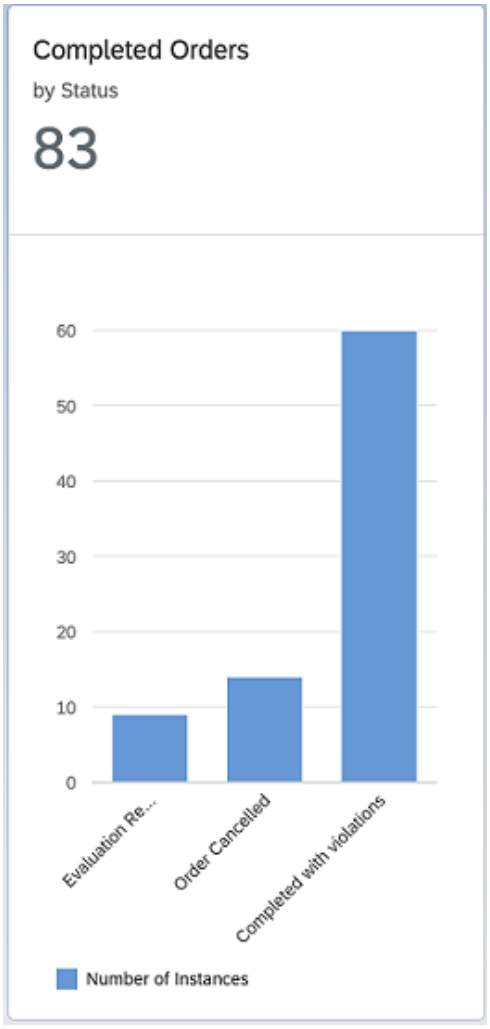


Completed Orders by Status

This example showcases how to display a measure (Cycle Time) grouped by a dimension (Status) in a column chart.

- **Title** - Cycle Time
- **Sub-Title** - by Status
- **Representation** - Column Chart
- **Measure** - Cycle Time (SC_Elapsed_Time)
- **Dimension** - Status (SC_Status)
- **Filters**- State (SC_State) is not equal to Open

The pictorial representation of the example is as follows:



Orders Without Design Evaluation

This example showcases how to display a measure (Number of Instances) in the header area of the card.

The example also describes how to use calculated attributes in performance indicators. In this example, **No Design Evaluation** is a calculated attribute of aggregation role Measure and Data Type Integer. It is evaluated to 1 if the design evaluation has not occurred, or 0 if the design evaluation has occurred. Apply a filter on **No Design Evaluation** to filter out instances where there is no design evaluation.

Select the measure for this performance indicator as **No Design Evaluation**, to count the number of instances where **No Design Evaluation** is equals 1.

- **Title** - Orders without Design Evaluation

- **Representation** - Header
- **Measure** - Number of Instances (SC_Number_Of_Instances)
- **Filters**- No Design Evaluation is equal to 1

The pictorial representation of the example is as follows:

Completed Orders by Product Category

This example showcases how to display measures (Cycle Time, Number of Instances) grouped by a dimension (Status) in a table card.

- **Title** - Completed Orders
- **Sub-Title** - by Product Category
- **Representation** - Table
- **Measure** - Product Category, Cycle Time (SC_Elapsed_Time), Number of Instances (SC_Number_Of_Instances)
- **Dimension** - Status (SC_Status)
- **Filters**- State (SC_State) is equal to Completed

The pictorial representation of the example is as follows:

Import a Visibility Scenario

Import a visibility scenario from your local system.

Prerequisites

- You've created a business process project. For more information about how to create a business process project, see Create and Manage Projects.
- You've exported a visibility scenario (.zip file) from SAP Workflow Management. Your visibility scenario is saved on your local system.

Context

Import an exported visibility scenario (.zip file) in SAP Build Process Automation.

Procedure

1. In the **Overview** tab of your business process project, choose **Import** > **Visibility Scenario**.
2. In the **Import Visibility Scenario** dialog box, click **Browse** to search for the visibility scenario (*.zip) file in your local system.

   Note

   While importing the visibility scenario (*.zip) file from your local system, the .zip file is validated and then, the following happens:

   - The destination in phases, attributes, and actions of the visibility scenario is reset. You need to configure the destination using an **Environment Variable** of type **Destination**.
   - The metadata of the visibility scenario is updated.
   - Any errors that exist inside the visibility scenario are displayed in the design console.

3. Select the file, then click **Import**. This action imports and opens the visibility scenario. The visibility scenario is automatically locked to prevent other users and applications from modifying it.

   Note

   You might encounter an unexpected error during the import of a visibility scenario for the following reasons.

   - A zip file doesn't contain the scenario model.
   - The scenario model contains special characters.
   - The visibility scenario already exists.

   Note

   The deployment of the project fails if you import the same visibility scenario into another business process project.

Gain Visibility Using Process Workspace

You can use **Process Workspace** to gain visibility on end-to-end processes for line-of-business users to understand the performance of the processes.

You can track processes in real-time, search and filter instances, analyze performance indicators, and view detailed information about an instance. This gives you a holistic view and insight into critical business-related situations. It also enables you to take corrective action to move the process forward. Continuous real-time analysis enables you to observe and act promptly to ensure that your business operations run smoothly.

Prerequisites

- You have the `AdvancedUser` and `PVOperator` roles assigned to your role collections.

  To have restricted access within a specific scenario, use the `PVRestrictedOperator` role instead of `PVOperator`.

- You have access to the **Process Workspace**. You can access the process workspace in the following ways:
  - You can use the dashboard icon for the respective visibility scenario within the **Monitoring** > **Visibility Scenarios** tile. For more information, see Visibility Scenarios.
  - You can configure the **Process Workspace** on SAP Build Work Zone. For more information, see Configure SAP Build Work Zone Content.

Context

By accessing the **Process Workspace**, you can view a list of visibility scenarios under the **Scenarios** table. To gain visibility on a visibility scenario, choose the visibility scenario from the list. The following are the key features of the **Process Workspace**:

Key Features

- View performance indicators to understand the performance of the process and gain entity level insights from the **Process Workspace**. Based on the relationship defined while configuring entities, you can drill down to view data and associated details. For more information, see Add Entities.
- View performance indicators in the context of a filter criteria. To apply global filters for your visibility scenario, choose the *Expand Header* icon. If you are unable to view the filter by default, you can choose to filter the performance indicators using the standard set of filters or extend the filters by choosing the **Adapt Filters** option.
- View instances in the table view grouped by key attributes.
  - View instances that contribute to a performance indicator in the table view grouped by key attributes.
  - Search for instances using a free text search. The search works on all the attributes except for the attributes of the timestamp data type.
  - Filter instances by using filter conditions.
  - Sort instances by an attribute in both ascending and descending order.
  - Download instances using the **Export to Spreadsheet** button based on the filter applied and the attributes chosen by the user.
- Choose the *Cluster View* icon to group instances based on attributes to analyze the process performance. Instances will be classified into groups based on the measure and dimension that you have selected from the dropdown lists. Upon selecting a specific group, you will be redirected to the instances page where the grouped instances are listed.

  Note

  - Currently, only the attributes of type String and Boolean are supported.
  - Grouping instances based on Active Steps, Active Phases, Completed Steps, and Completed Phases is unsupported.

- Navigate to the details view of the scenario instance by choosing an instance from the table. The scenario instance details view provides detailed information about the selected instance. It contains the following sections:
  - **Header**: Displays the attributes that are configured by the user and the actions available for the scenario instance if the defined conditions are satisfied. By default, you can only view the top six attributes. You can choose to expand the header to view the other attributes. For more information about attributes, see Attributes.
  - **Phases**: Displays the progress of an instance in the defined phases. This section provides details such as phase name, status, start time, end time, and duration for the defined phases.

    Note

The **Phases** section is displayed only if at least one phase is configured for a deployed visibility scenario.

- **Path**:
  - Displays the sequence of events for an observed process.
  - Displays a parallel flow when a process or a workflow contains parallel gateway as part of the process execution. For more information, see Parallel Flow in Process Path.
  - Provides step-level insights on choosing a specific step in the path. You can view the following information for a step:
    - **Step Status**: The status of the step.
    - **Cycle Time**: The total time taken to complete the step.
    - **Due On**: The expected date and time for the completion of step.
    - **Processor**: The person responsible for the completion of a user task in the step.
    - **Recipients**: Users and/or the group of users who are authorized to work on the task.
    - **Outcome**: Outcome of the user task from SAP BTP Workflow/SAP Build Process Automation.

      For tasks based on forms in SAP BTP Workflow or SAP Build Process Automation process, the values can be either Approved or Rejected.
    - **Log**: Details on the lifecycle of a step.

    Note

    The referenced subprocess in the process path is represented as a step. You can click on the respective step in the process path to view the associated process flow triggered.
  - You can view the path in both the graphical and the tabular view.
- **Action Logs**:
  - Display the activity logs of actions of the type **Trigger Workflow**.
  - Display the name of the action, the status of the action, the user triggering the action, and the time when the action was triggered.
  - Display the reason for the failure on clicking the failed icon in the action status.

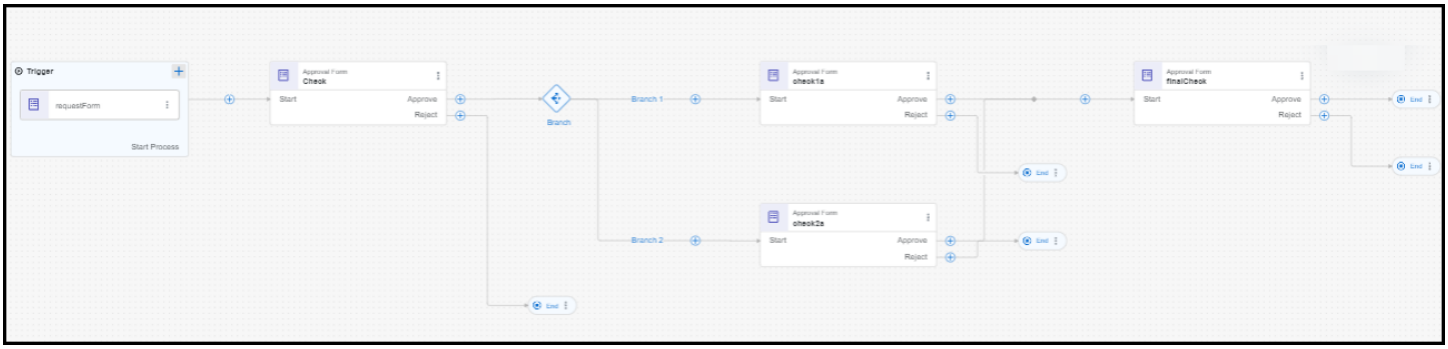
Note

The **Action Logs** section is displayed only if at least one action of the type **Trigger Workflow** is configured for a deployed visibility scenario.

Parallel Flow in Process Path

Processes that have a parallel gateway (modeled using SAP Build Process Automation or SAP Workflow Management) are depicted as parallel flows with two or more branches in the process path.

The following is a sample process containing a parallel flow:



Note

- If the parallel path cannot be depicted as modeled, the path is shown as a sequential flow.
- The following patterns are not supported for parallel flow depiction:
  - Nested parallel flows
  - The workflow model containing unsupported activities like service tasks, mail tasks, script tasks, and so on as part of any parallel flow section.
  - The workflow model containing an exclusive gateway.
  - The workflow model containing a parallel fork followed by an activity that act as a join.

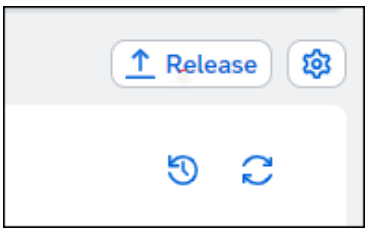
Note

For parallel flow depiction, a workflow model must only have a parallel fork followed by a parallel join. No other activity can act as a join.

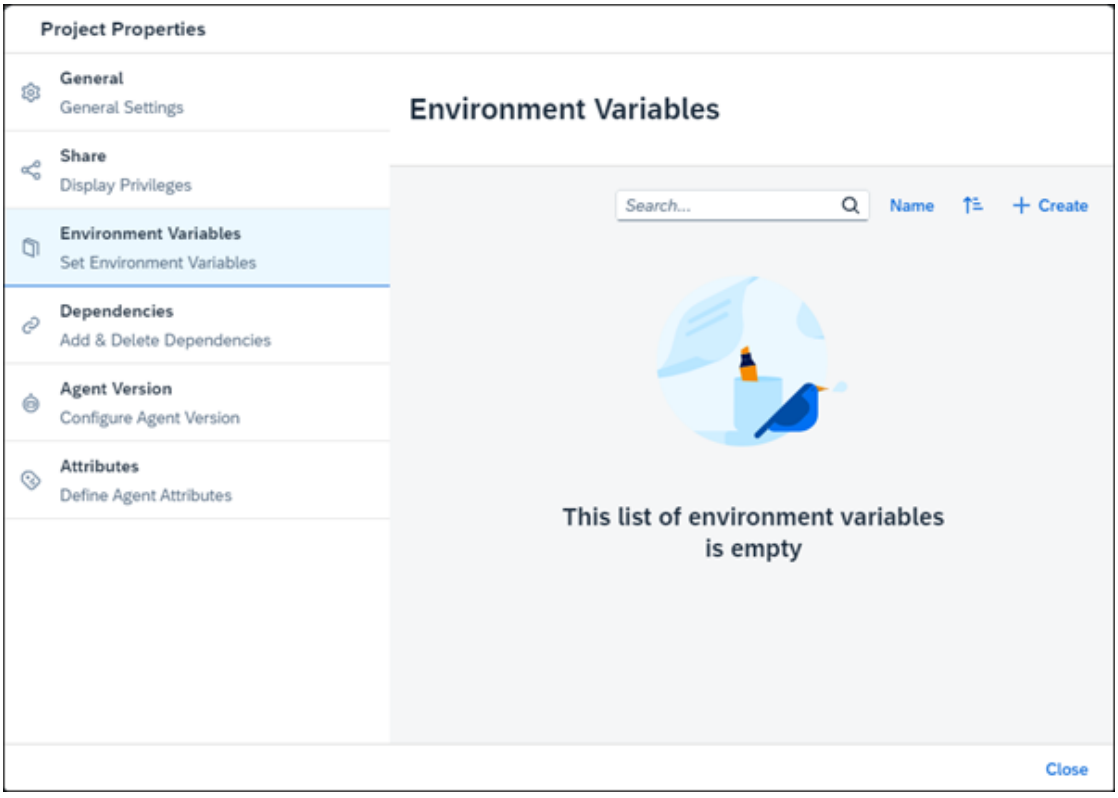
Configure Project Properties

Configure the properties of your business process project in the project Overview.

To configure the project properties, in the project Overview, choose *Manage the project properties*.



When configuring project properties, you have the following options:



The settings under **General** and **Share** are only displayed in this view. You can make changes to these properties in the **Lobby**. For more information, see Manage Project Members .

For more information about configuring project properties, see:

Create an Environment Variable

Project Dependencies

Manage Project Dependencies

Configure Agent Version

Agent Attributes

Agent Attributes

Working on Automation Projects

Deploy your automation projects and configure triggers to run your automation jobs.

Prerequisites

You have created and designed your automation as described in Create and Design Automations.

Context

To run attended or unattended automation jobs with your agent, you must first release and deploy your automation project, and configure a trigger.

Related Information

Release an Automation Project
Add an Automation Trigger to a Project

Release an Automation Project

Prerequisites

If you have tested your automation, your agent is currently in **Test** mode.

If your agent is in **Test** mode, it cannot run attended or unattended automations. To end the **Test** mode, restart your Desktop Agent.

Context

To execute attended and unattended jobs on your project, you must first release it.

Procedure

1. Choose **Release** on the project page.
2. Choose **Release** on the pop-up window.
3. You can see that your project is now labeled as **Released**.

Results

Your project is now released.

Next Steps

To execute attended or unattended jobs, you must deploy the project and add a trigger first.

Deploy an Automation Project

Deploy a project so that it can be run by end users.

Prerequisites

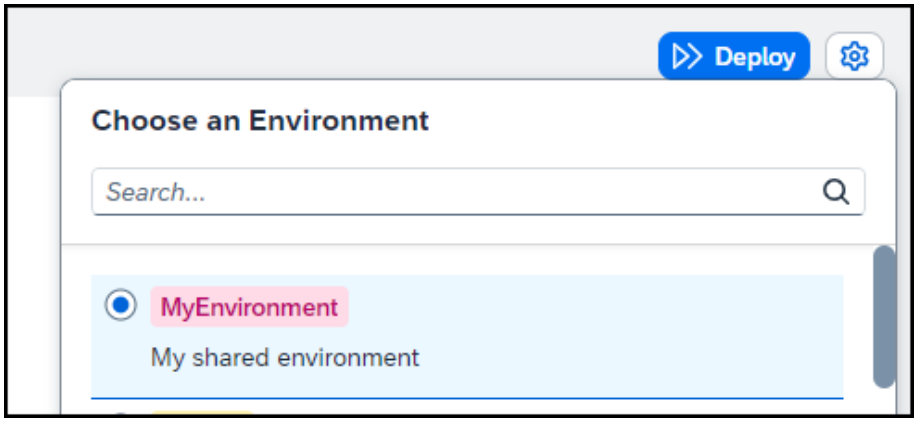
- If you want to deploy your project to a shared environment, you need to have access to this environment. Access is granted by administrators. For more information, see Create an Environment and Share an Environment.
- Your project has been released.

Context

You deploy a released project so that it can be run by end users.

Procedure

1. In the **Project Overview**, select a released version of the project.
2. Choose **Deploy**.
3. In the **Choose an Environment** screen, select the environment in which you want to deploy the project.



4. If you have created variables for your environment, set the values for these in the **Define Variables** screen.

   For more information, see Set Values for Environment Variables

5. Choose **Deploy**.

Results

The version is now deployed. The same project can now be deployed in a different shared environment, regardless of its version.

If you make changes, you can upgrade to a higher product version or redeploy the project.

Related Information

Upgrade
Redeploy a Project

Add an Automation Trigger to a Project

You can add a trigger to a project. A trigger is a rule that defines how and when the agent executes the project.

Prerequisites

You have deployed the project.

If you want to add a scheduled trigger to a project, you need to be assigned the role **ProcessAutomationAdmin** or **ProcessAutomationDeveloper**. For more information, see Authorizations.

Context

You can add one or several triggers to an automation project during project creation or at a later time. The available triggers are:

- **API**: An API trigger opens a dedicated endpoint that allows an external application to execute an automation project using an `HTTP POST` call.
- **Attended**: An attended trigger distributes the deployed automation project to a specific group of agents. An attended trigger is triggered manually from the desktop agent.
- **Scheduled**: A scheduled trigger creates jobs that are based on the schedule you define in the trigger. These jobs run fully automatically.

Create an Attended Trigger

Context

With an attended trigger, the deployed project is distributed to a specific group of agents and the automation can be run manually by the user.

Procedure

1. In SAP Build, choose **Control Tower**, then choose the **Environments** tile.
2. Select your environment and go to the **Attended Trigger** view.
3. Choose **Create an Attended Trigger**.
4. Select a project from your environment and enter a name and a description, if required.
5. Choose **Schedule** and enter a start date. You can also enter an end date but this is optional .
6. Select the time zone.
7. Use the time range picker to select the working time during which the project is made available to the user.
8. Choose **Distribution** and select your distribution restriction policy for the project. If you select **Agents matching attributes**, specify the attributes and the values for your restriction.

Add a Scheduled Trigger

A scheduled trigger does not require the user to start it manually. It creates jobs that are based on the schedule you define in the trigger. These jobs run fully automatically.

Prerequisites

You need to be assigned the role **ProcessAutomationAdmin** or **ProcessAutomationDeveloper** in order to be able to add a scheduled trigger. For more information, see Authorizations.

Procedure

1. In SAP Build, choose Control Tower, then choose the Environments tile.
2. Select your environment and go to the Unattended Trigger view.
3. Choose Add Trigger choose a project, and choose Next.
4. Choose Scheduled and Next.
5. Configure the trigger settings and optionally, choose Input Parameter, and Distribution to configure the agent attributes.
6. Optionally, deactivate New job.

   Note

   A new job is not created until a job with the same input parameters has been completed on the same trigger.

7. Optionally, activate Broadcast.

   You can broadcast an automation to run on multiple eligible agents coming from the same trigger. The broadcast feature allows you to create multiple jobs for one trigger. Once the automation is triggered, jobs are distributed to all eligible agents and executed. An agent is eligible if it meets the following criteria:

   - Agent is declared in environment or agent management.
   - Agent is connected in unattended mode.
   - Agent version can execute an automation.
   - Agent attributes are defined at the trigger level.

   Caution

   If you create or update a trigger with Broadcast enabled without the attribute distribution policy, it can lead to massive deployment.

8. Choose Create.

Add an API Trigger

Context

A trigger of type API opens a dedicated endpoint that allows an external application to start the execution of an automation in a specified deployed project via an HTTP POST call. As such, an API trigger is always attached to one automation in one deployed project.

Note

You can add, modify or remove a trigger of type API at any time after the creation of a deployed project, providing flexibility to open or remove new channels at any point of time.

Procedure

1. In SAP Build, choose Control Tower, then choose the Environments tile.
2. Select your environment and go to the Unattended Trigger view.
3. Choose Add Trigger choose a project, and choose Next.
4. Choose API and Next.
5. Configure the trigger settings and optionally, the agent attributes.

   Note

   The maximum expiration time is limited to 30 days or 4 weeks. Once the expiration time is passed, the trigger appears in Expired status. The job will be expired in one of the following scenarios:

   - the job is not distributed within the expiration window.

- the job is running but doesn't respond within the expiration time, followed by a timeout period and a 30-second delay.

6. Choose **Create**. You can now **Copy** the API trigger information and **Close**.

Execute an API Trigger

Configure and test an API trigger.

Prerequisites

You have a service key from the SAP Business Technology Platform (BTP).

Procedure

1. Create a new POST request in Postman with the trigger URL.
2. Define the payload in the body using the json sample from the API trigger definition.

   If there is no JSON sample when you create the trigger, add at least an empty object {} in Postman.
3. Provide a trigger token or API key: For deprecated triggers, add a trigger token header to your request and set the generated token as its value. For all other triggers, add an API key header to your request and set an API key of your environment as its value.
4. Go to the **Authorization** tab and choose type **OAuth 2.0**.
5. Click **Get New Access Token**.
6. Choose **Client Credentials** from the **Grant Type** dropdown.
7. Enter the client ID, the client secret, and the access token URL.

   To get the access token URL, copy the URL property in the service key and add "/oauth/token".

Sample Code

```
{
        "uri": "https://tenant-url... .com",
        "uaa": {
                "uaadomain": "authentication.sap.hana.ondemand.com",
                "tenantmode": "shared",
                "sburl": "https://uaa-domain... .com",
"clientid": "abcdef!a12345|xyz!a1234", <<---- this is the client ID
                "verificationkey": "-----BEGIN PUBLIC KEY----- ... -----END PUBLIC KEY-----",
                "apiurl": "https://api.authentication-domain... .com",
                "xsappname": "abcdef!a12345|xyz!b1234",
                "identityzone": "identity-zone-name",
                "identityzoneid": "identity-zone-id",
"clientsecret": "xxxxxxxxxx=", <<---- this is the client secret
                "tenantid": "tenant-id",

        "url": "https://uaa-url-domain-.. .com" <<---- this is the url property



                }
}
```

8. Request the token.
9. Choose **Use Token** to add the token to your request.
10. Execute the request.

Results

A job is scheduled. In the **Responses** field of the public API, the *<jobUid>* variable is returned to execute an API trigger.

Related Information

Creating Service Keys in Cloud Foundry

Create an Action Project

Create an Action project to encapsulate APIs as actions in your business scenarios.

Prerequisites

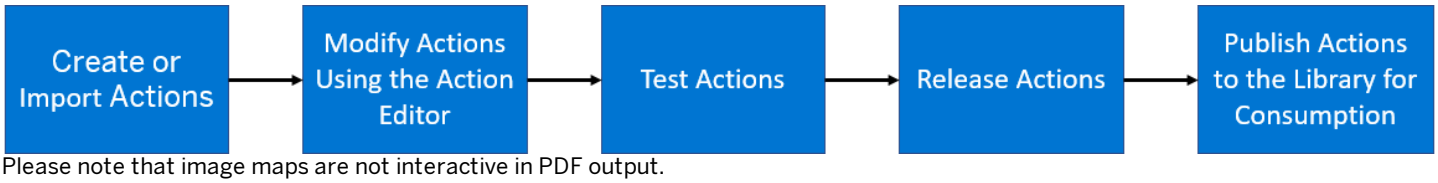
You've read and understood the restrictions of the **Actions** project. For more information, see Restrictions on the Actions Project.

Context

You can embed external skills and capabilities into your SAP Build Process Automation projects using the **Actions** . The **Actions** project contains specific action artifacts that can be used in managing your business processes.

An Overview of the Actions Project Lifecycle

This image is interactive. Hover over each area for a description. Click highlighted areas for more information.



Please note that image maps are not interactive in PDF output.

Procedure

1. In the navigation pane, choose **Connectors** > **Actions**.
2. Choose **Create**.
3. Select the following options from the **Choose an API Source** window that appears:
    a. In the **Live API** section, the following options are available:
        - Choose **Graph** , choose a specific instance, and click **Next**. For more information, see Using Graph.
        - Choose **SAP Cloud Application Programming Model**, if you want to consume APIs from the SAP Cloud Application Programming Model (CAP). For more information, see Using SAP Cloud Application Programming Model.
        - Choose **ABAP RESTful Application Programming Model**, if you want to consume APIs from the ABAP RESTful Application Programming Model (RAP). For more information, see Using ABAP RESTful Application Programming Model.
        - Choose **OData Destinations**, if you want to consume OData APIs. For more information, see Using OData Destinations.
        - Choose **SAP Systems**, if you want to consume APIs from the different SAP applications. For more information, see Using SAP Systems.
    b. In the **API Specification** section, following options are available:
        - If you select **Business Accelerator Hub**, choose a specification from the list of available standard APIs and choose **Next**. For more information, see Using SAP Business Accelerator Hub.

- If you select **Upload API Specification** , drag and drop your import file into the import box. You can also select **Browse Files** to find and open your import file and choose **Next**. For more information, see Uploading API Specifications.

4. In the **Create an Action** project dialog box, enter the **Project Name** and a **Short Description**.
5. Choose **Create**.

   The action editor opens immediately after creating a new Action project.
6. The dialog box for selecting actions appears. Choose one or more Actions from the list.
7. Choose **Add**.

Using Graph

Using the Graph destination, you can create an Action project based on the consumption of business data within a single semantically connected data graph. For more information about Graph, see What Is Graph?.

Prerequisites

- You've subscribed to the SAP Integration Suite and activated the Graph capability. For more information, see Activating and Managing Capabilities.
- You've created a Graph service instance and downloaded a service binding. For more information, see Create a Graph Service Instance.
- You've copied the `clientid`, `clientsecret`, and token service URL from the service binding. This information is required for creating a destination.
- You've created a destination for the Graph instance and added the following additional properties. For more information, see Create a Destination.

Destination Configuration

Field	Value
Name	Enter the name of your choice.
Type	HTTP
Description	Enter an optional description for your destination.
URL	URL of the SAP Graph instance.
Proxy Type	Internet
Authentication	OAuth2JWTBearer
Client ID	Client ID found in the service binding. For more information, see Create a Graph Service Instance.
Client Secret	Client Secret found in the service binding. For more information, see Create a Graph Service Instance.
Token Service URL Type	**Dedicated**
Token Service URL	`https://<url-retrieved-from-service-binding>/oauth/token?grant_type=client_credentials`

Additional Properties

Property	Value
`AppgyverEnabled`	`true`

Property	Value
`sap.applicationdevelopment.actions.enabled`	`true`
`sap.processautomation.enabled`	`true`
`sap.build.usage`	GRAPH

Context

The Graph instance allows you to consume all business data in the form of a semantically connected data graph accessed via a single unified API.

Procedure

1. Navigate to **Connectors** > **Actions**.
2. Choose **Create** > **Choose an API Source** > **Live API**.
3. Choose **Graph**.
4. Choose the graph instance, system, and package that best suits your subscription.
5. Choose **Next**.
6. In the **Create an Action** project dialog box, enter the **Project Name** and a **Short Description**.
7. Choose **Create**.

> **→ Remember**
> The URL prefix and destination are automatically filled in with the chosen Graph destination.

Using SAP Cloud Application Programming Model

Use SAP Cloud Application Programming Model (CAP) as a destination to create an Action project.

Prerequisites

You've created a destination in SAP BTP cockpit for the SAP Cloud Application Programming Model and added the following additional properties:

Property	Value
`sap.applicationdevelopment.actions.enabled`	`true`
`sap.build.usage`	CAP

For more information about creating and managing destinations, see Managing Destinations for the Action Project.

Context

If youneed to consume APIs from the SAP Cloud Application Programming Model, you can use a destination approach. This approach requires the creation of a destination in the SAP Business Technology Platform (BTP) cockpit to establish the connection. Once the destination is established, you can usean Action project to consume APIs from the SAP Cloud Application Programming Model.

> **ⓘ Note**

- When the CAP service is a REST service and not an OData service, $metdata doesn't exist, so the REST-based CAP service can't be consumed.

- Regularly review destinations and update them as needed, to ensure they remain secure and effective for your Action project.

Procedure

1. Navigate to **Connectors** > **Actions**.
2. Choose **Create** > **Choose an API Source**.
3. In the **Live API** section, choose **SAP Cloud Application Programming Model**.
4. Search for or select the required destination.
5. Choose **Next**.
6. In the **Create an Action** project dialog box, enter the **Project Name** and a **Short Description**.
7. Choose **Create**.

Using ABAP RESTful Application Programming Model

Use ABAP RESTful Application Programming Model (RAP) as a destination to create an Action project.

Prerequisites

You've created a destination in the SAP BTP cockpit for the RAP and added the following additional properties:

Property	Value
`sap.applicationdevelopment.actions.enabled`	`true`
`sap.build.usage`	RAP

For more information about creating and managing destinations, see Managing Destinations for the Action Project.

Context

RAP defines the architecture for efficient end-to-end development of intrinsically SAP HANA-optimized OData services (such as Fiori apps). It supports the development of all types of Fiori applications as well as publishing Web APIs. It's based on technologies and frameworks such as Core Data Services (CDS) for defining semantically rich data models and a service model infrastructure for creating OData services with bindings to an OData protocol and ABAP-based application services for custom logic and SAPUI5-based user interfaces. For more information, see SAP - ABAP RESTful Application Programming Model.

Once the destination has been created in the same subaccount where SAP Build Process Automation has been subscribed to, the Action project consumes all the exposed APIs from the RAP.

Procedure

1. Navigate to **Connectors** > **Actions**.
2. Choose **Create** > **Choose an API Source**.
3. In the **Live API** section, choose **ABAP RESTful Application Programming Model**.
4. Search for or select the required destination.
5. Choose **Next**.
6. In the **Create an Action** project dialog box, enter the **Project Name** and a **Short Description**.
7. Choose **Create**.

Using OData Destinations

Create an Action project using other SAP BTP destinations like OData to consume OData APIs.

Prerequisites

You've created an SAP BTP destination and added the following additional properties:

Property	Value
`sap.applicationdevelopment.actions.enabled`	`true`
`sap.processautomation.enabled`	`true`
`sap.build.usage`	`odata_gen`

For more information about creating and managing destinations, see Managing Destinations for the Action Project.

Context

If you're consuming OData APIs in your Action project, then you need to establish a connection between the OData system and the SAP BTP subaccount using destination.

Note

It is expected that customer extensions are available as part of the API definition for consumption in Action when using this option.

Procedure

1. Navigate to the **Connectors** > **Actions**.
2. Choose **Create** > **Choose an API Source**.
3. In the **Live API** section, choose **OData Destinations**.
4. Search or select the required destination.
5. Choose **Next** and proceed with project creation.

Using SAP Systems

Create an Action project using SAP Systems to consume APIs from SAP applications, such as SAP SuccessFactors, SAP Ariba, and SAP S/4HANA cloud, that are registered with a global account in SAP BTP.

Prerequisites

- You've added an SAP system and then registered it to establish a connection between the SAP system and SAP BTP. For more information about adding a system, see Registering an SAP System.
- You've included your registered systems in a formation and selected **Integration with SAP Build** as the formation type under the **General Information** section and selected the subaccount where SAP Build Process Automation is subscribed. For more information about creating a formation, see Including Systems in a Formation.

Context

Adding a system to the system landscape list in a global account of SAP BTP is essential for making it available as an API source for creating Action projects. To access the system's APIs that are exposed, you must register and include the system in a formation. Only then can you view all the APIs of the registered system.

Among the SAP systems available, there are options such as SAP S/4HANA Cloud, SAP SuccessFactors, and SAP Commerce Cloud.

Note

- It is expected that customer extensions are available as part of the API definition for consumption in Action when using this option.
- The **SAP Systems** option is not available in the CN40 data center subscribed service when selecting an API source for Action project creation.

Procedure

1. Navigate to **Connectors** > **Actions**.
2. Choose **Create** > **Choose an API Source**.
3. In the **Live API** section, choose **SAP Systems**.
4. Search for or select your registered system.
5. Select the relevant API bundles.
6. Choose **Next**.
7. In the **Create an Action** project dialog box, enter the **Project Name** and a **Short Description**.
8. Choose **Create**.

Next Steps

To test the API actions consumed from the SAP System, create a separate destination for the respective system in the SAP BTP subaccount. For example, if you are consuming APIs from the SAP S/4HANA Cloud system, see Create a Service Instance to Consume the SAP S/4HANA Cloud APIs.

Using API Business Hub Enterprise

Use the API Business Hub Enterprise to consume APIs from a centralized API catalog managed by your organization.

Prerequisites

- You've access to the API Business Hub enterprise application. For more information about the set-up, see Set Up API business hub enterprise Application Using the Standalone Tile or you can set up the API business hub enterprise capability from the Integration Suite launchpad. To set up an API business hub enterprise from Integration Suite, see Setting Up API Management Capability from Integration Suite.
- You've generated a service key and copied the `url` (application URL), `clientid`, `clientsecret`, and `tokenUrl`. For more information about creating a service key, see the API Access Plan for API Business Hub Enterprise.
- You've created a destination for API Business Hub Enterprise and added the following additional properties. For more information, see Managing Destinations for the Action Project.

Destination Configuration

Field	Value
Name	Enter the name of your choice.
Type	HTTP
Description	Enter an optional description for your destination.
URL	URL of the API Business Hub Enterprise application.
Proxy Type	Internet

Field	Value
Authentication	OAuth2ClientCredentials
Client ID	Provide the value of the `clientid` field from the service key you created. For more information, see Creating Service Key.
Client Secret	Provide the value of the `clientsecret` field from the service key you created. For more information, see Creating Service Key.
Token Service URL Type	Dedicated
Token Service URL	Provide the value of the `url` field from the service key you created. **Example** `https://<url-retrieved-from-service-binding>/oauth/token`

Additional Properties

Property	Value
`sap.applicationdevelopment.actions.enabled`	`true`
`sap.build.usage`	`apihub_enterprise`

Context

The API Business Hub Enterprise application allows you to explore APIs connected across different landscapes and consume them in the business process via Actions, depending on the business scenario. If you want to consume APIs that are managed by you, you must create those APIs in the API Business Hub Enterprise application.

Note

ODATA services can only be consumed through Actions when your API Business Hub Enterpris application is connected.

Procedure

1. Navigate to Connectors > Actions.
2. Choose Create > Choose an API Source > Live API.
3. Choose API Business Hub Enterprise.
4. Choose the product and select the API that best suits your requirement.
5. Choose Next.
6. In the Create an Action project dialog box, enter the Project Name and a Short Description.
7. Choose Create.

Using SAP Business Accelerator Hub

When creating an Action project, use the standard list of API specifications available in the SAP Business Accelerator Hub.

Context

The SAP Business Accelerator Hub offers a variety of standard APIs that can be consumed directly in your Actions project. By embedding these APIs into your business processes through Actions, you can increase efficiency and minimize the need for manual intervention. With

the ability to easily access and leverage these APIs, you can quickly create, configure, and test your Actions.

> **Note**
>
> The customer extensions of APIs are not available, since Business Accelerator Hub does not provide API definitions for customer-specific APIs, but rather SAP's standard API definitions that can be used by anyone.

Procedure

1. Navigate to the **Connectors** > **Actions**.
2. Choose **Create** > **Choose an API Source**.
3. In the **API Specification** section, choose **SAP Business Accelerator Hub**.
4. Select the relevant product and package from the list of available options.
5. Choose **Next** and proceed with project creation.

Uploading API Specifications

To create an Action project, you can directly upload the API specification file.

Context

This option is helpful if you've already obtained and customized an API specification file. You can easily upload it to be consumed within your business processes through Actions.

Procedure

1. Navigate to the **Connectors** > **Actions**.
2. Choose **Create** > **Choose an API Source**.
3. In the **API Specification** section, choose **Upload API Specification**.
4. Drag and drop the API specification file to upload.

   > **Note**
   >
   - Only EDMX, XML, and JSON files are supported and the file size is limited to 5 MB.
   - Open API specification files with versions 2.x.x and 3.x.x of JSON type are supported.
   - When API specifications are extracted from Business Accelerator Hub, customer extensions will not be available. The customer extensions are expected to be available if the API specification is extracted from a live API (for example, using *</$metadata>* for OData APIs).

5. Choose **Next** and proceed with project creation.

Build API Actions from Scratch

Create API actions from scratch using the action editor.

Context

When developing API actions, the **Build from Scratch** option refers to the creation of an API specification from scratch without referencing any existing templates or predefined structures. In most cases, this approach is beneficial when a unique set of API actions is needed, and the API specification must be customized according to the project requirements or business processes.

Procedure

1. Navigate to the Connectors > Actions.
2. Choose Create > Choose an API Source.
3. In the API Specification section, choose Build from Scratch.
4. Enter a project name and description.
5. Choose Create.

   Upon successful creation of the project, you're navigated to the action editor where you can add custom actions.

Overview of an Action Editor

The Action Editor is where you can create, modify, and control actions.

The Action Editor enables you to view and edit the data blocks that you defined as actions in the editor. It gives you a simplified view of the API specifications. The editor can list all of an object's action artifacts at once. Additionally, it offers details about individual actions, including input and output parameters, as well as testing capabilities.

The action editor makes API operations available to actions while also simplifying input and output values. You can also use the editor to test the operation of an action. To obtain reusable action artifacts, release and publish the Actions project to the library.

Setting Up Actions

To consume APIs from different sources, the following methods are used to set up Actions.

- Use APIs from the Graph, SAP Application Programming Model, and ABAP RESTful Application Programming Model.
- Consume API packages from systems registered with a global account in SAP BTP.
- Use the standard list of APIs available in the SAP Business Accelerator Hub.
- Upload your own API specification file.

> **Note**
>
> To work in the action editor, we recommend that you upload Open API, EDMX, or XML specification file. The API specification is a format for describing REST and OData APIs. You can explain your complete API in an API specification file, which includes endpoints that are available (/instances) and the operations that can be performed on each endpoint (GET, POST). Each action in the actions project has parameters associated with it, such as inputs and outputs.

Properties of an Action

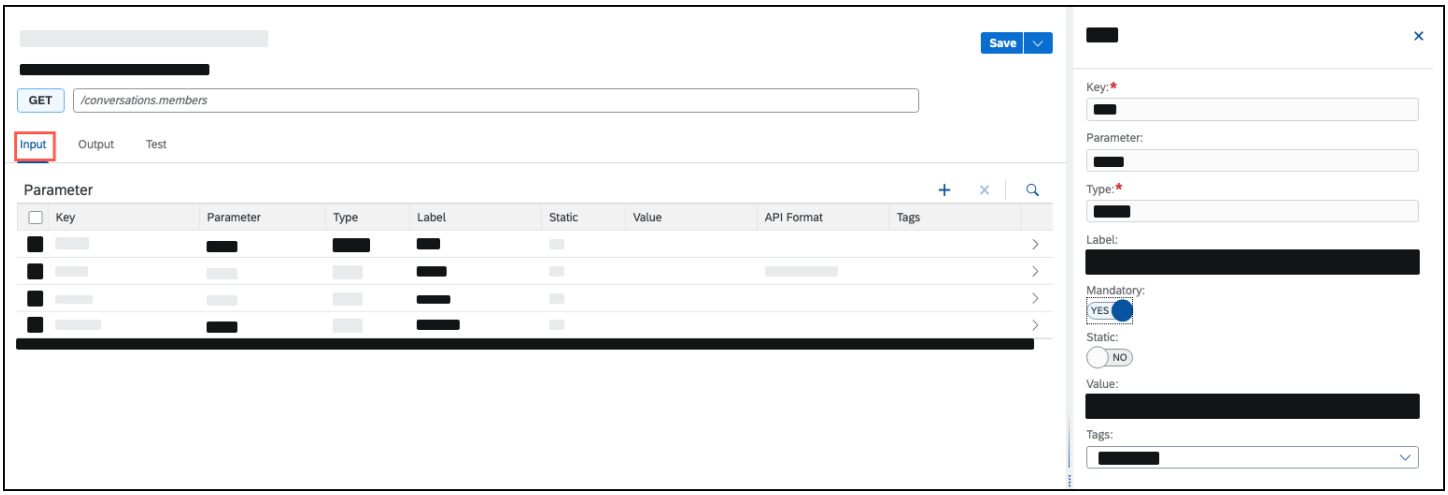
- You can edit the name of an action as per your requirements.
- You can view and customize Input and Output values of API operations.

Properties of Input Parameters

The Input tab contains the following sections:

- Parameter
- Body

You can define the following attributes for Input parameters:



- Label: A name that you define to help you manage different types of parameter.
- Parameter: Determines where the key is located. For example, `query`, `header`, and `path`.
- Key: Unique identifier for a parameter.
- Type: The data type.
- Static: Determine if the request parameter-mapping value is constant.
- Value: Provides instructions for a specific course of action. It's a static or default value used for testing Actions.
- API Format: You can specify an API format for the date and time at the parameter level. It's possible to change the date, time, or date-time format in the parameter.
- Tags: You can assign a list of tags to each input parameter. Input parameter supports multiuse tags.

  Available tags are as follows:

  - $search
  - $top
  - $skip
  - $at
  - $from
  - $to
- Description: Once you've imported the API specification file, you're able to select the APIs you need and receive the associated descriptions field. It helps you make the right decisions for your business and streamline the process.
- Max Length: This parameter property specifies a maximum number of characters.

Properties of Output Parameters

API specification files contain output parameters that are fetched from the response of an API call. Your output attributes determine how these parameters are formatted.

The following attributes are defined for Output parameters:

- Output : Indicates the status of the request in the drop-down. For example, 200, 400, default etc., The status code 200 implies that the request has been successfully processed. As per your requirements, you can modify and create a new HTTP response status code output.
- Key: Unique identifier for a parameter.
- Type: The data type.
- Label: A name that you define to help you manage different types of parameters.

- **API Format**: You can specify an API format for the date and time at the parameter level. It's possible to change the date, time, or date-time format in the parameter.
- **Tags**: It only supports single-use tags, which are array tags.

In the action editor, you can modify input and output parameters. You can also add custom parameters. For more information, see Managing Input and Output Parameters.

Configure Actions Using the Action Editor

When you're in the action editor, the following options are available to you:

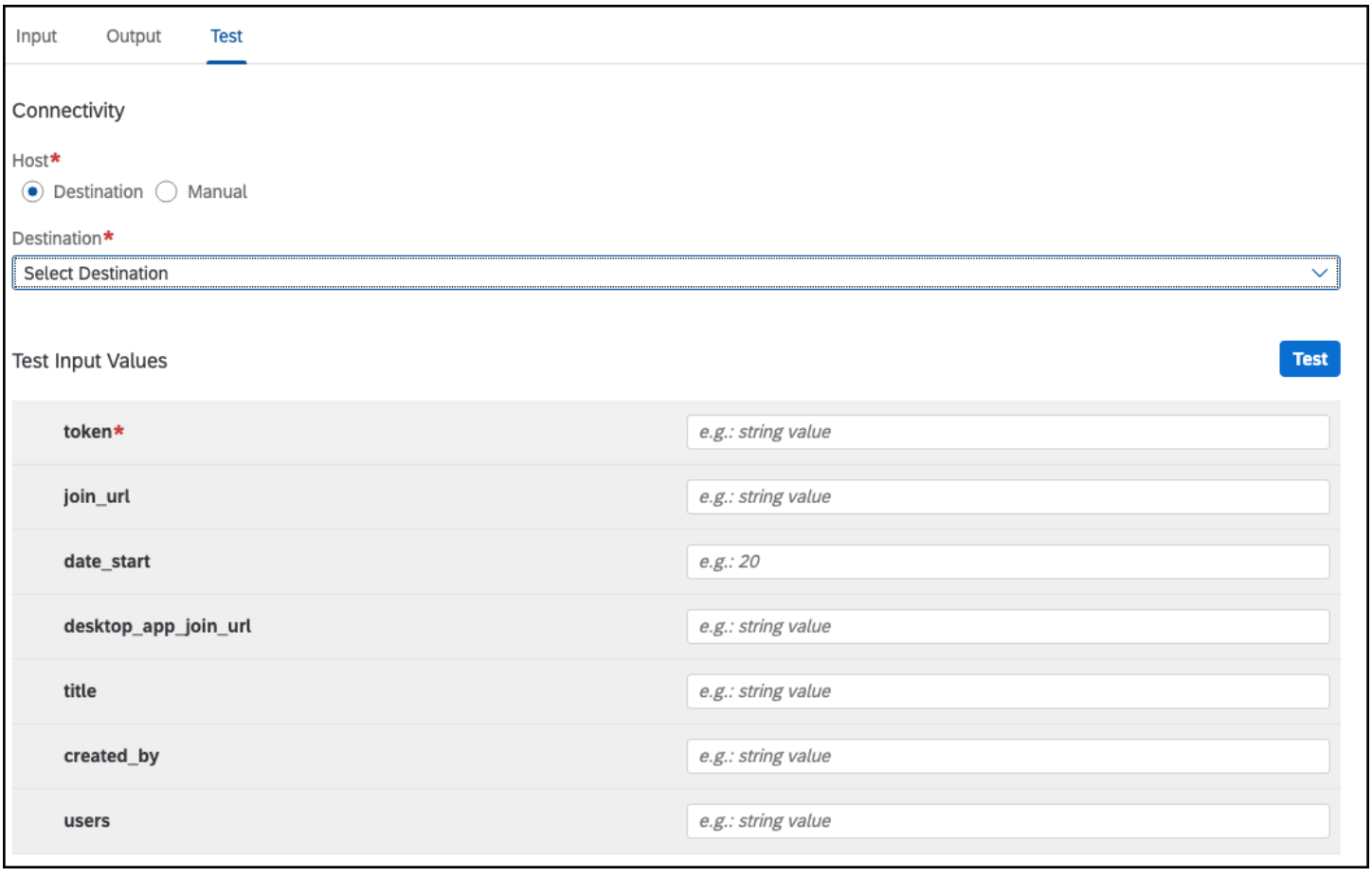
- Upload the API specification file to import a new set of actions.
- Update by adding or deleting the actions. For more information, see Adding Actions.
- Search for, sort, and filter actions.

Updating API Actions

Import a new version of the API specification file to update the API actions with a new set of parameters. For more information, see Updating Actions.

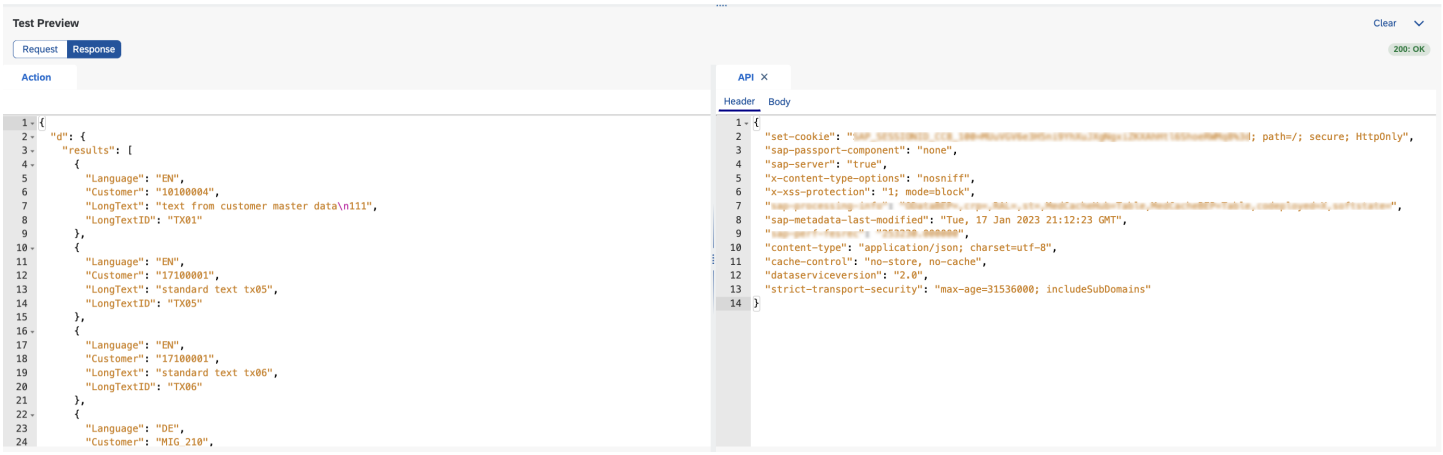
Test API Actions

In the action editor, you can test your API actions using the **Test** console. It's important to test the API actions to better understand and verify the action's behavior before it's consumed. In the **Test** console, you can explore and execute API operations based on the resources associated with the action. For more information, see Testing Actions.



Preview API Actions Request and Response

In the **Test Preview** area in the editor, you can test API requests and view raw responses before releasing them to the library. The preview function can assist you in troubleshooting API requests that time out, create too many calls, or otherwise result in errors.



[Managing Input and Output Parameters]

The Input and Output parameters view allows you to manage parameters in actions. You can perform several operations, such as adding, removing, and showing parameters.

[Adding Actions]

You can add actions from the API specification file into the action editor. You can then modify the action according to your needs.

[Testing Actions]

Use the Test console to test your API actions.

[Updating Actions]

The update feature allows you to modify the Actions Project according to your business requirements.

Managing Input and Output Parameters

The Input and Output parameters view allows you to manage parameters in actions. You can perform several operations, such as adding, removing, and showing parameters.

Prerequisites

You've imported actions into the editor.

Context

After importing an action, you can manage the input and output parameters. Input parameters describe which information is provided to an API before it can perform its function. You can set the output parameters before calling an API because the API gives information to the application in the output parameters. When an API runs without any issues, the application accesses the information returned in output parameters.

You can configure your actions to display only the parameters that you want as input parameters.

The action editor also provides the capability to pass custom parameters. With the custom parameters, you can add additional capabilities beyond those parameters maintained in the API specification.

Procedure

1. In the left navigation pane, choose **Actions**.
2. In the **Parameter** section, choose **Add**.
3. In the **Add New Field**, the following options available to you:
    a. In the **Available Fields** tab, select available parameters from the list and choose **Add**.
    b. To add the new parameter, navigate to **New Field** tab and enter the details.
4. You have the option to include additional fields in the **Body** parameters. Choose **Add** to insert parameters using the following options:

- **Field**: Select this option to add existing fields or create new fields based on your requirements.
- **Fields from Sample JSON**: Select this option if you want to insert fields from a sample JSON file into your action.

Flattening Objects

Using the flattening object feature, you can transform nested objects into flat objects, where all nested objects are transformed into a single key-value pair.

Context

Flattening API objects refers to the process of transforming nested or hierarchical data structures into a flat structure. This can be helpful if you need to simplify data for easier processing in your business process.

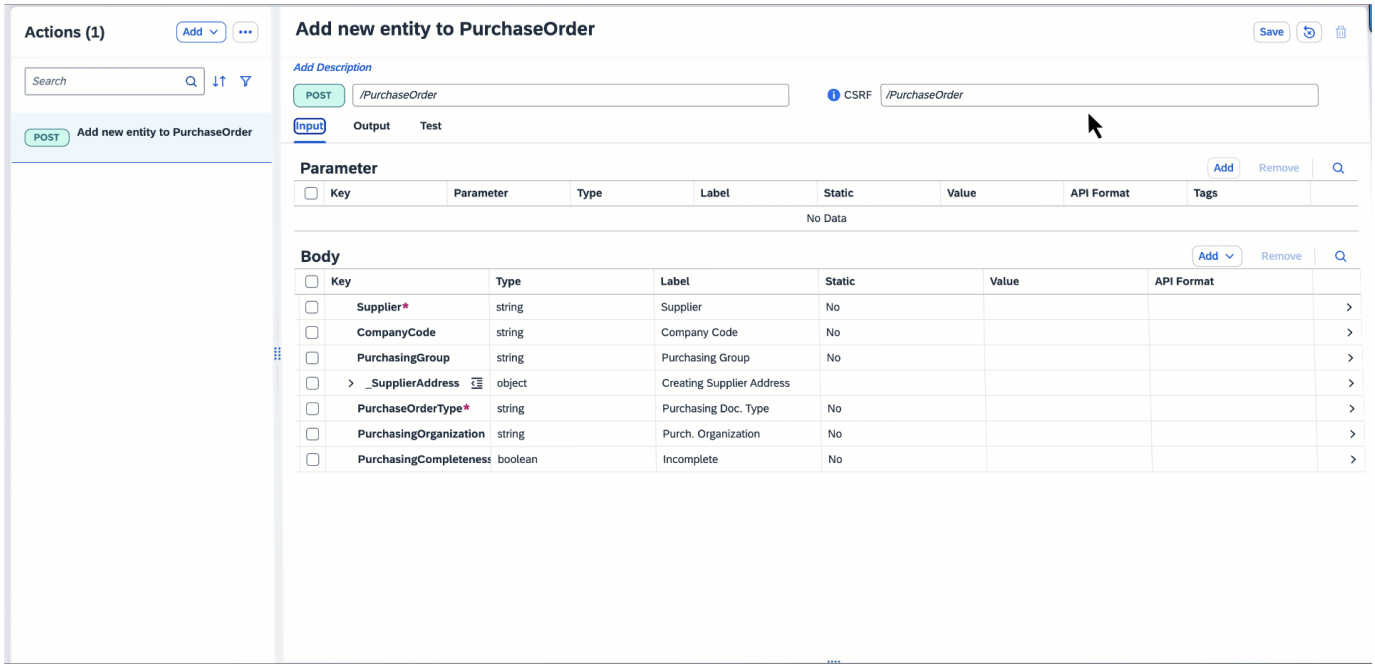
As a result of flattening, citizen developers are able to take advantage of the nested input and output body parameters of your action by moving up the flattened fields one level in the object hierarchy.

Flattened API objects can be tested in the **Test** console just like any other specification. It's important to ensure that your OpenAPI specification accurately represents the flattened structure of API objects. If necessary, update the input or output body parameters section.

The following are steps for flattening API objects:

Procedure

1. Choose either the **Input** or **Output** section according to your API actions.
2. In the **Body** section, select the objects and choose *Flatten Object* icon to flatten the object structure.



Note

The flattening feature is supported for objects only.

Add Fields From the Sample JSON to the Input or Output Parameters

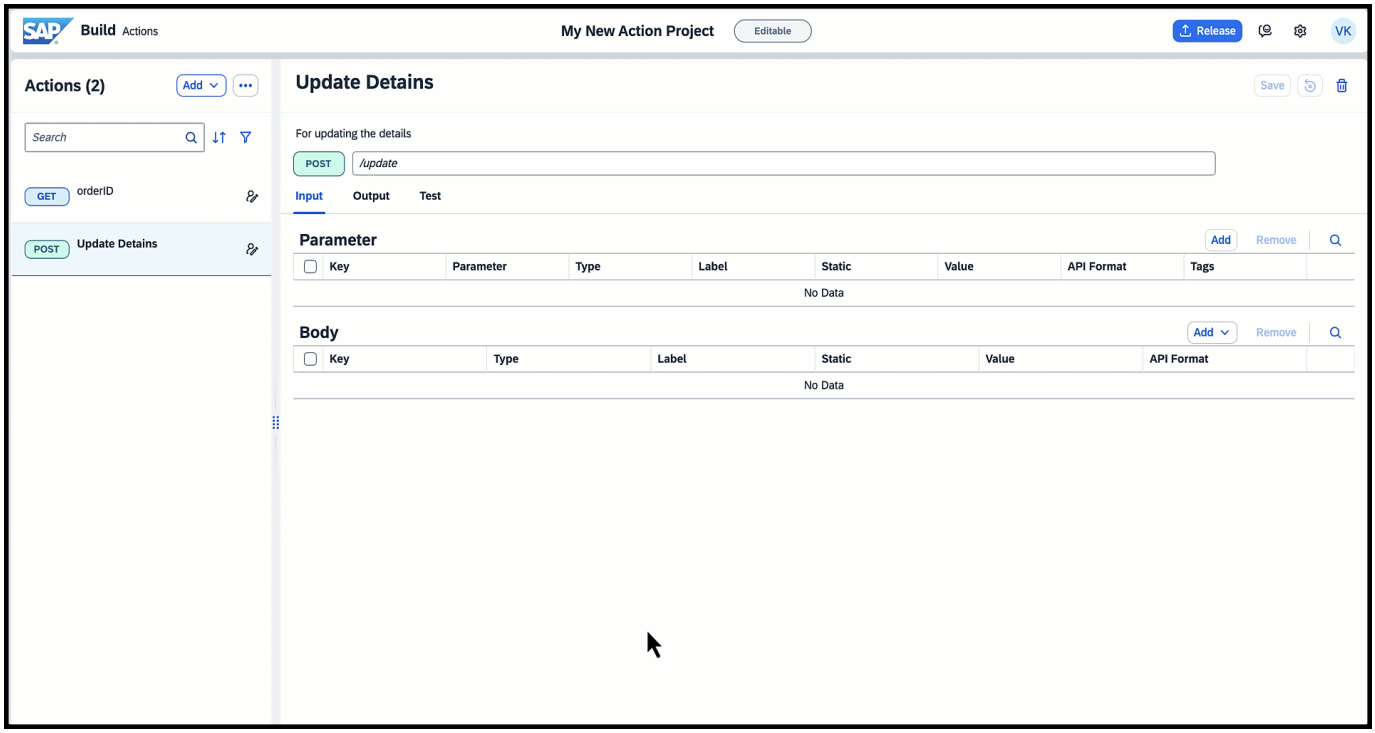
Upload or paste the necessary fields from your own JSON file into relevant actions for managing input/output operations.

Context

When working with custom API actions, it's necessary to manage input and output operations effectively. To simplify this process, you can use a sample JSON file that contains predefined fields and values. This enables you to seamlessly integrate and test your API actions with consistent data.

Procedure

1. Go to the **Body** section and choose **Add**.
2. Choose **Fields from Sample JSON**.
3. In the **Add Fields From Sample JSON** dialog, select **Choose File** to upload the file from your local system or you can just paste it directly into the editor window.
4. Choose **Next** and review the input fields. At this step, A warning message displays if there are any conflicts with the uploaded fields.
5. Choose **Add**.



Related Information

Adding Actions
Testing Actions
Updating Actions

Using Filter Conditions ($filter)

Restrict the data to be retrieved using filter conditions and the filter conditions are validated before being applied.

The actions project supports Open Data Protocol (OData) query parameters, like `$filter`. For various purposes, query parameters are inserted into the URL. We recommend following the OData URI conventions that require the parameters to be passed correctly. The following is one of the system query options.

$filter: A filter is an object that describes a multidimensional extract of data from a data set. Query option **$filter** adds filtering capabilities to the OData service. So that you can filter the OData service feed or collection based on the fields available in the entity type set.

The expression language used in **$filter** operators supports references to properties and objects. Object values can be numbers, strings enclosed in single quotes, or boolean values.

For example, `/Suppliers?$filter=City eq 'Redmond'`. This retrieves all of the suppliers assigned to *City, Redmond*. The **$filter** can support additional operators such as equal, not equal, greater than, less than, etc.

ℹ Note

A filter condition without the filter parameters, such as filter property, filter operator, and filter operand in the request, is not supported.

Procedure

1. Go to the **Action Editor** and select the **$filter** query parameter from your action.
2. Navigate to the right-hand side panel and choose the *Value Help*.
3. Select one of the following options:
    - **Condition Editor**: Choose this interface to define the group that will be applied to the input data. You can set the condition and criteria depending on your requirements. See Using the Condition Editor.
    - **Text Expression**: Choose this interface that enables you to use text expressions to define filters for pattern-matching against the texts and conditions being evaluated. See Using the Text Expression Editor.
4. Choose **OK**.

Using the Condition Editor

Use the condition editor to set conditions.

To add an inner group condition to an existing condition, identify the existing conditions and determine which condition needs to be added. Once the condition has been determined, use the condition editor, or open up the existing group and add the inner group condition. Ensure that the condition input is syntactically correct. Once the condition is saved, test the condition to make sure that it's functioning properly and make changes as needed.

The format of valid filter conditions is as follows:

- Single filter condition:

    ### Example

    ```
    <filter_property_name> <filter_operator> <filter_operand>
    ```

- Multiple filter conditions:

    ### Example

    ```
    <filter_property_name> <filter_operator> <filter_operand> and <filter_property_name> <filter_operat
    ```

Filter Parameter	Values
`<filter_property_name>`	Any of the properties on which the filter condition is allowed. An invalid property name is considered an invalid request.

Filter Parameter	Values
`<filter_operator>`	The following binary filter operators are supported: **Operator** / **Description** `eq` — is equal `ne` — isn't equal `gt` — is greater than `ge` — is greater than or equal `lt` — is less than `le` — is less than or equal
`<filter_operand>`	Valid operands are strings, integer, float, true, false, null, date, datetime, and timestamp. It also supports custom parameters.

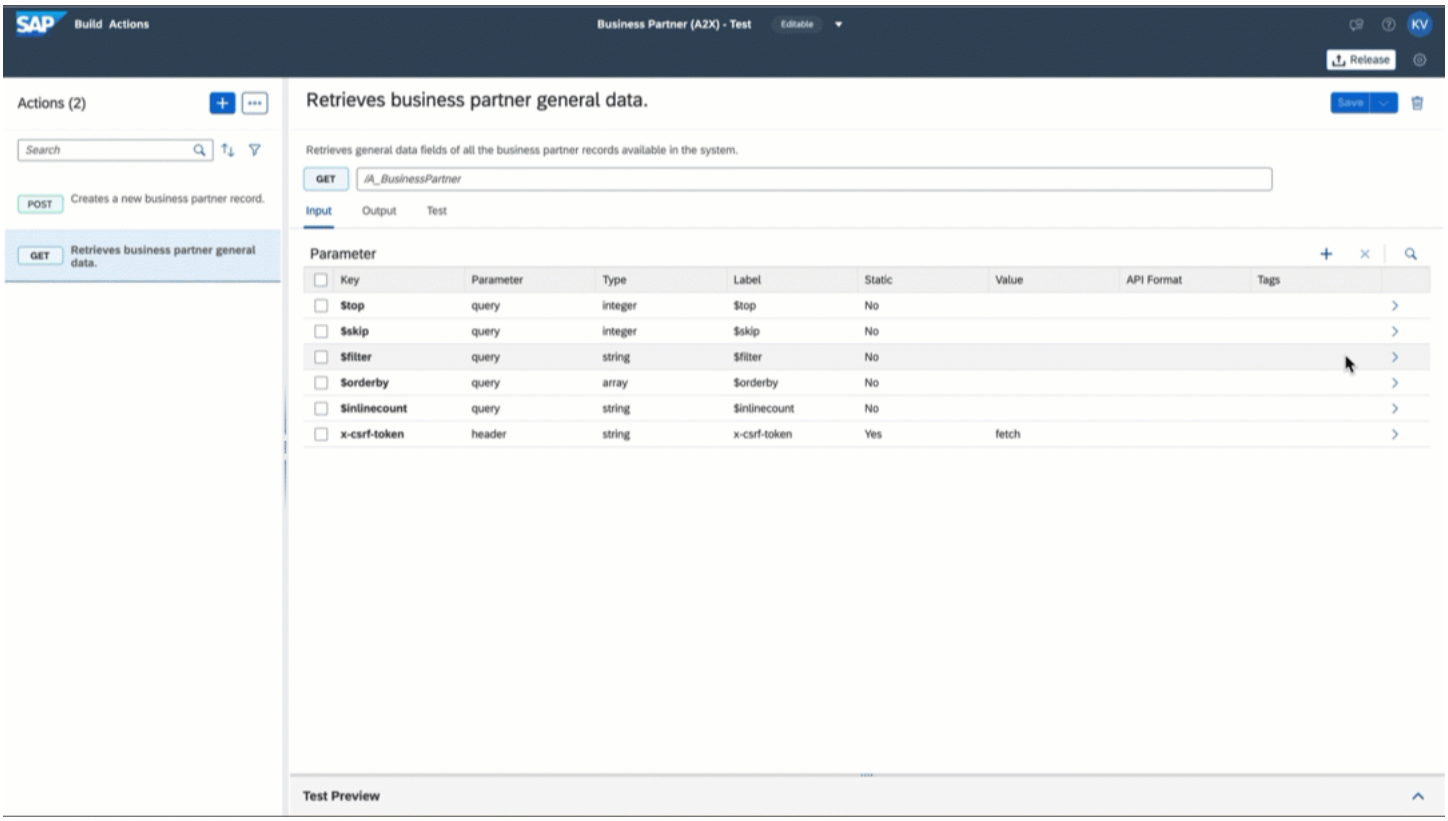
To add a condition:

1. Choose the *Value Help*.
2. Select the **Condition Editor** tab.
3. Choose **Add Group**.
4. Search for or select the conditions, logical operators, and criteria from the dropdown lists.
5. Choose **OK**.

Example

Request URL:

```
/A_BusinessPartner?$filter=((BusinessPartner%20eq%20'51')%20or%20(BusinessPartner%20eq%20'1'))
```



Using the Text Expression Editor

Use the text expression editor to set the filter conditions to retrieve the data.

Text expressions are strings of characters that consist of operators and conditions. They can be used in a variety of ways, including as input to functions, as output, or as part of a larger expression.

In a text expression, some of the valid filter conditions are formatted as follows:

Group	Operator	Details
Conditional AND	`and`	logical And
Conditional OR	`or`	logical Or
Equality	`eq`	equal to
	`ne`	not equal to
Relational	`gt`	greater than
	`ge`	greater than or equal to
	`lt`	less than
	`le`	less than or equal to

For example, the query with filter conditions looks like this:

Example

```
substringof(<'search string'>,<string field>) eq 'true' or endswith(<string field>,<'search string'>) eq
```

To add a text expression:

1. Choose the *Value Help*.
2. Select the **Text Expression** tab.
3. Enter conditions in the editor.

   Example

   ```
   BusinessPartner eq '51' or BusinessPartner eq '1'
   ```

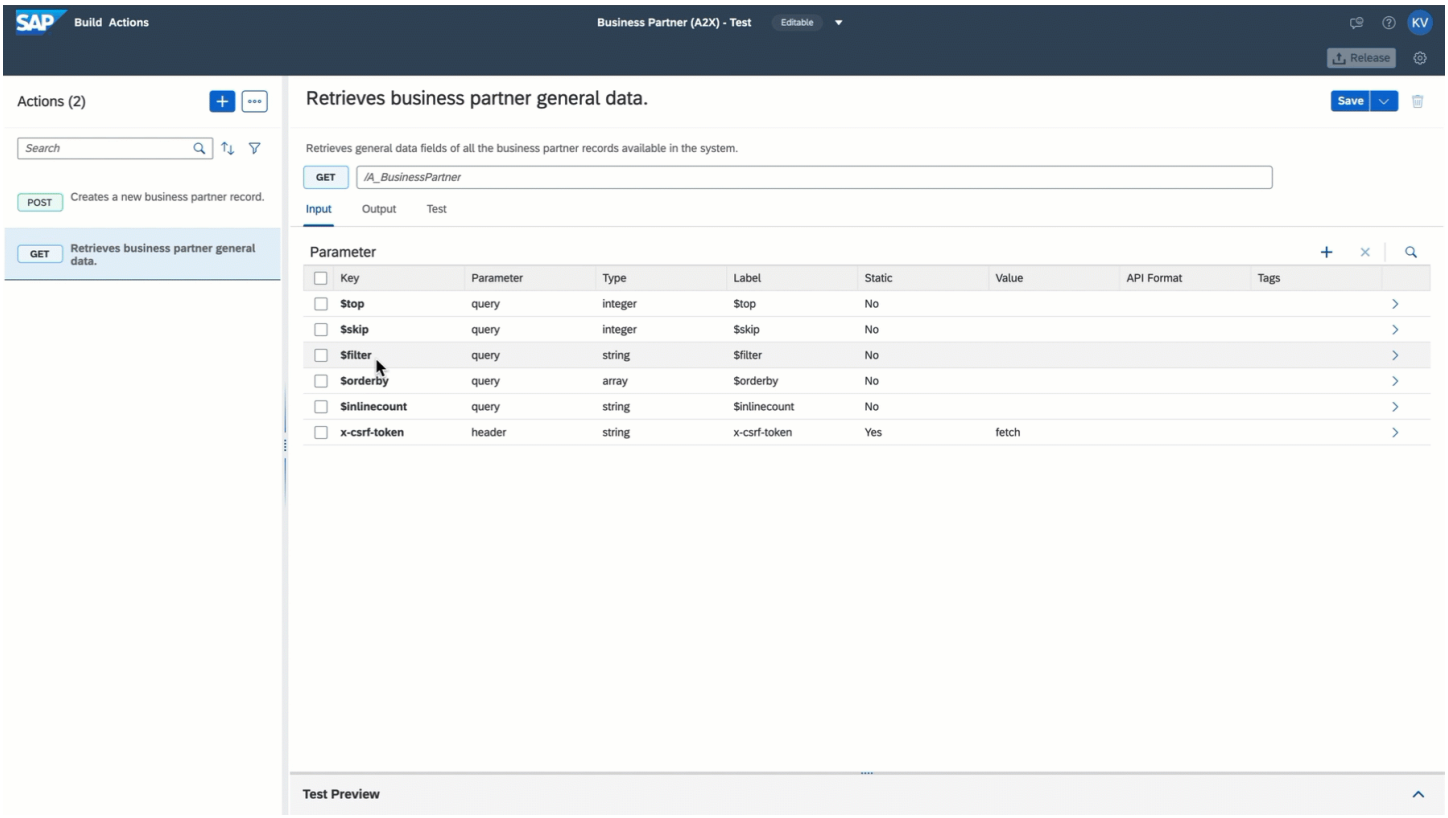
   - Example 1:

     ```
     contains(Location/Address, 'San Francisco')
     ```

   - Example 2:

     ```
     not endswith(Name,'lines')
     ```

4. Choose **OK**.



Adding Actions

You can add actions from the API specification file into the action editor. You can then modify the action according to your needs.

Prerequisites

You've created the Actions Project. See Create an Action Project.

Procedure

1. In the left navigation pane of the action editor, choose **Add**. You can choose between the following options:

   a. Choose **Available Action** to add actions that already exists in the project.

b. Choose **Custom Action** to create an action from scratch. For more information about creating custom action, see Add a Custom Action.

2. In the **Add Actions** dialog, select the actions that you want to add, and choose **Add**.

Related Information

Managing Input and Output Parameters

Add a Custom Action

When creating actions from scratch, you can use a custom type and display it with custom control.

Procedure

1. In the **Add Custom Action** dialog, select the API operation.
2. Enter the display **Name** and optional **Description**.
3. Enter a valid **Endpoint**.
4. Choose **Add**.

Testing Actions

Use the Test console to test your API actions.

Prerequisites

You've created an HTTP destination and added the following additional properties as it's mandatory to test your actions if you're using destinations. For more information, see Create HTTP Destinations.

Property	Value
`sap.applicationdevelopment.actions.enabled`	`true`
`sap.processautomation.enabled`	`true`

Context

The action editor includes a test console where you can test your actions. It's important to test API actions before releasing them for consumption. You can ensure that your API actions are giving the expected response, and if you come across any errors, it helps you to fix them.

> → **Remember**
>
> You can only test on-premise systems with a destination scenario. Testing from an on-premise system requires configuring the cloud connector and using the destination maintenance for the corresponding system or API.

Procedure

1. Select your Action in the action editor and go to the **Test** console.
2. There are two options available in the **Host**:
   - Destination
   - Manual

3. If you choose **Destination**, select the **Destination**'s name from the dropdown menu you created in your SAP BTP subaccount. For more information, see Configure SAP Build Process Automation Destinations for SAP S/4HANA Cloud.

> **Note**
>
> The values for properties `sap-language` and `sap-client` maintained at the destination always takes precedence over the values maintained at the action level.

4. If you choose **Manual**, do the following:

    a. If you have the URL of the service that contains the API, enter the service URL.

    b. Choose **Authentication Type** to select the required type of authentication:

    - **None**: No authentication is required.
    - **Basic Authentication**: Provide a user name and password.

5. **Optional:** If you're using a CSRF token in your action, make sure that it's enabled. Ensure that the CSRF token is sent with every request. Additionally, always check the response for the CSRF token before processing any requests. See CSRF Token.

6. In the **Test Input Values**, enter the values.

> **Example**
>
> The following values are input for the fields.

Field	Values
`firstName`	`Jhon`
`lastName`	`Smith`
`username`	`Jsmith`
`id`	`2345`

7. Choose **Test**.

    A preview of the API response appears in the response preview window.

Generate Output for API Responses

Using the response generated from an incoming request, you can generate output for your API actions.

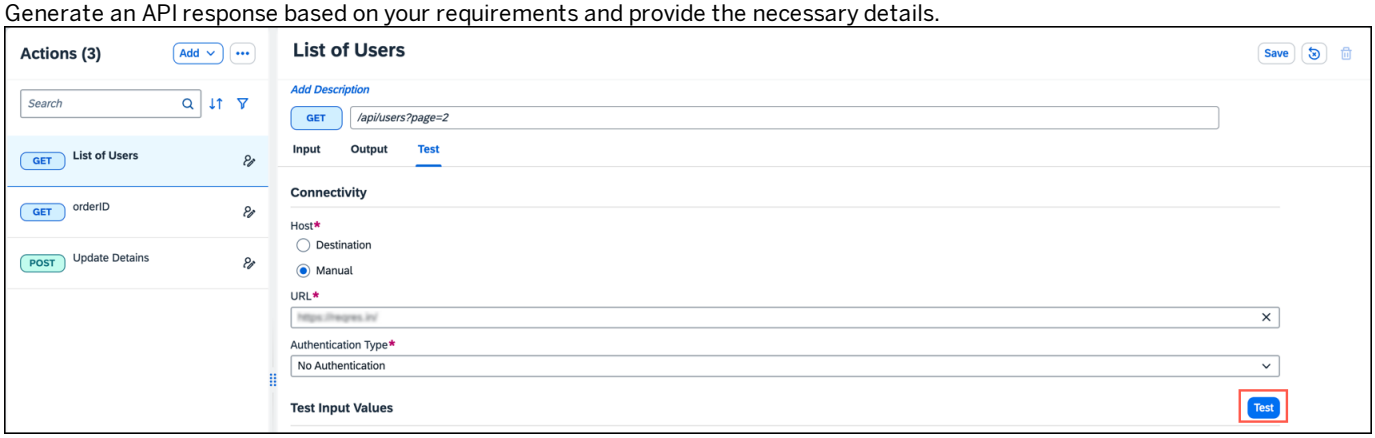
Context

Before generating output for your API action, it's important to test it by providing all the necessary details. Once you receive a successful response, you can then generate an output to retrieve the parameters required for the body.

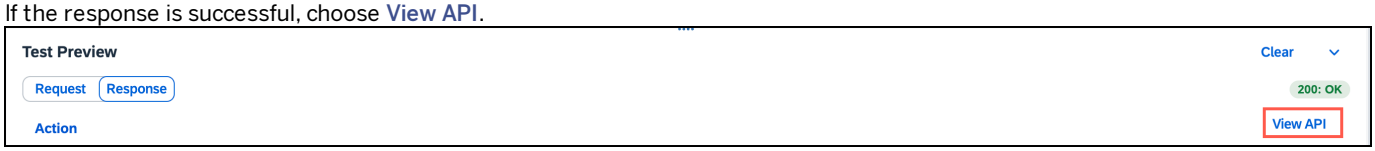
Procedure

1. Navigate to the **Test** console.

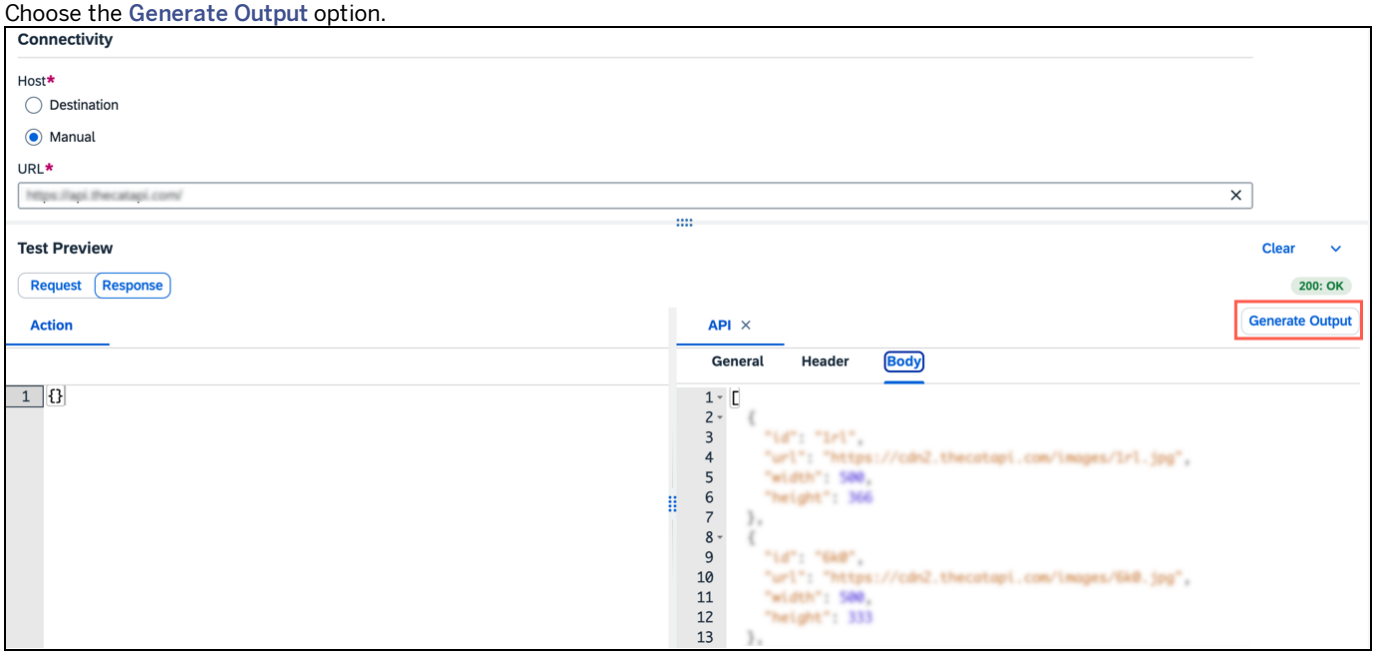
2. Generate an API response based on your requirements and provide the necessary details.



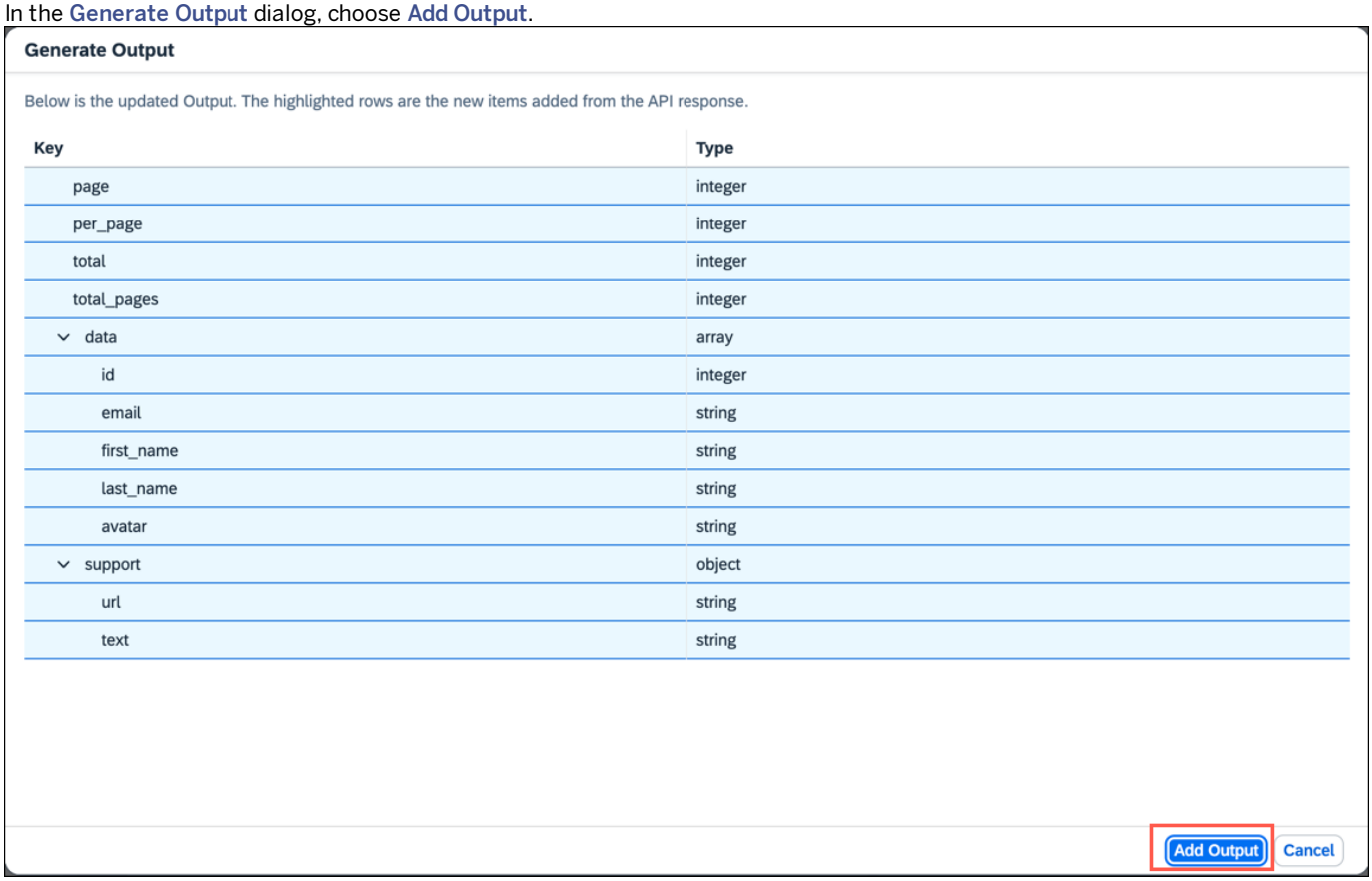
3. If the response is successful, choose View API.



4. Choose the Generate Output option.



5. In the Generate Output dialog, choose Add Output.



An output is generated. Navigate to the Output tab and select the relevant output response code to view.

Related Information

Release a Project
Project Options

Updating Actions

The update feature allows you to modify the Actions Project according to your business requirements.

Prerequisites

You've created an Actions Project. For more information, see Create an Action Project.

Procedure

1. Open your project in the Action Editor.
2. In the Actions window, choose *More Options* and select Update API Specification option to upload your new or modified version of API specification file.

Note

- To download the previously uploaded file, you can use the Download API Specification option at the bottom of the window.
- This feature only supports the Open API specification version 2.x.x. and 3.x.x of JSON type to upload. The API specification also supports EDMX and XML file types.

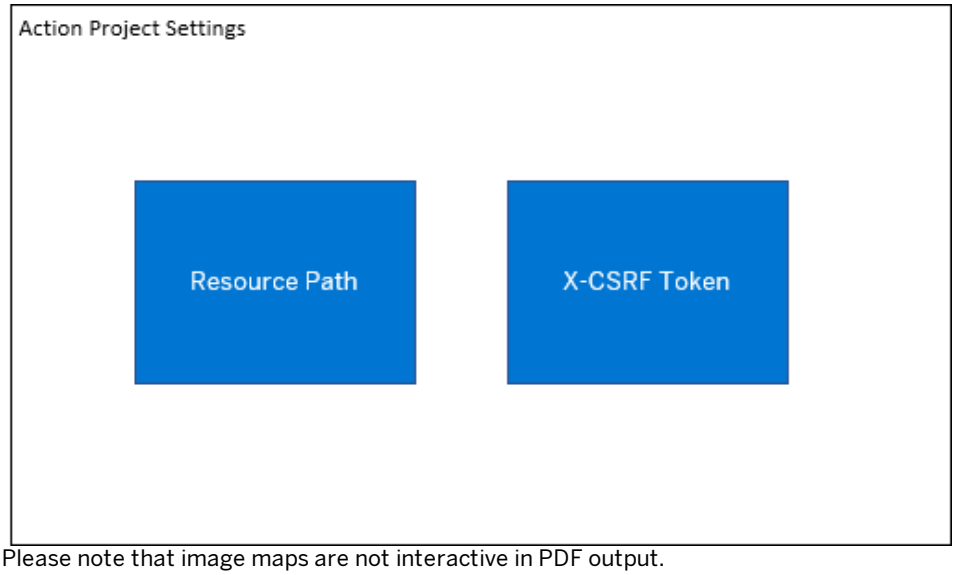
The Actions Project has been updated.

Action Project Settings

The Project Settings option allows you to define the settings for your action project.

You can choose any of the following settings to view detailed instructions.

This image is interactive. Hover over each area for a description. Click highlighted areas for more information.



Please note that image maps are not interactive in PDF output.

CSRF Token

Enter a valid cross-site forgery attacks (CSRF) token fetch end point for the Action in your Actions Project.

Resource Path or URL Prefix

Use the URL resource path when setting up an Actions Project.

Managing Destinations for the Action Project

To establish a secure and reliable connection between API sources, such as SAP Cloud Application Programming Model (CAP), ABAP RESTful Application Programming Model (RAP), and other SAP applications with Actions, create a destination in the same subaccount where you've subscribed to SAP Build Process Automation.

Error Messages

The table contains the following error messages that the user interface might display during the configuration and consumption (in Process Editor or Monitor) of Actions.

Manage Existing Projects

Once created, existing SAP Build Process Automation business process projects can, for example, be accessed, released, published, shared, and deleted from the Lobby.

Procedure

To access the manage options, choose **... (Options)** alongside the respective project.

Import a Project

You can either import a project from your local system or choose from the list of available projects.

Prerequisites

You've exported a project (ZIP file) from the SAP Build Process Automation lobby, or from the SAP Intelligent RPA Studio. Your project is saved on your local system.

For more information on how to export a project from SAP Intelligent RPA, see Export a Project.

Context

You can import an exported project multiple times to create multiple new identical projects in SAP Build Process Automation. You can also import a template project and customize it according to your needs. A project can contain multiple automations or processes as per your business need.

Procedure

1. Navigate to the **Lobby**.
2. Choose *Import*
3. In the **Import** dialog, choose **Browse Files** to search for the `*.zip` or `*.mtar` file in your local system or choose projects from the list for import, if available.

   > **Note**
   >
   > The maximum file size that can be imported is 400 MB.

4. Select the file, then choose **Import**.

   After successfully importing the file, your projects are available in the lobby.

Restore a Project

You can restore the content of a project from package versions generated from the project. The content of the chosen package version replaces the project.

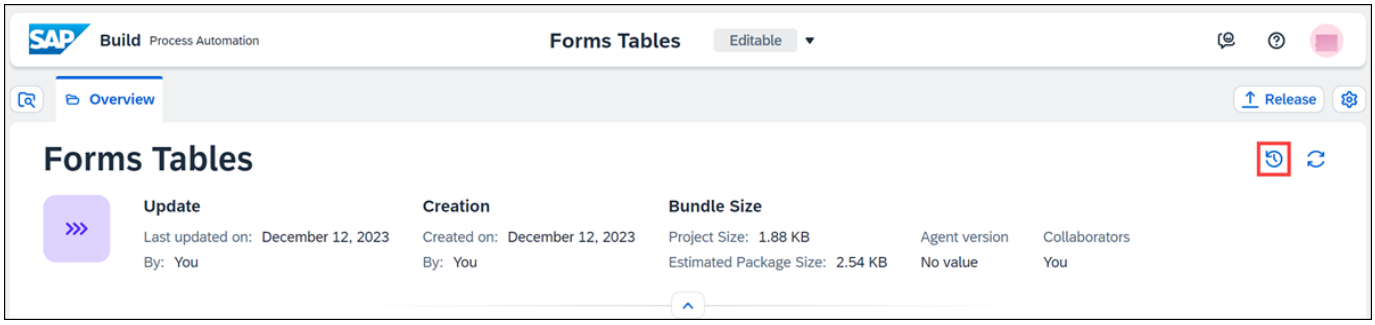
Prerequisites

There are restrictions on the versions that can be used to restore a project:

- You can only restore a package with the same major and minor version as the latest built package.
- You cannot restore a project that contains a decision artifact.
- You cannot restore a project if one of its artifacts is being edited.

Procedure

1. In the **SAP Build Lobby**, open the project.
2. In the project overview, choose (**Restore a version**).



3. In **Select version to restore**, choose the version and **Confirm** to restore it.

Desktop Agent User Guide

About the Desktop Agent 3

The desktop agent 3 (or agent 3) is a component that is installed locally on user desktops. It runs automation projects that launch and run applications of various kinds, read information from screens, enter data, choose options, and process data.

Automation projects are assigned to tenants running on the desktop agent 3. You can see what your desktop agent 3 is doing by accessing the menu from your computer's taskbar while your desktop agent 3 is ready or active.

When it's installed, the desktop agent 3 is configured to start at Windows logon by default. The agent can run in attended or unattended mode. In attended mode, you start the automations by choosing them, and in unattended mode, the automations run automatically without your intervention.

Install and Update the Desktop Agent 3

Before you start to install and use the desktop agent 3, check the following requirements:

- Make sure Microsoft.NET framework 4.7.2 is installed on your machine.
- To run projects with BAPI activities, your IT administrator must install a Windows component on your PC. If you have administrator access, please install Microsoft Visual C++ 2010 Redistributable Package x86 on your machine. If you don't have this access, please ask your administrator to install the component.
- Update the Core SDK dependency in your project to version 1.26 or higher. For more information see, Update a Dependency.
- If you want to install the agent service to keep the agent running when using unattended mode, you need administrator rights. For more information on the agent service, see How the Agent Service Works.

Supported Browsers

Category	Web Browser	Version	Cloud Factory	Cloud Studio	Captured Web Applications	Browser Extension	Store
Chromium-based browsers	**Google Chrome**	**94 and greater**	Supported	Supported	Supported	Supported	Supported
	Microsoft Edge	94 and greater	Supported	Supported	Supported	Supported	Supported

Enable Agent Download and Automatic Updates

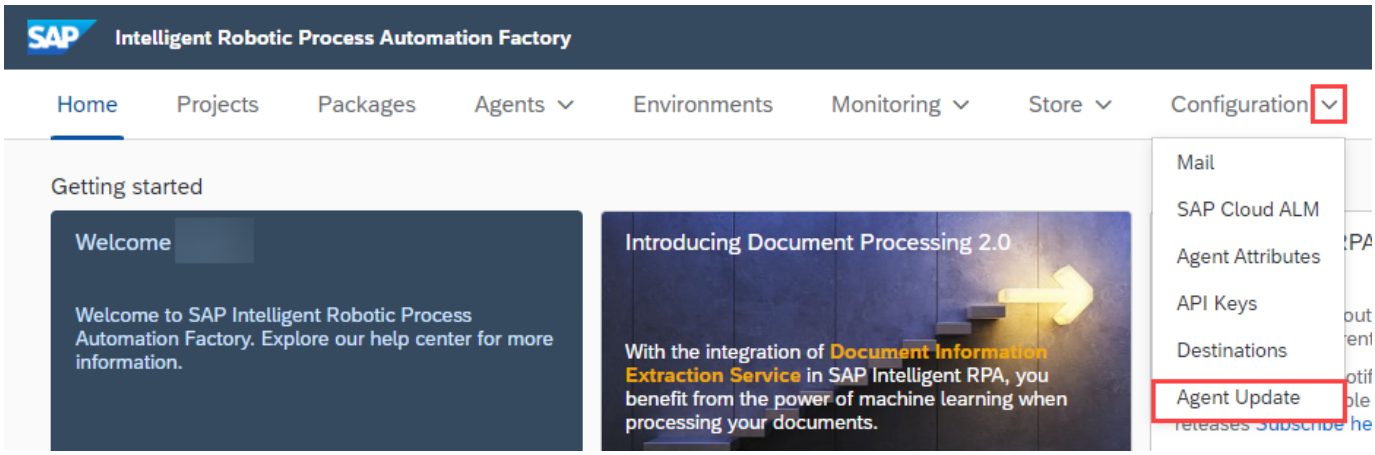
Context

In order to download and install the desktop agent 3, you must first create a technical user in the **Repository-Based Shipment Channel** portal (or **RBSC portal)** and copy its secret ID to your tenant.

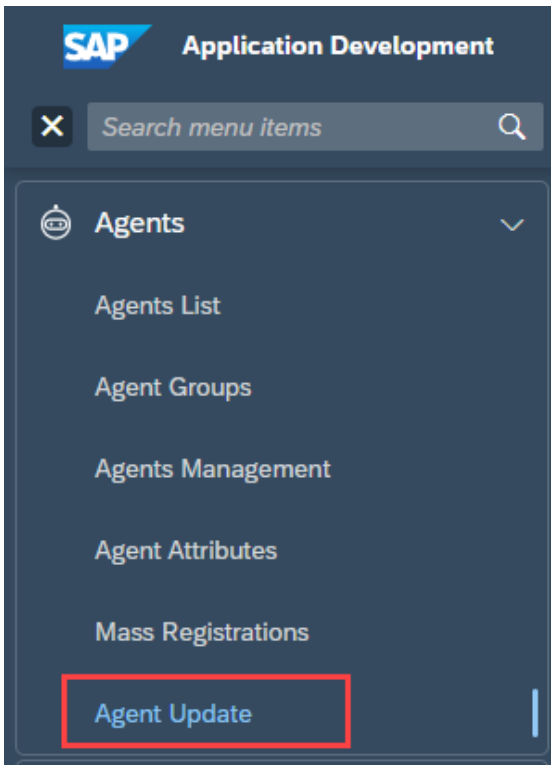
This secret ID is also required to update your agent 3 automatically when a new version is released. For more information, see Enable Automatic Updates.

Procedure

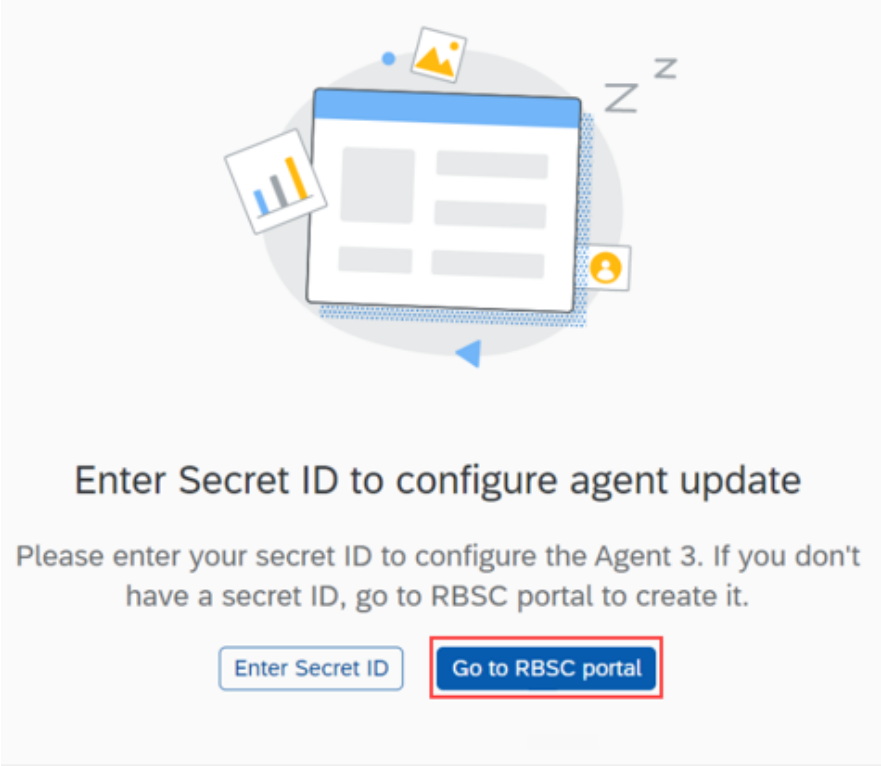
1. **If you use an RPA tenant:** In the Cloud Factory, choose the dropdown arrow next to **Configuration** and choose **Agent Update**.



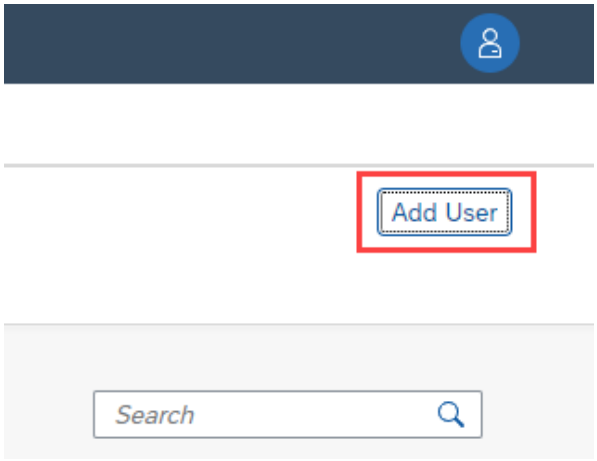
If you use an SPA tenant: In the tenant, go to Settings, and choose Agent Update.



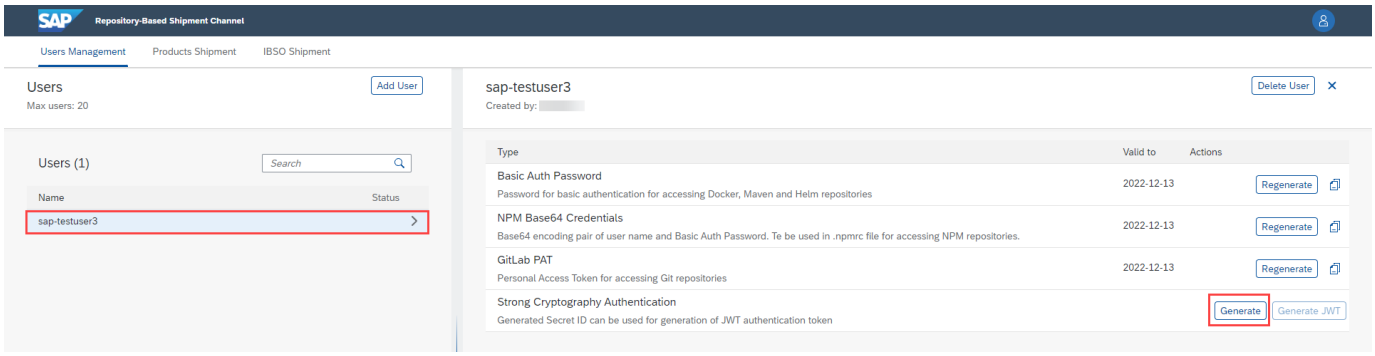
2. Choose Go to RBSC Portal.



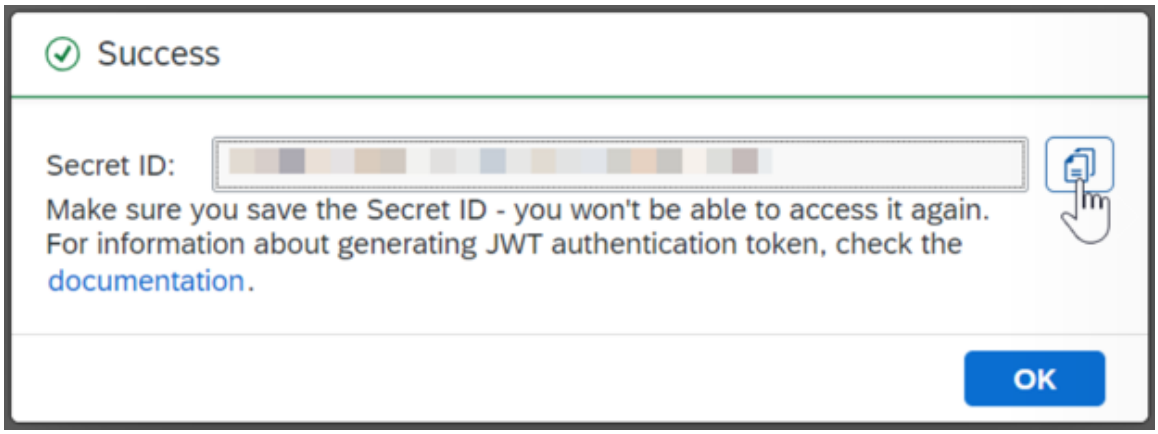
3. Choose **Add User**.



4. Enter a name to create a technical user and choose **Add User**.
5. On **Users** panel, choose the technical user you've created.
6. Go to the **Strong Cryptography Authentication** section and choose **Generate**.

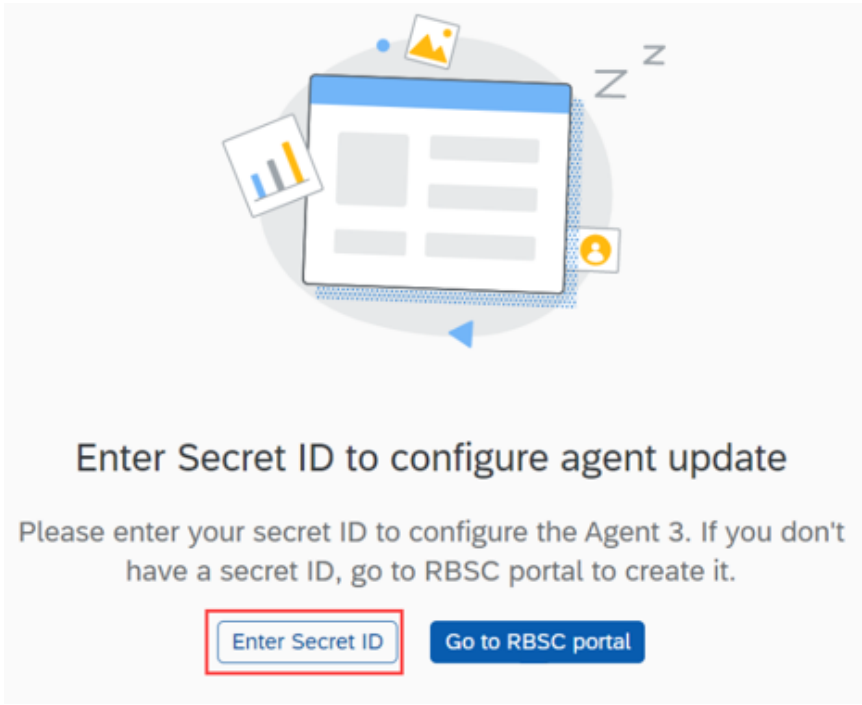


7. Choose on the **Success** window that appears to copy the generated secret ID. Choose **OK**.



You've added a technical user and generated a secret ID.

8. Enter both the technical user and the secret ID in the Agent Update page: **If you use an RPA tenant:** In the Cloud Factory, choose the dropdown arrow next to Configuration and choose Agent Update.**If you use an SPA tenant:** In the tenant, choose Settings and choose Agent Update.
9. Choose Enter Secret ID.



10. Enter your technical name in the Name field and paste the secret ID you copied from the RBSC page in the Secret ID field. Choose Confirm.

Results

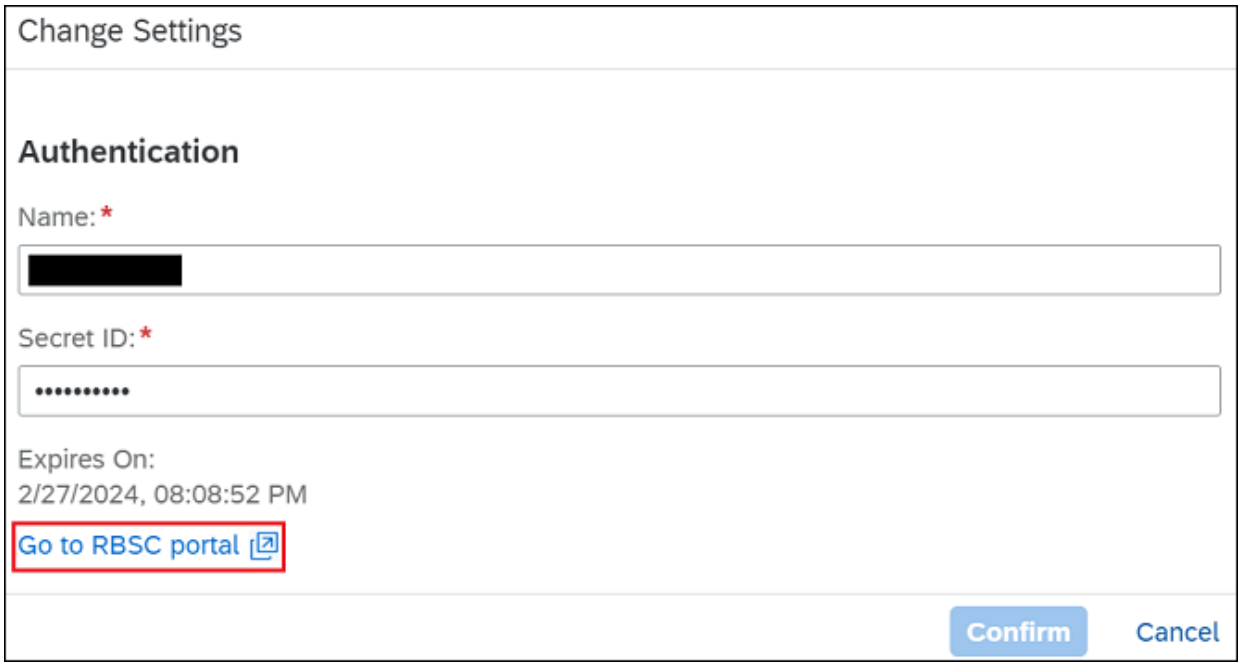
You can now install the desktop agent 3 or copy the link to the download page to send it to other users.

Update Expired RBSC Secret ID

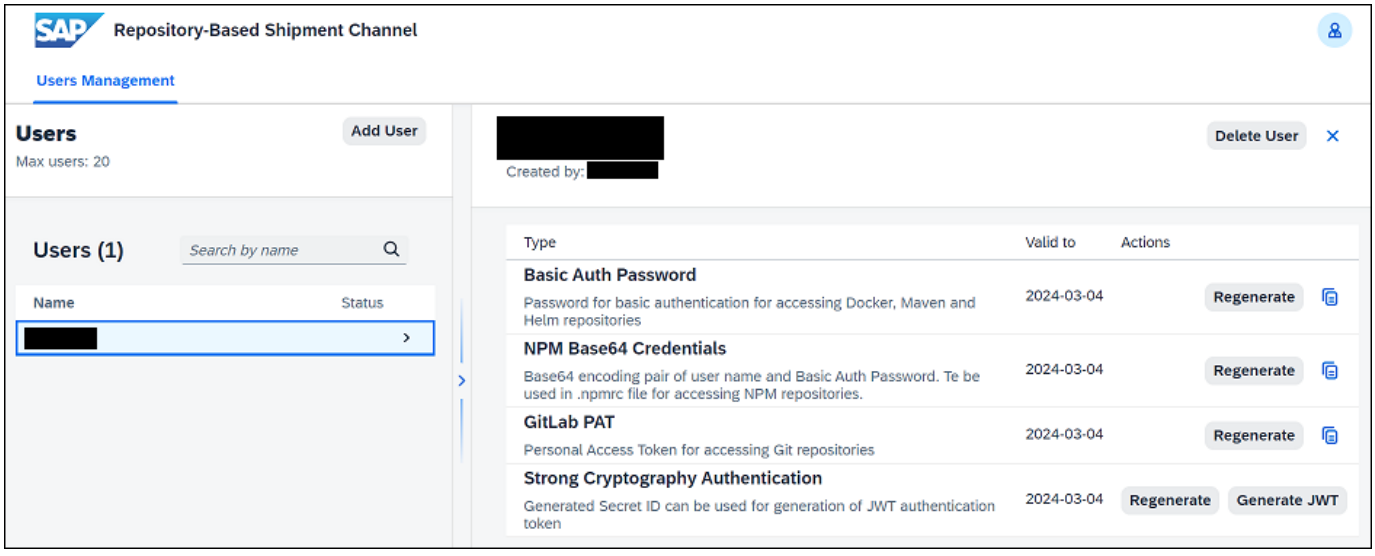
The RBSC secret ID has a six-month expiration period. Therefore, you must generate a new secret ID once it expires.

Procedure

1. In SAP Build, choose **Control Tower**.
2. Choose **Agent Update**.
3. On the **Agent Update** page, choose **Change Settings**.
4. On the **Change Settings** pop-up window, choose **Go to RBSC portal**.



5. On the RBSC portal, generate your new secret ID.



Your new secret ID is generated.

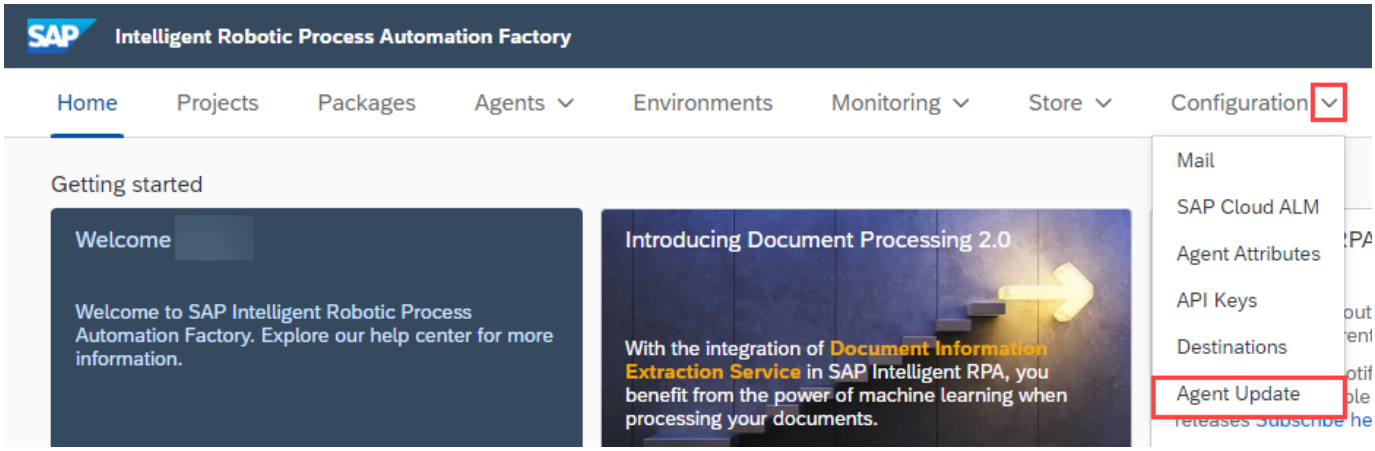
Install the Desktop Agent 3

Context

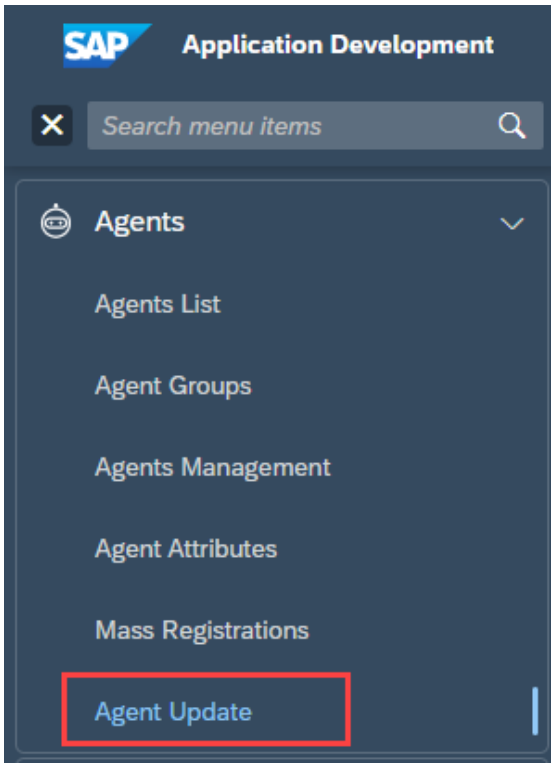
You've generated a secret ID and pasted it in your tenant. You can now download and install desktop agent 3.

Procedure

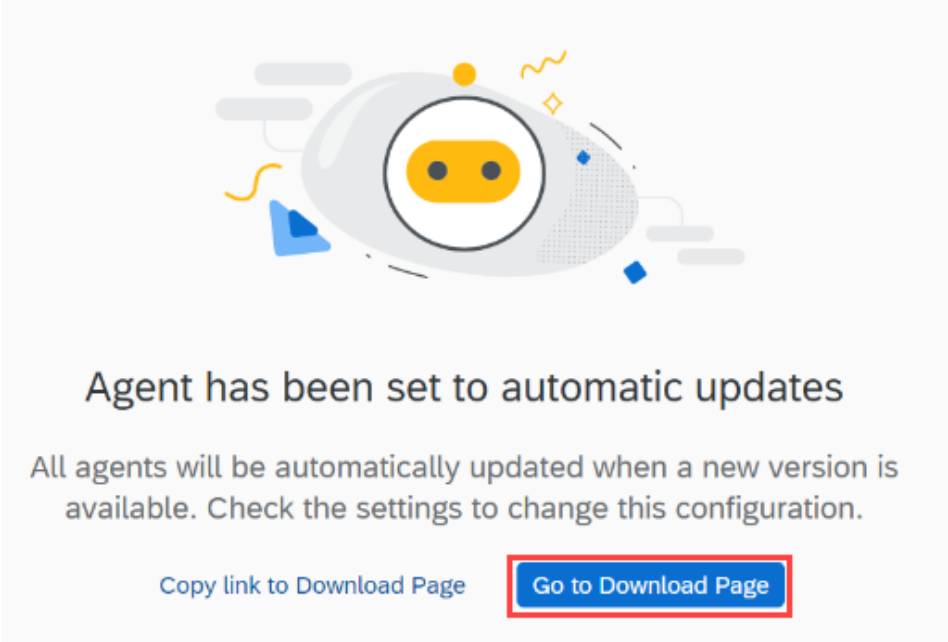
1. **If you use an RPA tenant:** In the Cloud Factory, choose the dropdown arrow next to **Configuration** and choose **Agent Update**.



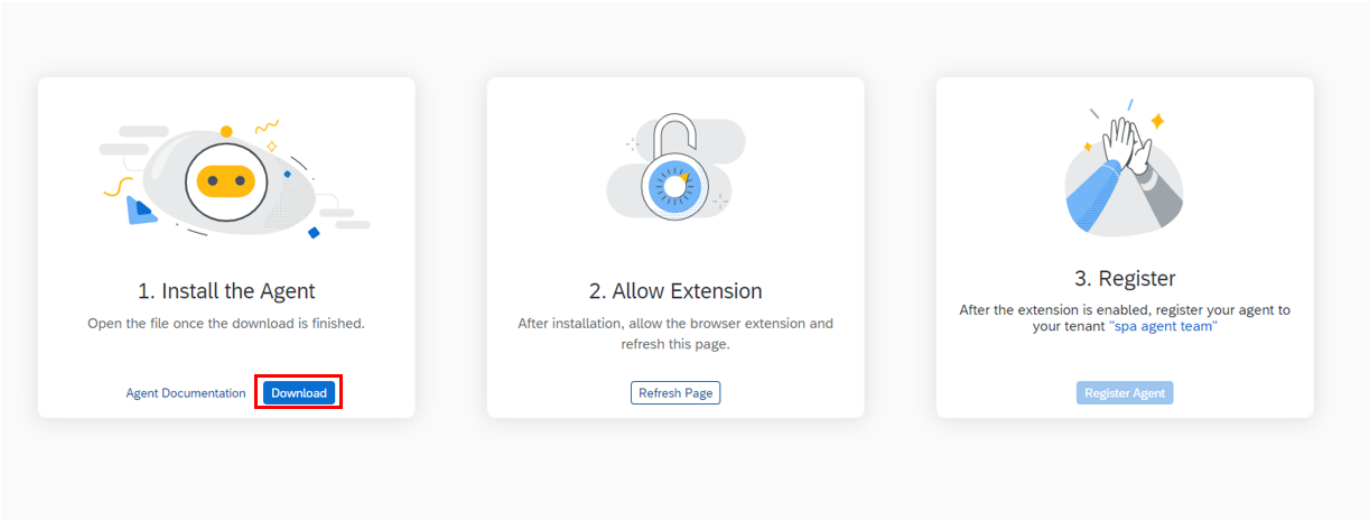
If you use an SPA tenant: In the tenant, go to **Settings**, and choose **Agent Update**.



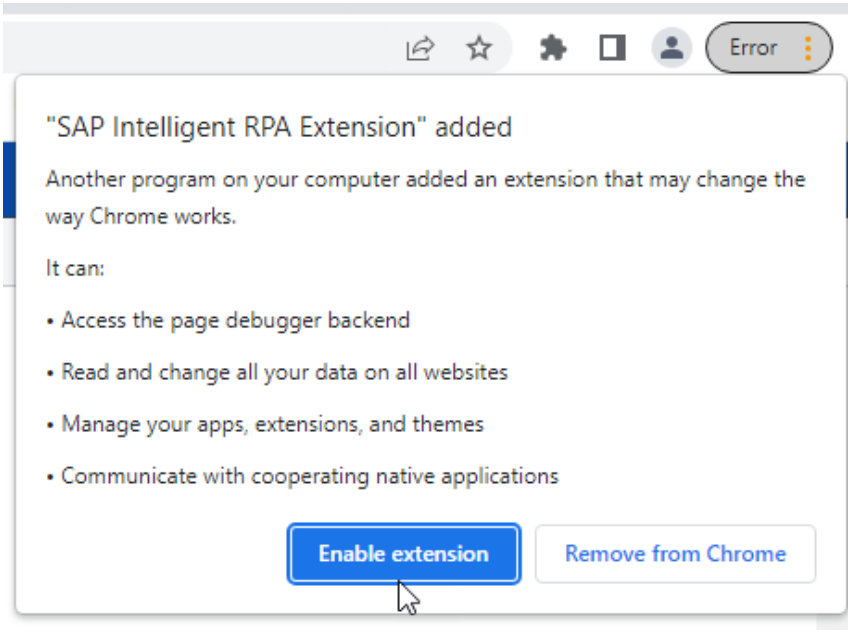
2. Choose **Go to Download Page**.



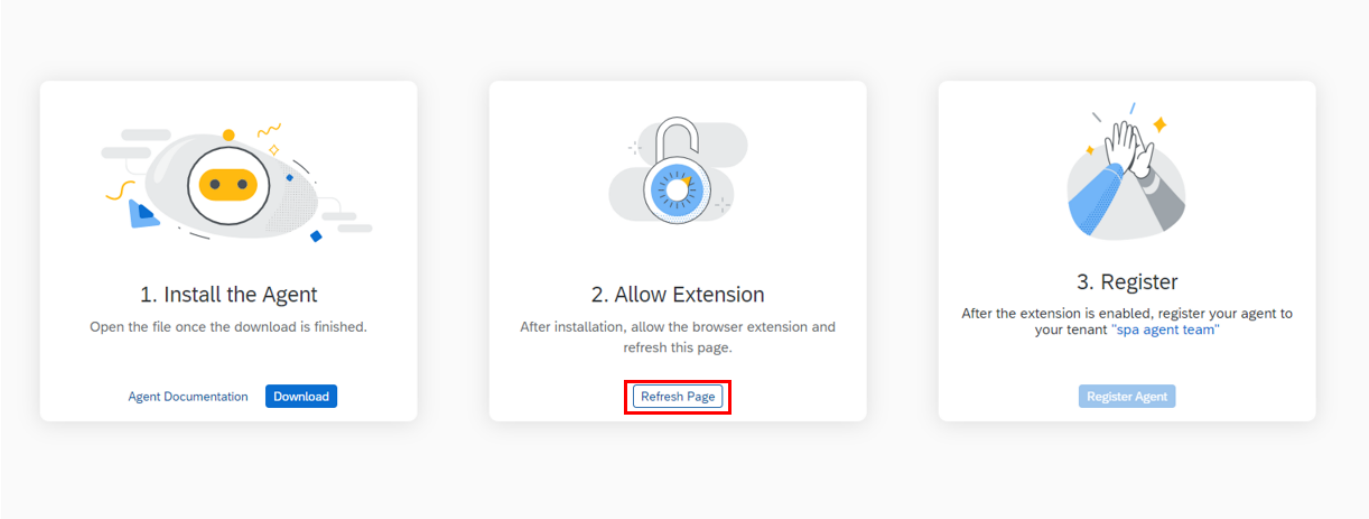
3. Choose **Download**.



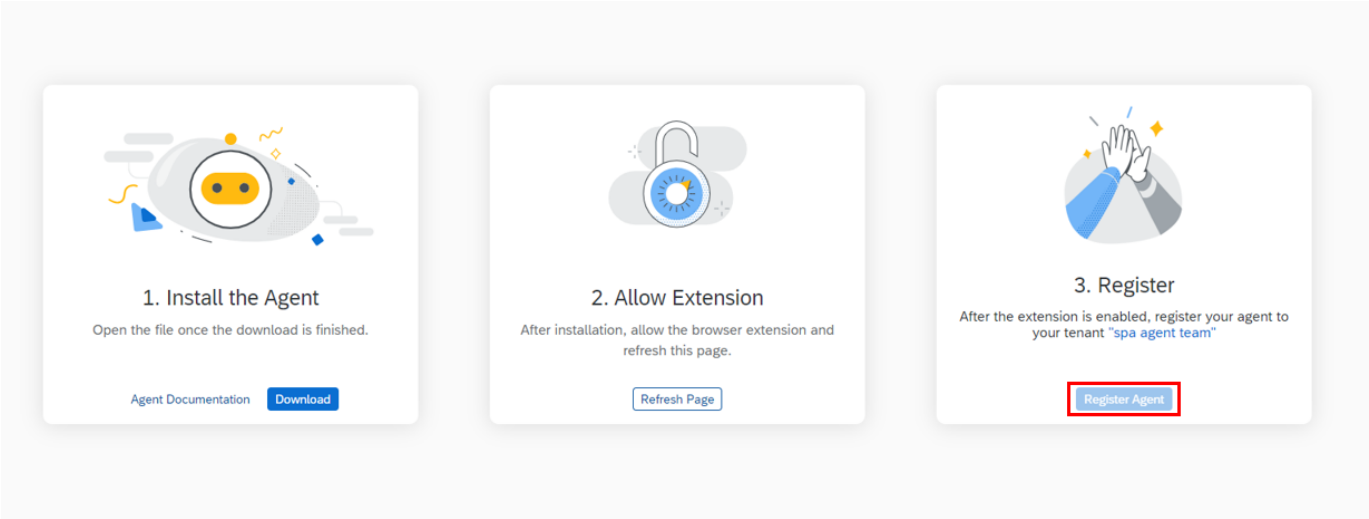
4. Open the file and wait until the installation of the agent 3 is complete.
5. Enable the extension in your browser.



6. Choose **Refresh Page**.



7. Choose **Register Agent**.



8. Open your agent to finalize the setup and accept the tenant configuration.

Results

You've added your tenant to your agent and registered it. You can now start working on your projects.

Register Desktop Agent 3

After you have installed desktop agent 3, you need to register your agent and connect it to a tenant before you can run automations.

Prerequisites

You have installed desktop agent 3 as described in Install the Desktop Agent 3.

Procedure

1. In SAP Build, choose **Control Tower** and **Agents**.
2. Choose **Register New Agent** and **Copy and Close**.
3. Open the agent, choose **Tenants**, and **Add Tenant**.
4. Enter a name for your tenant in the **Name** field, paste the copied URL into the **Domain** field, and **Save** your changes.
5. **Activate** the tenant and authenticate in the browser pop up.
6. Optional: If the authentication window does not appear, choose **Settings / System**, and uncheck **Use browser for tenant registration**. Then repeat the above step: **Activate** and authorize.

Results

Your agent is now connected to the tenant.

Update the Desktop Agent 3

New versions of the desktop agent 3 are frequently available.

By default, the agent 3 is configured to be automatically updated. This way, the newly available version is automatically pushed to your agent. For more information, see Enable Automatic Updates.

You can also choose to manage agent updates for testing purposes for example. For more information, see Manage Updates.

Enable Automatic Updates

Prerequisites

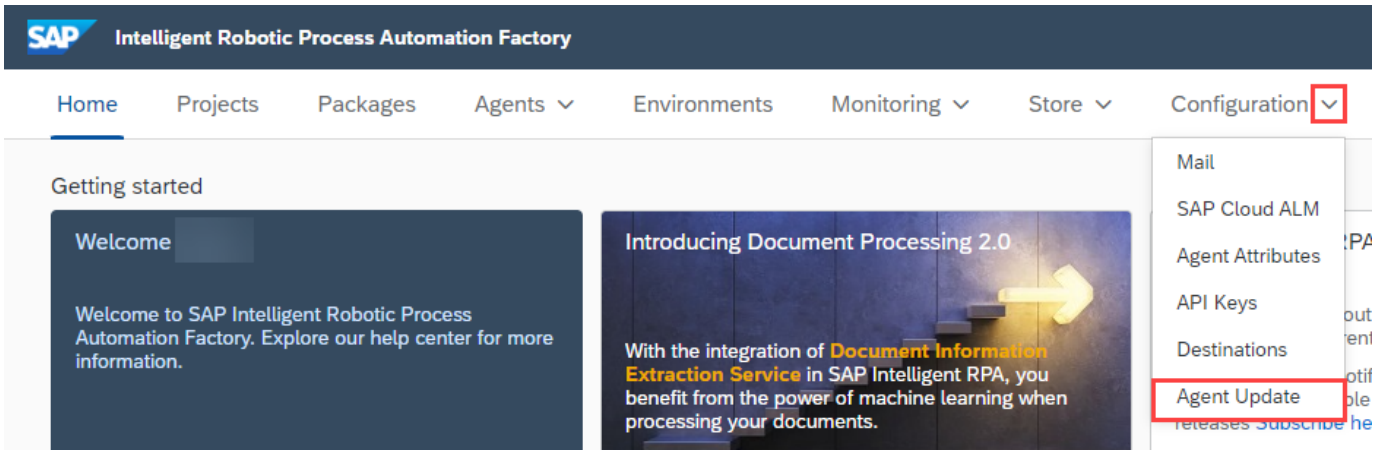
You have generated a secret ID from the RBSC portal as described in Enable Agent Download and Automatic Updates.

Context

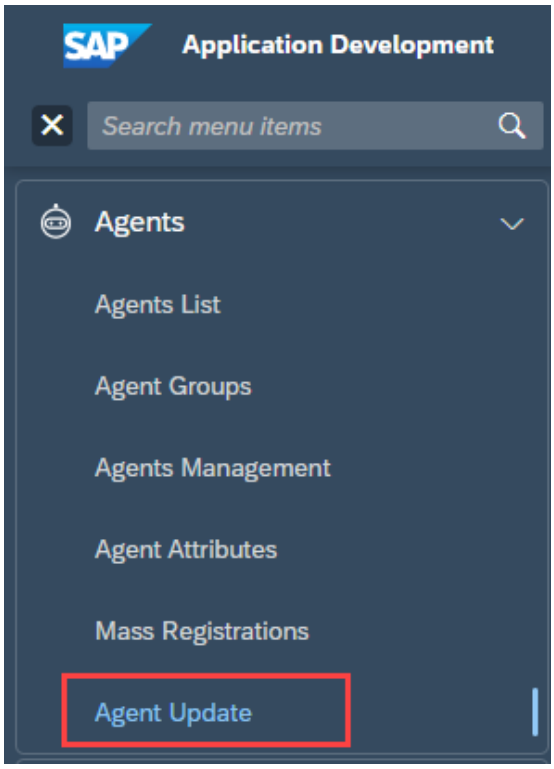
Automatic updates deliver the latest version of the agent 3 directly to your machine or to all your users' machines.

Procedure

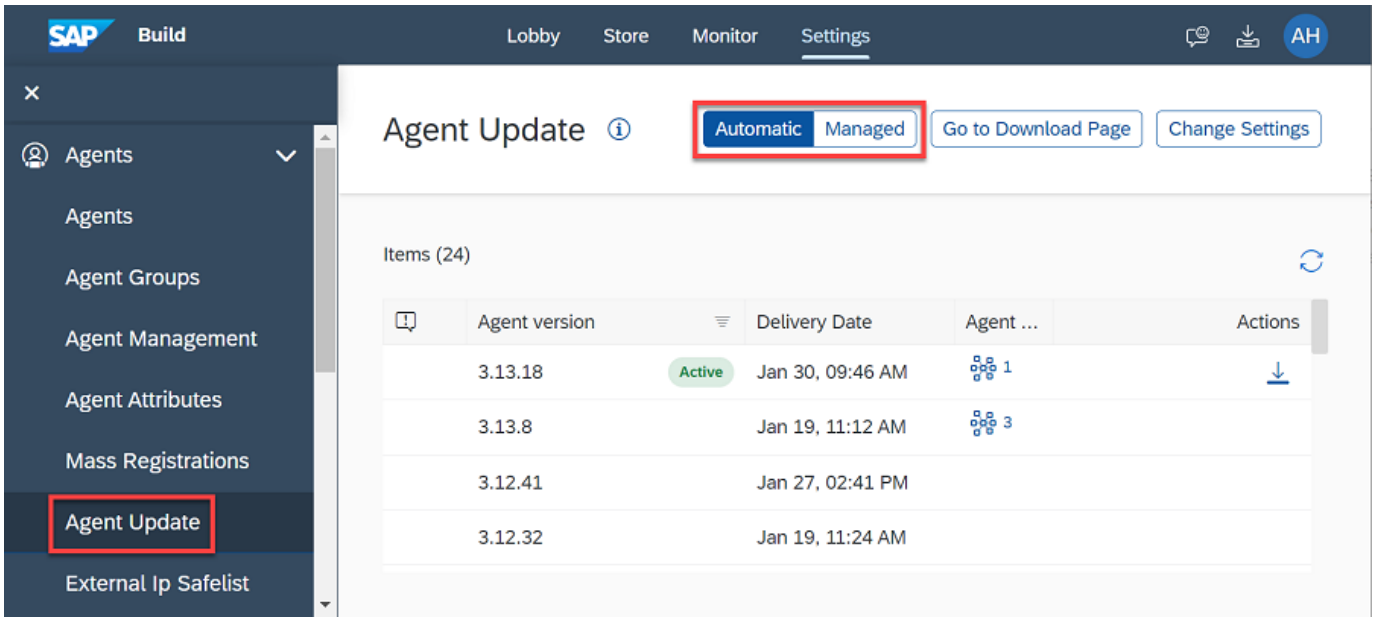
1. **If you are using an RPA tenant:** In the Cloud Factory, click the dropdown arrow next to **Configuration** and select **Agent Update**.



If you are using an SPA tenant: In the tenant, go to **Settings**, and select **Agent Update** on the left-hand side panel.



2. Click Change Settings.
3. In the Change Settings dialog, check Enable automatic updates.



Results

The updates are automatically delivered to your agent if it is in unattended mode.

If you agent is in attended mode, when a new update is available, it is automatically downloaded, and you receive a notification prompting you to open your agent. You can then choose to update your agent now or tonight by selecting one of the two options on the pop-up window that appears.

Manage Updates

Prerequisites

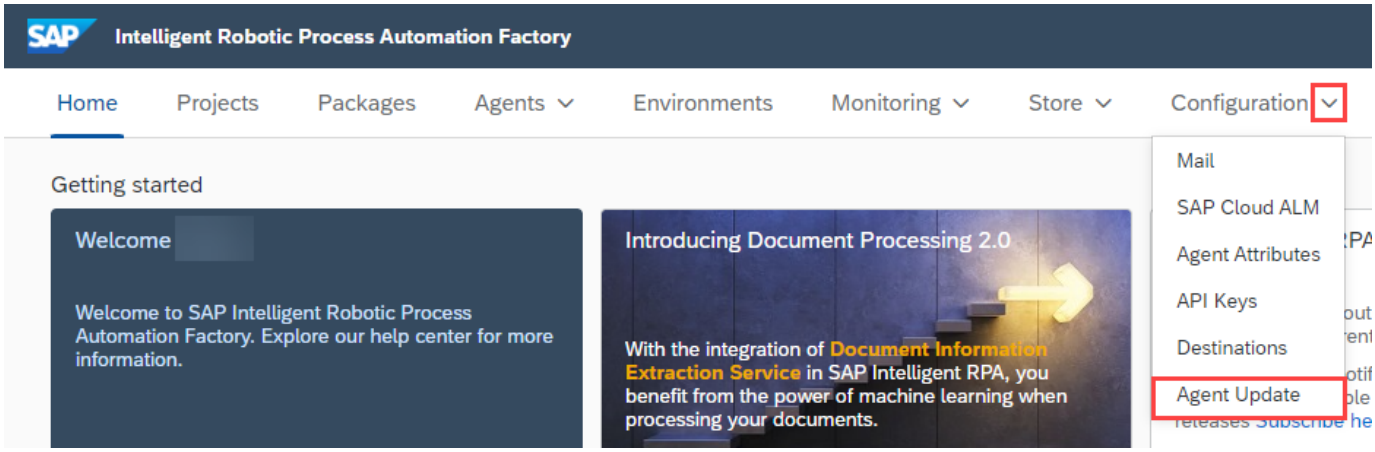
You have generated a secret ID from the RBSC portal as described in Enable Agent Download and Automatic Updates.

Context

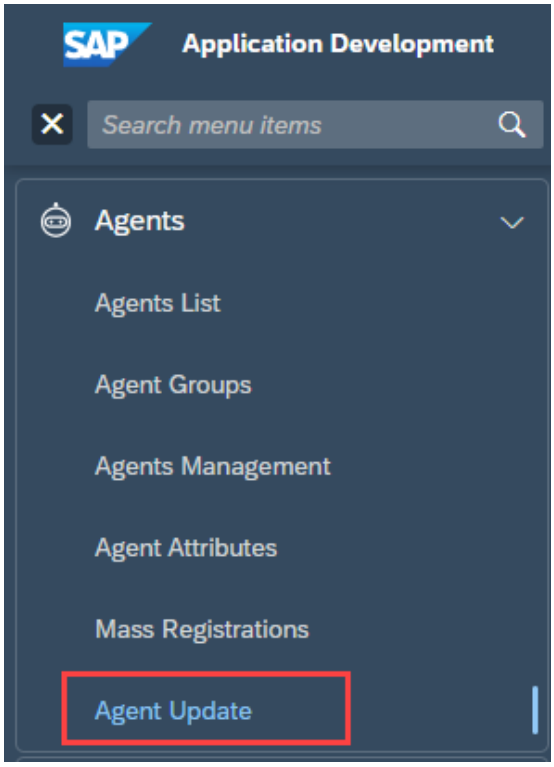
If you decide to manage updates, you can pause these updates to test them, for instance. You can then choose which versions you want your users to receive on their machines.

Procedure

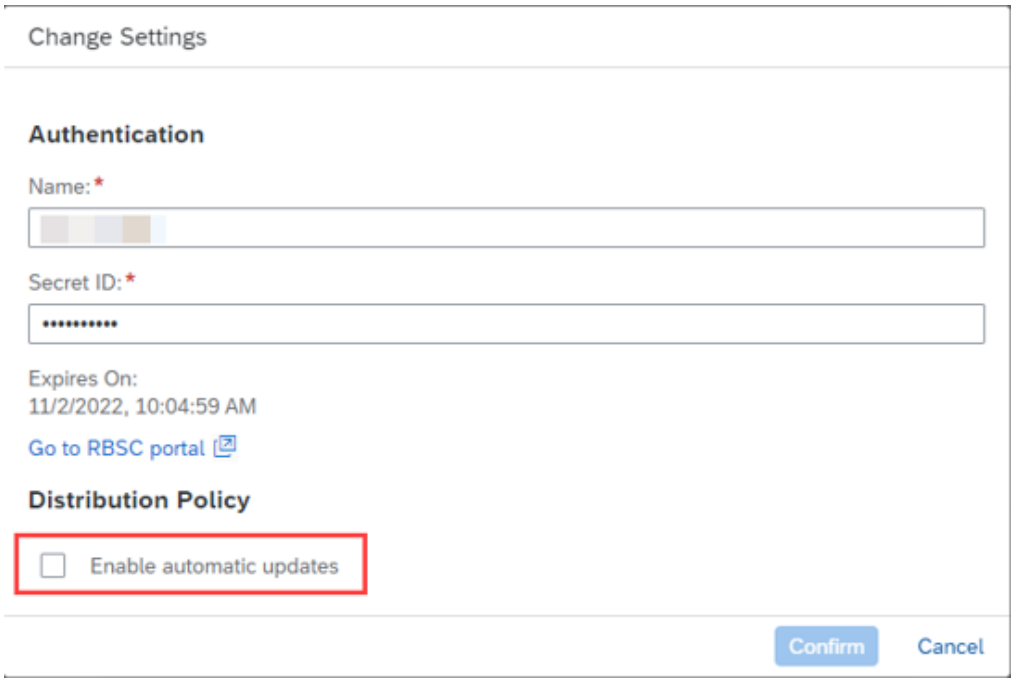
1. **If you are using an RPA tenant:** In the Cloud Factory, click the dropdown arrow next to Configuration and select Agent Update.



If you are using an SPA tenant: In the tenant, go to Settings, and select Agent Update on the left-hand side panel.

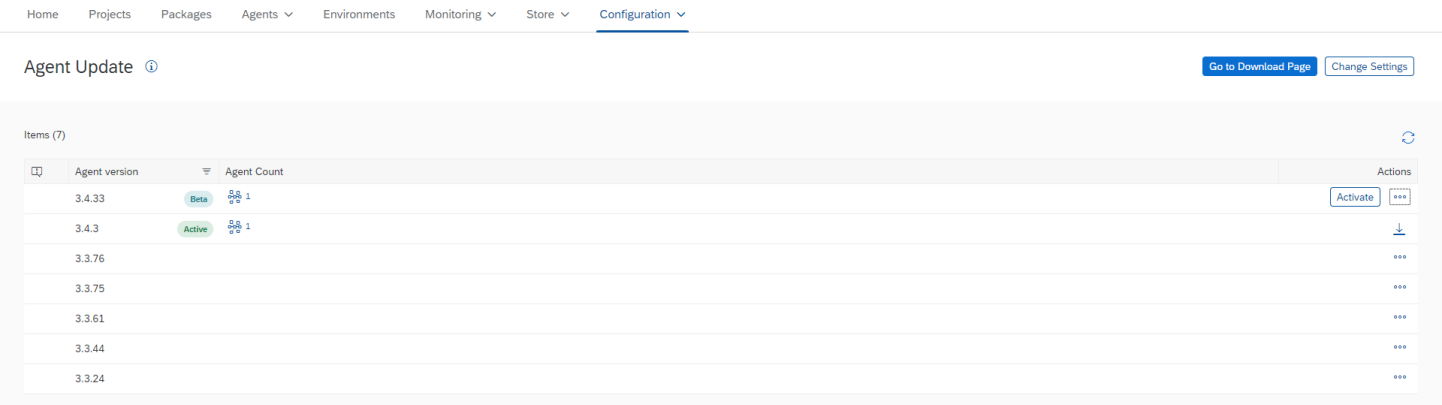


2. Click Change Settings in the top right-hand corner.
3. In the Change Settings dialog, deselect Enable automatic updates and click Confirm.



Results

On the **Agent Update** page, you can you can see the list of the last published agent versions and manage updates.



You can perform several actions such as:

- Activate an agent version.

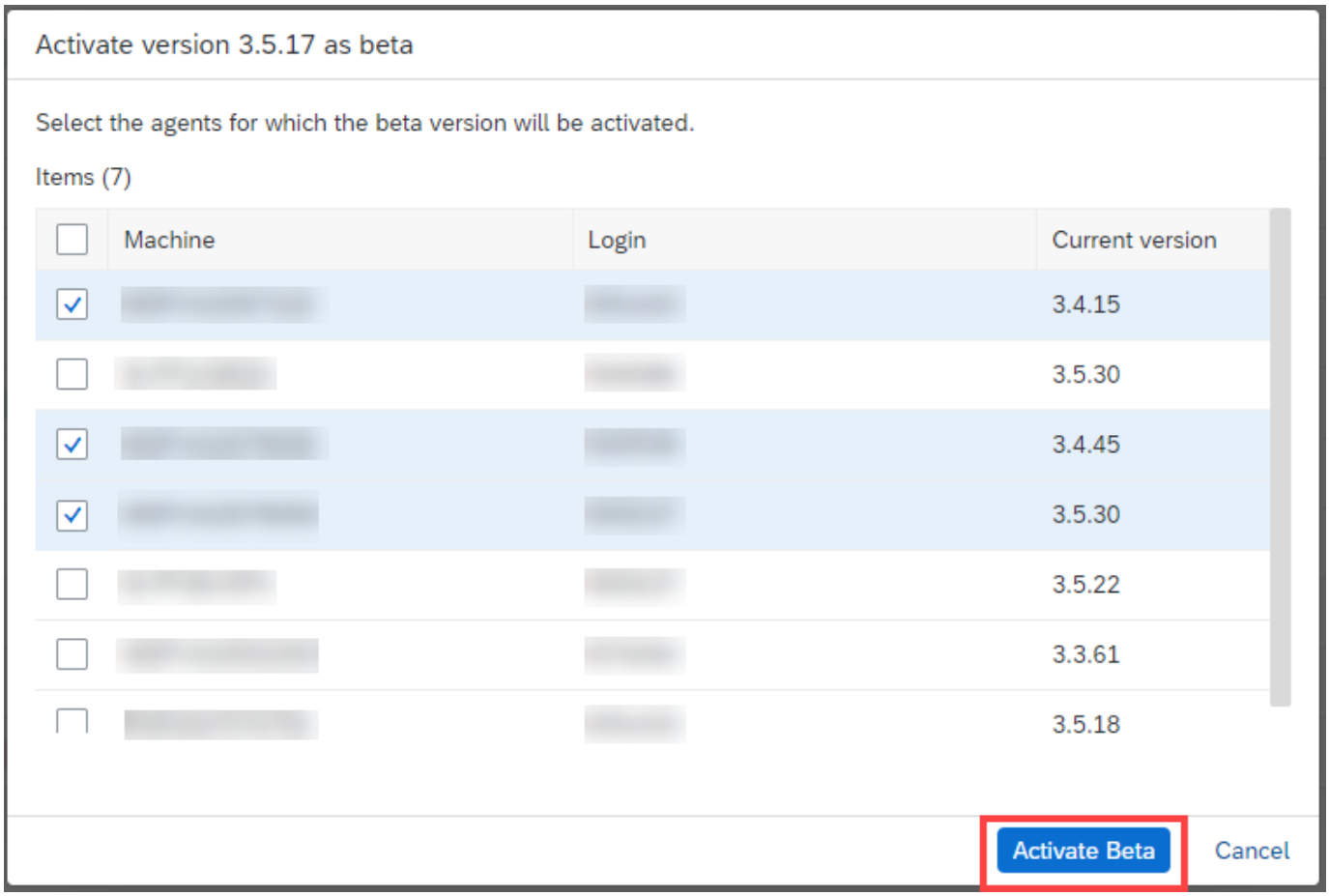
  To do so, click ⋯ under **Actions** next to the version of your choice and select **Activate**.

  You can also click the **Activate** button next to the beta version to set this version as the active version.

- Set an agent version as a beta, if you want to test this version for example.

  To do so, click ⋯ under **Actions** next to the version of your choice and select **Activate as Beta**.

  You can then select to which agents this beta version will be pushed and click **Activate Beta**.



You can edit the agents for which the beta version is activated by clicking ▯ under **Actions** and selecting **Edit Beta Agents**.

- Download an agent version on your machine.

  To do so, click ▯ under **Actions** next to the version of your choice and select **Download**.

  You can also click ▯ next to the active version to download it.

By clicking 1 next to an agent version, you can see which agents are registered in this version.

How the Agent Service Works

The service ensures that the agent is always running and is required when the agent is running in unattended mode.

When the agent is installed on a machine, a service named SAP`DesktopAgentService.exe` can also be installed. This service runs under the local system identity. It is independent of the agent that runs under the session user identity.

To check if the agent service is installed on your machine, choose **About** and **Agent Details** in the agent. If the agent service is not installed on your machine, the service state is **Not detected**, and the service version is **Unknown**.

What Is the Purpose of the Agent Service?

The service ensures that the agent is always running and monitors it for issues. Therefore, if you are using an RDP environment, we recommend that you install the service.

The service is also required when the agent is running in unattended mode. The agent sometimes needs to unlock and lock the session to perform activities and for security. Enable the agent to automatically unlock a session by entering the credentials in the **Connection** tile. For more information, see Configure the Connection Settings .

How to Install and Update the Agent Service

As an admin, you can install the service on your users' machines at the same time as you install the desktop agent 3. To do so, you must run the setup as administrator and check the corresponding checkbox on the installation window. If you don't run the setup as administrator, the checkbox is not visible.

You can also install the service using the following command line (if your script has been started as an administrator): `xxx.exe /s --service`

When a new version of agent 3 is available, a new version of the service is also available.

ℹ Note

You must update the service manually. If automatic updates are enabled, only the agent is automatically updated.

ℹ Note

If you are using desktop agent version 3.17 and under, make sure that the SAP`DesktopAgentService.cfg` file has been configured correctly. For more information, see SAP Note 0003388923.

Enable Batch Installation

Context

As an administrator (`IRPAOfficer` role), install and configure automatically the desktop agent 3 on several machines with the batch installation.

This procedure shows you how to create a PowerShell script that will automatically install and configure the desktop agent 3 the first time a user logs into a machine (Windows Desktop or Windows Server sessions).

ℹ Note

Agent 3 is an installation per user, not per machine. It means that one independent agent is installed for every session.

Procedure

1. Define a shared location within your company environment, accessible from the different machines, to store the files required for the batch installation of the desktop agent 3. It can be:
   - within your company network: `//...`
   - in a public folder: `%programdata%\SAP` for instance
2. Download the .exe installation file of the desktop agent 3 as described in the step 3 of the installation procedure. Store this file in the shared location defined in the step 1.
3. Create and download a registration token as a .json file as described in the registration procedure of the Mass Agent Registration. Store this file in the shared location defined in the step 1.
4. Create a configuration .json file named `agentSettings.json` with the attributes you want to configure, following this pattern:

```
{
        "generalSettings": {
                "language": "en",
                "agentMode": "unattended",
                "startOnWindowsLogon": true
                }
}
```

   Choose between the following values:

   - language: "en", "fr", "en_US_sappsd", "es", "de", "it", "ja", "ko", "pt", "ru", "zh_CN", "zh_TW"

- agentMode: "attended", "unattended"
- startOnWindowsLogon: true, false

Store this file in the shared location defined in the step 1.

5. Create the PowerShell script that will automatically install and configure the desktop agent 3 on a machine. According to you specific needs and methods, you can write or adapt your own script. For example:

```
Write-Host "------------------------------------------------------"
Write-Host "-               Agent 3.x User Deployment            -"
Write-Host "------------------------------------------------------"

$notInstalled=$true

$Folder = $HOME+'\AppData\Local\Programs\SAPDesktopAgent'
$File = $Folder+'\SAPDesktopAgent.exe'

if (Test-Path $Folder) {

        if (Test-Path $File){

                $notInstalled=!$true

                Write-Host "agent 3.x already installed"

                Write-Host "------------------------------------------------------"

                }

        }

if ($notInstalled) {

        if (Test-Path "$PSScriptRoot\SAPDesktopAgentSetup*.exe") {

                foreach ($files in Get-ChildItem $PSScriptRoot) {

                        if ($files.Name -match "SAPDesktopAgentSetup-.*.exe") {

                                $setup = $files.FullName

                                        }

                                }
                                Write-Host "agent 3.x is installing ..."

                                $ps = [System.Diagnostics.Process]::Start($setup, "/S")
                                $ps.WaitForExit()

                                Write-Host "agent 3.x has been installed correctly."

                                Write-Host "------------------------------------------------------"

                                }

                        else {

                                Write-Host "No setup in the folder"

                                Write-Host "------------------------------------------------------"
```

```
}


}
```

6. On all the targeted machines, add the run registry key that will be used to launch the execution of the PowerShell script. We recommand to use `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run`.

   For example:
   `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce\SAPDesktopAgentSetup` with the value `< definedLocation> \SAPDesktopAgentInstallation.ps1 /s`.

Results

When the specified machines start, the PowerShell script will automatically install and configure the Desktop Agent 3.

> **Note**
> Even is the agent is connected and configured, the user still needs to activate the browser extension. For more information, see Install the Desktop Agent 3.

Use the Desktop Agent 3

Using the desktop agent 3, you can perform different actions to run and monitor your projects such as:

- Add and Manage Tenants
- Configure the Connection Settings
- Change the Agent Mode
- Monitor the Agent
- Manage Traces

Start Jobs With a Disconnected Agent

You can trigger jobs when desktop agent 3 is disconnected from the cloud.

When a job is running on desktop agent 3 and the agent is disconnected from the cloud, it can still complete the running job. You can start a new job from a disconnected agent, if the following requirements are met:

- The agent is in attended mode. You can change the agent mode in your agent under **Settings / Mode** .
- The project has been used in the last two weeks. Go to **Projects** to see your most recent projects and **Start** a project. If a project is grayed out, it has not been used in the last two weeks.
- The timer in **About / Agent Details** has not run out. If you start a job and the timer runs out, the job can still finish. However, to start another job you have to wait until the timer resets at midnight.

As soon as the agent is reconnected, the agent and tenant synchronize.

Add and Manage Tenants

You use the **Tenants** tab to see information about the tenants associated with your agent.

Context

Add a new Tenant

The tenant your agent is connected to is at the top of the **Tenants** list and is labeled as **Active**.

By clicking  next to a tenant in the **Tenants** list, you can activate, edit, duplicate, or delete it.

You can associate multiple tenants with your agent.

To associate a new tenant to your agent:

Procedure

1. Click **Add Tenant**.
2. Enter a name for the tenant in the **Name** field.
3. Provide the URL of the tenant in the **Domain** field.
4. Click **Save**.
5. If you want to connect your agent to this tenant and click  next to your newly added tenant and select **Activate**

   Your tenant will move at the top of the **Tenants** list.

Configure the Connection Settings

You can configure the service to unlock a session and start the agent automatically when a machine boots.

Prerequisites

Agent service 3 or higher is installed and running.

Context

Configure the connection settings of the agent to allow the service to open the Windows session automatically when booting. If the agent is in unattended mode, the service can, after a reboot, automatically open the Windows session and start the agent.

By default, no credentials are entered and the service will not login to the session automatically. So when you start the machine, you need to login manually.

Procedure

1. In your agent, choose **Settings** and **System**.
2. In the **Connection** tile, choose **With user account**, enter your username and password, and **Save Settings**.

Results

If your session locks, the service is able to unlock it automatically.

Note

- If the connection tile is disabled, the agent service is either not installed, not running, or its version is below agent service 3.
- If you disable **Automatically start at Windows logon**, the agent will not start.

Run Unattended Jobs with Activities on Graphic Sessions

The agent sometimes performs graphic activities, for example, a mouse click, on a running automation. If the computer screen is locked, the agent cannot perform graphic activities.

When unattended bots are configured on open Windows Remote Desktop Protocol (RDP) sessions and the sessions remain active but without graphic sessions, the agent cannot unlock the sessions for security reasons.

The RDP manager mode allows you to keep open Windows RDP sessions or to automatically open them when needed. Enable RDP manager mode to register and manage Windows RDP sessions on different computers to automatically perform activities when an automation is running.

If your automation includes the following activities, you can use the agent service to unlock the sessions, and the RDP mode to reconnect to graphic sessions:

Graphic Activities

Activity	Technology
Screenshot	All
Keystroke	All except WebGUI
Set Element	All
Click	UI Automation
ClickSync	UI Automation
MouseClick	All
ClickDouble	All
ClickRight	All
Maximize	All
Minimize	All
Restore	All
Iconify SAP GUI	All
SetFocus (Element)	All

Key Points

- SAP desktop agent service manages unlocking the sessions.
- The RDP manager keeps sessions open.

Register Users to Unlock User Sessions

Register the users.

Context

The users of the RDP sessions must register in the Windows SAP desktop agent service. There's one service per machine. If several sessions are running on the same machine, they use the same service. Each session for which the user is registered can be unlocked.

The following user roles are available:

- UNLOCK TYPE: Only allows to unlock the session.

- LOGON TYPE: Includes the creation of the session by the service and the management of the UNLOCK role.

ℹ Note

Only one user of type LOGON can be registered per machine. We recommend that you register the user of the RDP manager session with the LOGON TYPE role.

Procedure

1. Open Command Prompt.
2. Go to the directory `C:\ProgramFiles(x86)\SAP\IntelligentRPA\SAPDesktopAgentService`.
3. Run the following commands.

    a. For the LOGON user: `CxStoreCred.exe userId= ****** password=******* type=LOGON`

    ### ℹ Note

    Alternatively, you can register the LOGON user directly in the desktop agent under **Settings** > **System** > **With user account**. Enter the user name and password, and set **Automatically start at Windows login**.

    b. For the UNLOCK user: `CxStoreCred.exe userId= ****** password=******* type=UNLOCK`

    ### ℹ Note

    - If you don't specify the user type in the command line, it's LOGON by default.
    - The `userId` and the `password` are limited to 32 characters each.
    - Special characters in the password must be in straight quotes (""), for example, "cbkugˆˆcna".

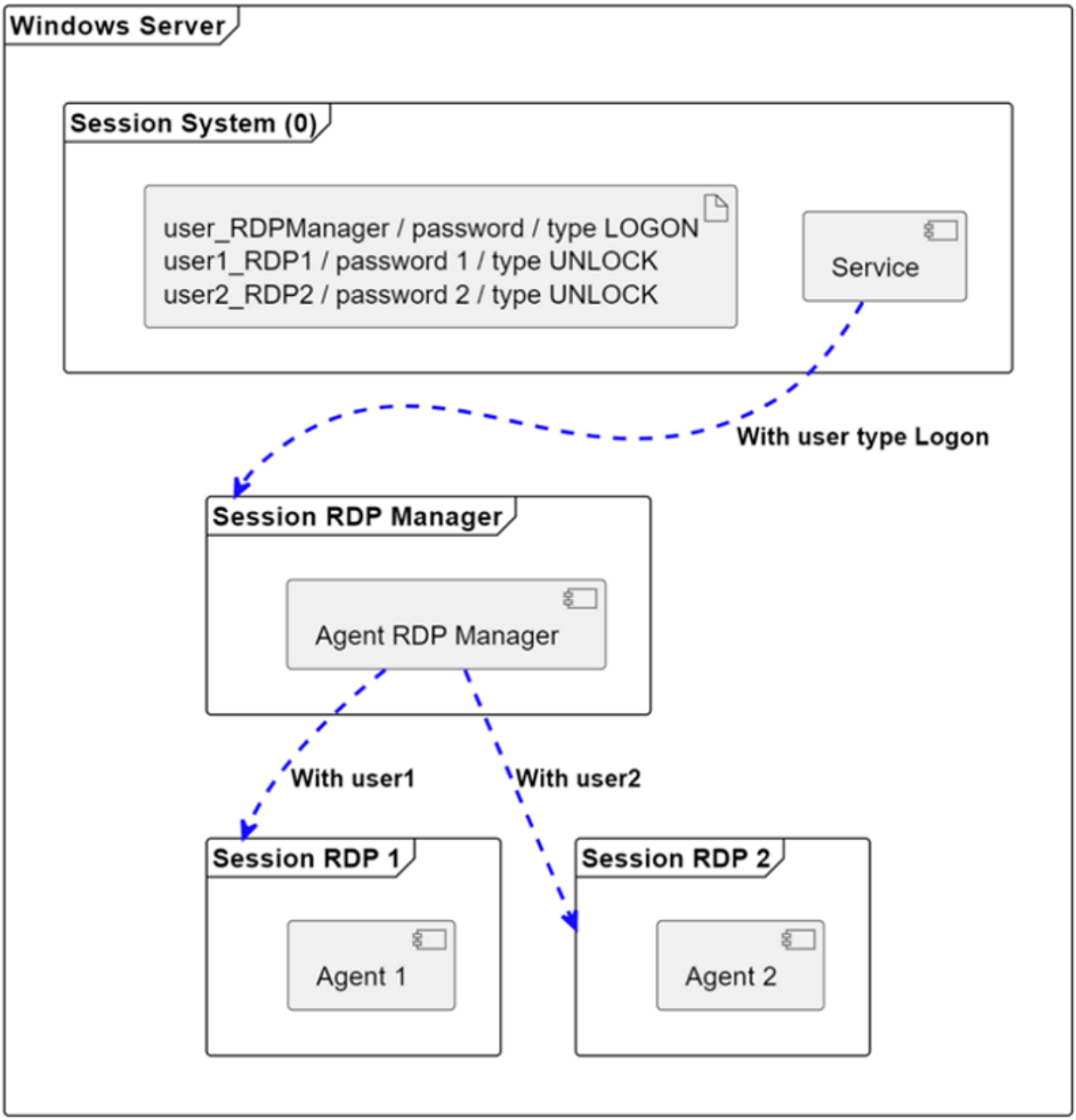
Example

Users can be registered in the agent service from any session on their machine. Once registered, the user is stored in the session system through the service. In the example, you can therefore:

- Register all users in one of the sessions on the machine, that is, RDP manager session, RDP session 1, or RDP session 2.

  Register each user from their own session. user_RDP1Manager in session RDP manager and user1_RDP1 in session RDP 1, and user2_RDP2 in RDP session 2.

Example: Configuration of an RDP Manager Session that Opens Two Sessions



Keep Sessions Open with the RDP Manager

The RDP manager is a mode of the agent available from version 3.8.

Prerequisites

The users are registered. See Register Users to Unlock User Sessions.

Context

RDP manager mode is only available for version 3.8 and higher. However, an agent in RDP manager mode can manage sessions with other running agents of version 2 and 3.

ℹ Note

The session that runs the agent in RDP manager mode can't run automations. It's dedicated to run the agent in RDP manager mode.

The RDP manager can keep sessions open on the local machine or a remote machine.

You can temporarily open an RDP session on another machine. If the session is open, the RDP manager doesn't try to open it. But if you close the RDP session, the RDP manager automatically detects that and opens it again. The RDP manager checks every 30 seconds whether the session is open.

To configure the RDP manager mode, do the following:

Procedure

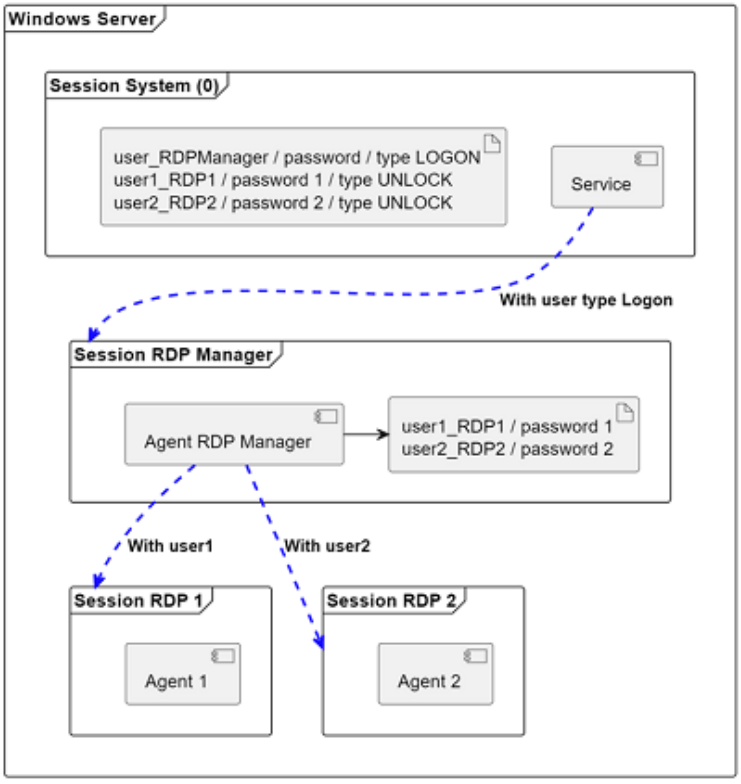
1. Open desktop agent 3.
2. Choose **Settings** > **System**, and set **Automatically start at Windows logon**.
3. Choose **Settings** > **Mode**, and choose **Activate** on the **RDP Manager Mode** bar.
4. Choose **RDP Sessions** and **Add an RDP session**.
5. Enter the required data. The **Session Name** is for information only.
6. Choose **Add**.

Example on How to Register Users in the Service

Find two possible configurations for Windows SAP desktop agent service.

Configuration 1

- Uses a single Windows server machine to open multiple RDP sessions.
- Automatically opens of the RDP manager session.
- Session RDP 1 runs with agent version 2 and Session RDP 2 runs with agent version 3.



This configuration has the following options:

- Option 1: Register all users in the service in the session where the RDP manager is running.

  Sample Code

  ```
  C:\Program Files (x86)\SAP\IntelligentRPA\SAPDesktopAgentService\CxStoreCred.exe userId= user_RDPMa

  C:\Program Files (x86)\SAP\IntelligentRPA\SAPDesktopAgentService\CxStoreCred.exe userId= user1_RDP1

  C:\Program Files (x86)\SAP\IntelligentRPA\SAPDesktopAgentService\CxStoreCred.exe userId= user2_RDP2
  ```

- Option 2: Register each user in each session.

  Session RPD Manager:

  Sample Code

  ```
  C:\Program Files (x86)\SAP\IntelligentRPA\SAPDesktopAgentService\CxStoreCred.exe userId= user_RDPMa
  ```

  RPD Session 1:

  Sample Code

  ```
  C:\Program Files (x86)\SAP\IntelligentRPA\SAPDesktopAgentService\CxStoreCred.exe userId= user1_RDP1
  ```

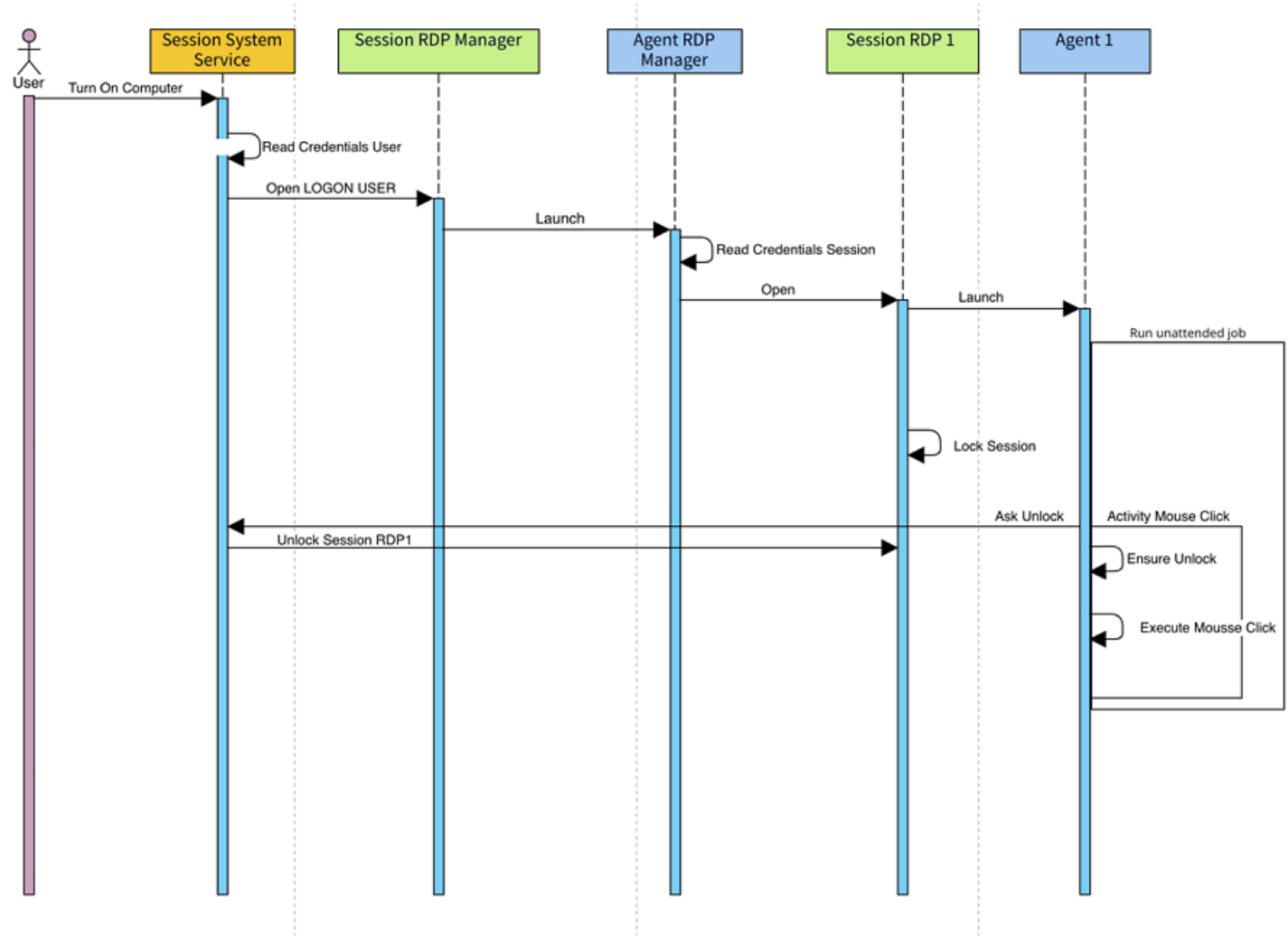
  RDP Session 2:

  Sample Code

  ```
  C:\Program Files (x86)\SAP\IntelligentRPA\SAPDesktopAgentService\CxStoreCred.exe userId= user2_RDP2
  ```

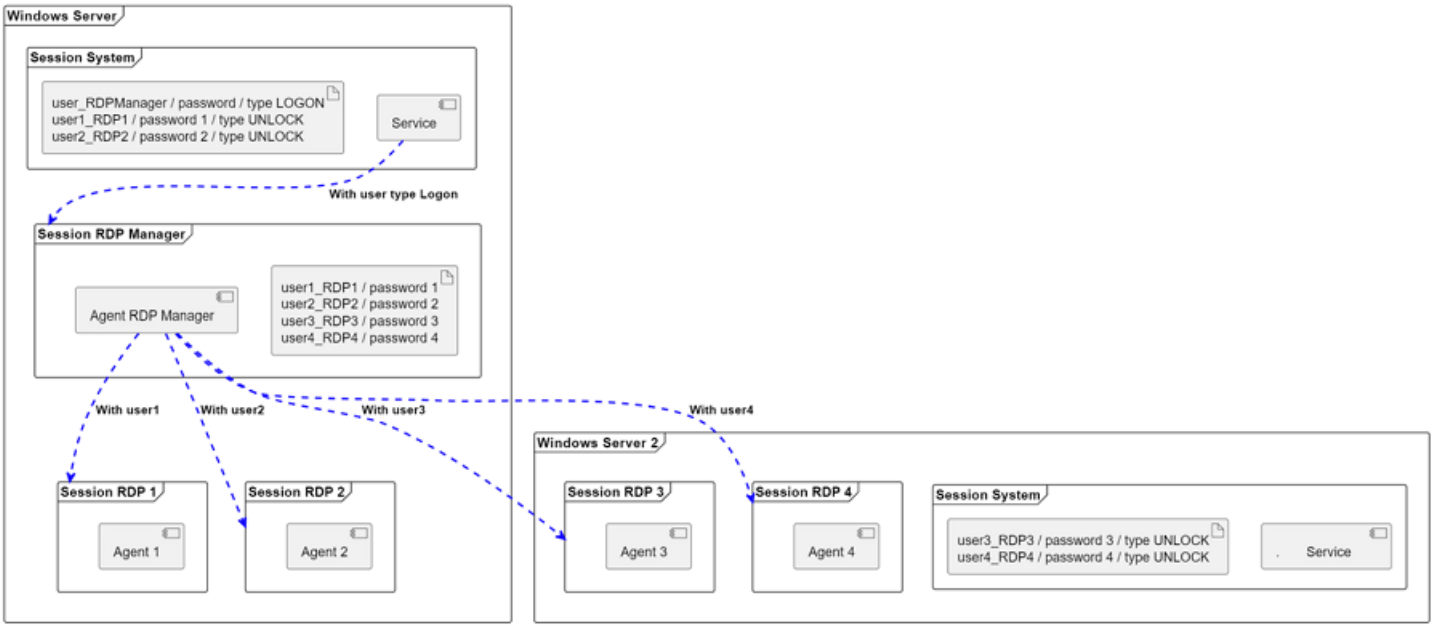
Flow Diagram

This is a graphical representation of the interactions during the execution phase of an unattended job. Agent 1 and the session RDP manager are on the same machine.



Configuration 2

- Uses two machines with multiple sessions on each machine.
- Automatically opens the RDP manager session.



For this configuration, the following applies:

- User1_RDP1 and user2_RDP2 are registered in the Windows server machine.

- User3_RDP3 and user4_RDP4 are registered in the Windows server 2 machine.

Change the Agent Mode

Using the desktop agent 3, you can change the mode of your agent.

The agent can be in three different modes:

- **Attended mode**: In this mode, you can launch the automation jobs by yourself or by configuring events to trigger them. This mode allows you to manually restart a project.
- **Unattended mode**: In this mode, the agent runs automations automatically and without your intervention.
- **RDP Manager mode**: In this mode, you can register and manage different Windows sessions on your agent with the RDP manager. If your agent is in RDP Manager mode, it cannot run automations.

  Note

  You can only choose to activate the attended, unattended and RDP Manager modes from the Agent 3. For more information, see Activate the RDP Manager Mode.

- **Design mode**: In this mode, you can design and troubleshoot your automations.

  Note

  This mode is only activated automatically when you test an automation.

  The agent will exit design mode and return to the previously-chosen mode (attended or unattended) after 30 minutes of inactivity.

Change the Agent Mode

If you want to put your agent in attended, unattended or RPD Manager mode:

1. Click **Settings** and select **Mode Settings**.
2. Hover over the mode you want to activate.
3. Click **Activate**.

You've successfully changed the mode of your agent.

Activate the RDP Manager Mode

Activate RDP Manager Mode to register and manage different Windows Remote Desktop Protocol (RDP) sessions on different computers to perform actions when an automation is running.

RDP Manager Mode is best suited for unattended jobs. Use it to run agents installed on remote machines.

An automation is running on a remote machine and the agent performs graphical actions such as a mouse click. If the computer screen is locked, the agent cannot perform a graphical action. The **RDP Manager** mode solves the previously mentioned issue. The mode allows you to register and manage secondary Windows sessions directly from your main session agent. In each session you add, an agent can run automations and perform graphical actions.

Note

- The RDP Manager uses a Windows Remote Desktop Connection to keep RDP connected.
- The RDP Manager checks every 30 seconds if the declared connections are open, and opens them if they're closed.

- The RDP Manager retries each declared connection twice and stops if it fails on the second try.
- If no user is connected, the RDP Manager uses the Windows remote desktop connection to reopen the RDP connection again. If a user is connected, the RDP Manager does not disconnect that user.

⚠ Caution

We recommend using only one agent in RDP Manager mode to manage the same set of connections. Using more than one can work but is not guaranteed by SAP, as it relies on the underlying Microsoft tools and protocol.

Register, Manage, and Activate RDP Sessions

Register, manage and activate RDP sessions in desktop agent 3.

Procedure

1. From your agent in the systray, choose **Settings**, **Mode** and **Activate** the RDP Manager Mode.
2. A new tab **Sessions** appears in the agent where you can see all of your registered RDP sessions.
3. Choose **Add an RDP session**.
4. Enter a name for your session, the user name and password that the agent needs to open an RDP session on another computer. Choose **Add**

   i Note

   Opening sessions using the secondary users closes the session used with the RDP Manager.

Results

The agent opens the RDP session that you have created. You can open the window of the RDP session by choosing and **Activate**.

Once the RDP session is connected, go to the **RDP Sessions** in the agent. The tool tip of the icon in the **State** column indicates the connected client.

Monitor the Agent

You can directly monitor your agent from the **History** tab.

You can click the **History** tab and select **Event History**. From this tab, you are able to see all the events related to the activity of your agent.

If you click the **History** tab and select **Job History**, you can see all the details about the jobs executed by your agent.

Manage Traces

Desktop agent 3 continiously records traces. You can use traces to monitor the agent and jobs.

Prerequisites

You have activated the support mode. See Activate Support Mode

Context

Traces are continiously recored. However, you can only access the tracer settings, if you activated the support mode.

Traces are encrypted and stored locally by default and can later be pushed to the cloud. If you disable encrypted traces, as described below, unencrypted traces are stored locally and cannot be sent to the cloud.

The trace recorder in the desktop agent traces different types of traces of several components.

> **Note**
>
> You can also manage tracer settings on the trigger, that means the tenant. If you manage traces on both, the agent and the tenant, the tenant has priority over the agent.

Procedure

1. In the desktop agent, choose **Settings** and **Tracer**.
2. **Optional:** Disable **Encrypt Traces**. This can be helpful, if you want to troubleshoot locally in case of a connection issue.
3. **Optional:** Enable **Show Trace Viewer**, if you want to show traces in real time.
4. **Optional:** Choose the components you want to record. You can further enable **Detailed** tracing for each component. However, we recommend that you do not enable **Detailed** tracing because it can slow down the trace recorder.
5. **Save Settings**.

Results

- If you changed and saved the default settings, the trace recorder stops and starts a new trace recording with your settings.
- You can view the completed recordings by choosing **History** and **Reports**.

More Information

You can view the ongoing trace by choosing **History** and then **Reports**.

To stop the ongoing trace and begin uploading it, choose *More* and then click **Upload Trace**.

If the agent is connected to a tenant, you can upload encrypted traces by choosing *More* and then clicking **Upload Trace**. However, if the agent is not connected to any tenant, you can choose *More* and then click **Get Traces for Manual Upload** to download the traces and manually upload them to the cloud service.

You can open unencrypted traces in the trace viewer by choosing *More* and then clicking **Open in Trace Viewer**. To download traces, choose *More* and then click **Get Traces**. If the traces fail to be sent to the cloud service, a warning message is displayed by clicking the *Warning* icon.

To reset all filters, click the *Reset All Filters* button.

If the support mode is not activated during the execution of an automation in the Cloud Studio, you can open the screen tester on design mode on the agent side. To open the screen tester, choose *Tools* and then click **Open Screen Tester**.

Map Agent Groups to an Identity Provider

If you use an identity provider (IDP) to manage details about your users such as names, departments, and roles, you can map your IDP to agent groups to ease user management in SAP Build Process Automation.

Context

If you map agent groups to your organization's IDP once, the agent groups can use the information stored in your IDP.

Procedure

1. In the SAP BTP cockpit, establish trust between your IDP and the one used by SAP BTP.
2. Choose a SAML attribute you want to map to the agent group, for example **Division** and expose it in the SAML assertion:
    a. Create a custom role based on the `IRPAAgentUser` role template.
    b. After creating the role, include the role in the role collection. Edit the **AgentUserIdp**, which is in the role you created by choosing it: In the **agent_group** attribute, carefully type in the name of your SAML attribute. We recommend that you copy and paste the attribute name, because if the typed name does not match the attribute, the mapping does not work.
    c. Create a custom role collection by using the standard BTP procedure, and choose the role that you just created.
3. In SAP Build, choose **Settings** and **Agent Groups** to create agent groups, to map them to the values of your SAML attribute.
    a. Choose **Add Agent Group**.
    b. Enter the exact name as the value of the SAML attribute, choose **Type: Any** and **Create**.

Results

You have mapped your agent group to your IDP.

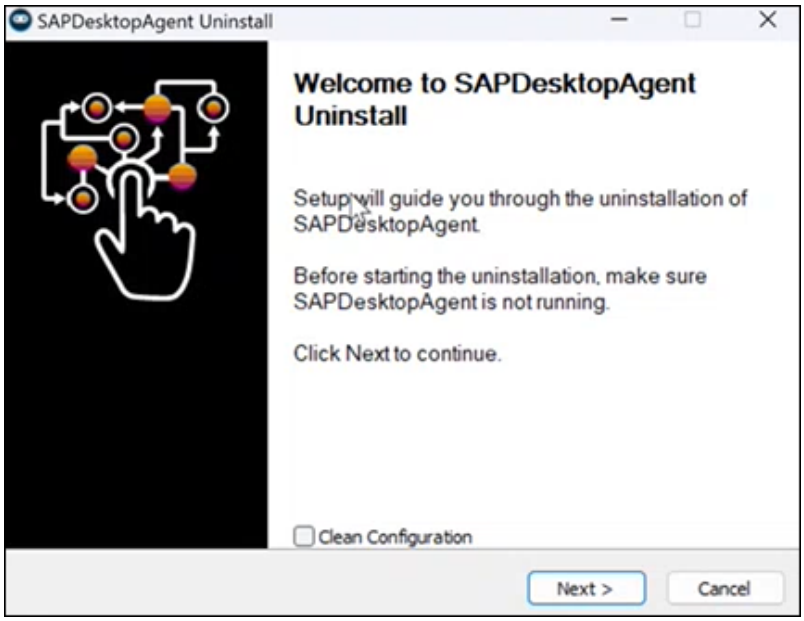
Best Practices for the Desktop Agent 3

In this section, you will find our recommended best practices for using the agent 3.

- We strongly recommend that you uninstall the desktop agent 2 from your computer before installing the desktop agent 3.

  However, if you want to have both desktop agents on your computer at the same time, for example if you are testing the migration from desktop agent 2 to desktop agent 3, note that there are some limitations. For more information about these limitations, see SAP Note 3238616.

- You can choose to clean up any sensitive or personal stored data such as storages, local cache, registry keys, credentials and so on while uninstalling agent 3 from your computer. To do so, you must check the **Clean Configuration** checkbox on the uninstallation window.



Register Authentication on Agent 3

The list of authentications defined in the Cloud Factory for a specific tenant is displayed in agent 3 when it is connected to that tenant.

- To view the list of authentications, go to the **Control Tower** tab and select **External Authentication**.
- Initially, the **Status** is displayed as **Not registered**. For instance, the authentications with the type **GOOGLE** are the ones specific to Google.

To register any authentication with Not Registered status, click the specific authentication and then, enter the email ID with which the input field will be connected to activate the Register button of that authentication. Next, click the save () button.

Once you click the **Register** button, the browser is launched for authentication and authorization. If the Chrome browser is installed, it is launched; else, the Edge browser is launched. Once registered, the status is displayed as **Registered**, and the authentication can be used in automation.

> **Note**
>
> It is recommended to enable the SPA extension on the Chrome or Edge browser so that the browser tab opened for authorization is closed automatically. Otherwise, the browser tab must be closed manually.

- The first authentication with the **Registered** status is set as a **Default** authentication automatically. To set a different authentication as a **Default** authentication, click (More options) under **Action** and then click **Set as default authentication** corresponding to that authentication.
- To unregister any existing registered authentication, click **Unregister**.

For more information on how to authenticate authentications using agent 3 UI, see Create External Authentication.

About the Desktop Agent 2

The Desktop Agent is a component of SAP Intelligent Robotic Process Automation that is installed locally on a computer. The agent starts automation projects that launch and run applications, reads information from screens, enters data, chooses options, and processes data.

Projects are assigned to tenants running on a Desktop Agent. You can check the status of your desktop agent in the system tray (systray), that is accessible while your desktop agent is ready or active.

After installation, the desktop agent is configured to start at Windows logon automatically. Don't change this configuration because your agent can be assigned background (unattended) jobs.

The first time you launch the Desktop Agent on your workstation, you're prompted to log in. If you're unsure of your login details, contact an administrator.

Related Information

The System Tray
Running Projects
Status of the Desktop Agent

Status of the Desktop Agent

The icon of the systray indicates the status of the desktop agent.

Icon's Appearance	Meaning
Gray	The desktop agent is in idle mode - no project is loaded.
Purple	A project is loaded.
Blue circle superimposed on purple icon	The desktop agent is running a job in interactive mode.
Red circle superimposed on purple icon	The job failed.
Red circle superimposed on gray icon	An error occurred.
Orange circle superimposed on gray or purple icon	A warning about the job was issued during the connection phase.

> **Tip**

You can read error and warning messages in the About window.

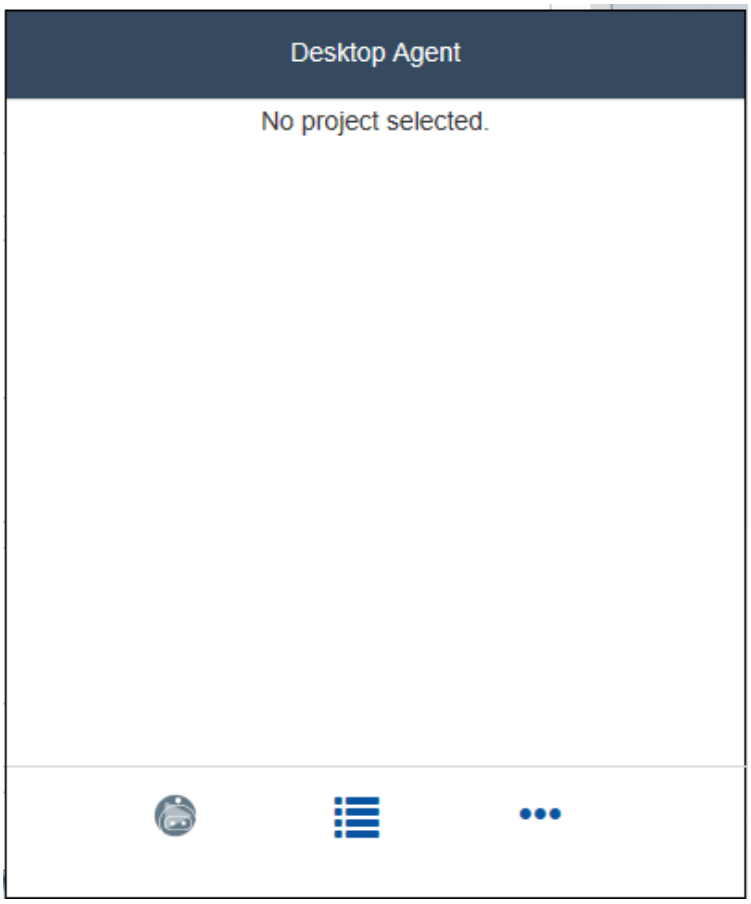
Related Information

The About Window

The System Tray

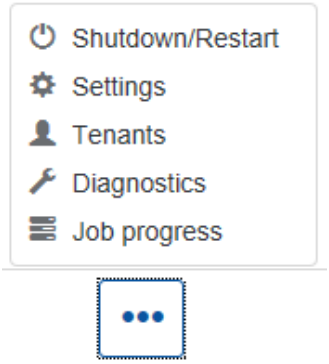
The system tray (systray) gives you access to the desktop agent.

When no project is selected, the desktop agent window looks like this:



- Choose the desktop agent icon to open the About window.
- Choose the projects icon to open the projects window.
- Choose the three dots for more actions.

  The following list appears:



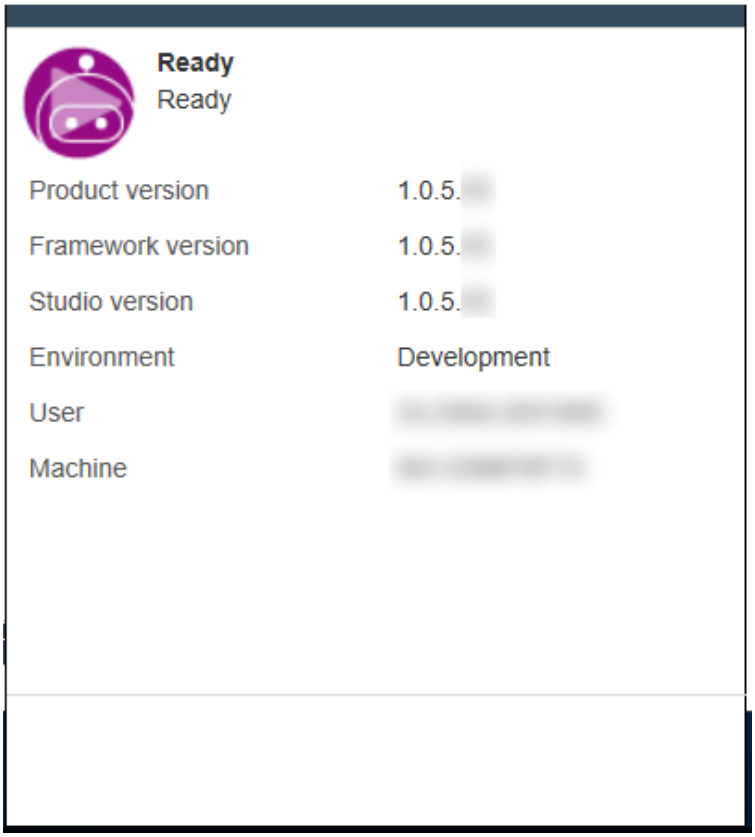
When a project starts, the desktop agent window displays project-specific menus to run jobs.

Related Information

The About Window
The Projects Window
More Actions

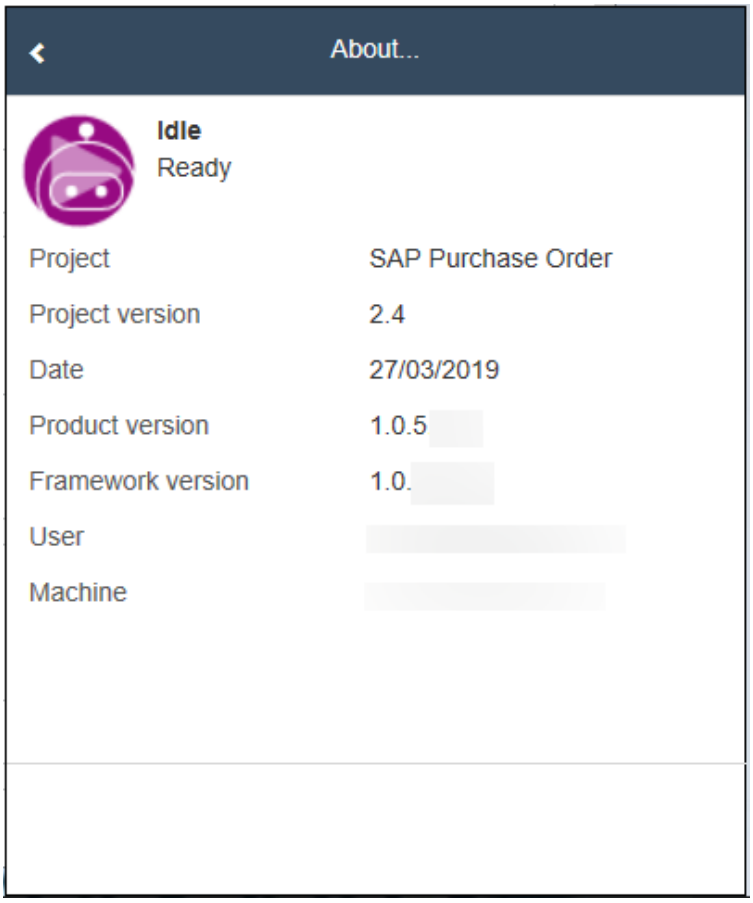
The About Window

If no project has been started, the About Window looks something like this:



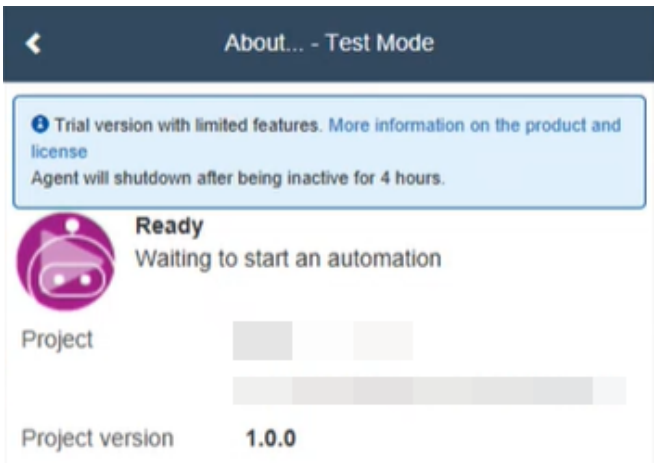
The icon's color indicates the status of the desktop agent, which is also displayed at the top of the window next to the icon (for example, Paused or Ready).

When a project has been started but no jobs are currently running, the About window looks like this:



During or after jobs are executed, error or warning messages are displayed in the About window. If a warning occurs, an orange circle appears superimposed on the desktop agent icon. If there's an error, the circle is red.

If your desktop agent (from version 2108) is running, it automatically shuts down after 4 hours of inactivity. If the desktop agent is running jobs, the agent will shut down after the end of the last job. A message strip is displayed at the top of the systray to explain that the desktop agent will shut down after 4 hours of inactivity.



If the desktop agent (from version 2108) can't disconnect from the Cloud Factory, the Cloud Factory tries to disconnect it.

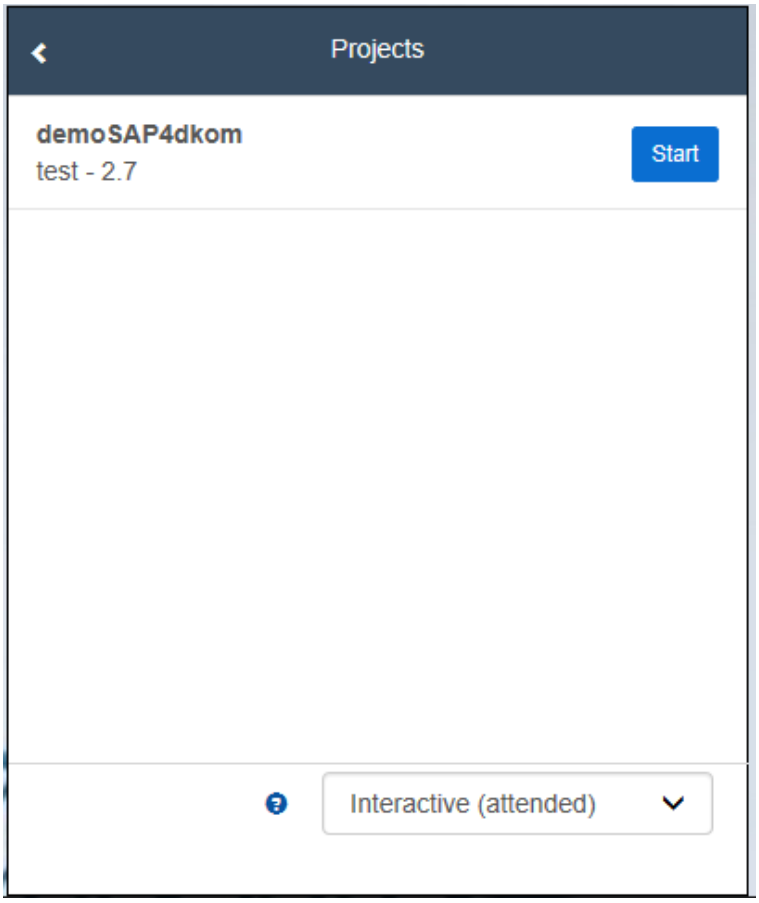
Desktop agents whose version is older than 2108 gets disconnected by the Cloud Factory after 4 hours of inactivity. However, you must manually restart them.

Related Information

Status of the Desktop Agent
Running Projects

The Projects Window

If one or more projects are available, the Projects window looks like this:



Press the **Start** button next to the project you want to start.

Note

You can switch modes using the dropdown list at the bottom of the window. By choosing the question mark icon to the left of the dropdown list, you can view details of the modes. For further information, see Running Projects.

Confirm that you want to switch to this project by choosing **OK** in the next window.

The desktop agent icon in the notification is of the Windows taskbar turns pink and its hover text is the name of the project.

To stop running a project, return to the Projects window and press the **Stop** button.

More Actions

Additional actions can be selected from the list displayed when you press the three dots at the bottom right of the main desktop agent window.

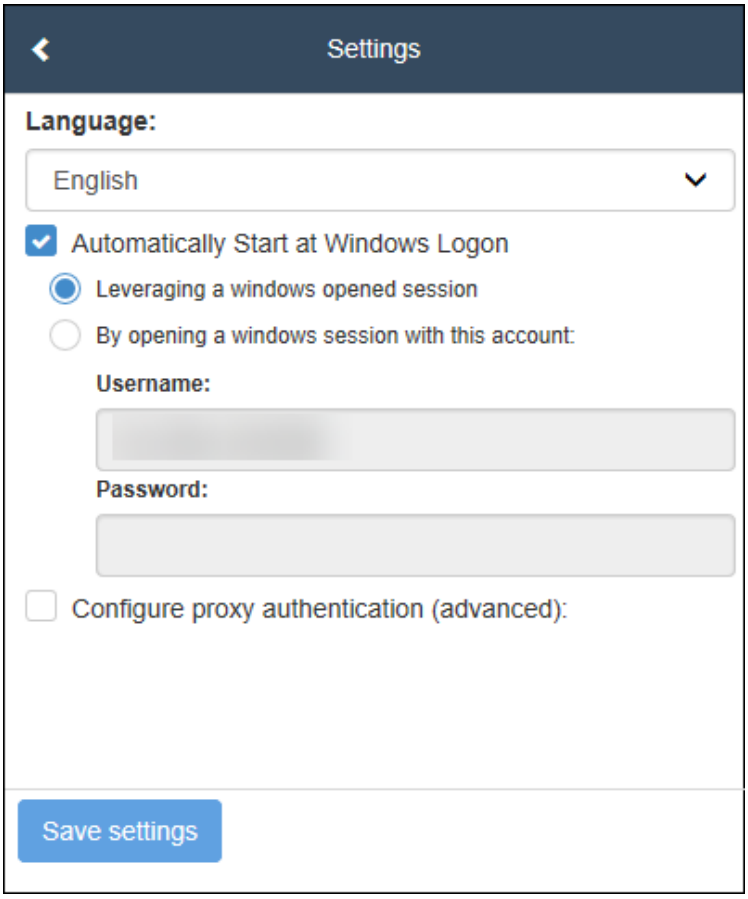
Shutdown/Restart

If you choose this action, you can either shutdown or restart desktop agent, or press **Cancel** if you decide to do neither.

Caution

To avoid data loss, stop the running project from the Projects window before shutting down or restarting desktop agent.

Settings



In the Settings window, you can specify whether desktop agent should automatically start at Windows logon. You can also configure a proxy authentication.

⚠ Caution

Keep this setting checked if jobs will be run on your agent in background (unattended) mode.

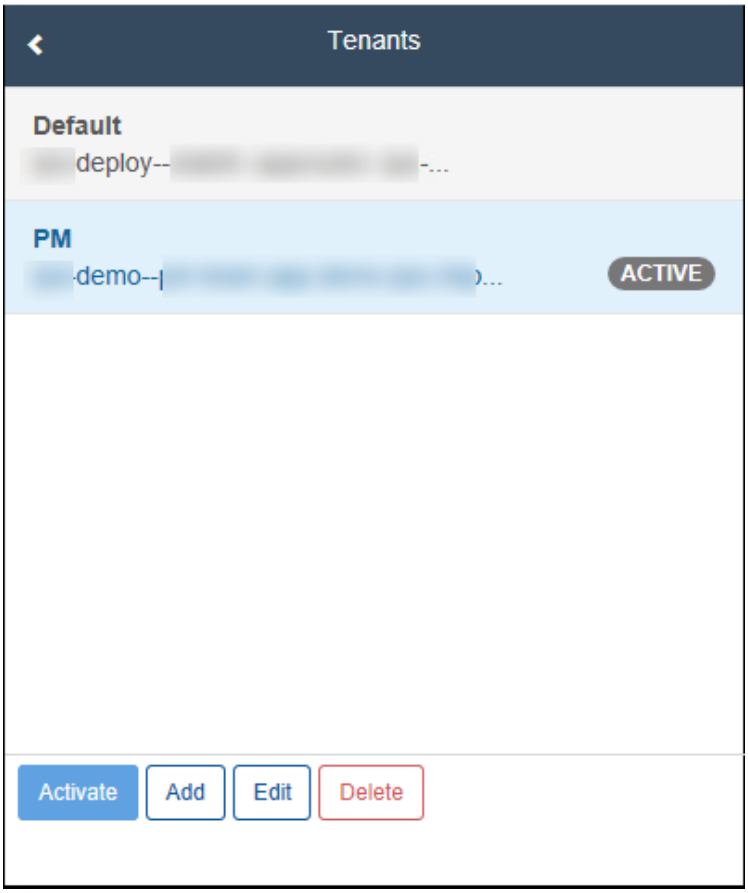
You can also select the language via the dropdown list.

When you change any settings, the Save Settings button is enabled so that you can save your preferences.

→ Tip

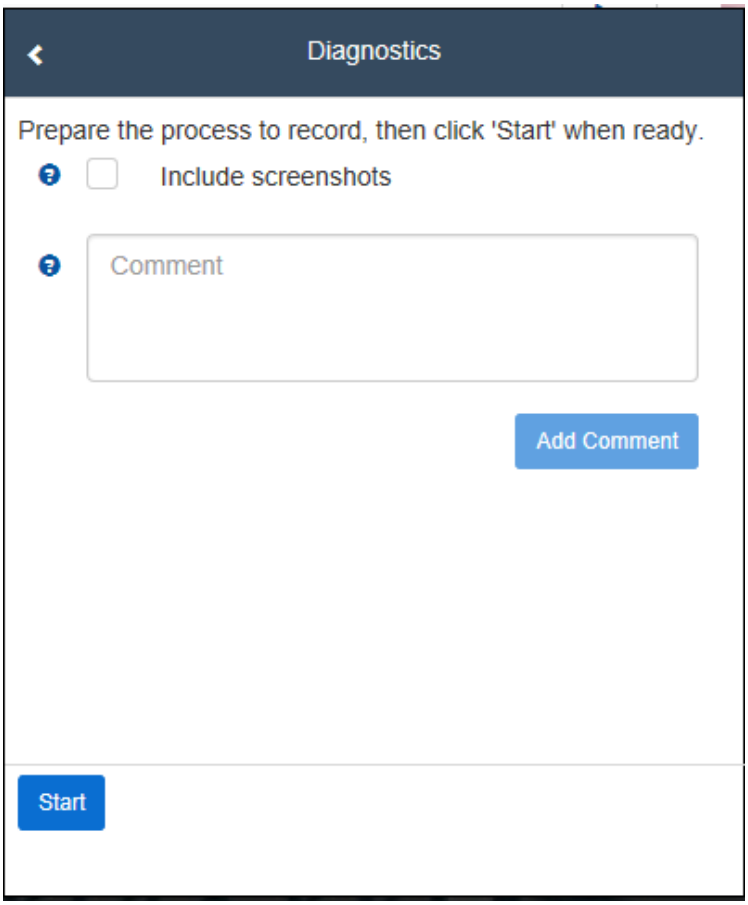
Press the arrow at the top left to leave the Settings window.

Tenants



If there are multiple tenants associated with your desktop agent, you can activate, add, edit, or delete tenants in this window.

Diagnostics



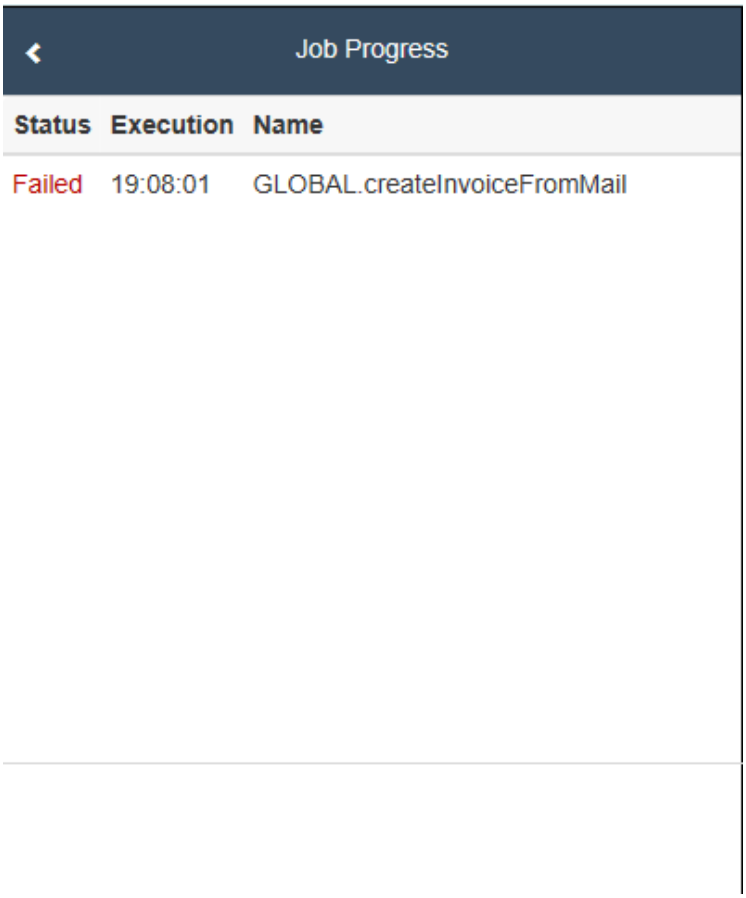
If a job isn't running correctly, you can record its execution from the **Diagnostics window**. Press the **Start** button at the bottom to begin recording. The recording is saved to a diagnostics file that you can forward to the support team. You can include screenshots by choosing

the corresponding checkbox, and add comments about the issues encountered in the box provided.

→ Tip

Press the arrow at the top left to leave the Diagnostics window.

Job Progress

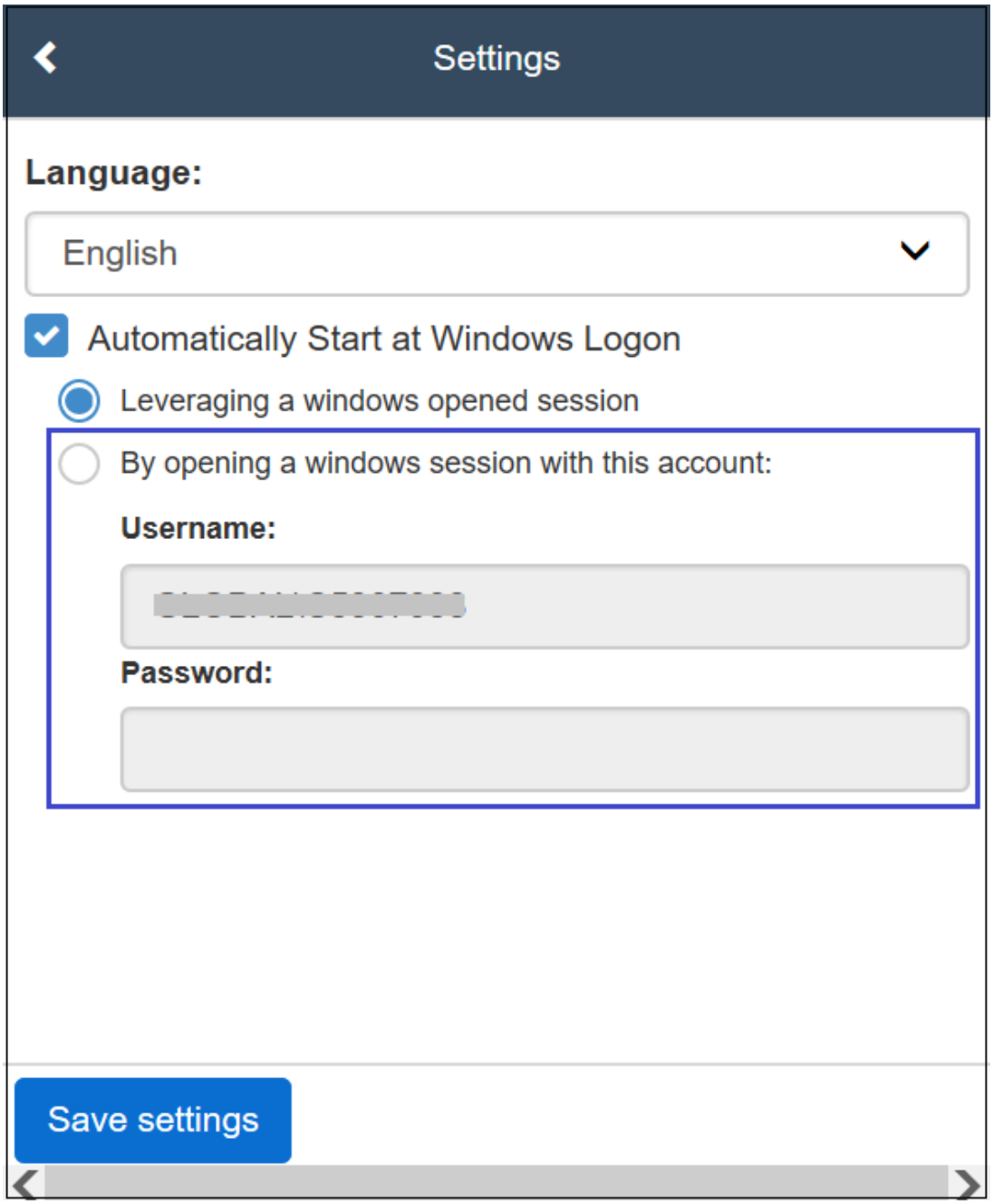


While a project is running, you can view the status of jobs in the Job Progress window. The time of execution is shown in the column below the label Execution. Name refers to the name of the job, which can be a scenario or a user task, for example.

Set Windows Password in Desktop Agent

You can set your Windows password in desktop agent.

To set your Windows password, the desktop agent must be configured to run in unattended mode. The Windows password set option is only available if the agent is configured in unattended mode.



Note

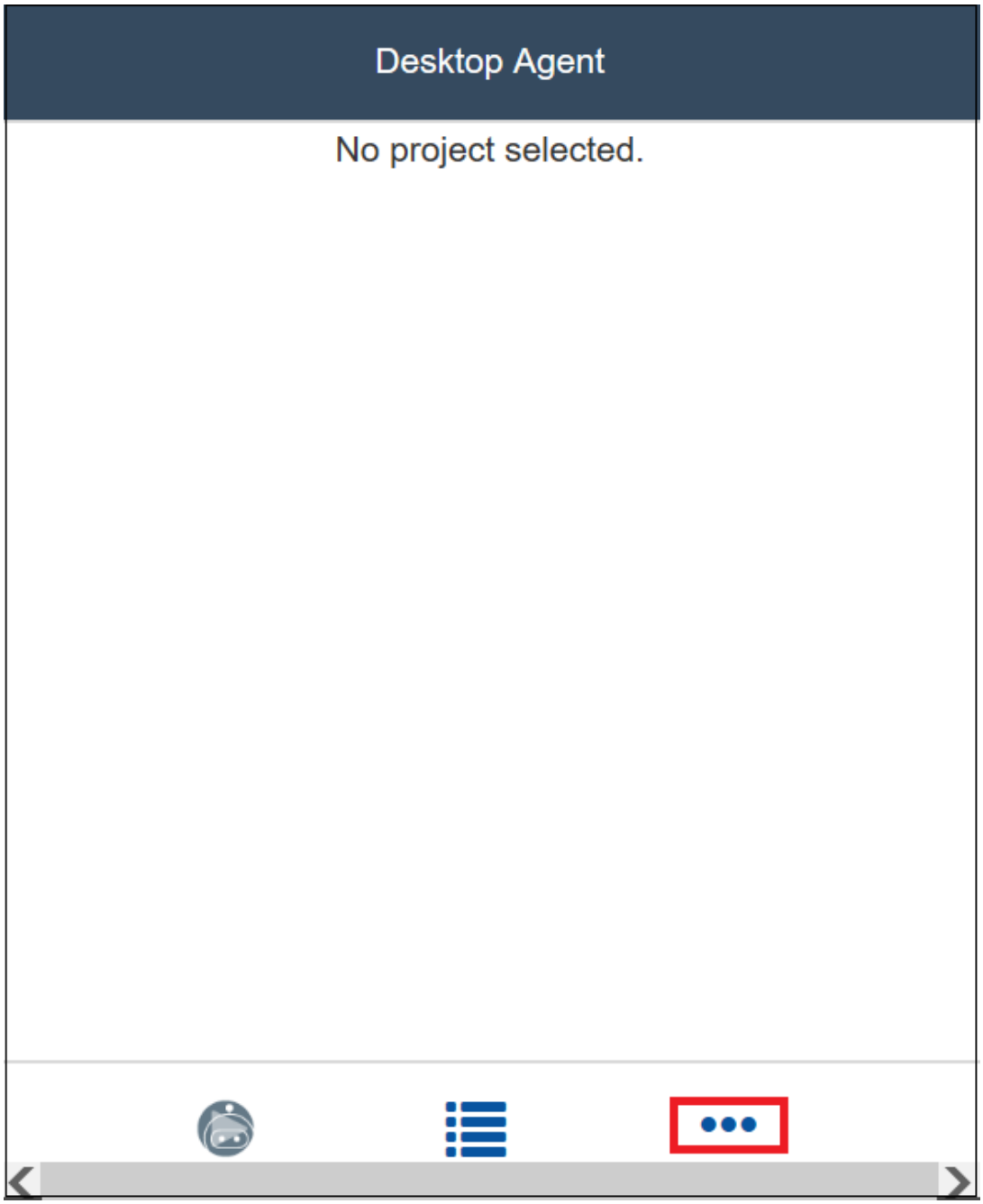
To configure unattended mode, on the **Projects** screen, select **Background (unattended)**.

Note

If the default option, leveraging a windows opened session, is selected, then you must open Windows manually for the desktop agent to start automatically.

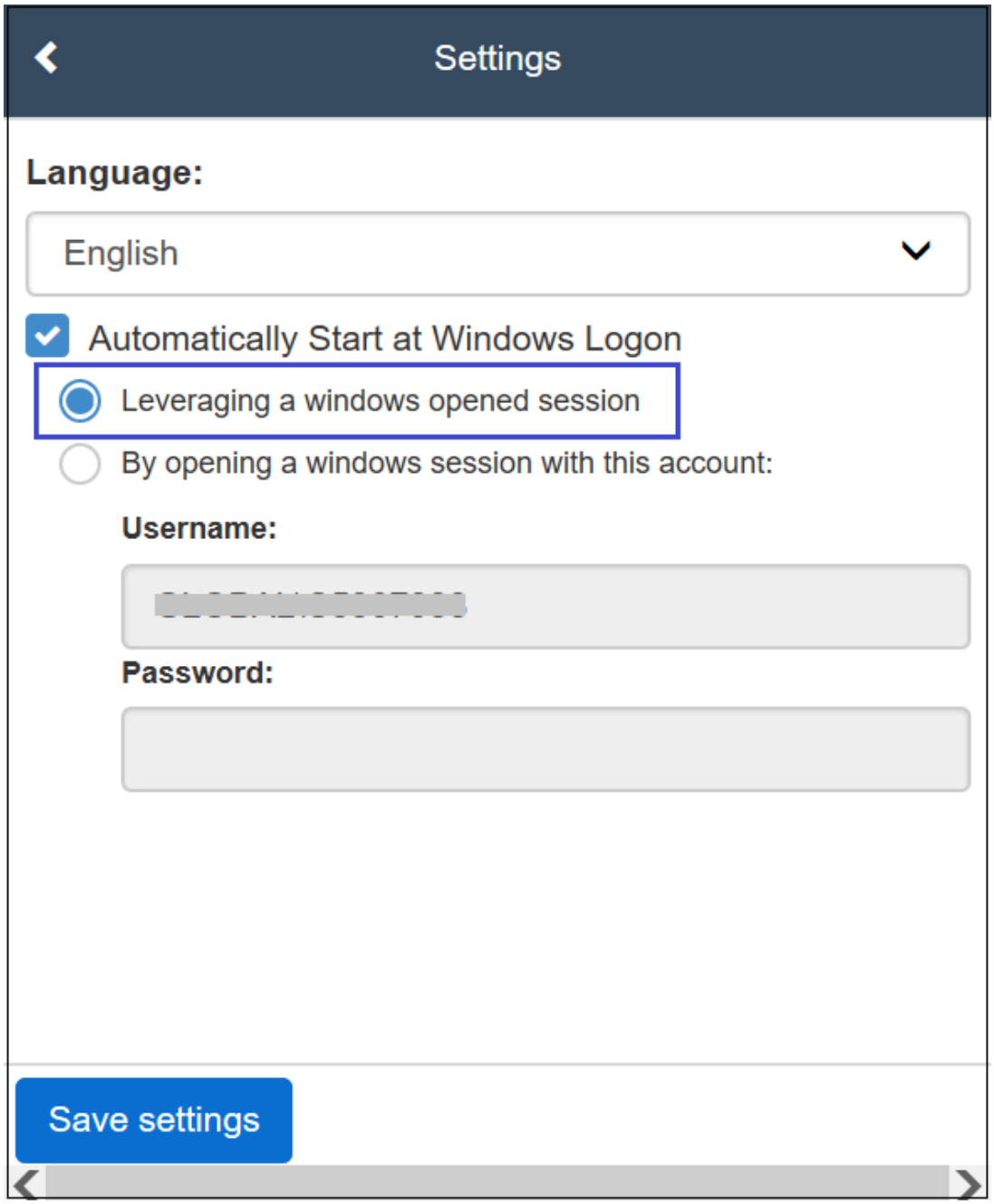
Procedure

1. Open a desktop agent Systray.
2. Choose the More Actions button.



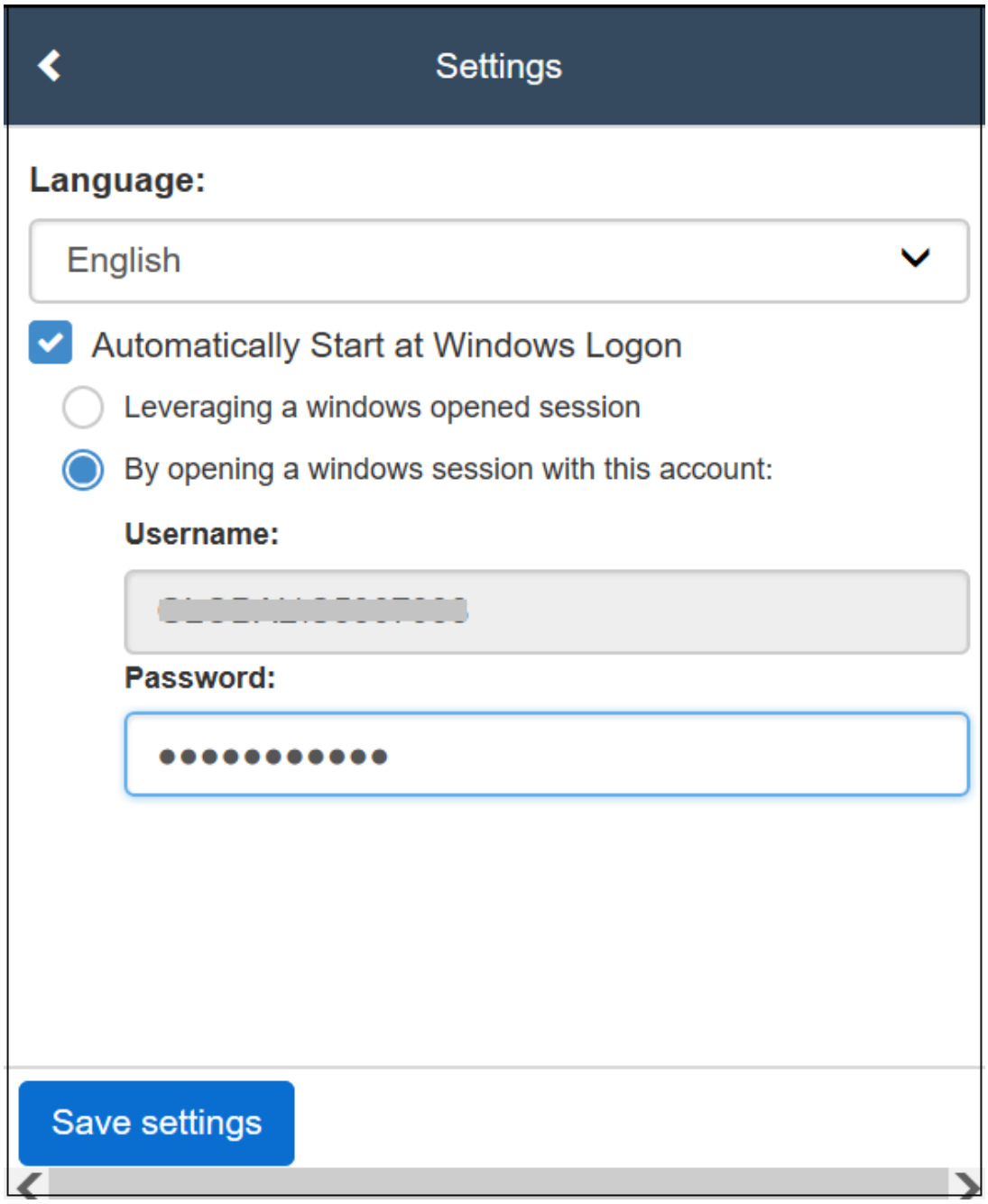
And then choose **Settings**.

The **Settings** window is displayed and the system default option is preselected.



3. Select the By opening a windows session with this account: option.

4. Enter your Windows password in the Password field.



Note

Username and Password are the credentials of the Windows session. These credentials (username and password) are stored in a hidden catalog of the Windows Credential Manager.

For more details about your Windows password, contact your system administrator.

5. Choose **Save settings**. The desktop agent window is displayed. For more details about the desktop agent window, see The System Tray section.

Note

You can store Windows credentials for launching desktop agent in unattended mode using the following command line:

```
cxStoreCred.exe <yourUser> <youPassword>
```

For more details, see the SAP Note, 3010368.

Remove Windows Password

1. Go to the **Desktop Agent Setting** screen.

2. Select **Leveraging a windows opened session**.
3. Choose **Save settings**. The Windows password will be removed from desktop agent.

ℹ Note

You can also remove the password using the following command line:

```
CxStoreCred.exe /forget
```

For more details, see**To remove a user/password** section in the SAP Note, 3010368.

Related Information

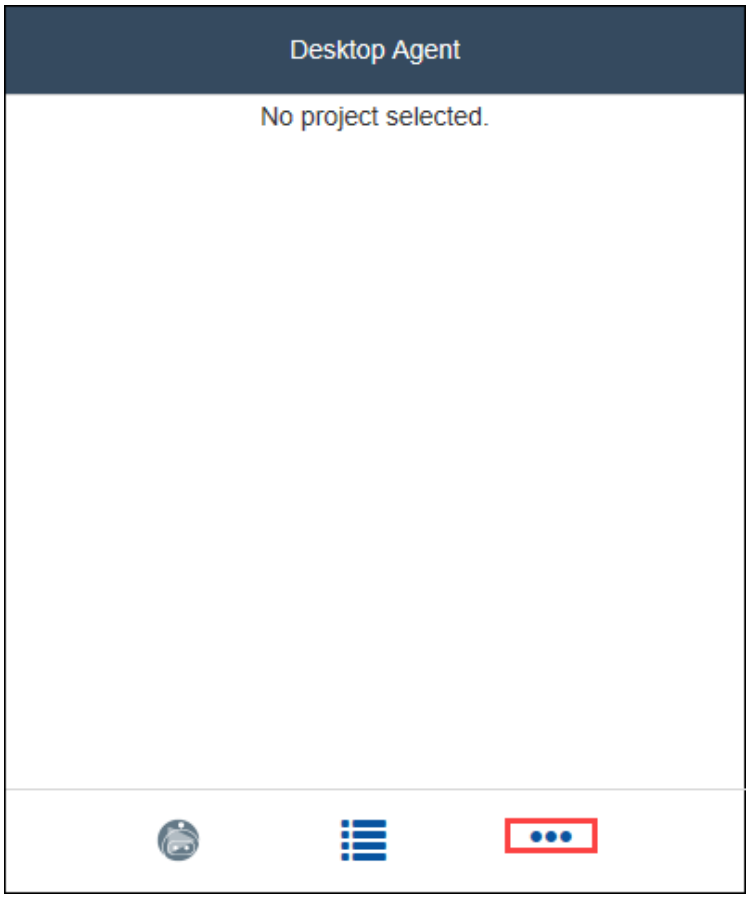
How to Set Up an Unattended Bot

Configure Proxy Authentication in Desktop Agent

If you use a proxy that doesn't use your default Windows credentials, you can configure the proxy authentication in Desktop Agent. It allows you to log on to the Cloud Factory.

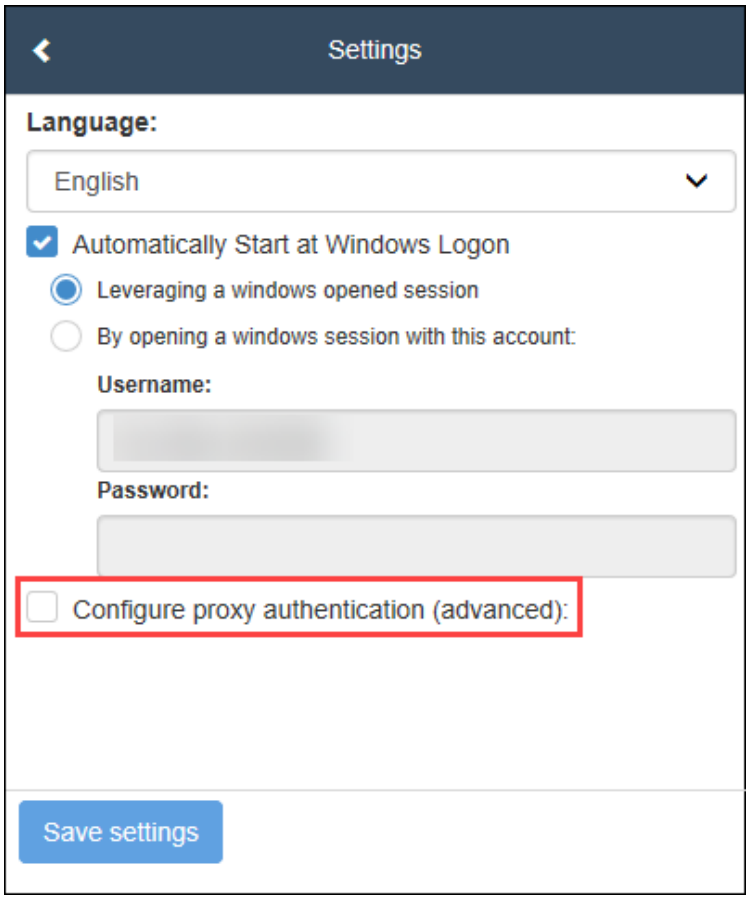
Procedure

1. Open a desktop agent Systray.
2. Choose the More Actions button.

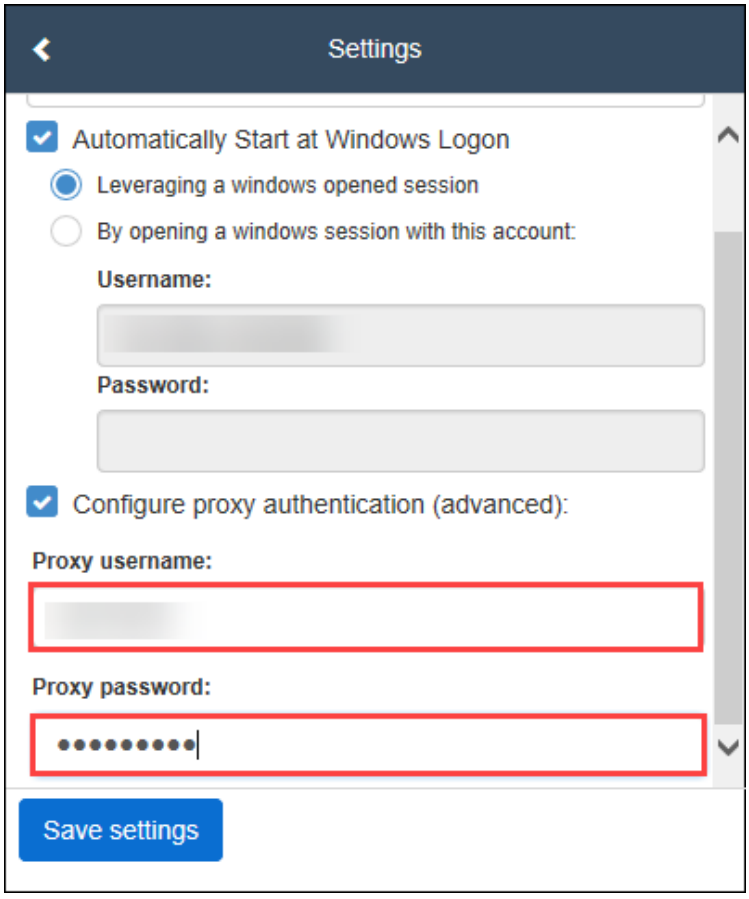


And then choose **Settings**.

3. Select the **Configure proxy authentication (advanced):** option.



4. Enter your Proxy username and Proxy password in the fields that appear.



5. Choose Save settings.

Running Projects

The desktop agent runs in two different modes: **Interactive (attended)** and **Background (unattended)**.

→ Tip

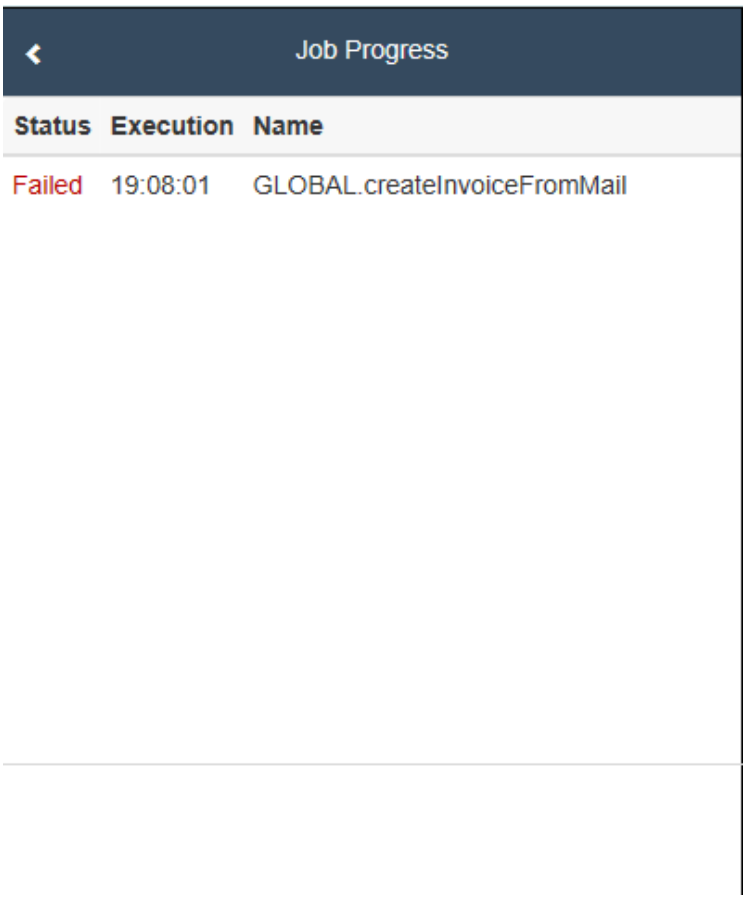
To view or change the mode, press the Projects icon to open the Projects window. At the bottom of this window, the current mode is shown in a dropdown list.

Interactive Mode

In Interactive mode, you have to select a project and run the project's scenarios manually.

To select a project, simply press the **Start** button next to the project name in the Projects window. When you have confirmed that you want to switch to this project, the main window of the Agent systray will change to display project-specific menus. There can be multiple menu items and sub menus from which you can run scenarios as jobs.

You can check the progress of a job by choosing **Job Progress** from the **More actions** menu (three dots). A window like this appears:



In this window, **Name** refers to the name of the scenario being run and **Execution** refers to the time when the scenario was executed.

To stop a project, press the Projects icon to open the Projects window and press the **Stop** button next to the current project.

Background Mode

When the agent is in Background mode, projects will run automatically according to a defined schedule, and you don't have to take any action.

Related Information

The System Tray

Best Practices for the Desktop Agent

When a desktop agent is running on a system, it causes limitations.

Here is a list of these limitations and the solution to them::

- You may face some errors while you're using the desktop agent. For more information on these errors and how to solve them, see: Desktop Agent.
- Within the desktop agent systray, error messages are reported in the About screen. For more information on these error messages, see: Desktop Agent Error Messages.
- Sometimes, the agent systray appears blank. For more information on how to solve this error, see: Desktop Agent Blank Systray.
- You may want to use your PC while a bot is running. For more information on how to do this, see: Using the PC while a Bot is Running.
- The system may reboot while the desktop agent is in unattended mode. For more information on how to start the desktop agent immediately after the system reboot, see: System Reboot during Unattended Mode.

How to Set Up an Unattended Bot

In some cases, it is useful to send jobs to a remote machine, for example, virtual machines.

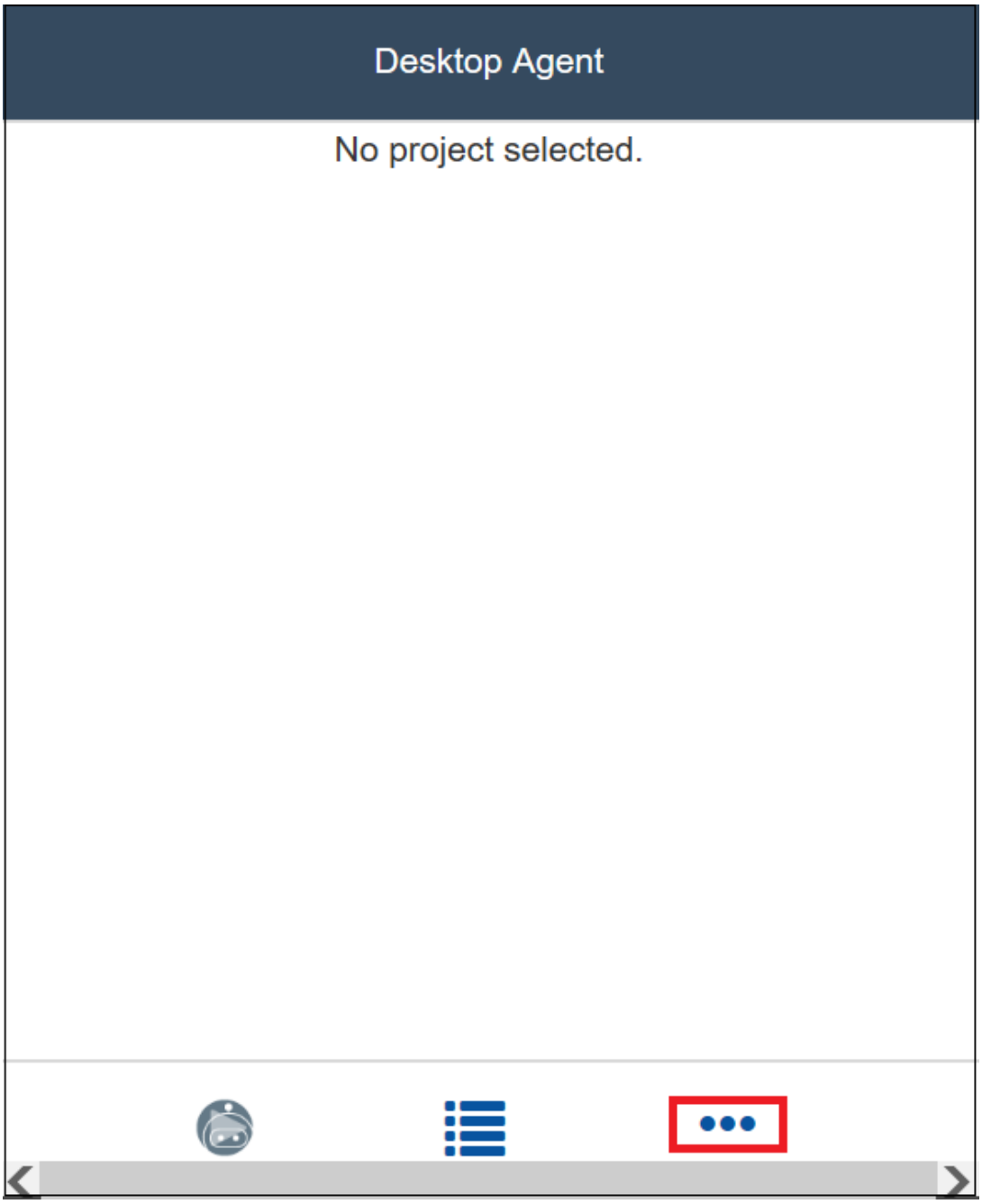
Context

If so, the desktop agent requires an opened Windows session to work properly. The following procedure ensures that the agent is correctly set up to perform the jobs on a remote machine.

> ⚠ **Caution**
> If the job to be executed contains some low-level actions, the agent must be able to unlock the Windows session. For more information, see SAP Note 3107785.

Procedure

1. Install the desktop agent on the remote machine. You can choose to install the components according to your needs. SeeOn-premise Components Setup.
2. Configure the SAP Intelligent RPA Factory and register the desktop agent.
    a. In the SAP Intelligent RPA Factory, create an Environment. See Create an Environment.
    b. In the SAP Intelligent RPA Factory, create an Agent Group. See Create an Agent Group.
3. Register the agent and then add the agent to an Environment. Follow the procedure below to register and add agent to an Environment.
    a. Open the desktop agent.
    b. Choose **More Actions**.

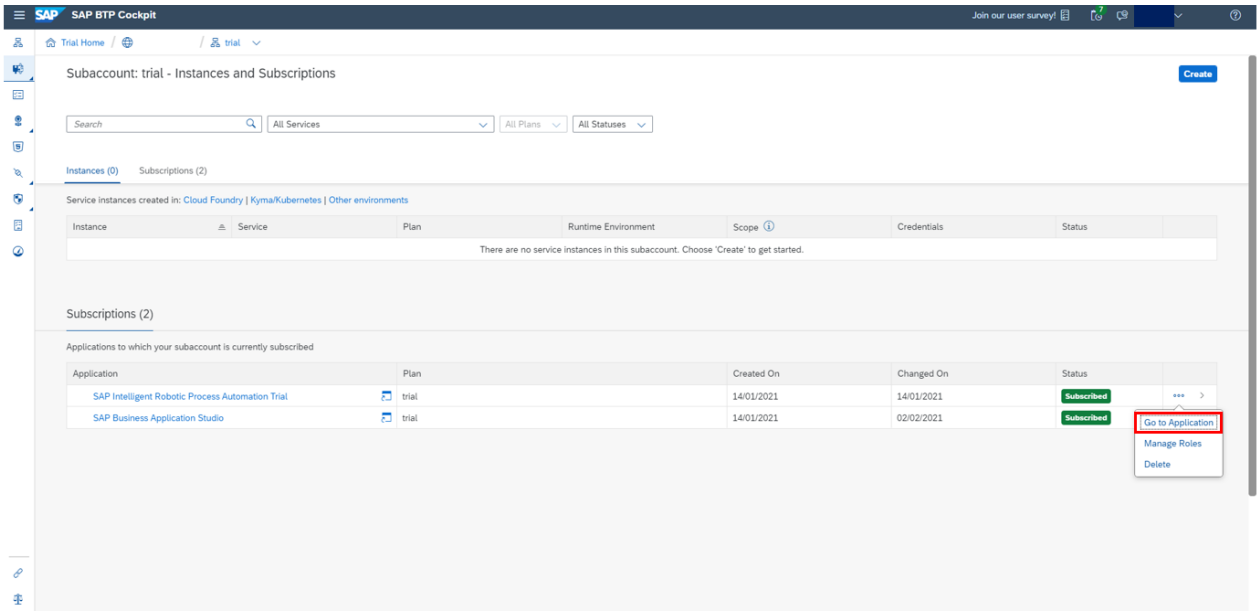


c. Select **Tenants**. The **Add** tenant screen is displayed.

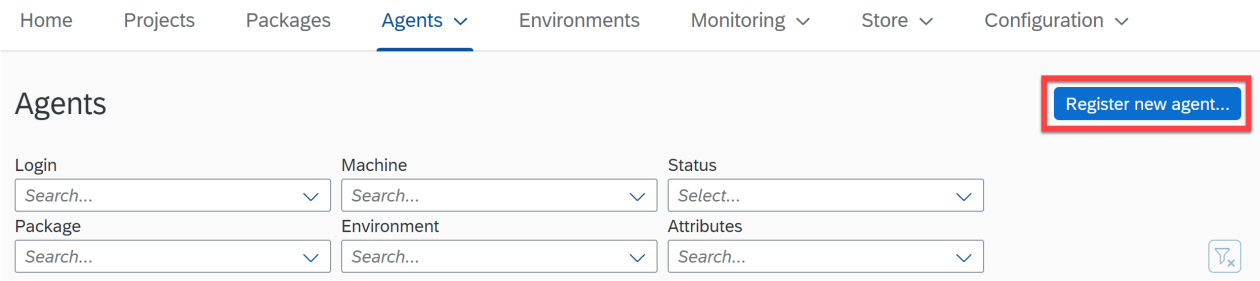
d. Provide the name of the tenant in the **Name** field and the tenant URL in the **Domain** field.

→ **Tip**

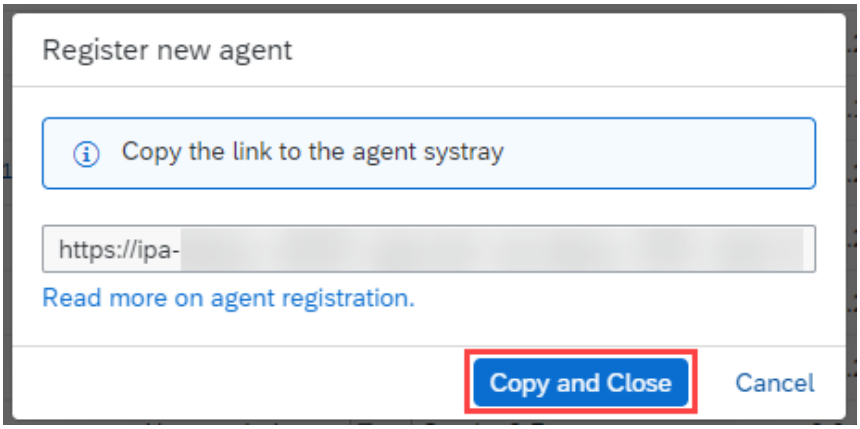
To find the SAP Intelligent Robotic Process Automation Factory tenant URL, select **Go to Application** in your SAP Intelligent Robotic Process Automation Factory account on the SAP BTP cockpit:



On the SAP Intelligent Robotic Process Automation Factory homepage, go to the **Agents** tab and then choose **Register New Agent** on the right-hand side of the screen.



When the pop-up opens, select **Copy and Close**.



Choose **Save**. The **Tenants** screen is displayed.

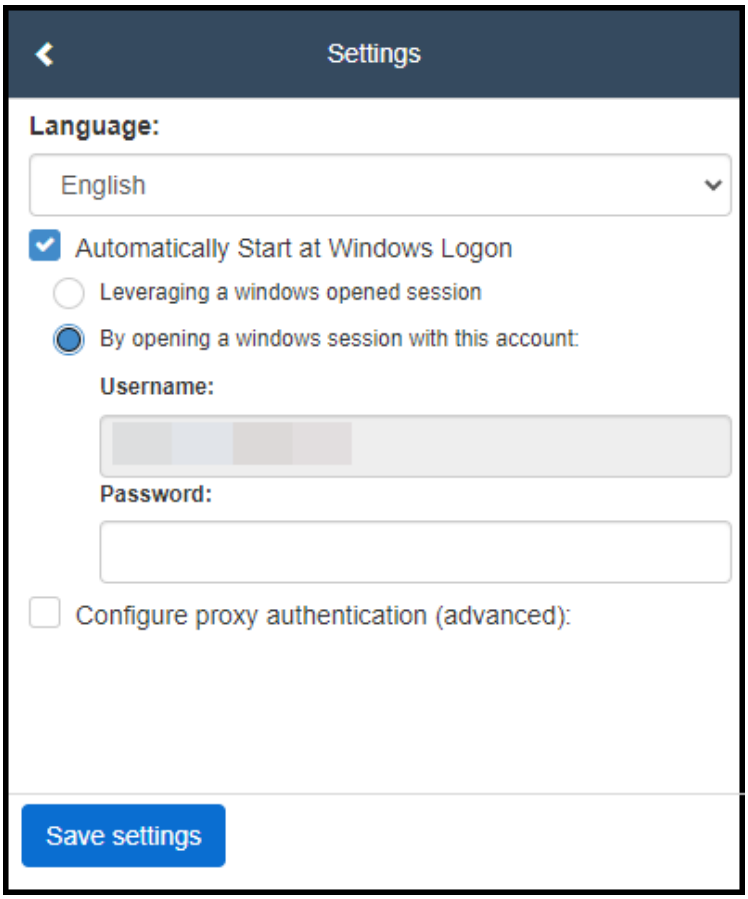
e. Select **Tenants**. The **Add** tenant screen is displayed.

f. Select the tenant and then click **Activate**. After successful processing, at the right side of the tenant, ACTIVE status is displayed

ℹ Note

The registration can be done automatically using the mass registration token. For more information, see Mass Agent Registration.

g. Once the desktop agent is registered, it must be configured to automatically start at Windows logon. To configure, select **Automatically Start at Windows Logon** and choose **By opening a windows session with this account**.



h. If needed, define the credentials of the account:

- Using CxStoreCred
  - Go to "C:\Program Files (x86)\SAP\Intelligent RPA\Desktop Agent".
  - Launch **CxStoreCred.exe** ***<user> <password>***.

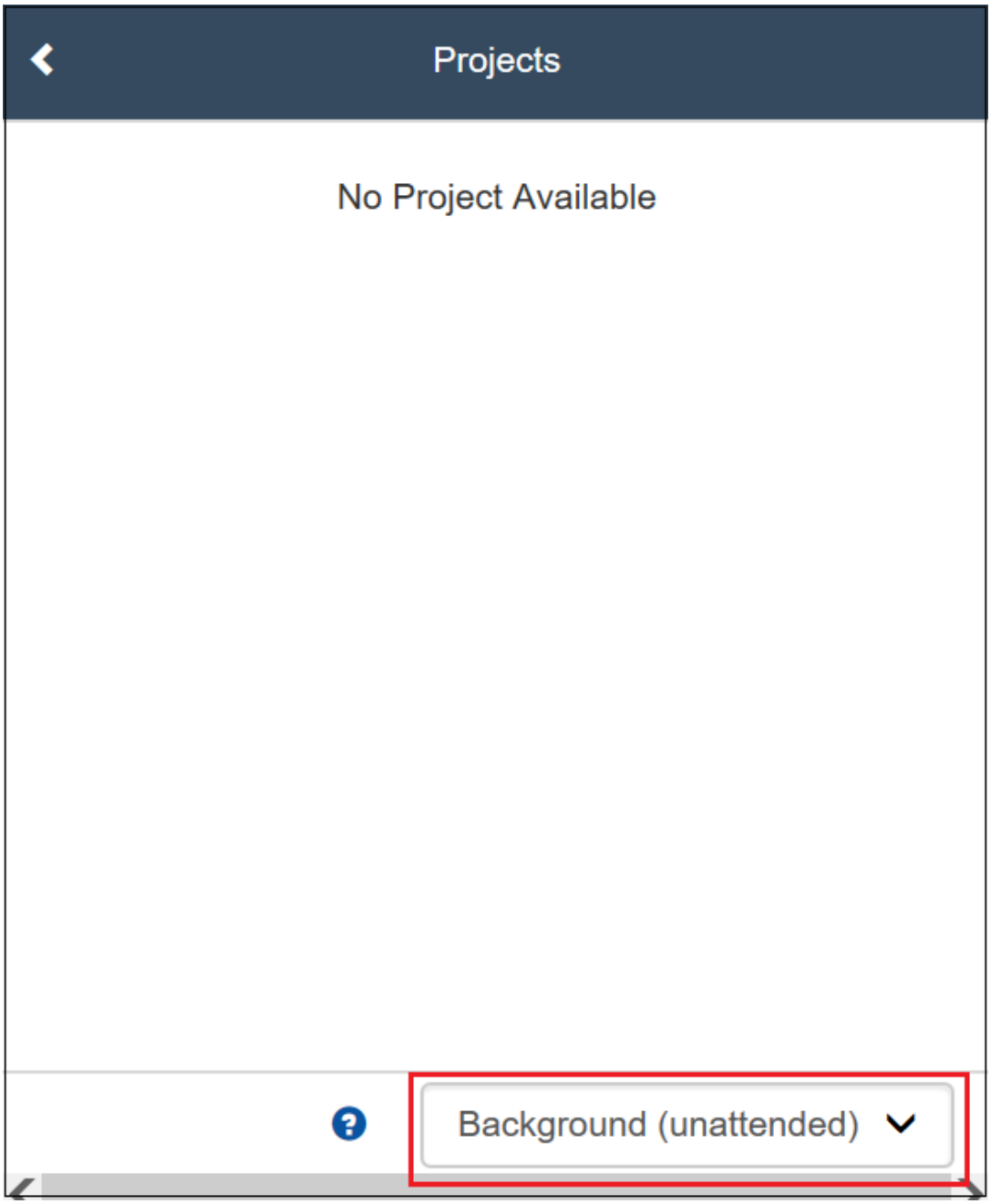
    You don't require the domain as Windows Domain is implicit. The password won't be verified in this command-line, you must ensure that the password is properly defined. After one attempt, the machine gets rebooted.
  - If you want to remove use: **CxStoreCred.exe** ***/forget***.
  - If you want to update an existing user password, simply enter again the command with the user and new password.
  - If you need to set or update a user (or a password) including space, you can use double quote such as **CxStoreCred.exe** "***My user***" "***My Password***".

    OR
- Using the settings from desktop agent. See Set Windows Password in Desktop Agent.

i. Ensure the desktop agent is running in unattended mode:

- Choose the **Desktop Agent Systray** icon.
- Choose the middle button (projects list).
- Select the option **"Background (unattended)"**.

j. The desktop agent must be configured to run in unattended mode. To do so, on the **Projects** screen, select **Background (unattended)**.



4. Check the Windows Local Security Policy:
    a. Open `secpol.msc` and check the **Security Options** in **Local Policies**.
    b. Interactive logon: **Do not require CTRL+ALT+DEL** should be **ENABLED**.
    c. Interactive logon: **Don't display last signed-in** should be **ENABLED**.
5. Check the registry.

    While the Window Local Security Policy should properly set up the machine, you can check the Windows registry too for the following values:

    - [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System]
        - "DisableCAD"=dword:00000000
    - [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System]
        - "dontdisplaylastusername"=dword:00000001

    Also, you may need to check whether a third-party software can use SAS (Secure Attention Sequence):

    - [HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System]
        - "SoftwareSASGeneration"=dword:00000003

The value 3 means that both Services and Ease of Access applications can simulate the SAS (Secure Attention Sequence).

6. **Optional:** Check Windows Settings.

   This should apply for machines usually not connected to a domain. However, in some cases, this could happen.

   - Go to **Windows Start** > **Settings**.
   - Go to **Accounts** > **Sign-in options**.
   - Scroll to the **Privacy** section and check the following option:

     Use my sign-in info to automatically finish setting up my device after an upate or restart must be DISABLED.

7. Your machine is properly set up for working in unattended mode and opens a session automatically.

   a. If the issue persists, set the agent in unattended mode. Run the automation and close all applications. Restart the remote machine.

   If you get a warning blue screen when starting, see the following steps to remove this screen:

   - Open the Registry editor in Virtual Machine.
   - Navigate to path `Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System`.
   - Remove **legalnoticecaption** and **legalnoticetext** keys.

   b. Store the Windows credentials for launching SAP Intelligent RPA desktop agent in unattended mode. For more information, see SAP Note 3010368.

   c. Reboot the machine. Do NOT log into the session after the reboot.

   ⚠ Caution

   If the Windows credentials are updated, then it is mandatory to manually update them in the Desktop Agent.

   - If the administrator logs in the same Windows session for which the credentials are stored, you need to reboot the machine after the operation.
   - If the administrator logs into another Windows session, and that the session for which the credentials are stored is closed, you need to reboot the machine after the operation.
   - If the administrator logs into another Windows session but the session for which the credentials are stored is left opened, there is no need to reboot the machine.

Related Information

Run Unattended Jobs with Activities on Graphic Sessions
Best Practices for Screen Lock and Disconnected Virtual Desktop Infrastructure
System Reboot during Unattended Mode

Best Practices when Migrating from Desktop Agent 2 to Desktop Agent 3

The following page will walk you through the steps required to use your Desktop Agent 2 projects with the Desktop Agent 3.

This applies when you have automations that run using the Desktop Agent 2 and you want to run them using the Desktop Agent 3.

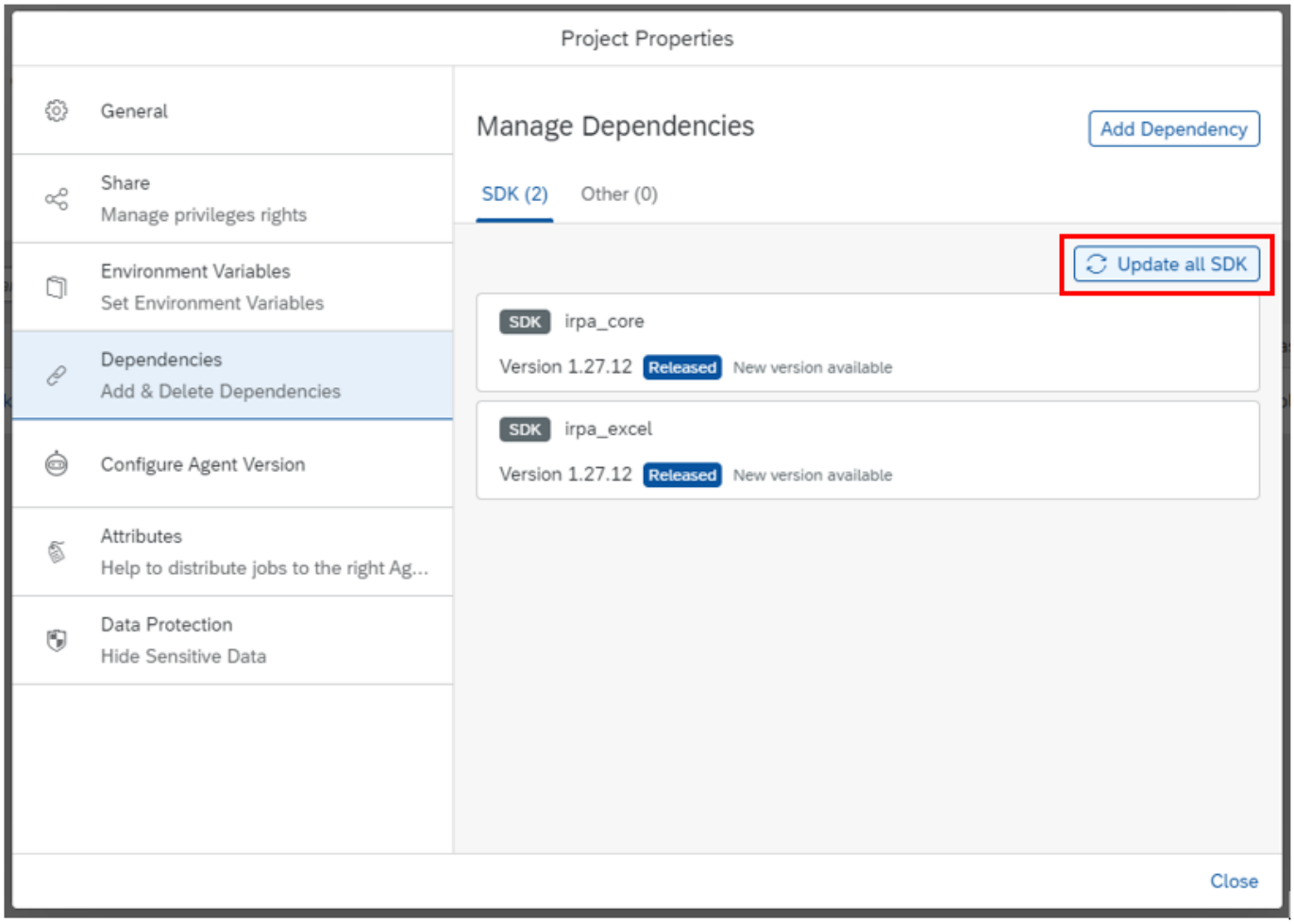
What are the prerequisites?

To use your automations with the Desktop Agent 3, you must update your Core SDK package to version 1.26 at least. If your project contains PDF or Documentation Information Extraction activities, you must update your PDF SDK version to version 1.26 at least at well.

To update your dependencies:

1. In the **Overview** page of your project, click to open the **Project Properties**.
2. Select the **Dependencies** tab.
3. Click **Update all SDK** so that all of your SDK packages are updated to the latest version.
4. **If you're using an RPA tenant:** Generate a new package from your project. For more information, see Generate a Package.

   If you're using an SPA tenant: Redeploy your project. For more information, see Deploy an Automation Project without a Trigger.



You can use your project with the Agent 3. Note that even after doing this update your project will remain compatible with the Agent 2.

Which projects are supported?

You will be able to use your project with the Agent 3 if it meets the following requirements:

- Your project must be a Cloud project.

  Note

  If you want to convert a Desktop Studio project to a Cloud project, see Migrate From Desktop Studio to Cloud Studio.

- The Core SDK package version of your project must be updated to 1.26 or higher.
- If your project contains PDF or Document Information Extraction activities, its PDF SDK package must be updated to version 1.26 or higher.
- The Desktop Agent 2 must not be installed on your computer.

  Note

  We strongly recommend that you uninstall the Desktop Agent 2 from your computer before installing the Desktop Agent 3. However, if you want to have both Desktop Agents on your computer at the same time, for example if you are testing the

migration from Desktop Agent 2 to Desktop Agent 3, note that there are some limitations. For more information about these limitations, see SAP Note 3238616.

How to know if your migration from the Desktop Agent 2 to the Desktop Agent 3 was successful?

- Your existing projects continue to work as before.
- When errors occur, the traces are correctly generated and visible in the Cloud.

Use the Store

The **Store** is available if you've subscribed to the service and are licensed to it.

Disclaimer:

SAP does not warrant the correctness and completeness of the store content. SAP shall not be liable for errors or damages caused by the use of content unless damages have been caused by SAP's gross negligence or willful misconduct. SAP is free to retire store content at any time from the Store. All content is versioned and only the latest version is available for download.

About the Store

The **Store** offers prebuilt content created and curated by SAP and authorized partners. The content available in the **Store** can be both paid and free. The **Store** can be accessed directly from the home page. The **Store** also offers a variety of filters, a search feature, and content descriptions to help you quickly discover the content based on your needs.

In the **Store**, you can choose from a wide range of pre-built content packages for different industries and business lines. It provides a catalog of ready-to-use content and templates that are directly accessible, configurable, and extendable. A content package contains a description, detailed artifact information, and a configuration guide that explains how to set it up. The content provided in the **Store** benefits you because it allows you to quickly use ready content that is tailored to your specific industry and needs. It also reduces the amount of time and effort required to develop content from scratch.

The **Store** contains different types of projects. The contents of the projects are easily customizable, making them an excellent resource for all types of business. Only the ready-to-use content is updated regularly based on the latest product changes and developments to ensure you've access to the updated content.

The Different Types of Projects and Their Descriptions

Project Type	Description
Business Process	Content created using SAP Build Process Automation. It can contain all artifacts and features associated with business process projects.
Actions	Use the Actions project to encapsulate APIs as actions in your business scenarios.
Task Automation	Content created using SAP Intelligent Robotic Process Automation. It can be used with the SAP Build Process Automation application and contains only automation artifacts.
Live Process	Process content created using SAP Workflow Management service. It can be used with SAP Build Process Automation, but it requires some additional configuration.

Tags for Different Project Format Types

To make it easier to recognize and arrange all of the projects, the following **Format Type** tags are used.

Tag Name	Description
Ready to use	These projects are ready to be used, having already been designed and are available for use in your business processes.
Template	A project template is a project that has prebuilt business process artifacts and actions that are commonly used in your industry-specific business processes. You can use the template to create new projects and it can then be customized and extended as needed.

Store Projects: More Information

The projects available in the **Store** consist of information listed in the following table:

Sections	Details
Header	• Name of the project • Type and sub type of the project • Publisher of the project • Version of the project • Catalog category of the project • Last modified details of the project

Sections	Details
Body	• Description that explains the purpose and prerequisites of the project. It can include the key process steps that are covered. • Media section to view additional material such as images • Lines of business and industries corresponding to the project • Associated artifacts • Associated documentation • What's New information • Version details • Dependent project details **Note** The project versioning is a combination of: • A first number corresponding to the number of the major version • A second number corresponding to the number of the minor version • A third number corresponding to the patch version **Example** `Version: 1.19.33` Here are a few types of artifacts that are included in the project: • Decision • Process • Visibility Scenarios • Automation • Form • Data Type • Files (such as documentation and test scripts) • Document Template • Dependencies, and so on

SAP Build Content Catalog

Explore the publicly available content catalog that consists of various projects under the SAP Build umbrella. With the catalog, you can learn more about line of business and industry-specific content that is designed specifically to enhance SAP applications such as SAP ECC, SAP S/4HANA, SAP Business by Design, SAP Ariba, SAP SuccessFactors. For more information, see SAP Build Content Catalog.

Related Information

Explore the Store

Explore the Store

In the Store, you can find the various projects that are relevant to your requirements.

Prerequisites

For live process projects, configure the destination required to work on them. See Configure Destination for Live Process Projects.

Procedure

1. To access the **Store**, access the SAP Build Lobby, and choose the **Store** from the side navigation.

   The Store window appears, with each project listed as a tile.

2. **Optional:** Use the filters or the search field to limit the packages displayed to the ones that are relevant to your needs:
   - Select the **Catalog** filters on the left to filter **Store** projects by **Automation SDK** or **Business Content** or **Learning Content**.

     The catalog consists of the following categories:

     - **Automation SDK**: It contains the different SDKs (activities, classes, modules, and data types) used for automation.
     - **Business Content**: It represents all SAP and partner packages that can be used as templates.
     - **Learning Content**: It's only intended to explain how to use SAP Build Process Automation capabilities and should be used only for learning and education purposes. It is not for productive use.

       Note

       - The learning catalog offers learning packages to get started with SAP Build. These packages allow you to learn best practices by reusing the most common flows to design your first project.
       - A content package containing the text **MI** (Mission) in the title indicates that it is linked to the SAP Developer Tutorial.

   - Select one or more filters on the left to filter the available Store projects by **Project Type** (for example, Process and Actions), **Format Type** (for example, Ready to use and Template), **Catalog** (for example, Business Content), **Product** (for example, SAP S/4HANA Cloud), **Publisher**, **Line of Business**, and **Industry**.
   - You can sort and filter your projects using the **Format Type** tags.
   - You can filter the content type based on paid or free content. The **Paid** filter helps you find packages available for purchase. You can further refine your results by selecting the types of packages you want to see.

     The **Paid** filter includes the following subfilters.

     - **Entitled**: You've purchased the package and you're privileged to add it to your tenant.
     - **Not Entitled**: The project is available for purchase but you've not purchased it yet.

   - Enter search terms in the search field and press Enter. Upon completion, the filters suggested in the search field match the filters listed on the left.

   Each item shown represents a single **Store** project. You can sort the results by **Last Updated** or by **Name** (ascending or descending).

3. Choose the individual project tile to open the overview page.
4. To add the selected project, choose **Add**.

   Note

   The **Add** button is visible only if you've chosen the **Ready to use** project type.

   The project is added to the Lobby, so you can navigate there and work on it. For example, you can add business process projects and navigate to the **Lobby** to work on them. See Working on Live Process Projects.

5. To view the project configurations and start using directly, choose **Open**.
6. To add projects based on templates, choose **Add** > **Create from Template**. You can also select **Create from Template** directly for the template only projects. For more information, see Create Projects from a Template.

Related Information

Create Projects from a Template
Update to the Latest Version

Create Projects from a Template

You can use a template to create a new project for your required application.

Context

Some of the projects in the Store can serve as templates for creating new projects. You can reduce the time required for project creation by using the templates from existing projects.

Procedure

1. Navigate to the **Store** page.
2. You can use the search function to find projects, or you can filter by relevant options.

   Projects are displayed with options like Create from template, Add, and Open.
3. Choose **Create from Template** for template only projects.

   Note

   You can see only one option displayed if filtered format type was a template only.
4. You can choose to **Add** or **Open** the selected projects.
   a. Select **Create from Template** from the dropdown menu.
5. In the **Create Business Process** dialog box, enter a new **Project Name** and provide a **Short Description** of your choice.
6. Choose **Create**.

   You get a success message that you've successfully created the project from the template.

Update to the Latest Version

You can update your project to the most recent version from the Store using the update feature.

Prerequisites

You've added a project from the **Store**.

Context

SAP offers preconfigured business projects to address different business scenarios. You can add these projects from the Store to your Lobby. SAP also publishes updates with new feature enhancements and bug fixes.

An icon is displayed on the respective project whenever there's an update.

Procedure

1. Navigate to the **Store**.
2. Search for the project or identify the projects with the **Update** icon.
3. Click **Update** icon to update your project to its latest version.

A confirmation message appears telling you that the project has been updated.

Monitor Processes

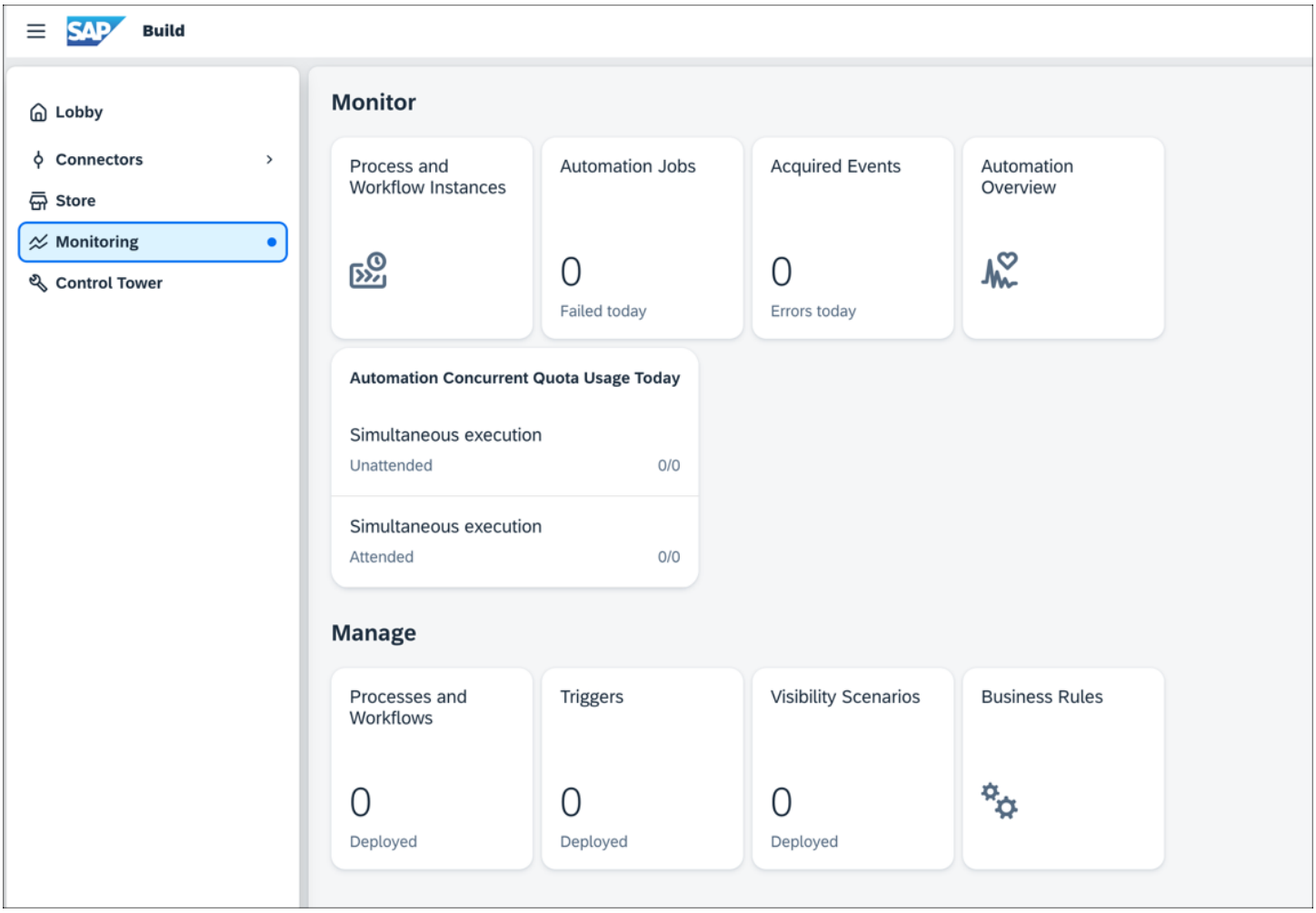
The **Monitoring** page collects events coming from the deployed workflows and automation jobs and presents them to the user. These events represent the state of the agents, the status of the jobs that have been run, and the business activity of monitoring events.

Prerequisites

You've been assigned to the `ProcessAutomationAdmin` role collection. For more information, see Authorizations.

Context

SAP Build offers several interlinked views for Process and Workflow Instances, Automation Jobs, Automation Overview, Visibility Scenarios, Business Rules, Triggers, and Acquired Events. All can be accessed using dedicated tiles in the SAP Build Work Zone.



Monitoring Process and Workflow Instances

In the **Process and Workflow Instances** view, you can see a list of all workflow instances and act on those instances.

For more information, see Process and Workflow Instances.

Monitoring Automation Jobs

In the **Automation Jobs** view, you can view all your automation errors, warnings, variables, and other information on the console.

For more information, see Monitor an Automation Using the Console.

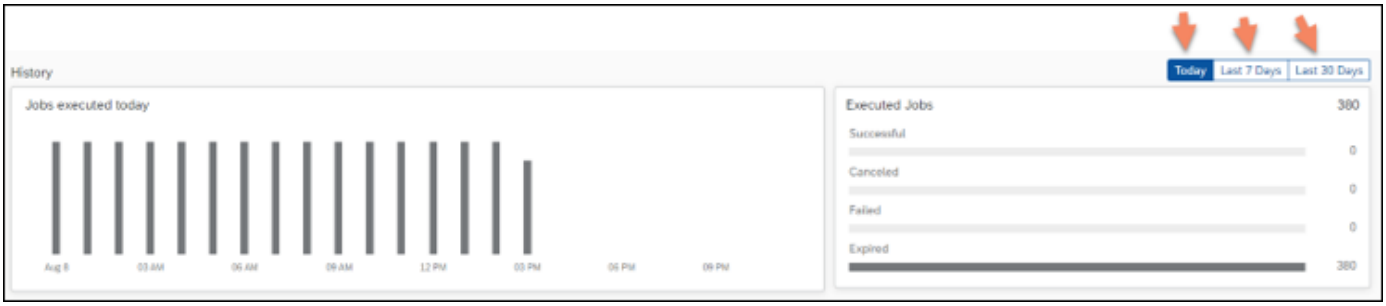
Monitoring Acquired Events

In the **Acquired Events** view, you can view the events that have been acquired by the process visibility capability.

For more information, see Acquired Events.

Monitoring Automation Overview

The **Automation Overview** page displays the status of your jobs. You can check whether any job has failed today, the last 7 days, or last month.



For more information, see Automation Overview.

Monitoring Visibility Scenarios

In the **Visibility Scenarios** view, you can manage your visibility scenarios by scheduling the job for processing the data and clearing the processed data. You can also view the processing log information and the details of a scenario definition in this view.

For more information, see Visibility Scenarios.

Managing Business Rules

Using the **Business Rules** tile, you can launch the **Manage Rule Projects** (MRP) application and perform advanced configurations on business rules projects that are imported as part of live process packages. Additionally, you can model complex scenarios and consume them using public REST APIs.

For more information, see Business Rules

Process and Workflow Instances

The **Process and Workflow Instances** view shows a list of all process and workflow instances and offers actions to work on these instances.

Search and Filters

Search for instances using any of the following criteria: instance ID, workflow definition ID, subject, business key, or the initiator of the instance.

> **Note**
> By default, all the instances that are not in **Canceled** or **Completed** status are listed.

To optimize the search results using additional filters, choose **Adapt Filters** and filter instances based on one or more of the following parameters:

Filter Parameter	Description
Status	Based on the status of the workflow execution. See Instance Status.
Project	The business process project to which the instance and definition belong
Definition	The definition ID of the instance

Filter Parameter	Description
Hierarchical Level	Filter only the root workflow instances or all workflow instances that include subflows. **Note** You can filter to display only the root instances by selecting Source Instances in the Hierarchical Level field.
Instance ID	The instance ID of the workflow.
Started After	All instances started on and after the selected date are displayed.
Started Before	All instances started on and before the selected date are displayed.
Started By	All the workflow instances started by one or more users are displayed.
Completed Before	All instance executions completed on and before the selected date are displayed.
Completed After	All instance executions completed on and after the selected date are displayed

Instance Details

You can view the details of the project by selecting the workflow or process instance from the list. To view the project information associated with a process instance, redeploy the project. See Deploy a Project.

- **Context**: View the current context of a workflow instance in the Context tab.
- **Logs**: View the executed activities in the workflow instance and their details in the Logs tab. For erroneous instances, this tab shows the error logs. The logs can also be downloaded in JSON format.
- **Definition**: Navigate to the workflow definition of an instance by choosing the Definition ID.
- **Parent**: Navigate to the parent workflow instance of a subflow by choosing the Parent ID.
- **Source**: Navigate to the root instance of a workflow or subflow by choosing Source ID.

> **Note**
> Root instances are main workflow instances that are started by an application or a user. Main workflow instances in turn start subflow instances.

Instance Status

Status	Description
Put on Hold	Pause the execution of the instance. You can suspend a running or erroneous workflow instance by putting it on hold.
Resume	Continue the execution of the workflow after putting it on hold. This operation also retries failed steps.
Retry	Retry the execution of failed steps of an erroneous workflow. This option is available only when there is an error in executing the instance.
Cancel	Terminate the execution of the instance.

More Options

- To view the subflow instances that are started by an instance, choose Show Subflow Instances.
- To view the tasks of a workflow instance, choose Show Tasks.

Automation Jobs

In the **Automation Jobs** view, you can view all information related to your automation jobs.

Overview

In the left-hand side panel, choose **Monitoring** and then the **Automation Jobs** tile.

In this view, you can see information about individual jobs, including their status, trigger, and duration.

Only automation jobs from productive projects are listed. If you want to include test jobs that ran during the project design phase, select the **Include Test Jobs** checkbox.

Filters and Buttons

You can filter the automation job information by selecting appropriate filters such as **Date Range**, **Automation**, **Status**, **Project** and **Machine**. You can apply one or several filters to get the desired job status information.

You can clear all filters by clicking the **Clear filters** button.

You can cancel a job with **Ready** status by clicking its checkbox to select the job and then by clicking the **Cancel jobs** button.

You can refresh the **Automation Jobs** table by clicking .

You can retry failed, expired, or canceled jobs by choosing **Retry**, if:

- they failed, expired, or canceled less than 1 week ago
- they are unattended
- they are not a child job
- the deployed package is still available

If the conditions are not met, the **Retry** button is not visible.

Automation Jobs Table

In the **Automation Jobs** table, you can view different columns that contain information about automation jobs:

Name	Description
	In this column, you can check the check box of an automation job to select it. If the automation job you have selected has the **Ready** status, you can cancel it by clicking the **Cancel jobs** button.
	In this column, you see warning notifications if the automation job contains errors.
Automation	In this column, you see the name of the automation that you have created in your project.

Name	Description
Status	In this column, you see the status of the automation jobs. The different statuses are: • **Canceled** • **Expired** • **Failed** • **Ready** • **Running** • **Successful**
Project & Version	In this column, you see the project and it's version that contains your automation and that you've deployed.
Environment	In this column, you see the environment in which the project containing your automation has been deployed.
Machine / Alias	In this column, you see the machine, or its alias, if applicable, on which the agent has executed the automation. Hover over the alias to see the machine name.
Login	In this column, you see the user login details such as user name or user ID.
Trigger	In this column, you see the trigger that you've created to execute the automation.
Last Updated	In this column, you see when the automation job was last updated. You can sort data displayed in this column in chronological or nonchronological order using or .
Duration	In this column, you see how long the automation took to be executed.
Traces	In this column, you see if traces are available for your automation job.

More Information

Traces

In the **Automation Jobs** table, if the traces are available for any specific job, the **Traces** column shows as **Available** with a direct link to the **Traces** view.

In the **Traces** view, the **Export Traces** button is displayed. Click the **Export Traces** button to export the trace information of your automation. Click **X** (close button) to go back to the job details page.

On the job detail page, the **View Traces** button is displayed. Click the **View Traces** button to view the full screen of the traces.

If the traces are not available for any specific job, the **Traces** column shows as blank.

If any specific job has traces but you aren't authorized to view it, a warning message icon is displayed under the **Traces** column in the **Automation Jobs** table. When you click the warning message icon, the cause of not being able to view the trace information is displayed as a warning message. For that specific job, the **View Traces** button is also disabled on the job detail page. When you hover over the **View Traces** button, you can view a warning message that displays the cause of not being able to view the trace information.

Acquired Events

Using the **Acquired Events** tile, you can view the events acquired by the process visibility capability and monitor errors while consuming the events pushed to the capability.

The Acquired Events tile provides the following functionalities:

- Visibility Events: Displays events that have been acquired by the process visibility capability. You can filter events based on the Process Definition ID, Process Instance ID, Event Type, and Timestamp attributes.

  You can import events into the process visibility capability. Choose *Import*, browse for the required JSON file containing an array of events and choose Import. The file to be imported can have an array of events as shown in the example:

  Example

```
[
   {
      "processDefinitionId":"PO",
      "processInstanceId":"PO11",
      "eventName":"PO Initiated",
      "timestamp":"2019-01-19T13:18:10Z",
      "context":{
         "PONumber":"190019191",
         "InvoiceNo":"390139951",
         "RemittanceNo":"1290029201",
         "Supplier":"SAME DEUTZ-FAHR",
         "Item":"Hybrid electric vehicles"
      }
   },
   {
      "processDefinitionId":"PO",
      "processInstanceId":"PO11",
      "eventName":"PO Sent",
      "timestamp":"2019-01-19T13:18:12Z"
   }
]
```

  Note

  - The timestamp you provide is taken from your time zone and converted to UTC.
  - By default, no events are displayed unless a filter is applied. When filters are applied, only the latest 5000 events are displayed. To see the remaining events, narrow down your search by adjusting the Timestamp value.
  - You can delete a maximum of 1 million events that match the filter criteria by choosing the *Delete* icon. Choose *Refresh* to view the updated status of deletion.

- Visibility Event Errors: Displays errors that have occurred while consuming the events pushed to the process visibility capability. You can filter the errors based on the **Timestamp** of their occurrence. The following details are displayed:
  - Messages: Displays the incoming event that could not be consumed.
  - Errors: Displays error messages specific to each event. The error log gets deleted automatically after 30 days.
  - Timestamp: Displays the time when a specific error has occurred.
- Business Events: Displays the business events consumed by SAP Build Process Automation in the last 7 days. For more information on business events, see Create Event Triggers.

  The following details are displayed for business events:
  - Timestamp: The time when a specific event has occurred.
  - Event Type: The type of the event.
  - Source: The source system from where the event has been created.
  - Data Received: The payload of the event received.

ℹ Note

You can download the visible business events and details in a CSV file by selecting the *Download Business Events* icon.

Automation Overview

This page shows an overview of the status of your agents and reviews historical data in the following sections:

- **Status**: Displays the number of ongoing jobs and of running agents by status.
- **History**: Displays the job history that contains count of agent per status.

Only automation jobs from productive projects are listed. If you want to include test jobs that were run during the project design phase, select the **Include Test Jobs** checkbox.

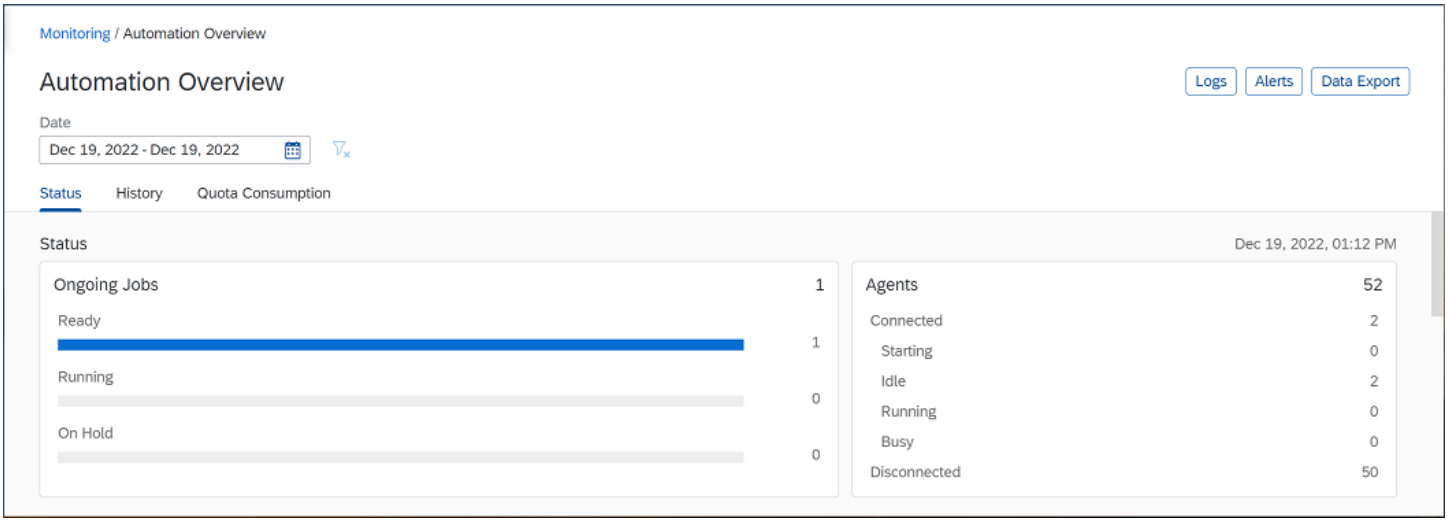
Status

Ongoing Jobs

This section displays the number of ongoing jobs. Click this section, to see the jobs with the status **Ready**, **Running**, **Pending**, or **Waiting**.

Agents

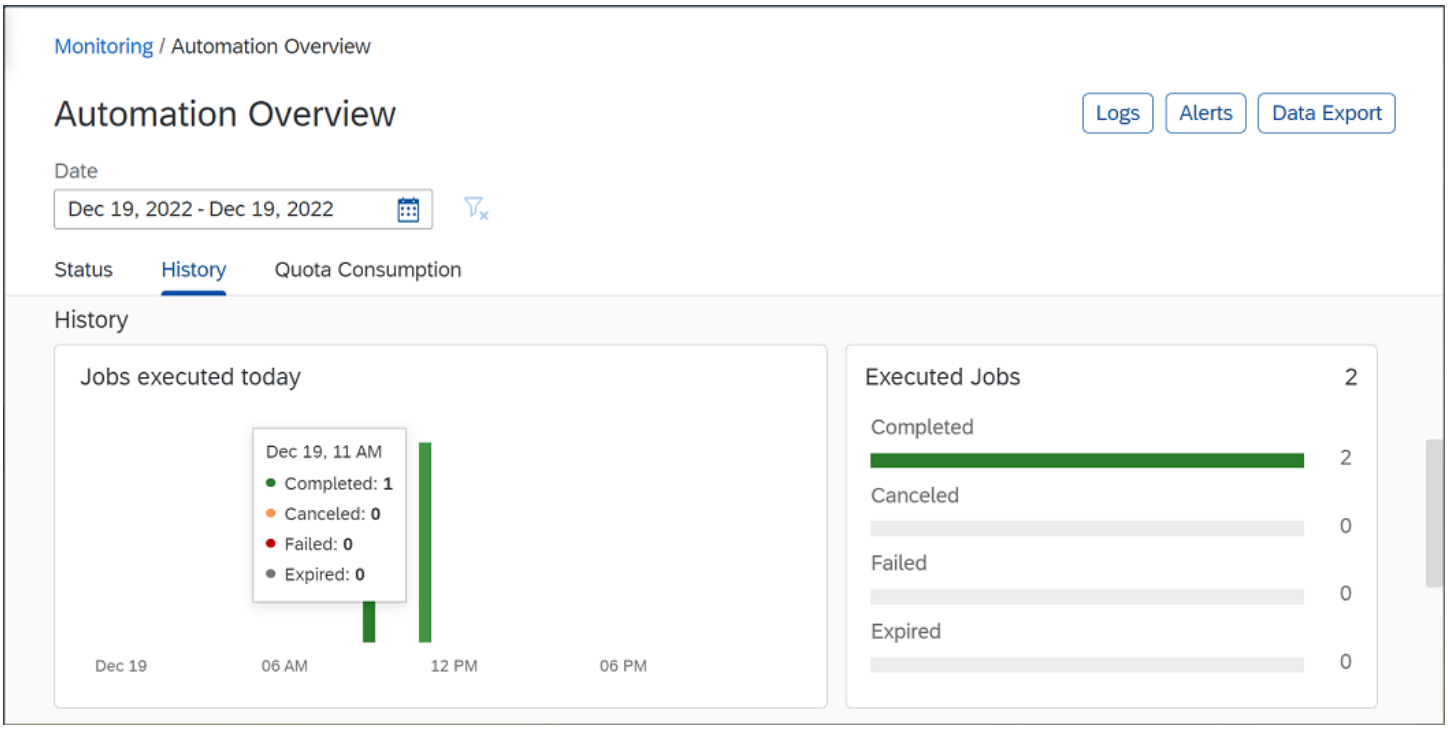
This section displays the number of agents and their status. Click this section, to see the agents that are connected with **Starting**, **Idle**, **Running**, or **Busy** status or have the **Disconnected** status.



History

Executed Jobs

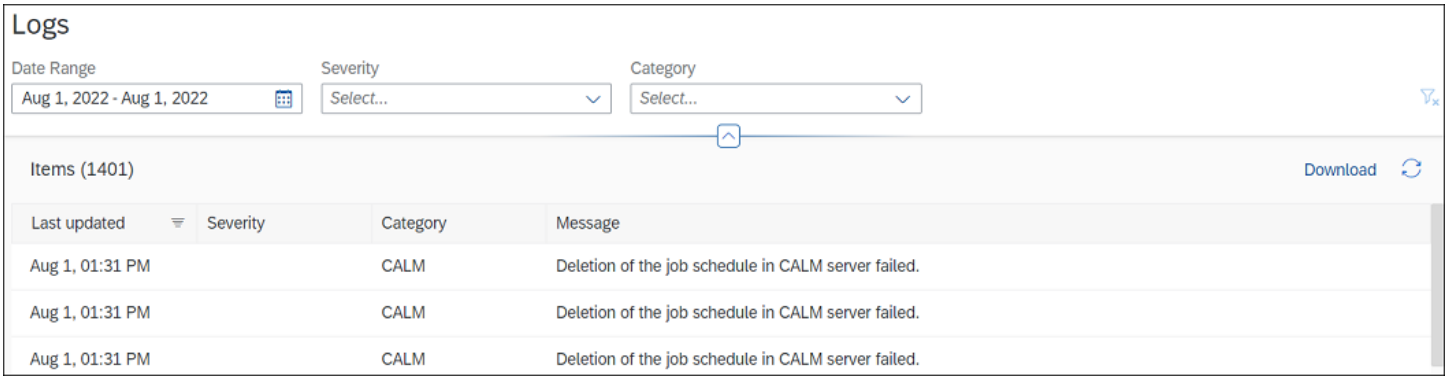
This section displays the number of finished jobs with the status **Successful**, **Cancelled**, **Failed**, or **Expired** for the time period you selected.



Logs

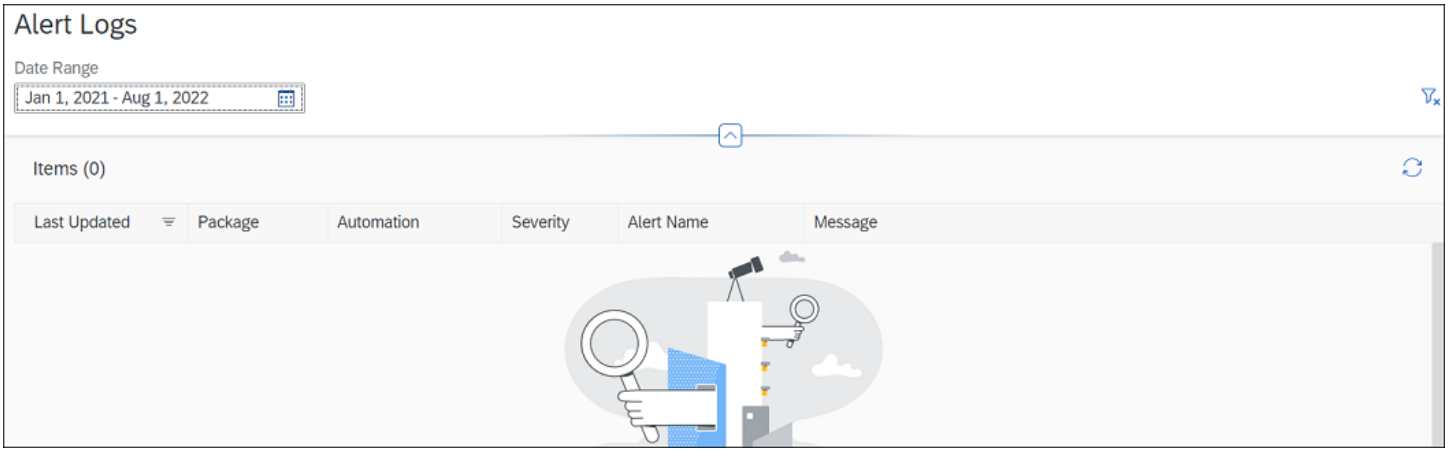
To display the general event logs of the tenant, choose the **Logs** button. You can view logged technical events, for example, connection errors.

You can filter the table by date range, severity, and category (jobs, agents, triggers, distributions, CALM, or alert handlers). To download logs, click the **Download** button.



Alerts

To display all logged alerts raised by users, click the **Alerts** button. You can filter the table by date range.



Data Export

To download a CSV file containing data records for review and analysis, click the **Data Export** button. You can download the following types of records:

- **Historical Job Data**: You can download archived data of run jobs.

  Note

  The historical job data record is limited to 2 million and stored up to 2 years.

- **Distribution**: You can download rebooted data of unattended jobs that have been distributed to agents.

  Note

  The distribution records are limited to 10,000 and stored for up to 30 days.

- **Business Activity Data**: The Business Activity Data or Business Activity Monitoring (BAM) allows you to generate useful information about your running automations. For more information, see Business Activity Monitoring (BAM).

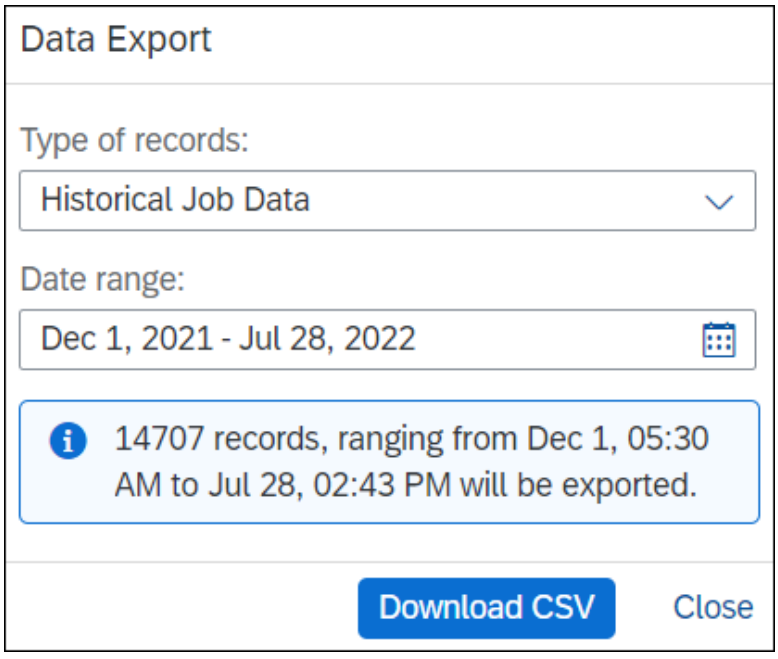
  Note

  The business activity data records are limited to 50,000 and stored for up to 90 days.

  Caution

  The CSV files downloaded using BAM can contain personal data, for example, logins and machine names. It's the responsibility of the user to ensure that the usage of this data complies with the Data Protection and Privacy regulations.

  According to the type of record you select, you can see the number of all the available records over a time period. You can choose a specific time period using the date range picker. It automatically refines the number of records corresponding to a selected time period.



Simultaneous Quota Usage for Automations

Check how many automation jobs you have run simultaneously in relation to the number permitted by the quota.

This view displays the number of attended and unattended automation jobs run simultaneously today and the quota usage over the last 30 days. Hover over a date to displays the quota usage for that day.

The maximum number of jobs you can run simultaneously is determined by the quota defined for your account. For more information, see Quotas, Restrictions, and Limits

This information is relevant for IT administrators.

Processes and Workflow Definitions

The **Processes and Workflow Definitions** view shows a list of deployed processes and workflow definitions.

Search and Filters

- Filter workflow definitions based on workflow definition's ID, name, or version.
- You can also choose the required project from the **Project** dropdown to filter the processes within that project.
- To navigate to the list of all instances of a definition, select the definition from the list and choose **Show Instances**.

Note

You can view the details of the process by selecting the process from the list. If the project information of the process is not visible, redeploy the project. See Deploy a Project.

Workflow Definition

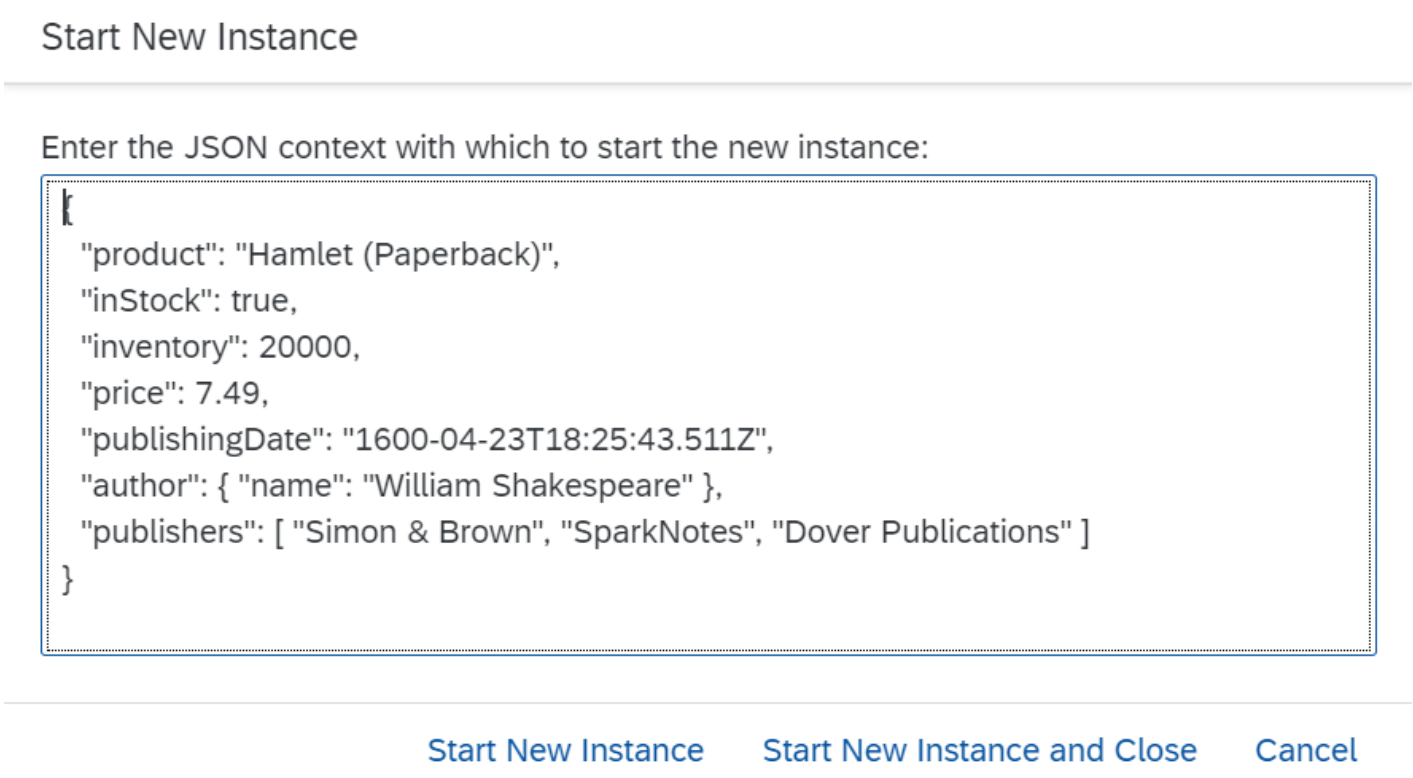
You can perform the following actions on the workflow definitions:

Start New Instance of a Workflow Definition

To start a new workflow instance, select a workflow definition and choose **Start New Instance**. You can also choose **Start New Instance and Close**, where the dialog closes upon starting an instance.

If you've configured a sample context while modeling a start event, this is shown as the context data while starting a new workflow instance in the **Start New Instance** window. However, you can also modify this JSON context data as required. For more information, see Configure Start Events.

Example



The JSON structure contains the content to be passed to the workflow context. A context node as a wrapper isn't required.

In the workflow context, use numbers where computations or comparisons are required. We recommend using a string instead of numbers as IDs, especially for business keys

Download the Workflow Model

To download the workflow model, select the definition from the list, then choose **Download Workflow Model**. The downloaded workflow model is of the latest deployed version of a workflow definition.

Notifiers

Create a notifier in order to receive information about the execution of an automation, or in the creation of a SAP Conversational AI chatbot.

Add an API Notifier

Add a CAI Notifier

Add an Email Notifier

Add an API Notifier

An API notifier sends a notification via a REST call when a job execution status reaches a specified state (START, SUCCESS, FAILURE, CANCEL) and contains the previously custom defined message, also called a payload.

Prerequisites

You have added a trigger to a released project. For more information on how to add an API trigger, see Add an API Trigger

Procedure

1. Go to the **Monitor** tab, then select **Triggers** under **Manage**.
2. Find the trigger for your project, click the and then select **Add notifier**.

   The Create Notifier menu pops up.
3. Select **API** and click **Next**.
4. Open the **Notify on** dropdown list, and choose the activation event:
   - **All status changes**: A notification is sent for any status change on the job.
   - **Start**: A notification is sent when the job starts.
   - **Success**: A notification is sent when the job is successful.
   - **Failure**: A notification is sent when the job fails. For more information on available variables for a failure notifier, see Available Variables for Notifiers.
   - **Cancel**: A notification is sent when the job is canceled.
   - **Expired**: A notification is sent when the job is failed or expired. A job expired event triggers under two circumstances:
     - When you have a ready job that does not run before the expiration date. This case concerns automations with a scheduled trigger.
     - When you have a job that is running but the timeout is reached, and the agent is not able to send the result due to some unforeseen challenges such as internet connectivity issue. This case concerns automations with an API trigger.
5. **Optional:** From the **Available variables** list, you can use the variables in the **Callback URL** or **Payload** field. For more information on available variables, see Available Variables for Notifiers.
6. Enter a name and a description.

   Note

   API Notifiers currently support only destinations with **Proxy Type** ***Internet*** and do not support ***OnPremise***.
7. Choose one of the two following options under **Target System**:
   - Use predefined destinations.

     If you have chosen this option:

     a. Select your **Destination** from the dropdown list.

     This list shows only destinations that are shared with the environment and have one of the supported authentication types (**No Authorization**, **Basic Authorization** or **OAuth2 Client Credentials**).

     b. Make sure that the base **URL** of the destination is automatically filled in.

     c. **Optional:** Specify a path to add to the base URL.

     Caution

     If the destination you have chosen is removed the notifier becomes invalid. You cannot execute it anymore.

   - Enter credentials.

     If you have chosen this option:

     a. Enter the **Callback URL** - the target endpoint of the system that you want to notify.

     Caution

     Enter a valid URL and make sure not to disclose any sensitive data.

     b. Select the type of **Authorization** you want to add to the notifier:
     - **None**: no authorization is passed with the request.
     - **Basic**: username/password pair is passed with the request.
     - **OAuth 2.0**: a client ID, client secret, and authorization URL are passed with the request. Enter the following:
       - **Auth URL**

- **Client ID**
- **Client secret**
- **Grant type**

8. Select the **HTTP Method**:
    - GET
    - POST
    - PUT
    - DELETE
9. **Optional:** Add a Header to the Notifier: enter the Key and the Value of the Header.
10. Write the **Payload** of the Notifier.

    You can enter any JSON payload in an API notifier. As part of your payload definition you can use available variables that you see under the Notifyon dropdown. For example, if you've selected **Success** event in the **Notify on** dropdown, you can use `${output}` variable in your payload definition. `${output}` variable will be resolved at runtime with the output data from the job execution.

    Any Notifier for an API trigger can refer to the `${invocation_context}` variable, which contains the `invocationContext` json object provided in the API trigger payload. This allows calling applications to correlate asynchronous trigger calls with the corresponding notifier callbacks.

    Variable references used in the notifier payload definition are resolved at runtime with the actual data. If there is an invocation context, the data type of the variable references is determined dynamically.

    Example

    Assume that the payload of the notifier contains variable `${invocation_context.taskId}`. If at runtime the provided trigger invocation context is `{taskId: "ABC"}`, then the notifier variable is resolved to "ABC" and is sent as a string. If it's `{taskId: 123}`, then it's resolved to an integer 123. If it's {taskId: {id: "xyz"}}, then it is resolved to the json object `{id: "xyz"}`.

Add a CAI Notifier

SAP Conversational AI and SAP Process Automation work together to create your chatbot, CAI (Conversational Artificial Intelligence) provides the natural language interface, while SAP Process Automation executes the tasks.

Prerequisites

When designing chatbots, you can use webhooks to trigger the execution of an automation. For this, you need to add an API trigger to the corresponding automation. The endpoint of the created API trigger can then be called in the webhook. For more information about how to use a webhook, refer to the SAP Conversational AI documentation.

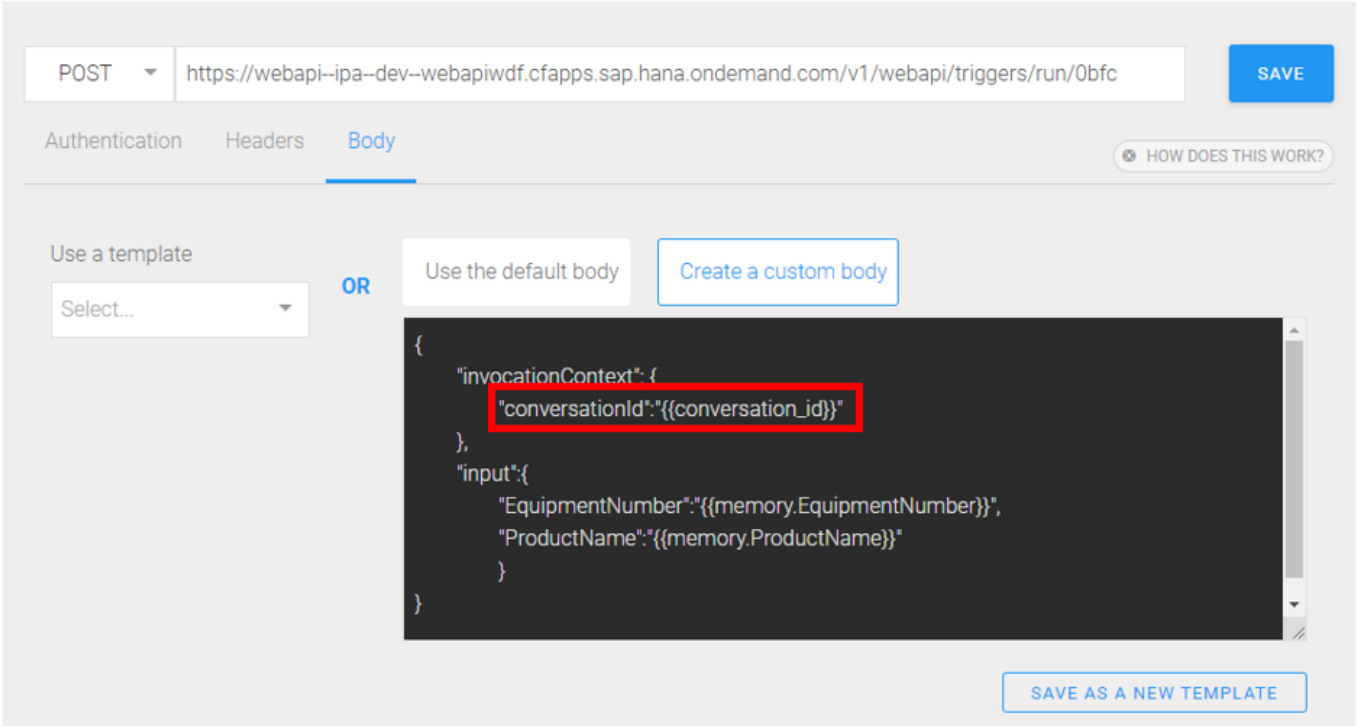
Procedure

1. Go to the **Monitor** tab, then select **Triggers** under **Manage**.
2. Find the trigger for your project, click the and then select **Add notifier**.

   The Create Notifier menu pops up.
3. Select **CAI** and click Next
4. Open the **Notify on** drop-down list, and choose the activation event:
    - **All status changes**: A notification is sent for any status change on the job.
    - **Start**: A notification is sent when the job starts.
    - **Success**: A notification is sent when the job is successful.
    - **Failure**: A notification is sent when the job fails.

- **Cancel**: A notification is sent when the job is canceled.
- **Expired**: A notification is sent when the job is failed or expired. A job expired event triggers under two circumstances:
  - When you have a job ready that does not run before the expiration date. This case concerns automations with a scheduled trigger.
  - When you have a job that is running but the timeout is reached, and the agent is not able to send the result due to some unforeseen challenges such as internet connectivity issues. This case concerns automations with an API trigger.

5. **Optional:** From the **Available variables** list, you can use the variables in the **Message Content** and **Memory** field. For more information on available variable, see Available Variables for Notifiers.
6. Enter a name and a description.
7. Refer to the **Conversation ID** passed as invocationContext in the body of the chatbot webhook call.



8. In the **Token** field, enter:
   - request token of the project if you want to send a message.
   - developer token if you want to update a conversation.

Note

There are two different types of actions that you can select in CAI notifiers:

- **Send a message**
- **Update a conversation**

You must provide the token and client credentials for both the actions.

9. **Optional**: Check the **Enterprise Edition** checkbox if you want to use the enterprise edition of CAI.

Note

If you want to use the integration of IRPA with the CAI enterprise edition, you must enable the developer token at the tenant level in CAI using a BCP ticket. For more information, see Enable Developer Token.

   - Provide the enterprise endpoint URL in the **Enterprise Endpoint** field. The URL of the enterprise endpoint must be without the path. Only the hostname must be provided in the URL of the enterprise endpoint.

10. In the **Auth URL**, **Client ID**, and **Client Secret** fields, enter the:
    - runtime client credential details when you want to send a message.
    - design time credential details when you want to update a conversation.

Note

- Runtime APIs are used to interact with the bot. These APIs work with the request, version, or environment token of the bot.
- Design time APIs are used to configure the bot. These APIs work with the developer token of the bot.

11. Select the action you want to perform with the notifier:
    - **Send a message**

      No additional implementation is required on the chatbot side. In the **Message content** area, define the message to be sent.
    - **Update a conversation**

      Here you can modify the chatbot **Memory** and if needed, the conversation **Language**. In your chatbot design you can react to these changes and decide on what is communicated to the user.

      You also need to provide the User slug, Bot slug and Version slug to enable this operation. You can refer to the available variables in the slug definitions, as well as in the memory specification. Consult the SAP Conversational AI documentation to learn where to find the slugs.

Add an Email Notifier

An Email notifier sends information about the execution status of an automation by sending an email to a list of recipients.

Prerequisites

You have configured a Simple Mail Transfer Protocol (SMTP) server. For more information, see Configure an SMTP Mail Destination.

Procedure

1. Go to the **Monitor** tab, then select **Triggers** under **Manage**.
2. Find the trigger for your project, click the and then select **Add notifier**.

   The Create Notifier menu pops up.
3. Select **Email** and click **Next**.
4. From the **Notify on** drop-down list, choose the activation event:
   - **All status changes**: A notification is sent for any status change on the job.
   - **Start**: A notification is sent when the job starts.
   - **Success**: A notifcation is sent when the job is successful.
   - **Failure**: A notification is sent when the job fails.
   - **Cancel**: A notification is sent when the job is canceled.
   - **Expired**: A notification is sent when the job is failed or expired. A job expired event triggers under two circumstances:
     - When you have a job ready that does not run before the expiration date. This case concerns automations with a scheduled trigger.
     - When you have a job that is running but the timeout is reached, and the agent is not able to send the result due to some unforeseen challenges such as internet connectivity issues. This case concerns automations with an API trigger.
5. **Optional:** From the **Available variables** list, you can use the variables in the **Subject** or **Message Content** field. For more information on available variables, see Available Variables for Notifiers.
6. Enter a **Name** and **Description** (optional).
7. Add the subject of the email in the **Subject** field, recipient email id in the **Recipients** field, content of the email in the **Content** field, and click **Create**.

Available Variables for Notifiers

For every notifier (API, CAI and Email), you can use variables in specific fields of your notifier. The types of available variables depend on the activation event.

- **All status changes**: `${output}`, `${error}` and `${job}`
- **Start**: `${job}`
- **Success**: `${output}` and `${job}`
- **Failure**: `${error}` and `${job}`
- **Cancel**: `${job}`
- **Expired**: `${job}`

Additionally, for **Notifiers** that are attached to API triggers, you can use `${invocation_context}` for every activation event.

The variable `${output}` refers to the output of your automation or process executed by the trigger to which the notifier is attached.

The variable `${error}` is static and has the following sub-properties:

- `${error.source}`
- `${error.category}`
- `${error.jobUid}`
- `${error.details}`
- `${error.details.url}`
- `${error.details.message}`
- `${error.details.exception}`
- `${error.timestamps}`
- `${error.timestamps.executionStarted}`
- `${error.timestamps.errorOccured}`
- `${error.timestamps.jobSubmitted}`

The variable ${job} is static and has the following sub-properties:

- `${job.status}`
- `${job.triggerUid}`
- `${job.uid}`
- `${job.triggerName}`
- `${job.botName}`
- `${job.monitoringURL}`
- `${job.duration}`

Note

`${job.triggerRunUid}` variable is renamed to `${job.uid}`.

Note

Autocomplete feature is added to the available variables. Entering the first letter of a variable triggers the autocomplete of the variable. For example, to get `${error}` or `error: ${error}` variable, type “e” and press “Enter”. You can access the autocompletion for the sub-properties of a variable by typing the “variable name” and “.”. For example, to get the sub-properties of `${error}` variable, type “error.” or “error.timestamps.”.

Note

You can list available variables by typing "${". To navigate the structure of a specific variable (for example, the `${error}` variable), type "${error.".

Note

Complex variables (for example, `${error}` and `${job}`) in text fields are replaced by `[Object]`. Reference complex variables (for example, `${error}` or `${output.myObject}`) are converted to JSON and displayed in full.

Visibility Scenarios

As an administrator, you can manage the deployed visibility scenarios using the **Visibility Scenarios** tile.

The **Visibility Scenarios** tile provides the following functionalities:

- You can search for the scenario by typing the keyword in the **Search** field. You can also choose the required project from the **Project** dropdown to filter the scenarios within that project.
- You can access the dashboard for the visibility scenario using the  icon to view the detailed performance indicators and the instance details.
- You can view the details of a scenario by selecting the scenario from the list. The following details are displayed:
    - Information such as project name, project version, person who has deployed the project, and when the project was deployed.
    - Processing information such as:
        - **Total Number of Instances**: The total number of instances processed.
        - **Number of Completed/Abruptly Ended Instances**: The total number of instances that are in **Completed** and/or **Abruptly Ended** state.
        - **Start Time**: The time when a specific job has started.
        - **Status**: The status of the processing job.
        - **Events Processed**: The number of events processed for a specific job.
        - **Events in Buffer**: The number of events in buffer.
        - **Instances Processed**: The number instances created/updated.

    ### Note

    - If the **Status** is **Failed** in the **Processing Log**, you can view the detailed error message for additional information.
    - The **Processing Log** gets deleted automatically after 14 days.

- You can process the acquired data of a scenario. To process the data, select the scenario definition from the list and then choose **Process Data**. This action correlates and processes the acquired data based on the scenario model to form an end-to-end process instance.

    ### Note

    - Every time you choose **Process Data**, the processing information is added to the **Processing Log**.
    - In case of any error while processing, a message strip will be dispalyed with hyperlink to view the process instances that were skipped during processing. To know details about the skipped instances, create a support ticket on the LOD-BPM-VIS component.

- You can download the archived data in a zipped file which contains the scenario instances that have been **Completed** or **Abruptly Ended**. For more information, see Download Archived Data.
- You can clear the existing processed data of a scenario. To clear the existing processed data, choose the scenario from the list and select **Clear Processed Data**. Choose *Refresh* to view the status of processed data deletion. The next processing run starts the processing of acquired data from the beginning.

  Note

  You can trigger **Clear Process Data** only for one scenario at a time.

- You can schedule data processing by toggling the **Schedule Job** button to **ON** located under the **Processing Job** tab. This action schedules data processing for every five minutes.

  Note

  - With the **Free** plan, you will not be able to use the scheduling feature for processing data.

- You can delete a scenario. To delete a scenario, select the scenario from the list and choose **Delete**.

Download Archived Data

You can download the archived data in a zipped file that can be used for further analysis if required. The archived data contains the completed and abruptly ended scenario instnaces that are either older than 24 months or have more than one million instances.

Context

The scenario instances with the state **Completed** or **Abruptly Ended**, that are either older than 24 months or have more than one million instances, are periodically archived. The archived scenario instances only contain scenario attributes, such as default attributes, context attributes, and calculated attributes. However, the information related to phase, path, entity, and raw events are not archived. The archived scenario instances are no longer displayed in the **Process Workspace**.

Note

The archived data is stored for two years from the date of archiving.

Follow these steps to download the archived data. You can use the custom date range picker to choose a specific time frame to refine the number of records.

Procedure

1. Select *Additional Options* and choose **Download Archived Data**.
2. In the **Download Archived Data** dialog box, select the time frame during which the scenario instances were completed.

Time Frame	Description
Latest Archive	Provides the latest file that was archived.
Custom Date Range	Provides the archived files within the specified date range.

3. Choose **Download**.

   A zipped file containing the archived files in .csv format is downloaded, and you can use it for further analysis.

Business Rules

Using the **Business Rules** tile, you can launch the **Manage Rule Projects** (MRP) application and perform advanced configurations on business rules projects that are imported as part of live process packages. Additionally, you can model complex scenarios in the imported projects and consume them using public REST APIs. See SAP Business Accelerator Hub.

> **Note**
>
> You must have either the **ProcessAutomationAdmin** or the **ProcessAutomationDeveloper** role to use this tile.

Using the MRP application, you can do the following:

- View the list of **Projects** created, their statuses, description and so on.

  A project is used to configure and manage the entities of business rules. Entities of business rules include data objects, rules, rulesets, and rule services.

- View the **History** of any project by selecting a project and then choosing **History**.

  > **Note**
  >
  > This project is a Business Rules project and is different from SAP Process Automation project.
  >
  > You cannot import a project from the **Manage Rule Projects** application to SAP Process Automation lobby and vice versa.

- Import a project by choosing **Import** . Importing from various sources is supported, for example, from your File System.
- Export a project in the form of a zip file by choosing **Export** . You can also delete any project by choosing **Delete** .

The projects that you create using the MRP application can be added as **Decisions** in a live process package. To know more about the Developer tasks for business rules using the MRP application, see Development.

Manage Control Tower

You use the Control Tower page to check the details and several settings required for your tenants.

You can find Control Tower as a separate page in SAP Build Process Automation. It includes various details, statuses, and error messages. You can maintain the several settings that are required to run your projects.

Overview of Control Tower

The **Control Tower** page lets you view the registered agents along with their status, versions, and modes. For more information, see Overview of Agents.

Tenant Configuration

Classify different types of tenants as an administrator.

Prerequisites

You've been assigned the **ProcessAutomationAdmin** role collection in your subaccount. For more information, see Authorizations.

Context

The type of tenants that are required for SAP Build depends on the needs of your business. As an administrator, you can decide what type of tenants are required and you can classify the tenant accordingly. The setting applies to all types of projects.

When you classify a tenant, you can maintain the tenant name and add a description. Furthermore, if you wish to provide a direct link to a policy or any other links the user should be aware of, you can do so.

The following tenant classification types are available to you:

- **Development**: A development tenant is designed to configure and customize the business scenarios as per the requirement before implementing into the main production tenant.
- **Test**: Used for testing new configurations within the assigned projects.
- **Production**: The tenant where your organization's live data is managed and maintained. It's important that only authorized users have access to the production tenant.

An Overview of Tenant Access for Users

User Access to Features	Development Tenant	Test Tenant	Production Tenant
Lobby	Explore all the features.	View, export, deploy, promote, and manage members of the shared projects.	Limited access
Connectors > Automation SDK	Complete access	Complete access	Complete access
Store	Add packages and create packages from a template.	View packages	View packages
Monitoring	Complete access	Complete access	Complete access
Control Tower	Limited access based on the authorizations	Limited access based on the authorizations.	Limited access based on the authorizations.

The steps for configuring your tenant are as follows:

Procedure

1. Go to the **Control Tower** and choose **Tenant Configuration** > **Tenant Details**.
2. In the **Tenant Type**, select a tenant type from the drop-down menu.
3. Enter a tenant name, a tenant description, and add a link.
4. Choose **Apply changes**.

Environments

An environment is a functional area in which you deploy and run your SAP Build Process Automation projects.

You use an environment to control how you deploy and execute projects. A project can be deployed in more than one environment and for different project versions. By creating shared environments, you can increase the security of deployed projects and the corresponding artifacts and resources, by limiting access and managing project members.

You can use two types of environments:

- **Shared**: A shared environment is created by users with the `ProcessAutomationAdmin` role and can be shared to users with the `ProcessAutomationDeveloper` role. By creating more than one shared environment on your SAP Build Process Automation tenant, you can isolate the deployment of different projects or project versions. This means you can deploy and run the same project with different configurations and make these different configurations available to different users, depending on the line of business and on security requirements. You can also restrict automation tasks to specific users and machines.
- **Public**: The public environment contains all of the existing projects already deployed in SAP Build Process Automation. The settings in the environment correspond to the settings already made for these projects. Access to the public environment is protected at the tenant level so every user who has access to the tenant has access to the public environment.

Each environment contains distinct configurations that define the conditions for running automation projects. These include:

- **Projects** - displays the projects in an environment

- **Attended Triggers** - to start a project manually
- **Unattended Triggers** - to start a project automatically
- **Agent Management** - to assign the machines used to run automation
- **Alert Handlers** - to configure mail alerts for steps in a project
- **Variables** - to reuse information across projects
- **API Keys** - to configure security access to endpoints

For more information about deploying projects in environments, see Deploy a Project.

References

When you begin working in the environment, the following references can be helpful:

- SAP Note - 3394899 - SAP Build Process Automation - Availability of Environments.
- Blog post on SAP Build Process Automation – Introducing Environments.

Create an Environment

Create an environment to deploy your process automation projects so that they can be run by your target users.

Share an Environment

Share an environment with users or user groups. It is only possible to share **Shared** environments and not any other type of environment.

Accessing Projects in Environments

You can access the projects that have been deployed in the chosen environments.

Unattended Triggers

Manage the scheduled triggers and API triggers used in your environment.

Attended Triggers

Manage the attended triggers for your environment.

Agent Management

Assign the agents for the projects in your environment.

Add Alert Handlers

Alert handlers provide the possibility to send emails for alerts raised in a deployed project. Alerts allow you to define business events in SAP Build Process Automation. Once an alert is created, it can be raised from an existing automation. That's why you need to create an alert handler to use the alert that you have created in your project.

Variables

View the variables created in the projects deployed in your environments.

Add API Keys to an Environment

You can create an API key within SAP Build Process Automation for an environment and use the API key to execute public endpoints bound to SAP Build Process Automation.

Create an Environment

Create an environment to deploy your process automation projects so that they can be run by your target users.

Prerequisites

You've been assigned the **ProcessAutomationAdmin** role collection in your subaccount. For more information, see Authorizations.

Context

Create the different environments required for your process automation projects. Configure your environments to match different requirements, depending on the line of business and security aspects.

Procedure

1. Navigate to the **Control Tower** > **Tenant Configuration** > **Environments**.
2. Choose **Create Environment**.
3. Enter a name for the new environment in the **Name** field.
4. **Optional:** Edit the **Identifier** field. By default, the identifier is the same as the name.
5. **Optional:** Select a **Color** from the drop-down list.
6. Enter a short description of the environment in the **Description** field.
7. Choose **Create**.

Results

The environment is listed on the **Environments** tile in the **Control Tower** and can be selected when you deploy a project in the process editor.

Task overview: Environments

Related Information

Share an Environment
Accessing Projects in Environments
Unattended Triggers
Attended Triggers
Agent Management
Add Alert Handlers
Variables
Add API Keys to an Environment

Share an Environment

Share an environment with users or user groups. It is only possible to share **Shared** environments and not any other type of environment.

Prerequisites

You've been assigned the **ProcessAutomationAdmin** role collection in your subaccount or you've been authorized as an administrator for the environment. For more information, see Authorizations.

Context

You can use a shared environment to deploy and run the same project with different configurations and make these different configurations available to different users. You can also increase the security of your projects by using a shared environment to restrict automation tasks to specific users and machines. Administrators can choose which users or groups have permission to monitor, deploy, view logs, and execute. These permissions in shared environments are detailed, offering IT administrators more authority.

Procedure

1. In the **Environments**, select a shared environment.
2. Open the environment and choose **Share**.
3. On the **Share Environment with Members and Groups** dialog, choose the drop-down menu in the field **Type** and select any one of the following options:

- **Users**: Share the environment with particular users. For more information about creating users in SAP Cloud Identity Services, see User Management.
- **User Groups**: Share the environment with a group of users. For more information about creating user groups, see Create a New Group.

4. Enter the user details (email address) or group name in the **Share with** field.
5. In the **Authorization** field, select the *value-help* to define the privileges you want to grant other users. You can select from:
   - **Monitor**: View environment details, view and monitor the deployed projects, variables, , and other associated resources.
   - **View Logs** View detailed execution logs, which can contain sensitive data.
   - **Deploy**: You are able to deploy and undeploy projects along with access to all **Monitor** rights.
   - **Execute**: Can launch forms, processes, and visibility scenarios.
   - **Administrate**: A user with administrative permissions can add agents or other resources to the environment. You can create, edit, share, and delete environments. It includes both the **Monitor** and **Deploy** permissions.

     Note

     Execute authorization isn't available with this permission.

Task overview: Environments

Related Information

Create an Environment
Accessing Projects in Environments
Unattended Triggers
Attended Triggers
Agent Management
Add Alert Handlers
Variables
Add API Keys to an Environment

Accessing Projects in Environments

You can access the projects that have been deployed in the chosen environments.

Prerequisites

- You've access to the environment.
- You've deployed a project in the environment. For more information, see Deploy a Project.

Context

Accessing your projects in an environment is a simple process that allows you to manage, collaborate, and monitor the deployed projects. Once you enter any environment, you see a list of your projects that are deployed. This centralized view makes it easy to locate and select the project you want to work on.

Each project consists of the following artifacts:

- **Unattended Triggers**: For more information, see Unattended Triggers.
- **Attended Triggers**: For more information, see Attended Triggers.

- **Agent Management**: For more information, see Agent Management.
- **Alert Handlers**: For more information, see Add Alert Handlers.
- **Variables**: For more information, see Variables.
- **API Keys**: For more information, see API Keys.

Procedure

1. To access your projects, navigate to **Control Tower** > **Tenant Configuration** > **Environments**
2. Select the required environment.
3. Select the project that you want to work on from the list.

Task overview: Environments

Related Information

Create an Environment
Share an Environment
Unattended Triggers
Attended Triggers
Agent Management
Add Alert Handlers
Variables
Add API Keys to an Environment

Unattended Triggers

Manage the scheduled triggers and API triggers used in your environment.

You can go to this view by choosing **Control Tower** and then the **Environments** tile. Choose your environment and then the **Unattended Triggers** view.

Filters

You can filter your triggers by:

- **Type**
- **Time range validity**
- **Project**
- **Attributes**

Triggers View

Column	Description
	Any warnings or notifications relating to the trigger
Trigger	The trigger that you have configured and the trigger type
Executing	The name of the automation or process to which you've added the trigger
Project &Version	The project and the version that contains the automation and that you've deployed

Column	Description
Next Execution	When the next execution of your deployed automation is planned
Last Updated	When the deployed automation was last updated You can sort data displayed in this column using or .
Attributes	The matching attributes associated with the triggers
Trace Activation	Any traces that have been activated on the trigger Click to see more information about the traces.
Actions	Click to see the following options for managing the trigger: • **Enable** or **Disable** • **Run now** This option is available for scheduled triggers only. • **Edit** • **Add Notifier** • **Delete** • **Activate Traces**

To add an unattended trigger to a project choose the **Add Trigger**.. For more information, see Add an Automation Trigger to a Project.

Parent topic: Environments

Related Information

Create an Environment
Share an Environment
Accessing Projects in Environments
Attended Triggers
Agent Management
Add Alert Handlers
Variables
Add API Keys to an Environment

Attended Triggers

Manage the attended triggers for your environment.

You can go to this view by choosing **Control Tower** and then the **Environments** tile. Choose your environment and then the **Attended Triggers** view.

To add an attended trigger to a project, choose **Create Attended Trigger** . For more information, see Add an Automation Trigger to a Project.

Filtering and Sorting

Choose *Filter* to filter your triggers by:

- **Time range validity**
- **Project**
- **Attributes**

Choose *Sort* to sort by **Trigger** and **Last Updated**.

Triggers View

Column	Description
	Any warnings or notifications relating to the trigger
Trigger	The trigger that you have configured
Executing	The name of the automation or process to which you've added the trigger
Project &Version	The project and the version that contains the automation and that you've deployed
Last Updated	When the deployed automation was last updated
Attributes	The matching attributes associated with the triggers
Trace Activation	Any traces that have been activated on the trigger Click to see more information about the traces.
Actions	Click to see the following options for managing the trigger: • **Enable** or **Disable** • **Edit** • **Delete** • **Activate Traces**

Parent topic: Environments

Related Information

Create an Environment
Share an Environment
Accessing Projects in Environments
Unattended Triggers
Agent Management
Add Alert Handlers
Variables
Add API Keys to an Environment

Agent Management

Assign the agents for the projects in your environment.

You can assign an agent or an agent group to specific automation jobs in a project within an environment. An agent can be added to more than one environment, including a public environment.

For more information about adding an agent to an environment, see Add an Agent

For more information about configuring agents, see Configure Agents.

Parent topic: Environments

Related Information

Create an Environment

Add an Agent

Add a single agent or multiple agents to an environment in SAP Build Process Automation..

Prerequisites

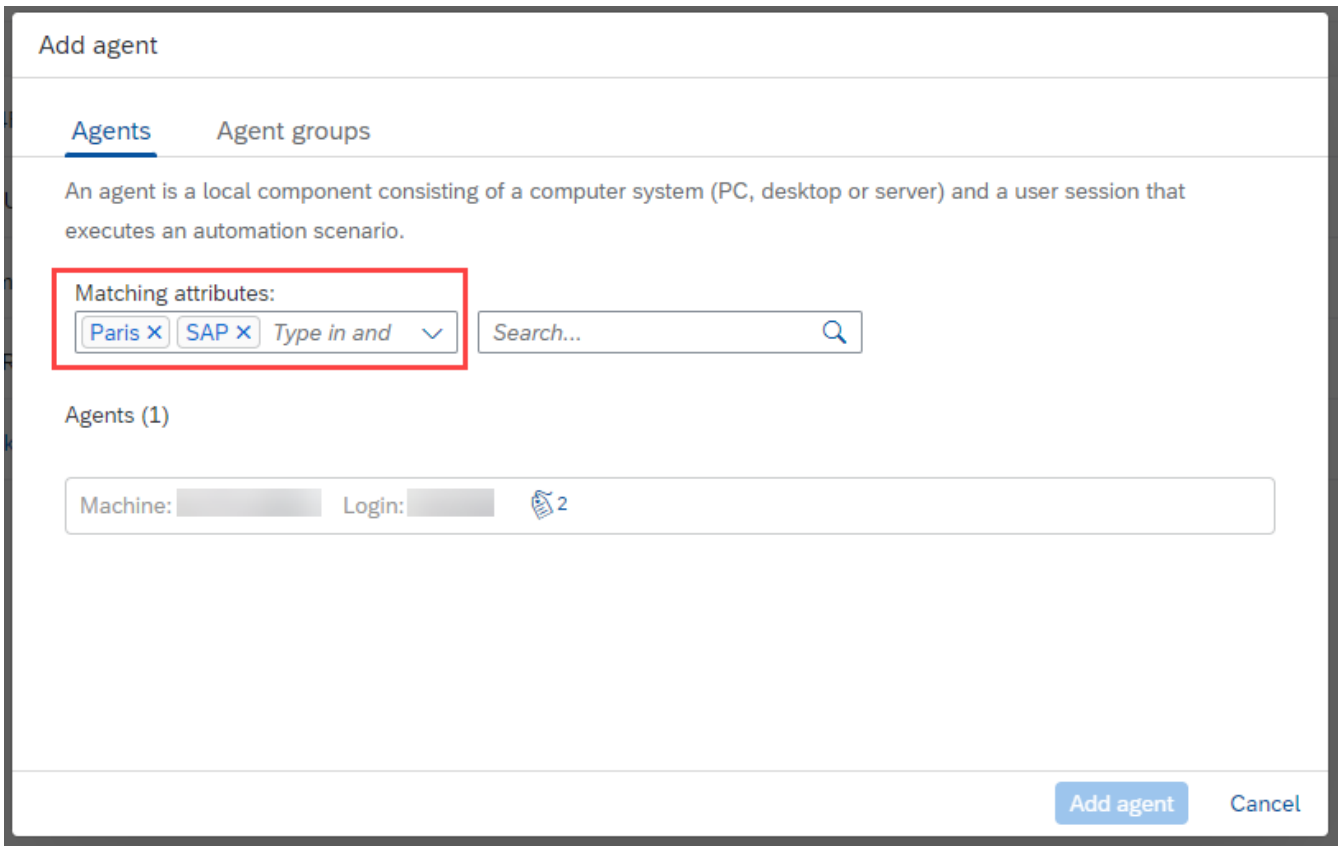
Make sure that your agent is registered in the **Agents List** tab first. For more information on how to register a new agent, see Register the Desktop Agent Tenant.

Procedure

1. In the **Control Tower**, go to the **Environments** view and choose the **Agents Management** view for your environment. Choose **Add Agent**.
2. In the **Add agent** dialog box, find your agent in the **Agents** list.
3. **Optional**: Look for a specific agent with the search bar.
4. **Optional**: Click on the **Matching attributes** dropdown list to select one or more matching attributes and filter the **Agents** list.

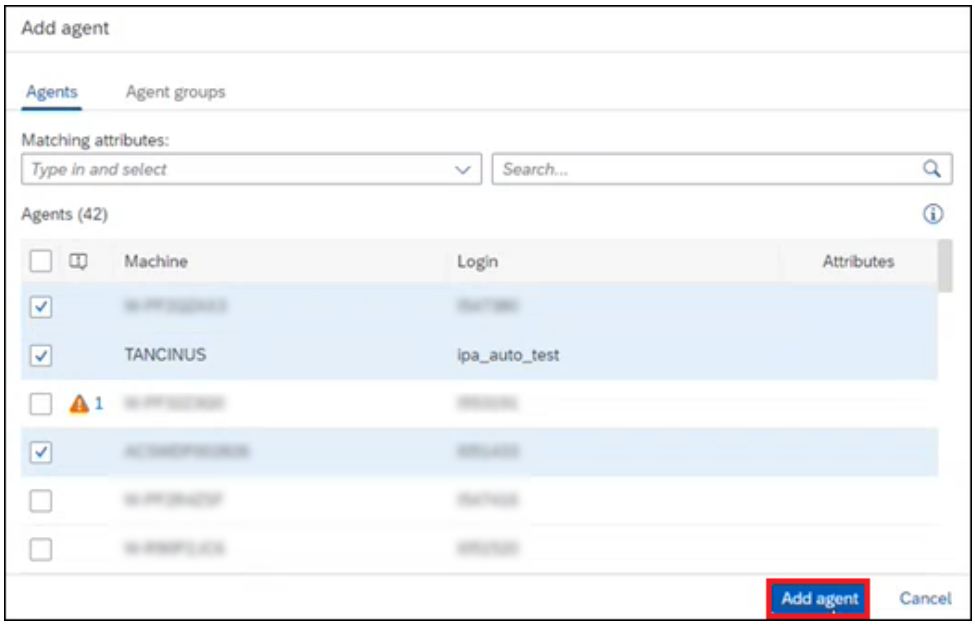
Note

From the **Matching attributes** dropdown list, when you select the respective checkbox beside the attribute value, the respective matching agent is displayed. When you select multiple checkboxes beside the attribute values under various attribute names, the respective matching agents are displayed. You can also enter the attribute name or the attribute value to search the respective matching agents.



On the **Agents** tile, you can see the number of attributes beside the (Show the attributes) icon. When you click on the number of attributes, the attributes that are defined on the agent are displayed.

5. Select a checkbox or multiple checkboxes to add an agent or multiple agents.
6. Click **Add agent**.



Results

Your agent or multiple agents are added to your environment.

Add Alert Handlers

Alert handlers provide the possibility to send emails for alerts raised in a deployed project. Alerts allow you to define business events in SAP Build Process Automation. Once an alert is created, it can be raised from an existing automation. That's why you need to create an alert handler to use the alert that you have created in your project.

Prerequisites

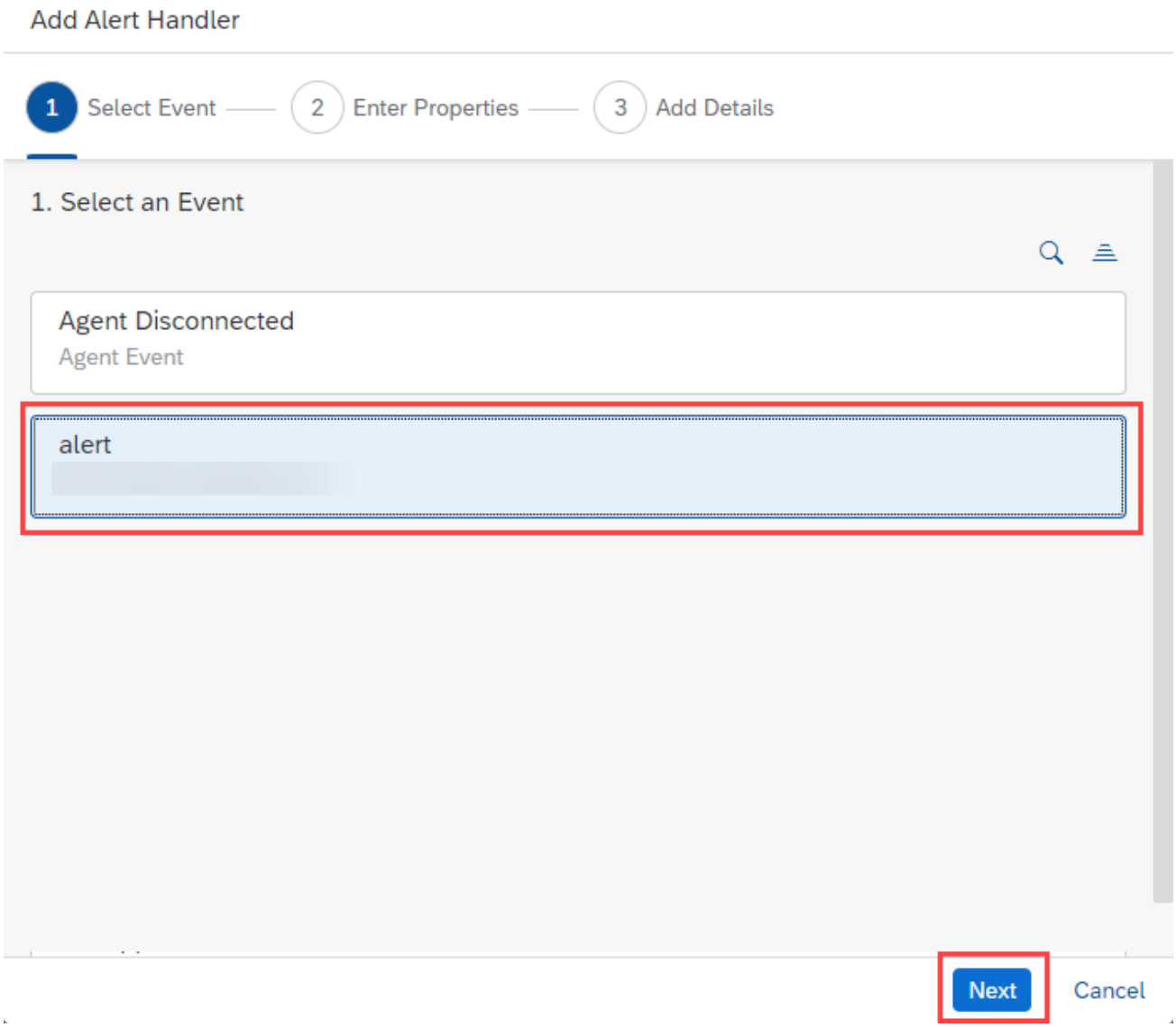
Before creating an alert handler, you first need to configure a Simple Mail Transfer Protocol (SMTP) server. For more information, see Configure an SMTP Mail Destination.

Note

You must deploy a project with an alert to create an alert handler. For more information, see Create an Alert.

Procedure

1. Choose **Control Tower**.
2. Choose **Alert Handlers** and choose **Add Alert Handler**.
3. In the **Select Event** dialog box, select the alert on which an email should be sent when it is raised in an automation. Click **Next**.



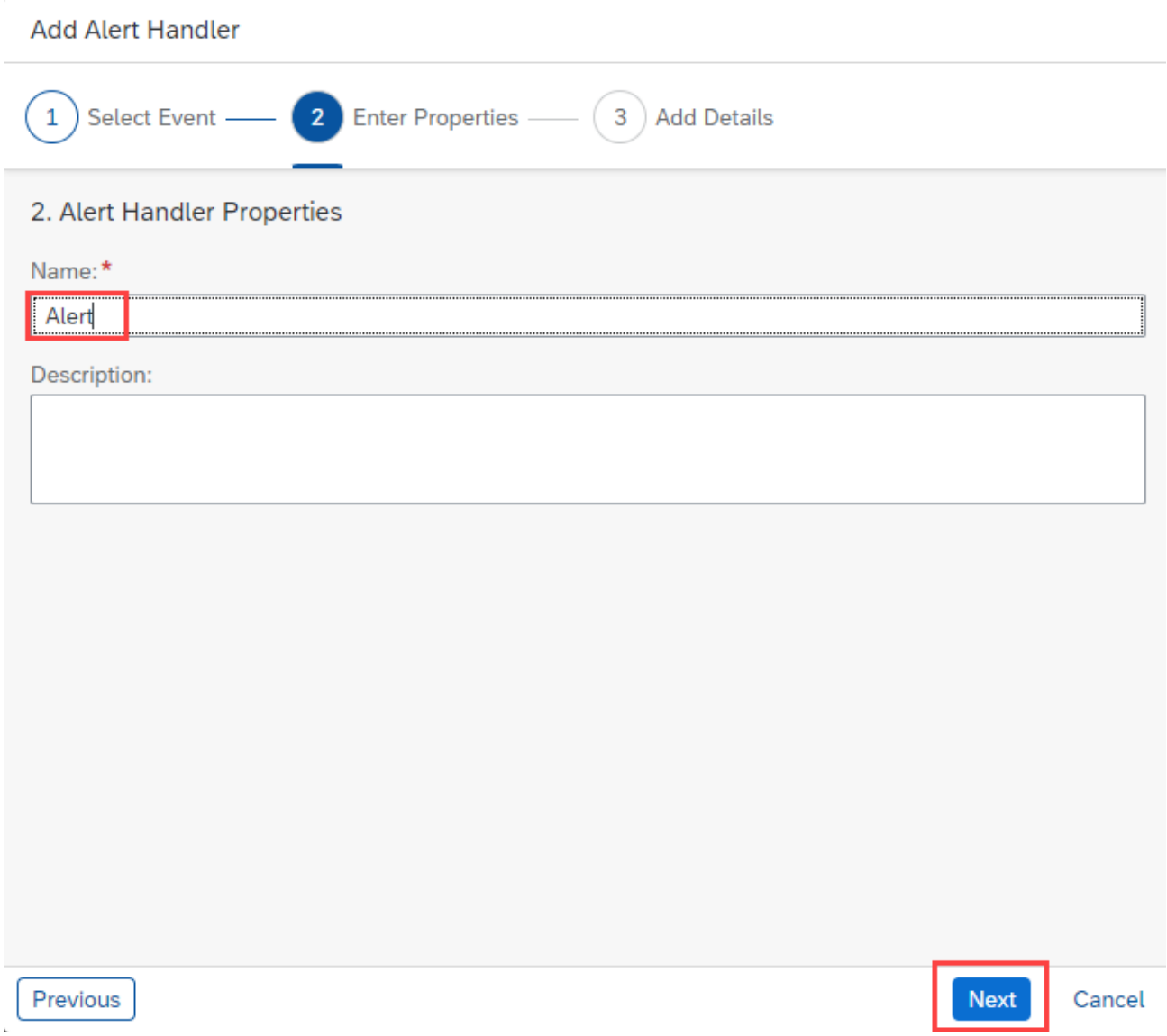
Note

Alert handlers are only triggered when the selected alert as an event source is emitted from an automation.

Note

Only alerts that are raised by an activity will become available as an event source for an alert handler after the deployment of the project.

4. In the **Enter Properties** dialog box, enter a name in the **Name** field.
5. **Optional:** Enter a short description of the alert handler in the **Description** field.
6. Click **Next**.



7. **Optional:** In the Add Details dialog box, from the Available variables **(1)** list, you can use the following variables in the Subject or Content field:
   - The `${alert}` variable has the following sub-properties:
     - `${alert.name}`
     - `${alert.message}`
     - `${alert.severity}`
     - `${alert.parameters}`
   - The `${automation}` variable has the following sub-properties:
     - `${automation.monitoringLink}`
     - `${automation.name}`
     - `${automation.packageName}`
     - `${automation.version}`
     - `${automation.triggerCreator}`

       Note

       The `${triggerCreator}` variable allows you to get information on the creator of the trigger that started the automation.

       ${triggerCreator} is not available for attended triggers and processes.
   - The `${context}` variable has the following sub-properties:
     - `${context.agentNames}`

- `${context.agentLinks}`
- `${context.environmentName}`
- `${context.environmentLink}`

- The `${event}` variable has the following sub-properties:
  - `${event.type}`
  - `${event.name}`
  - `${event.message}`
  - `${event.timestamp}`

Note

The sub-properties mentioned above refer to the raised alert and the context of the business event.

8. Enter the email address of the recipient in the **Recipients** **(2)** field.

Note

The **Recipients** field can have multiple email addresses. You can also provide email addresses using alert parameters, for example, `${alert.parameters.recipients}`.

In the **Recipients** field, you can also use dynamic recipients through variable(s) from the alert parameters. Only alert parameters can be used as a variable for your recipients. Other available variables such as `${context}` or `${automation}` or `${event}` are not possible to use in the **Recipients** field. A variable can hold one or multiple email addresses of the recipients.

Note

- Existing alert parameters can be accessed by directly using the curly braces syntax (for example, `${alert.parameters.recipients}` with a parameter named “recipients” in the alert).
- You can only use the “parameters” sub-property of the `${alert}` variable in the **Recipients** field.
- If you delete the parameter of an alert that is being used in the **Recipients** field, the alert handler is marked as “invalid” and a warning message is displayed on the screen to fix the issue.
- The alert message is sent only to the valid email address.

9. Enter the subject of the email in the **Subject** **(3)** field.
10. Enter the content of the email in the **Content** **(4)** field.

Note

Using the What You See Is What You Get (WYSIWYG) editor, you can visually format the content of your email and easily insert variables.

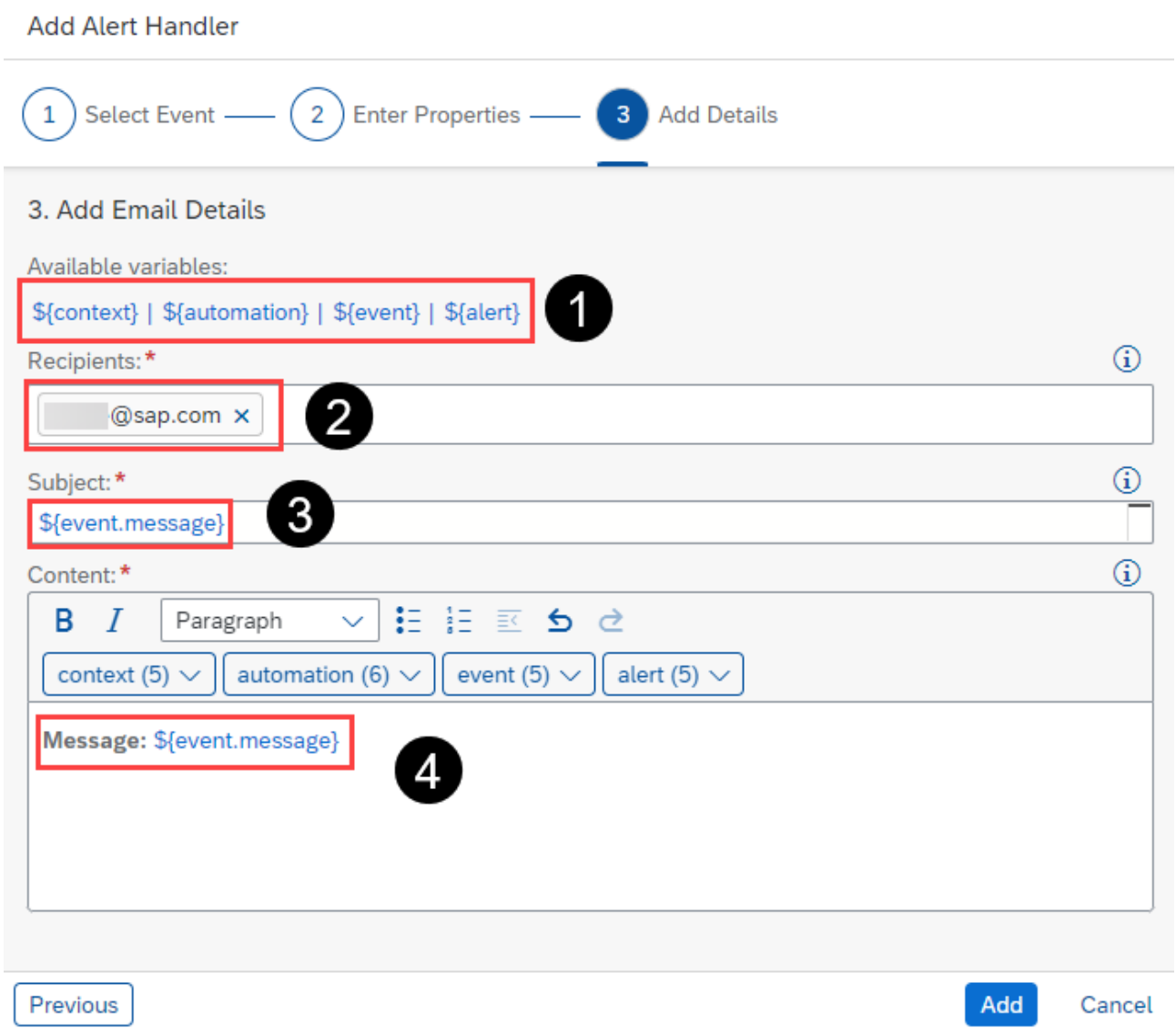
To start formatting, select the text or paragraph and then select the following formatting option which you want to apply:

Name	Button	Description
Bold		To bold any text. Applying the formatting twice removes the formatting.
Italic		To italicize any text. Applying the formatting twice removes the formatting.
Font-size dropdown		To change the font size of any text. By default, the **Paragraph** option will be selected. From the dropdown list, select the font size you want to apply: **Paragraph**, **Heading 1**, **Heading 2**, **Heading 3**, **Heading 4**, **Heading 5**, or **Heading 6**.
Bulleted list		To create a bulleted list.

Name	Button	Description
Numbered list		To create a numbered list.
Remove indentation		To remove the indent of any paragraph. By default, this button will be disabled. Once you add bullets or numbers to any paragraph, it will be enabled.
Undo		To undo the previous action. By default, this button will be disabled.
Redo		To restore the most recent **Undo** operation. By default, this button will be disabled.

You can select and insert variables in the content by clicking the dropdown button of the respective variable such as `context`, `automation`, `event` and `alert`.

11. Click **Add**.



Results

The created **Alert Handler** will be triggered whenever the selected event source is raised from an automation in the corresponding project.

ℹ Note

An alert handler becomes invalid when the definition of the alert gets changed, and the variables used in the alert handler details are no longer valid. Emails for invalid alert handlers will still be sent. Variables that are no longer available will be replaced with "undefined".

Task overview: Environments

Related Information

Create an Environment
Share an Environment
Accessing Projects in Environments
Unattended Triggers
Attended Triggers
Agent Management
Variables
Add API Keys to an Environment
Create an Alert

Alert Handler for the Agent Disconnected Event

Prerequisites

Add an agent to SAP Build Process Automation. For more information, see Add an Agent.

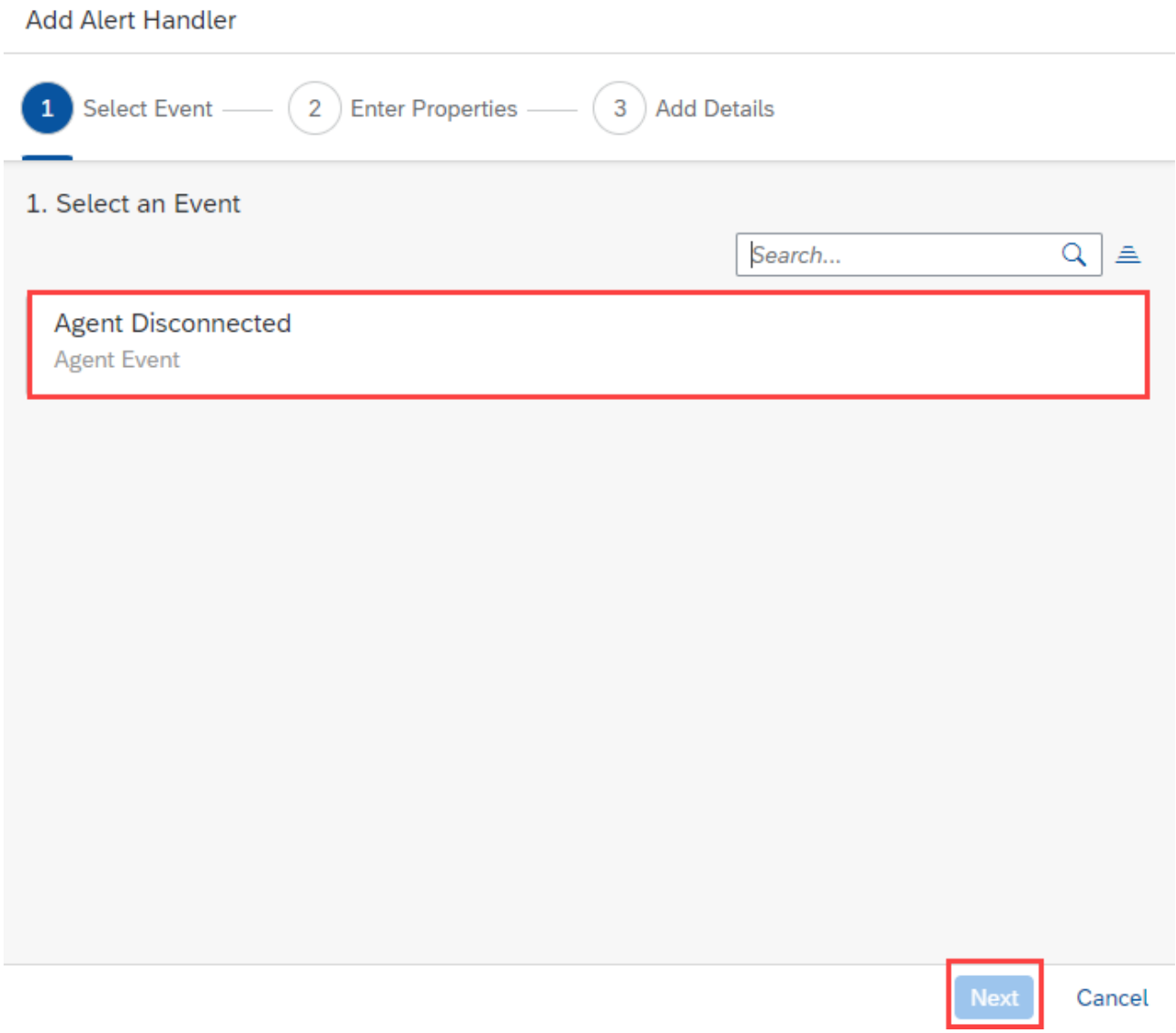
Before creating an alert handler, you first need to configure a Simple Mail Transfer Protocol (SMTP) server. For more information, see Configure an SMTP Mail Destination.

Context

You can define a notification that you want to receive when an agent of your environment is disconnected. You can add a constraint (such as the duration of the agent disconnection) so that you receive the notification only if the agent is disconnected for a defined period.

Procedure

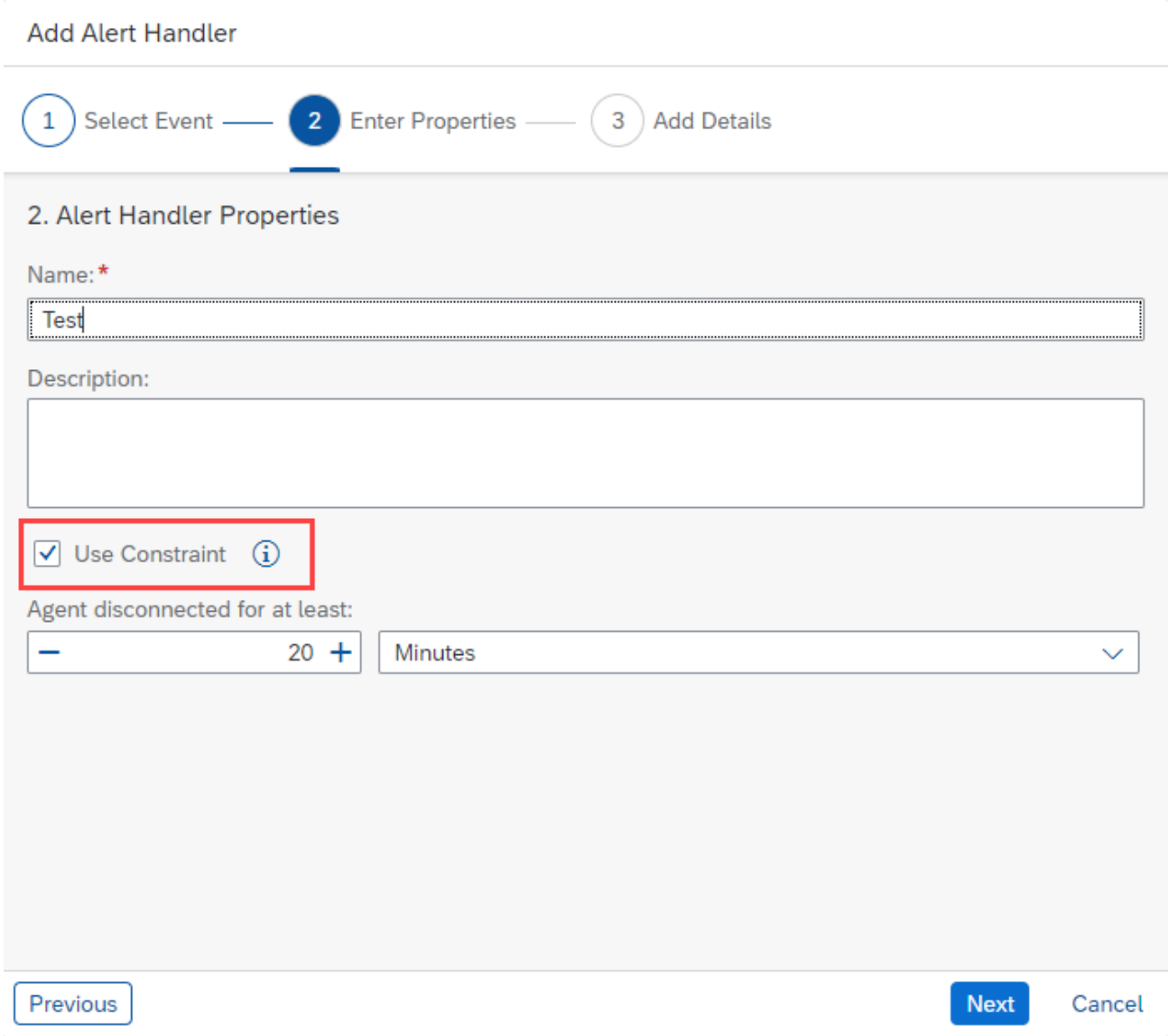
1. Go to the **Settings** tab.
2. Go to **Alert Handlers** and click **Add Alert Handler**.
3. In the **Select Event** dialog box, select the **Agent Disconnected** event and click **Next**.



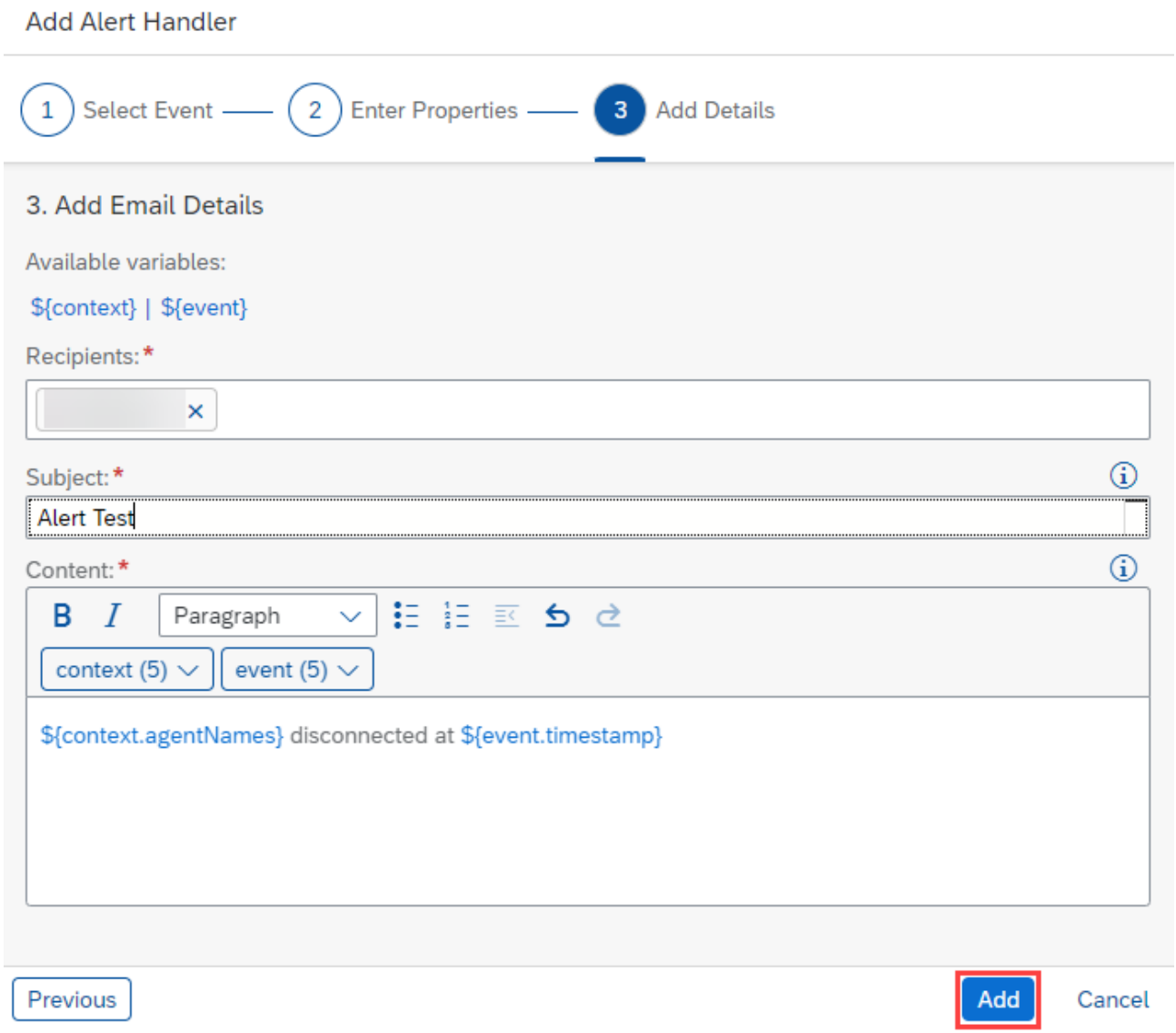
4. In the **Enter Properties** dialog box, enter a name in the **Name** field.
5. **Optional:** Enter a short description in the **Description** field.
6. **Optional:** Select the **Use Constraint** checkbox to set constraints so that the alert will be sent only if the agent is disconnected for more than the threshold value (in minutes or hours). For example, if the threshold value is set for 10 minutes, the alert will be sent 15 minutes after the agent disconnected at the latest. The threshold value can be defined in 5 minutes or 1-hour interval. By default, the **Use Constraint** checkbox is not selected.

ℹ Note

If you do not set any constraints, the alert will be sent every time up to 5 minutes after the event occurs.

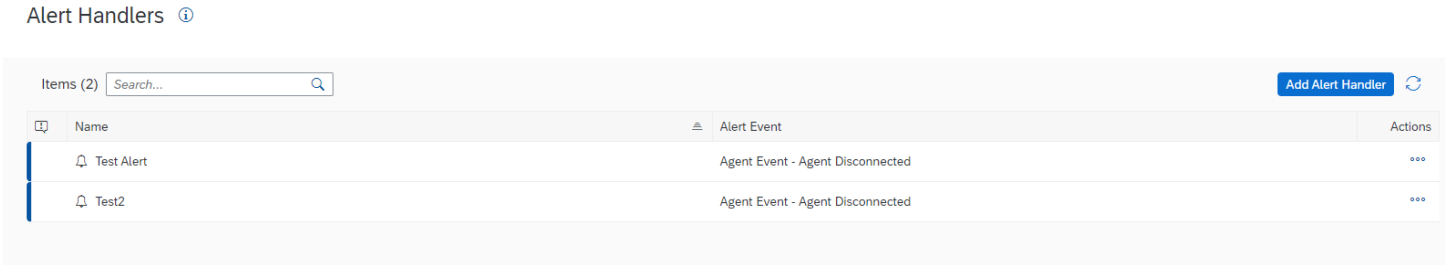


7. Click **Next**.
8. In the **Add Details** dialog box, enter the email address of the recipient in the **Recipients** field.
9. **Optional:** From the **Available variables** list, you can use the variables in the **Subject** or **Content** fields. For more information on how to format mail templates and insert variables in the **Content** field, see Add Alert Handlers.
10. Enter the subject of the email in the **Subject** field.
11. Enter the content of the email in the **Content** field.
12. Click **Add**.



Results

The created **Alert Handler** will be triggered whenever an agent is disconnected for more than the threshold value.



Variables

View the variables created in the projects deployed in your environments.

You can see a list of the variables created and their values in the **Variables** tab for the relevant environment. Choose **Control Tower** and then the **Environments** tile.

The name and type of each variable is listed. You can filter the list of variables by project. You can also see the project and version in which each variable has been created as well as when it was last updated. You can also see if the values of the variables are linked to other variables.

Choose *Display* in the **Actions** column,to see the **Variable Information**. This popup displays the value of the variable and the other variables to which this variable is linked, if applicable.

Parent topic: Environments

Related Information

Create an Environment
Share an Environment
Accessing Projects in Environments
Unattended Triggers
Attended Triggers
Agent Management
Add Alert Handlers
Add API Keys to an Environment

Add API Keys to an Environment

You can create an API key within SAP Build Process Automation for an environment and use the API key to execute public endpoints bound to SAP Build Process Automation.

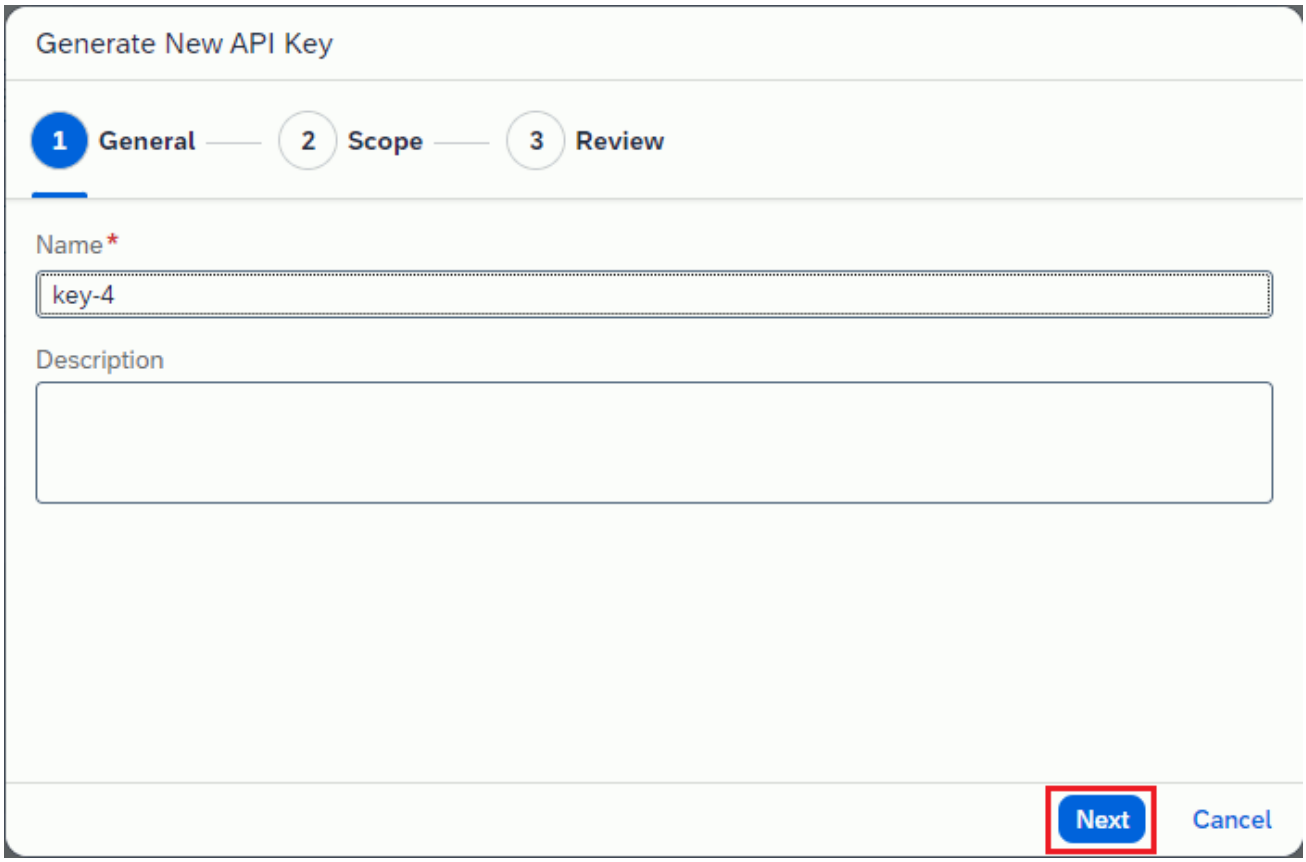
Context

While creating an API key, you will be able to see the complete API key, copy it to the clipboard once and paste it into a notepad or text file. As a best practice, you must store the API key in a safe place so that you can retrieve it in the future. You won't be able to copy/paste the key once it is added to your environment.

Perform the following steps to create API keys within SAP Build Process Automation:

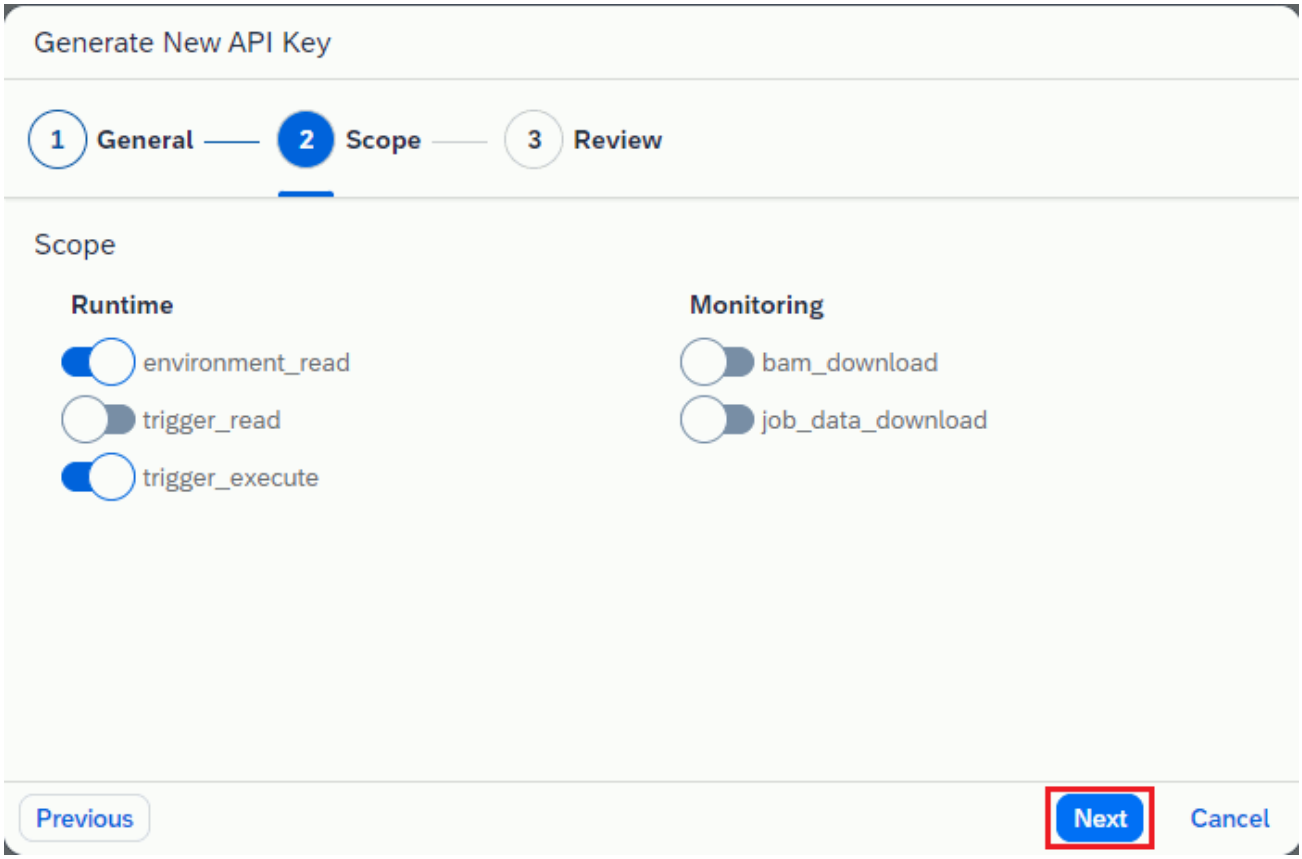
Procedure

1. Go to **Control Tower** > **Environments**. Choose an environment and then the **API Keys** view.
2. Choose **Add API Key**.
3. In the **General** section of the **Generate New API Key** window, enter an API key name in the **Name** field and a description (**optional**) in the **Description** field. Click **Next**.

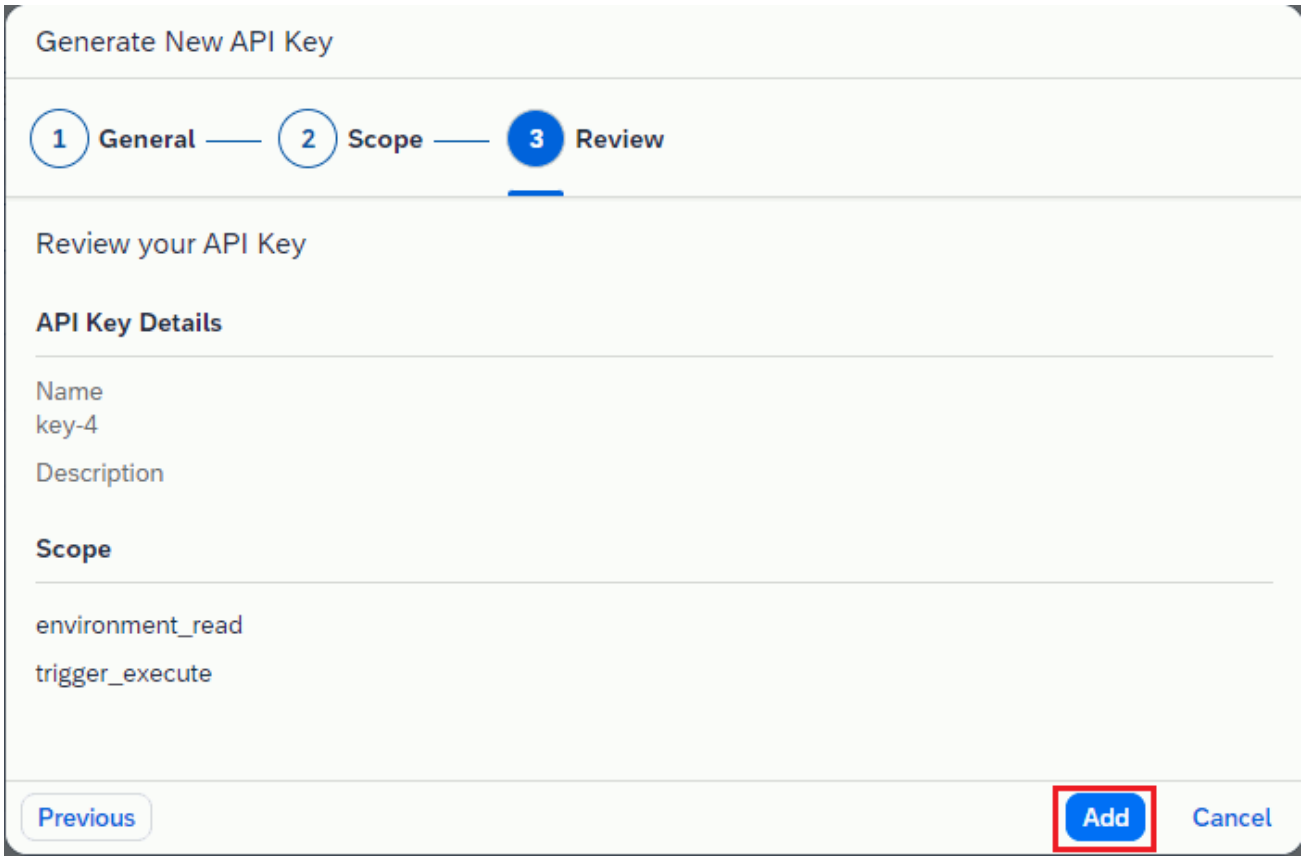


4. In the Scope section of the Generate New API Key window, click the toggle button(s) to choose a single or multiple scopes for your API key. Click Next.

   For more information about the scopes, see API Keys.

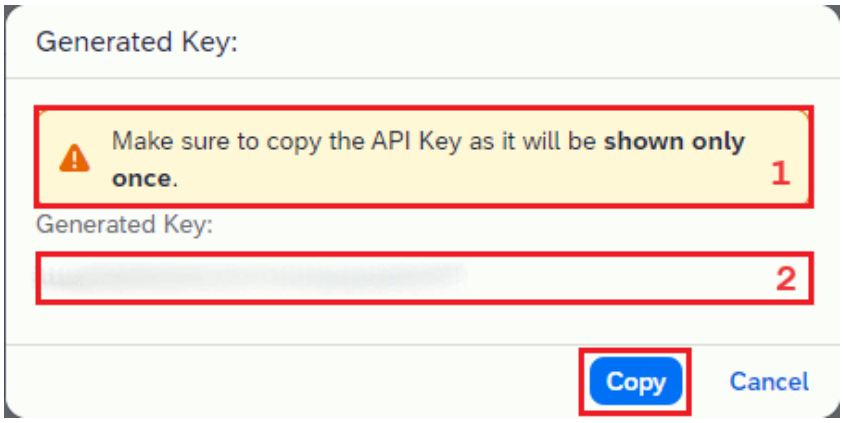


5. In the Review section of Generate New API Key window, you can review the setup of your API key. Click Add.



The Generated Key dialog box is displayed and you can view the generated API key.

6. Click Copy. The API key will be copied to the clipboard once and you can paste it into a notepad. You can also select the API key manually and copy it.



⚠ Caution

This is the only time when you can see and copy the API key. Later it won't be possible anymore to copy the API key once the dialog box is closed.

Results

The API key is generated for SAP Build Process Automation and appears in the API Keys table. In this table, you can view the name, scope(s) and author of each API key. You can also know when each API key was generated.

Next Steps

Click the icon to delete the API keys that are valid for SAP Build Process Automation.

Task overview: Environments

Related Information

Configure Agents

An agent is a local component that includes a computer system (PC, desktop, or server). It's identified by the name of the computer and the name of the user session that executes an automation.

You can configure your agents from the SAP Build Process Automation Lobby by going to **Control Tower** and then the **Agents** tile. Here you can:

- Create a new Agent or Agent Group.
- Register your agents to your project so that packages and jobs can be correctly distributed.
- See an overview of the **Agents List** and **Agent groups**, with their status, version, and mode.
- Register multiple agents on a specific agent group in the **Mass Registrations** tab.
- Enter an alias for your agent. If you don't enter an alias, or you remove the alias, the machine name is displayed.

ℹ Note

As a best practice we recommend that you always install the latest available version of the Desktop Agent to benefit from the latest features.

The following table displays the comparison between **Agents** and **Agent Groups**:

	Agents	**Agent groups (without Mass Agent Registration)**	**Agent groups (with Mass Agent Registration)**
Recommended number of Login/Machine users	Less than 5.	Between 5 and 20.	Greater than 20.
Registration	User must log in with credentials in the agent.	User must log in with credentials in the agent.	Registration is done via Mass Agent Registration. This allows you to register a large number of agents in an agent group and users won't need to log in.
Agent Sharing	Agent must be connected to your SAP Build Process Automation tenant before sharing to Agent Group.	Agent must be shared to the Agent Group and then approved.	Not needed. However, agents need to be approved.
Maintenance	New agents must be logged in.	New users only need to be added to the agent group.	New token needs to be generated to add new agents to the agent group.

Related Information

Add an Agent
Add an Agent Group

Overview of Agents

Manage registered agents.

In SAP Build, choose **Control Tower** then **Agents**.

Column	Description
Warning	Hover over or choose the symbol to see more details.
Machine / Alias	Provides the machine name or its alias. Hover over the alias to see the machine name.
Login	Provides the user details such as user name or user ID.
Status	Provides the agent status. An agent can have one of the following statuses: • **Disconnected** - The agent is not connected. • **Preparing** - The agent is connecting (transient status). • **Idle** - The agent is connected but no project has been loaded. • **Starting** - The agent is connected and is loading a project (transient status). • **Ready** - The agent is connected and a project has been loaded. The agent is ready to run a project. • **Paused** - The agent is connected and a project has been loaded, but the agent is unable to run the project (status initiated by the agent). • **Running** - The agent is connected and is running a project. • **Busy** - The agent is connected but can't do anything because of maintenance operation. → **Tip** After you have configured the system and before you execute any job, check the **Agent Page** to ensure that the agents you registered are connected correctly.
Attributes	For more details on attributes, see Agent Attributes.
Mode	Provides the mode name. There are two modes (in addition to the Test mode for debugging): • Attended (Interactive) • Unattended (Background) ⓘ **Note** For agents in version 1.0.5 or higher, the mode is read-only. You can only update the mode in the Agent systray on your machine. → **Tip** If your agent is in Test mode, it cannot run attended or unattended automations. Restart your Desktop Agent to end the Test mode and resume your running automations.
Version	Provides the agent version installed on the machine.
Project	Provides the agent's project name. For more information on creating a project, see Create and Manage Projects.
User	Provides the user name.

Shared	Provides the details of a shared agent. If an agent is shared, choose *to see the details.*
Last Updated	Provides the date and time of the last event (connected, disconnected, verifying token, token approved).
Actions	You can Share, Manage Attributes, Activate Support, Edit Alias, and Delete.

Further Details

Choose an agent in the list for further details.

From this window, you can disconnect the agent via the button at the top right.

Managing Disabled Automation Scenario

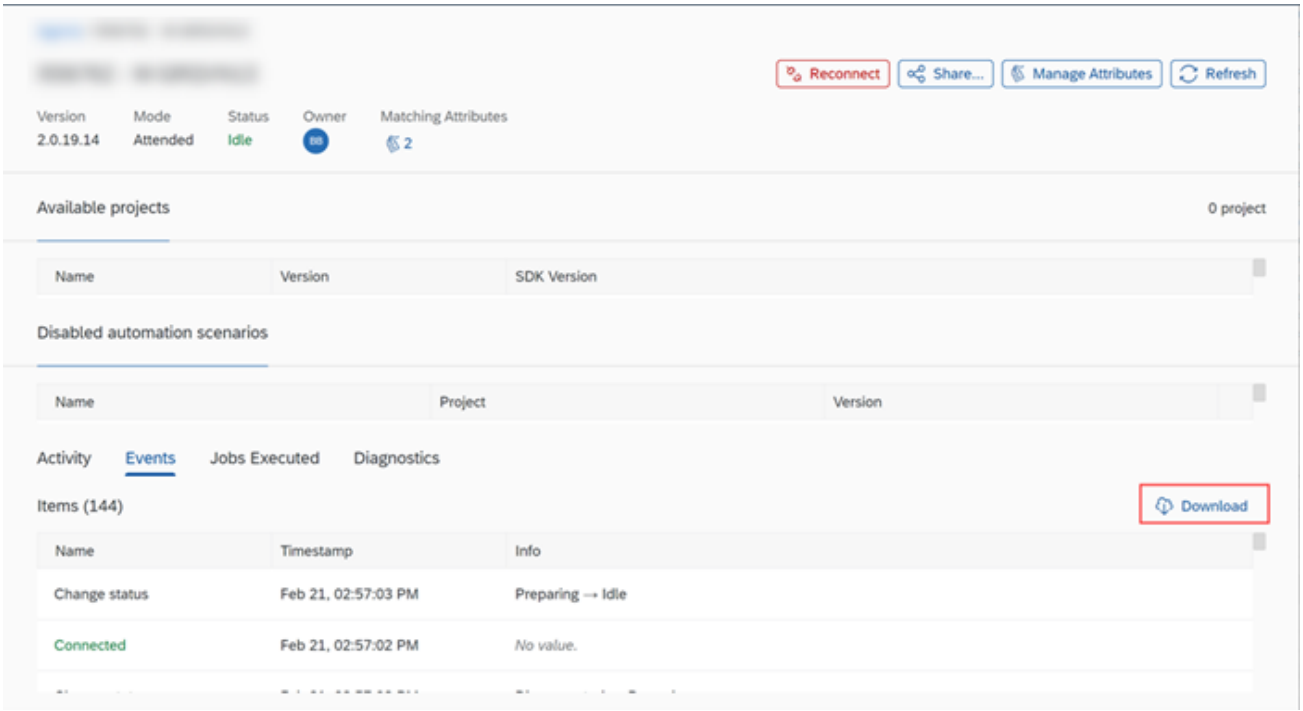
The Disabled Automation scenarios feature allows you to disable an automation scenario with a galaxy API if you want to exclude it from an agent. For example, an SAP logon application is not working or is not installed on your machine. So, you want to be sure that your agent doesn't get distributed jobs on the SAP logon application scenarios. You should then remove each automation scenario listed as disabled from a specific agent.

On the Agent page, you can see a table called Disabled Automation scenarios. It indicates the automation name, the project name, and the project version. You can also remove the disabled scenario from this table by clicking the button once you've adapted your agent to run the bots correctly.

Troubleshooting

Below the projects list is a history of events and status exchanged by SAP Build Process Automation and the agent.

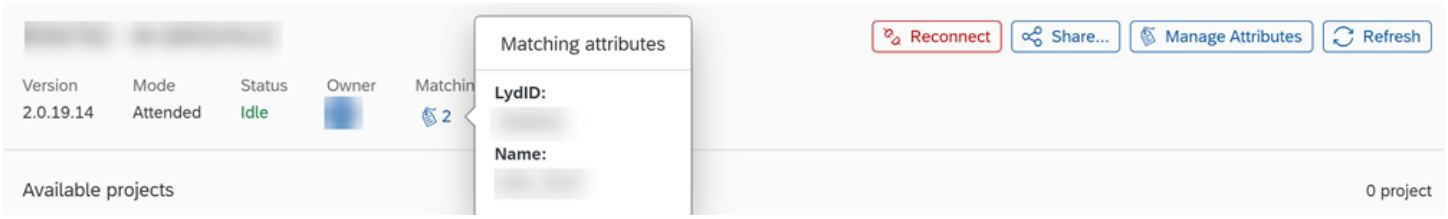
You can download the events history in CSV format via the Download button:



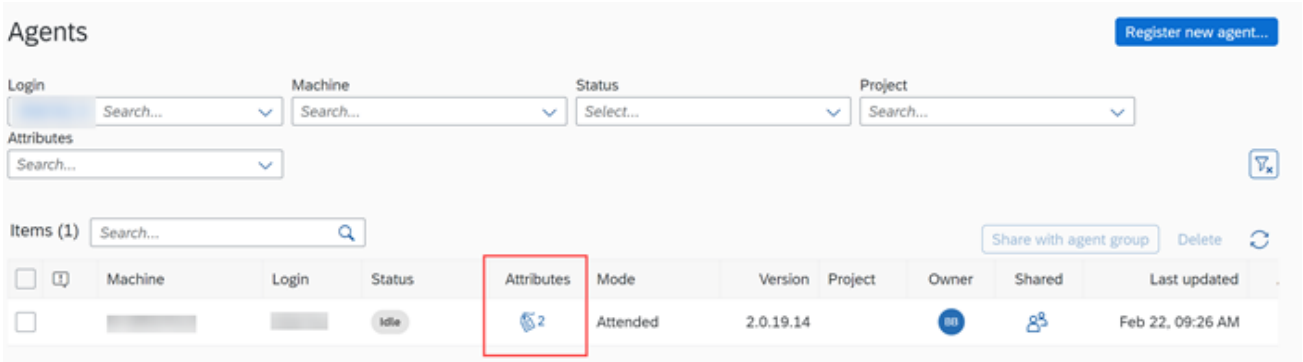
The Jobs executed tab shows the name, project name, and status of jobs executed by the agent.

Matching Attributes

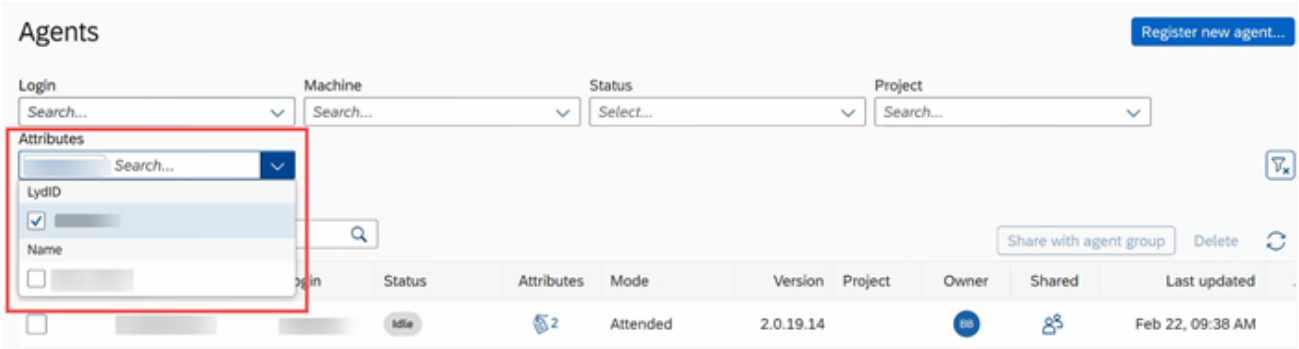
In the overview page of your agent, you can see the number of attributes beside the icon. When you click on the number of attributes, the attributes that are defined on the agent are displayed.



In the Agents view page, you can see the number of attributes beside icon. When you click on the number of attributes, the attributes that are defined on the agent are displayed.



You can select a specific attribute value, or values if you have multiple, from the Attributes search bar, and the respective corresponding agent is displayed.



Create an Agent Group

You use agent groups to control access to the automation tools when you need to deploy a great number of agents.

Prerequisites

Optional: You've modeled the agent group in a CSV (semi-colon separated) file that is less than 5 MB in size.

Context

Agent groups allow you to organize your agents in a parent-child relationship, using either:

- the computer name on which the agent is installed
- the login name of the person who will connect to the agent

You can group your agents via two criteria: groups of machines and group of logins. Machine groups allow any agent connecting from a given machine to use the automation tools. Login groups restrict usage to the specified list of users. By using both types of agent groups, you can create more powerful combinations and provide better control for administrators.

Before executing an automation, you need to create a machine or login agent group and add their agents to your group. You can then use the agent group in any automation.

> **Note**
> The creation of an agent group is a mandatory step to enable the registration of a desktop agent.

Procedure

1. Go to the **Settings** tab and click **Agent groups**.
2. Click **New agent group**.
3. **Optional:** If you want to import your group from a CSV file that is separated using semi-colons (;), click the button next to the **Agent group** field and choose your group file.
4. Enter a name for your agent group (max. 256 characters).
5. Choose the type of agent group (`machine` or `login`).
6. Click **Create**.

Results

Your agent group now appears in the list.

Further Details

On the **Agent Groups** overview page, you can view the number of registered agents in the **Registered Agents** column.

Add an Agent Group

You can add an agent group to SAP Build Process Automation.

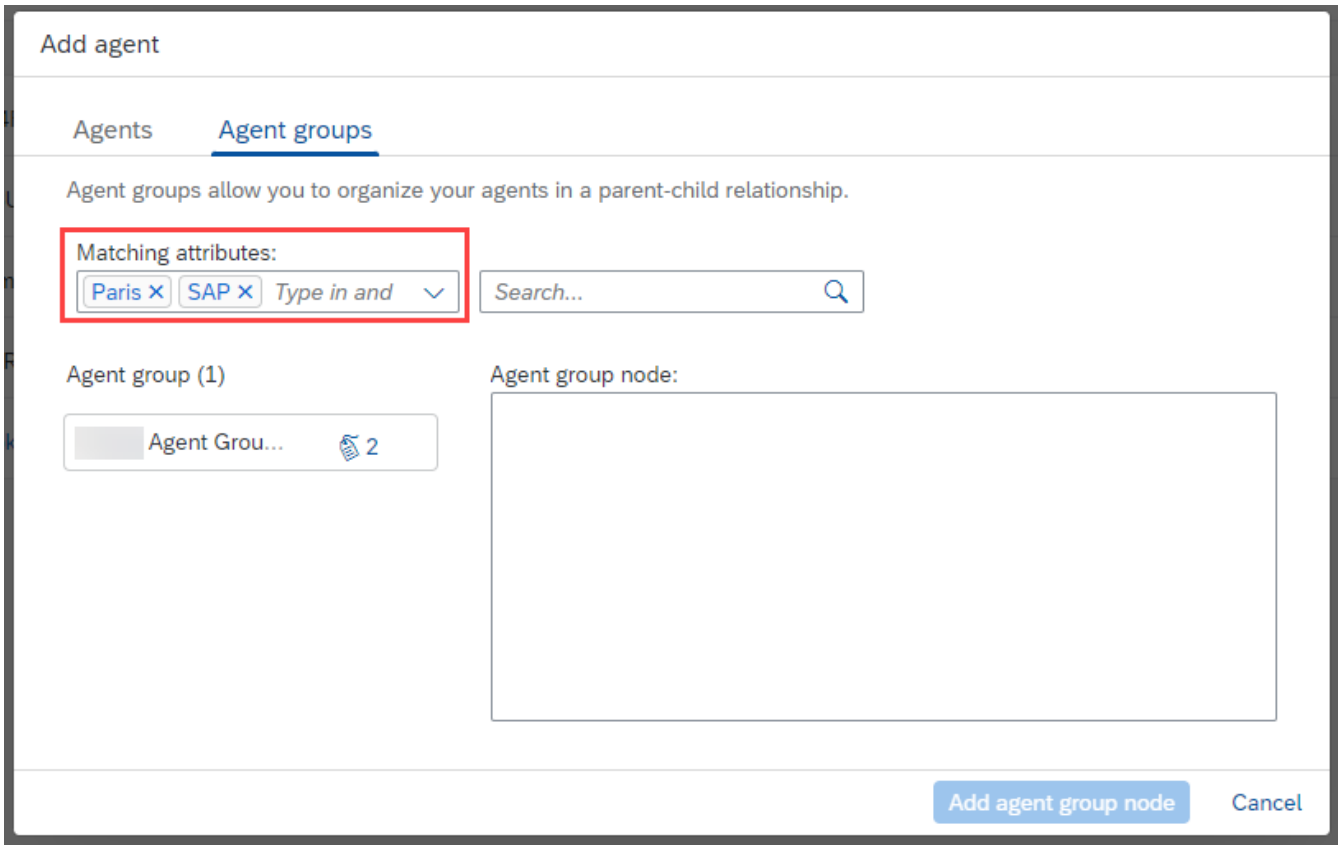
Prerequisites

Make sure that your agent is registered in the **Agents List** tab first. For more information on how to register a new agent, see Register the Desktop Agent Tenant.

Procedure

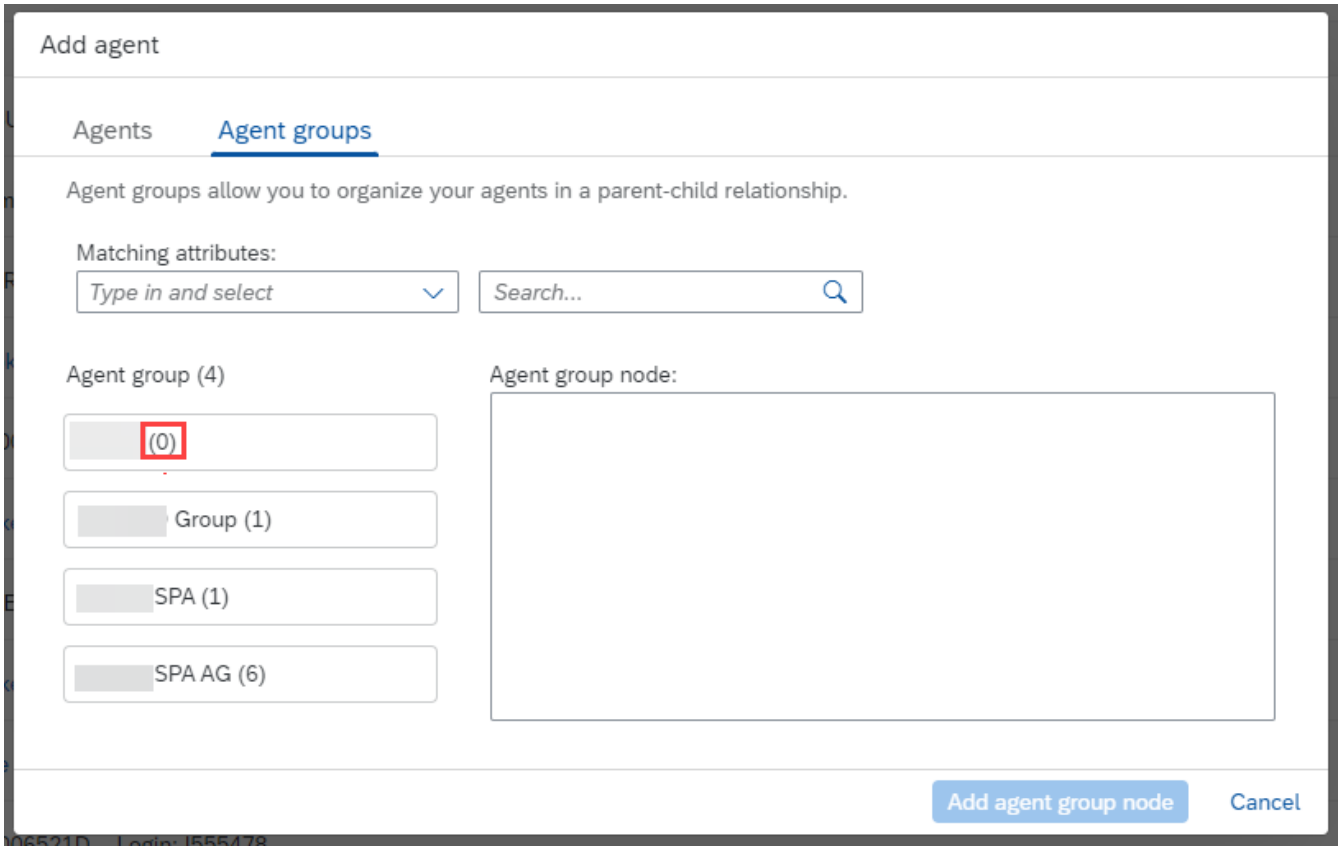
1. In **Control Tower**, go to the **Agents Management** tab and click on **Add Agent**.
2. Click on the **Agents Group** tab.
3. Find your agent in the **Agent group** list.
4. **Optional:** Look for a specific agent group with the search bar.
5. **Optional:** Click on the **Matching attributes** drop-down list to select one or more matching attributes and filter the **Agent group** list.

> **Note**
> From the **Matching attributes** drop-down list, when you select the respective checkbox beside the attribute value, the respective matching agent group is displayed. When you select multiple checkboxes beside the attribute values under various attribute names, the respective matching agent groups are displayed. You can also enter the attribute name or the attribute value to search the respective matching agent group(s).



On the **Agent Groups** tile, you can see the number of attributes beside the icon. When you click on the number of attributes, the attributes that are defined on the agent group are displayed.

Here, you can also view the count of matching agent groups besides the agent group names. The following screenshot shows the number of agents that are matching with the agent group.

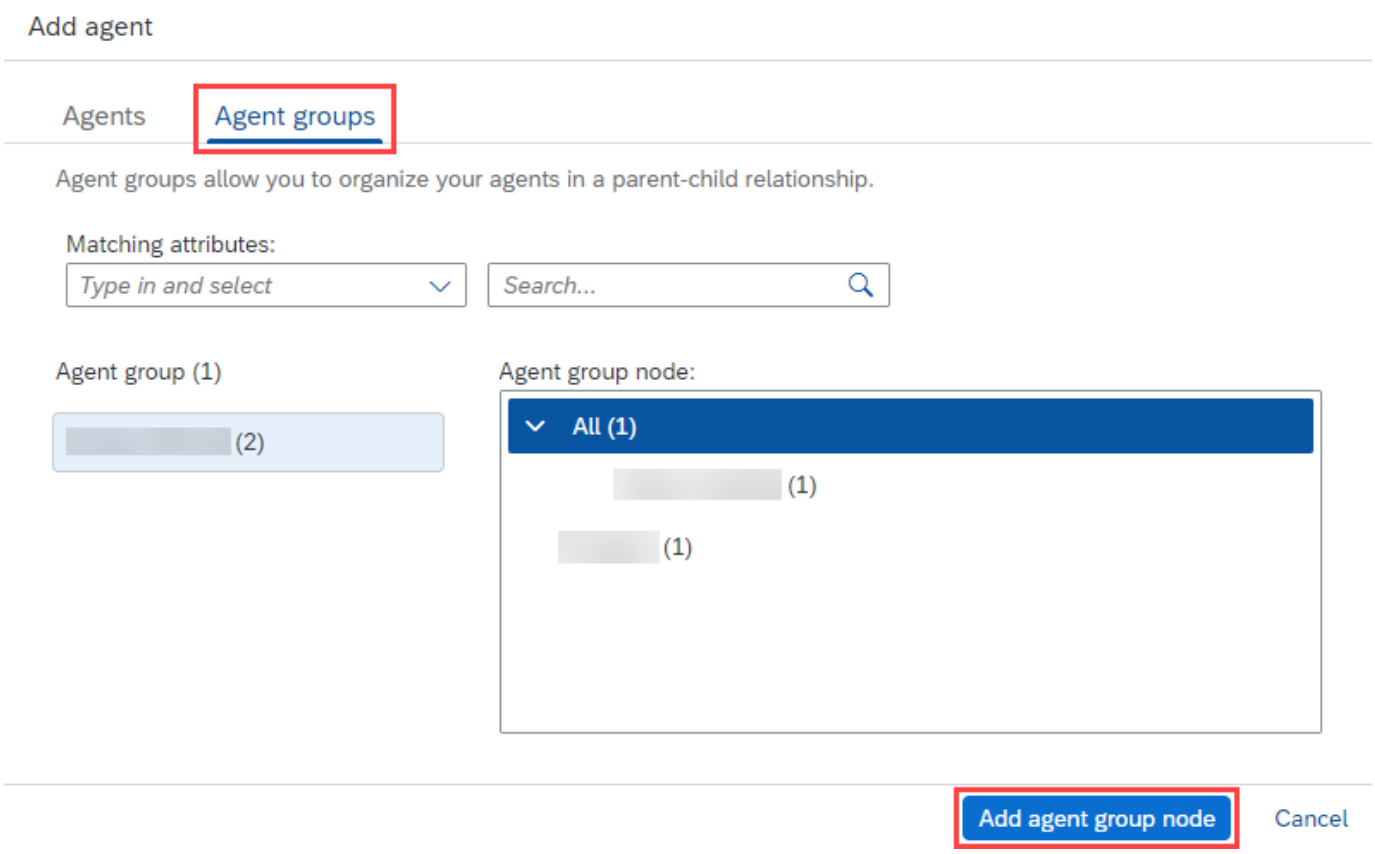


6. Select your agent group under **Agent group**.
7. Select a node in your agent group under **Agent group node**.

ℹ Note

You can select a whole agent group, a node or an individual (machine or login).

8. Click on **Add agent group node**.



Results

Your agent group is successfully added to SAP Build Process Automation.

Mass Agent Registration

The mass agent registration feature allows you to register a large number of agents on a specific agent group.

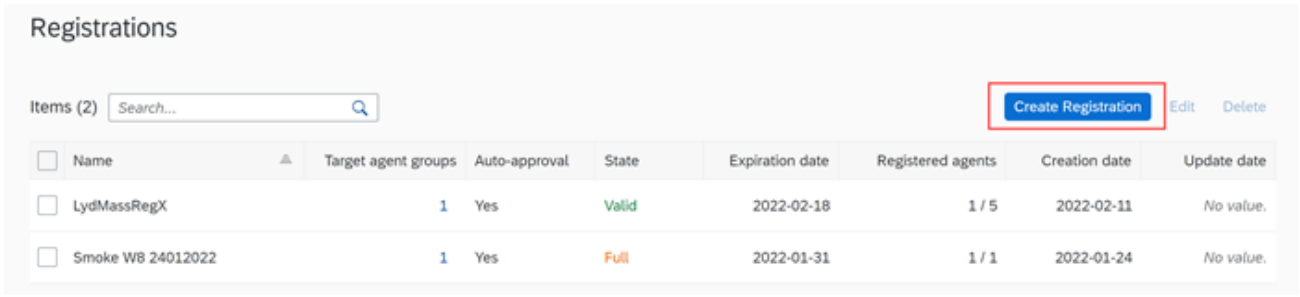
Prerequisite

You have created an agent group as described in Create an Agent Group.

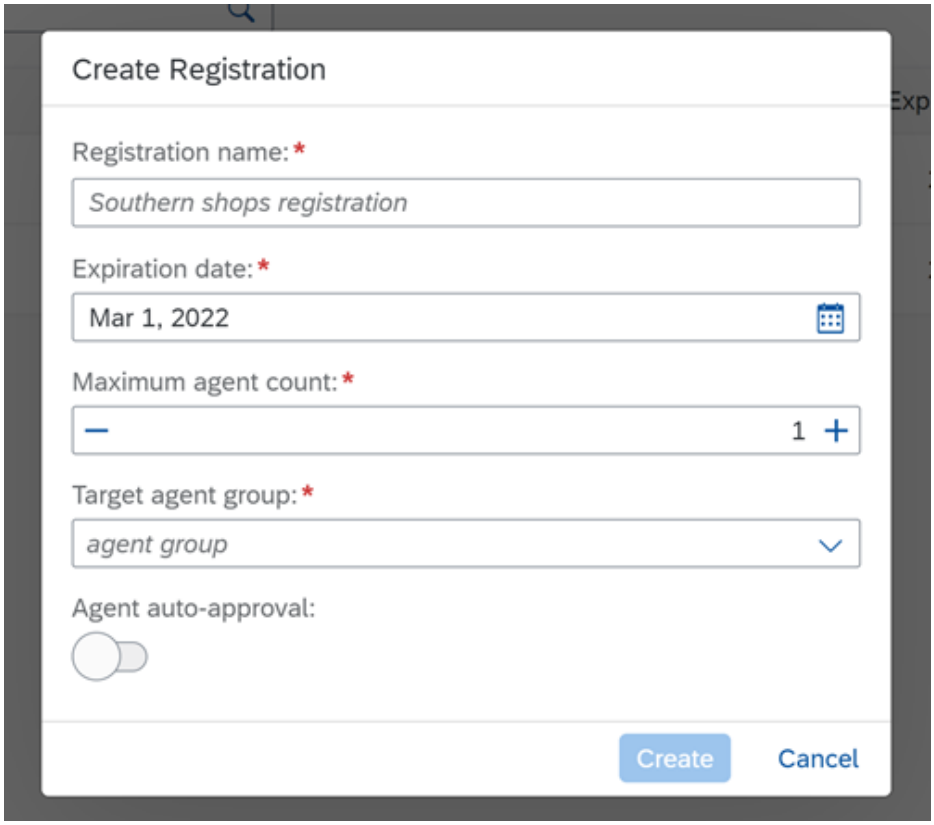
Registration Procedure

From the Lobby side

1. In **Control Tower**, go to the **Mass Registrations** tab.
2. Click **Create registration** on the right-side of the screen.



A Create Registration window pops up.



3. Enter a registration name (max. 256 characters).
4. Select the expiration date for the registration token (max. 1 year).
5. Select the maximum number of agents allowed to register with this token.
6. Select one or multiple target agent groups.

> **Note**
> You can select a maximum of 25 agent groups per registration token.

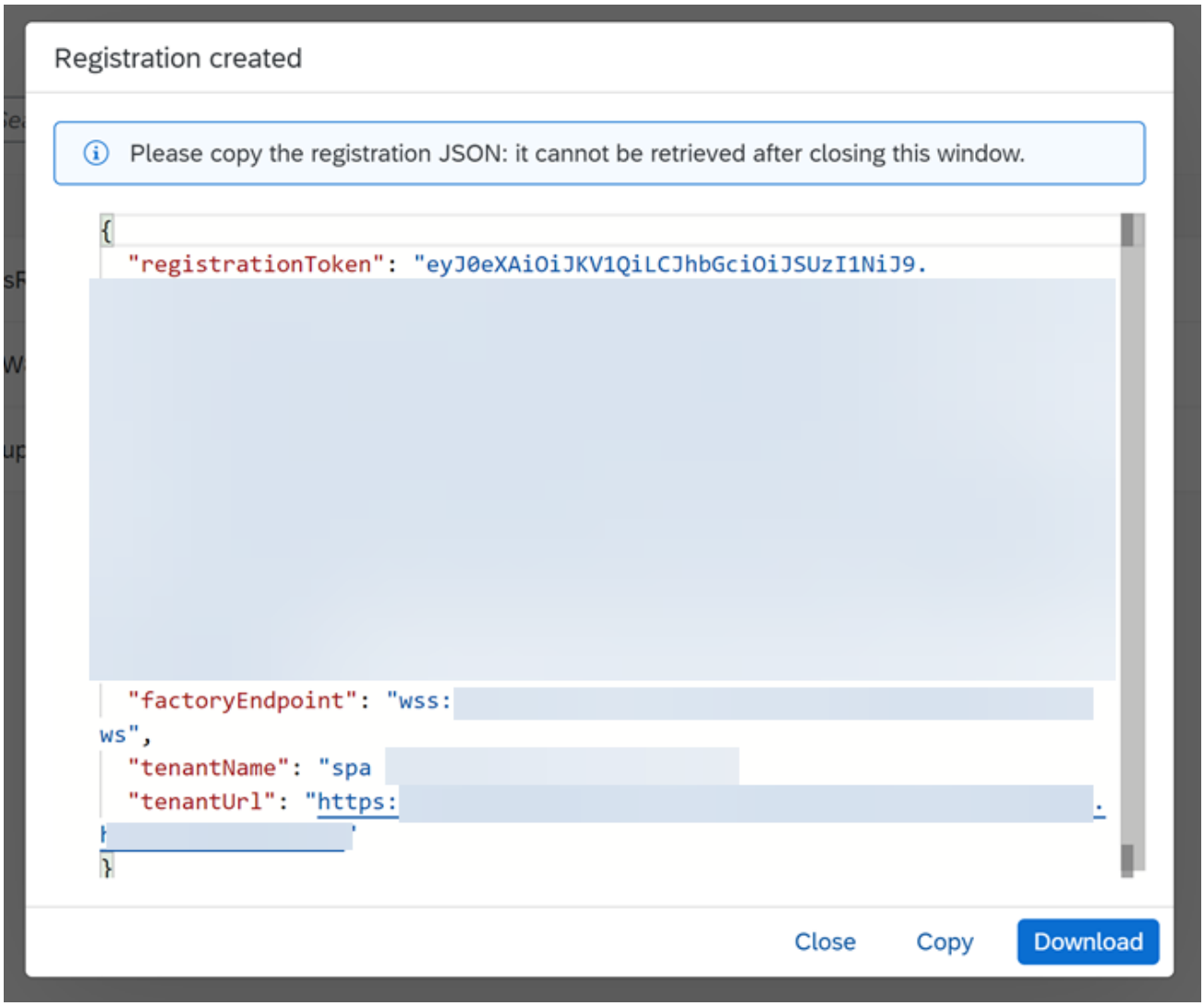
> **Note**
> If you use your agent with a registration composed of multiple target agent groups, the agent is automatically shared with all target agent groups.

7. **Optional**: Select the Agent auto approval to automatically approve the registered agents.

> **Caution**
> This option is less secured than the default behavior.

8. Click Create.

The registration token is created and generated in JSON format.



9. Click **Copy** to save the registration token and paste the token in a `mass_registration.json` file.
10. Click **Download** to download the registration token directly as a .json file.
11. Click **Close** to close the **Registration created** window.

On the Agent side

1. Once you have saved the registration token in a `mass_registration.json` file, the information can be retrieved from the following paths:
   - a local path

     For example, `C:\convenient\local\folder\mass_registration.json`
   - a Windows network path

     For example `\\remote\location\mass_registration.json`
   - a URL

     For example, `https://remoteserver/irpa/mass_registration.json`
2. You can register the agents in two ways :
   - with a command line

     Sample Code

     ```
     SAPDesktopAgent.exe --registration fullpathto/mass_registration.json
     ```
   - with a registry key

Sample Code

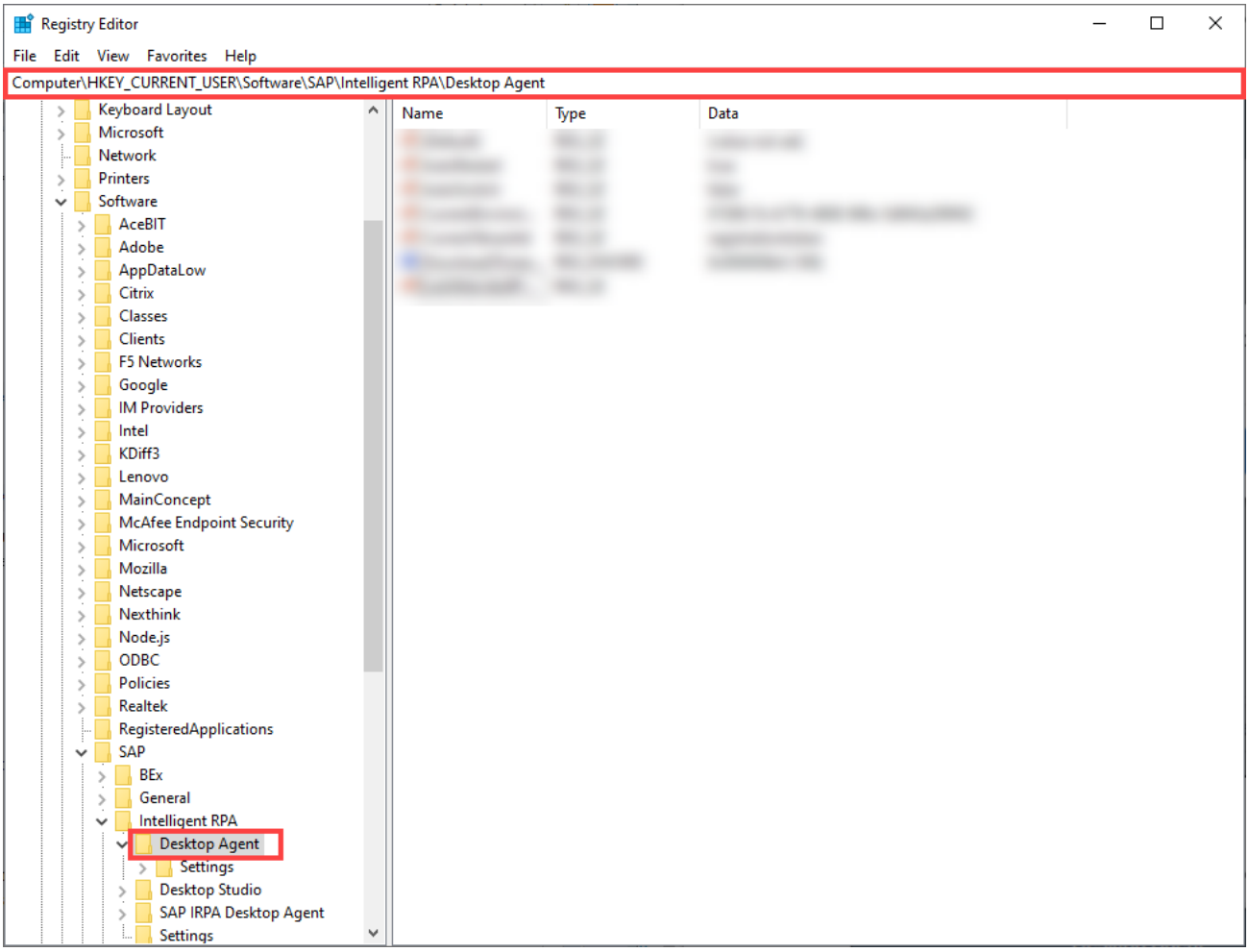
For x32 bits:

```
HKEY_LOCAL_MACHINE\SOFTWARE\SAP\Intelligent RPA\Desktop Agent
```

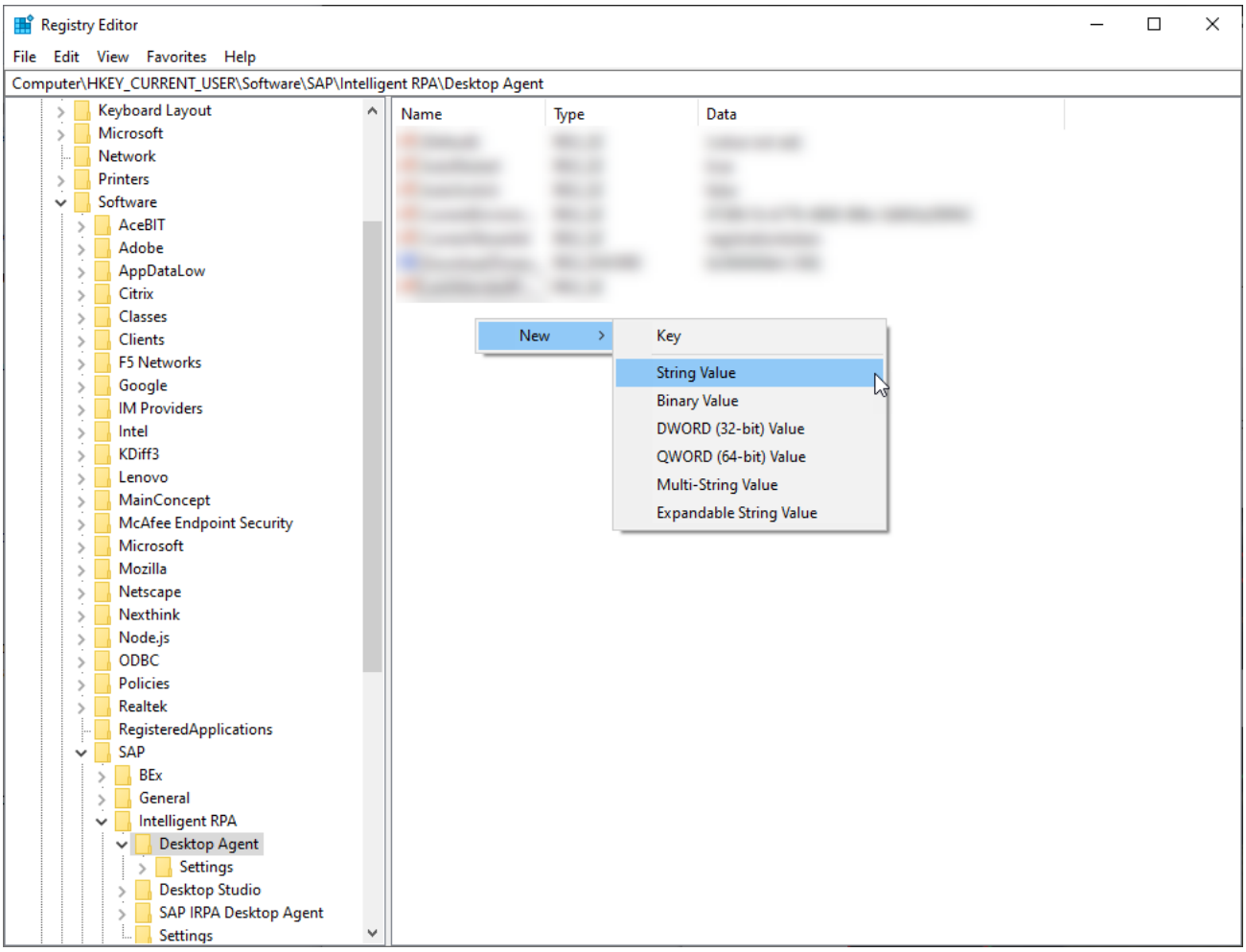
For x64 bits:

```
HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\SAP\Intelligent RPA\Desktop Agent
```

To create the registry key, run the command `regedit.exe` on your computer. Once you have opened the registry editor, copy the registry key in the computer path.

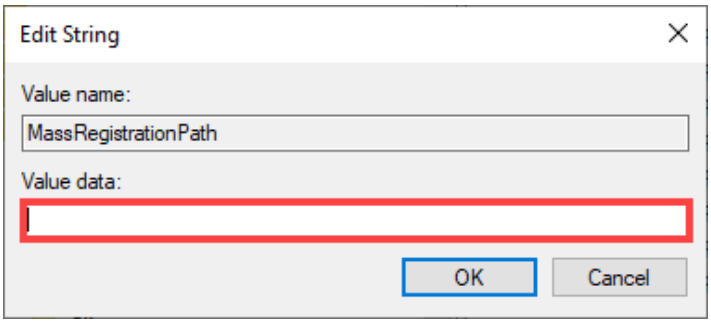


In the Desktop Agent folder, create a new string value `MassRegistrationPath`.



.

Double-click on the string to edit its value data and enter the full path to your `mass_registration.json` file.



Launch the agent.

Result

The agent is registered on the new tenant and awaits to be approved in the Lobby.

Further Details

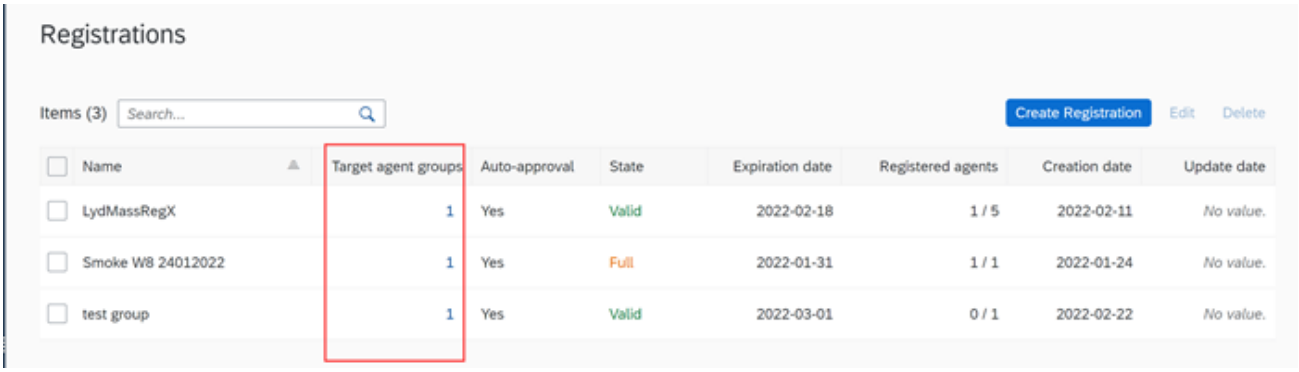
In the **Mass Registrations** tab, you can see a table displaying information on the registration tokens. It indicates the name of the registration token, the target agent group, the state of the registration, the expiration date, the number of registered agents, the creation date, and the update date.

A registration token has the following lifecycle state:

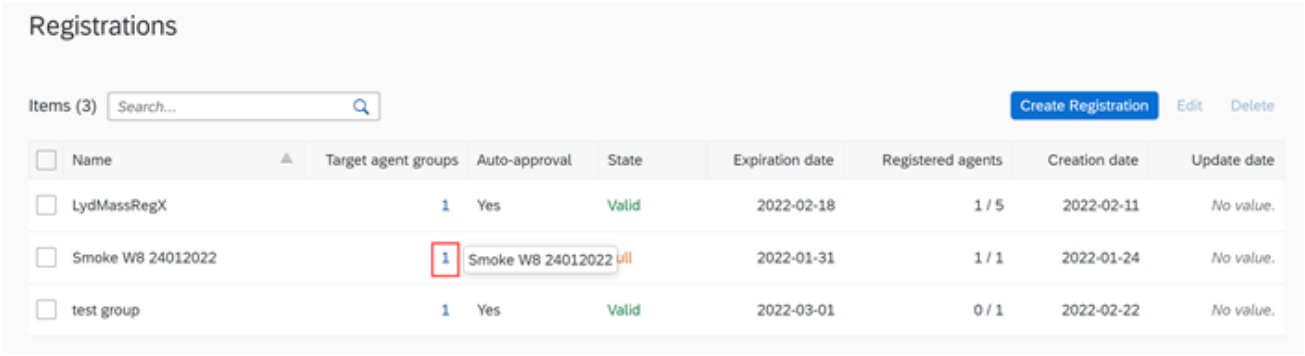
Status	Event
Valid	The registration token is ready for registration.

Status	Event
Invalid	• The target agent group has been deleted. • The registration token no longer has owners.
Expired	• The registration token has reached the expiration date. • The maximum number of registered agents has been reached.

Under **Target agent groups**, you can see the number of agent groups selected for a registration.

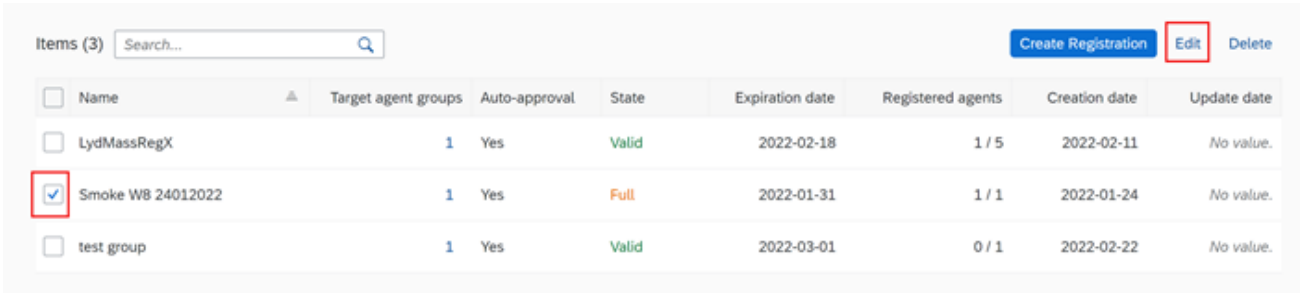


You can click on the number to display the agent groups.



By default, the **Edit** button is disabled.

You can modify the **Registration name** and **Target agent group** by selecting the respective check box beside the specific registration item and clicking the **Edit** button.



By default, the **Delete** button is also disabled. You can delete a single registration using the same procedure as you did to edit.

You can also delete multiple registrations by selecting the check boxes of the corresponding registration items and clicking the **Delete** button.

Change Agent Groups to the **Any** Type

You can change agent groups from type **Login** or **Machine** to type **Any** to make their content easier to manage.

Context

If you are using mass registration with an agent group of type **Login** or **Machine**, you may want to use type **Any** instead, if:

- You previously used mass registration to centrally manage a large list of users. You want the agents of those users to connect to your Process Automation tenant and be shared with an agent group.
- Your previous agent group was of type **Login** or **Machine**.

You can simplify your work by creating an agent group of type **Any** instead. This eliminates the need to manage the list of users manually, but still limits agent authorizations to users with the registration token.

Procedure

1. Create a new agent group of type **Any**.
2. Add this new agent group to all environments that use the previous agent group.
3. Choose **Control Tower** and **Mass Registrations**, edit the registration that has the old group, and replace the old group by the new group.
4. Optional: In the **Control Tower** > **Mass Registrations** view, select all the agents that were in the previous group, and share them with the new group.

Results

- You do not need to change the mass registration token used by all agents.
- New agents that connect will are authorized to the new agent group of type **Any**.
- You do not need to maintain the old agent group anymore.

Related Information

Create an Agent Group
Mass Agent Registration

Agent Attributes

Agent attributes allow you to filter down your agents and make sure that a specific agent executes the job of your choice.

Key Points

- Only an ITAdmin can create, edit, or delete attributes in the global tenant configuration. Anyone with edit access can use these attributes in the application (trigger or agent side).
- Once attributes are defined, a snapshot of the selected current attributes gets replicated on the application.
- If an attribute is modified or deleted in the global tenant configuration, there will be no impact on the attributes that are replicated on the application.
- Quotas are defined on attribute names and values across the tenant.

Subject	Value	Where to Apply
Attribute name / value (max. length)	25 characters	In the global tenant configuration
Attributes (max. count)	20	In the global tenant configuration
Attribute values (max. count)	20	In the global tenant configuration

Subject	Value	Where to Apply
Free attribute values (max. count)	20	In the application (trigger / agent)

- Attributes are case-sensitive.
- It's possible to create an attribute with space or special characters.
- Attribute names must be unique; however, attribute values can be same for different attributes.

Note

Currently attributes are only available for agents.

Agent Inventory

The agent inventory provides detailed information about an agent and a desktop. It provides the following information that helps you to troubleshoot a specific agent quickly:

- Agent Detail
- Agent Service Detail
- Desktop Detail
- MS Office Detail
- Browser Detail
- Antivirus
- Node Version

You can view the agent inventory in the following places:

At Cloud Factory – Agent

When the desktop agent starts, the local inventory is sent to the Cloud Factory. The local inventory from "Cloud Factory – Agent" contains only static information (for example, project list, active project, agent state and so on) that doesn't change during runtime. On the **Settings** page, choose **Agents** > ***Your Agent*** > **(More Actions)** and click **Inventory**. The **Inventory** popup window is displayed. Click **Copy information** to copy the inventory to the clipboard.

At Cloud Factory – Automation Jobs

When you run the job, traces that are recorded from the triggers include the local inventory. On the **Monitor** page, choose **Triggers** > ***Your Trigger*** > **(More Actions)** and click **Activate Traces**. The **Activate Traces** popup window is displayed. In the **Activate Traces** popup window, click **Activate** to activate traces. Now, on the **Monitor** page, choose **Automation Jobs** > ***Your Automation*** > **Traces** and click **Export Traces**. The inventory is downloaded to a log file.

At Desktop Agent

In the Desktop Agent, choose **About** > **Agent Details** > **(More Actions)**, and then select **Copy inventory to clipboard** or **Export inventory to file**. The inventory is copied to the clipboard or downloaded to a JSON file.

Manage Destinations

Destinations are predefined endpoints used with SAP Business Technology Platform (BTP). Destinations allow you to securely store sensitive information used to connect to a system, such as client credentials, keys, or certificates in the SAP BTP.

If you use SAP BTP destinations in SAP Build Process Automation, you don't have to store information separately in each system where you want to use it.

For example, you can use destinations to specify the target URL and authentication information of an API notifier. For more information, see Add an API Notifier.

Prerequisites

To use SAP BTP destinations in SAP Build Process Automation, you must first configure them in the SAP BTP cockpit. For more information, see Using the Destinations Editor in the Cockpit in the SAP BTP documentation.

If you want to use a destination in SAP Build Process Automation, make sure that:

- The destination contains the following property: `sap.processautomation.enabled` and that its value is **true**.
- The destination is added to SAP Build Process Automation.

⚠ Caution

To add destinations, make sure that you have the **IRPAOfficer** user role.

Destinations Section

To find the **Destinations** section, click the **Control Tower** tab and then click **Destinations** on the navigation panel.

The **Destinations** section shows all the destinations that are already added to SAP Build Process Automation.

For each destination, you see:

- The name and description of the destination,
- The type of the destination (only HTTP destinations are currently supported),
- The host address of the destination,
- The authentication type of the destination (only **NoAuthentication**, **BasicAuthentication**, and **OAuth2ClientCredentials** are currently supported),

On this page, you can also see warning messages that appear when:

- The destination has been deleted from the SAP BTP.
- The destination doesn't contain the `sap.processautomation.enabled` property or its value is **false**.

ℹ Note

You can delete the destination from SAP Build Process Automation by clicking the icon under **Actions**.

Buttons

Click the **Open in BTP Cockpit** button to navigate to the **Destinations** panel in the SAP BTP cockpit. To see and click this button, you must have the **IRPAOfficer** role.

Click the **New Destination** button to add a new destination to SAP Build Process Automation.

Add Destinations to SAP Build Process Automation

1. Click the **New Destination** button.
2. The **Add Destination** panel appears. You can see the list of destinations that you can add to SAP Build Process Automation, with their name, type, authentication type, and host address.

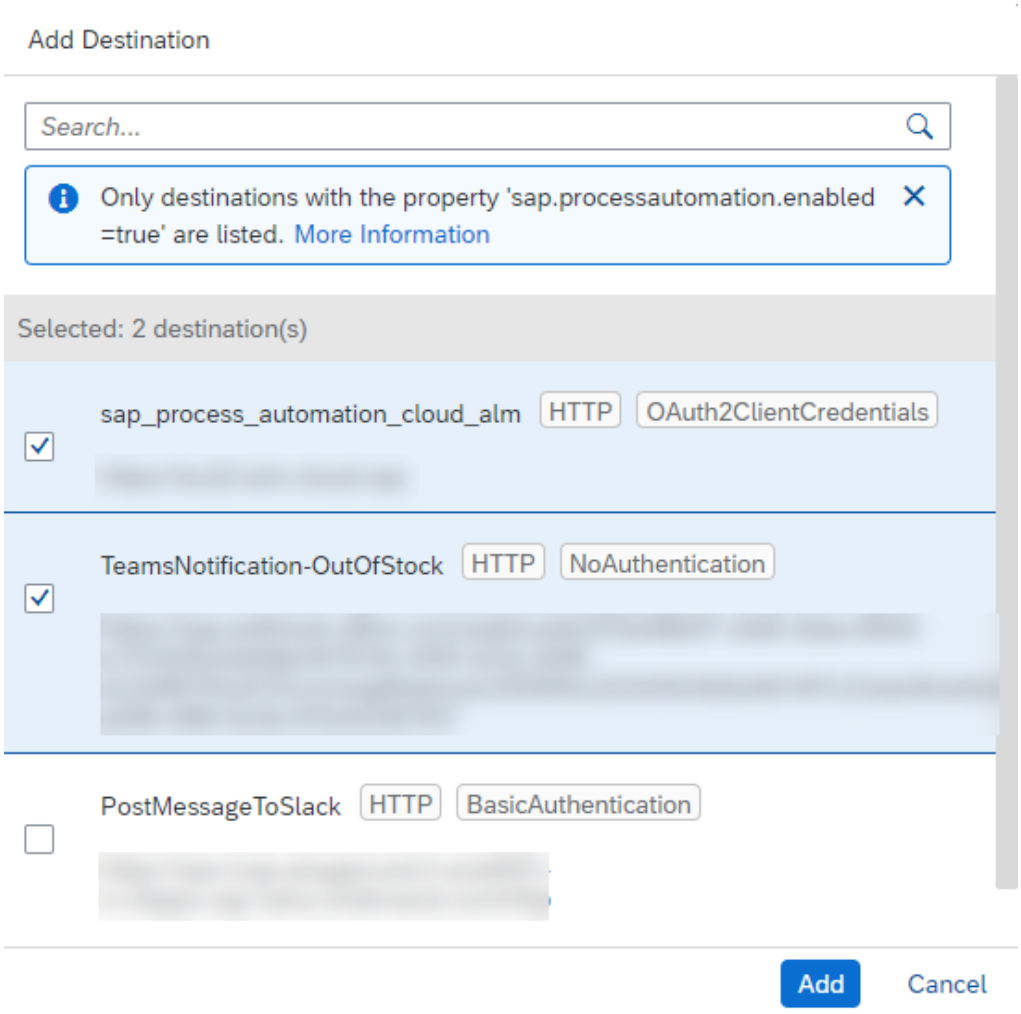
Note

On the Add Destination panel that appears, you can only view and select destinations:

- That have the `sap.processautomation.enabled` property that is set as **true**.
- That aren't yet shared with SAP Build Process Automation.

3. To add a destination to SAP Build Process Automation, check its checkbox and click Add.

   You can add multiple destinations at the same time.



You can notice that the destination or destinations you have added to SAP Build Process Automation now appear in the Destinations section.

API Keys

An API key is a special token that can be used as an authorization to execute API triggers and read access to API triggers and jobs. API keys are used in addition to client credentials for different public endpoints.

Context

You can view the list of API keys for an environment by navigating to Control Tower > Environments > Select an Environment > API Keys, and for multiple environments by going to Control Tower > API Keys. Afterward click Add API Key to create an API key and use it as an authorization to execute API calls. For more information on how to create API keys, see:

- Add API Keys to an Environment.
- Add API Keys to Multiple Environments.

Note

You only need to use API keys for triggers that execute automations.

API Scopes

An API key contains multiple scopes. Each scope is used as an authorization for one or more specific public endpoints. You must add only the required scopes to your API key. In our API reference on SAP Business Accelerator Hub, you can see which scope is required to access each endpoint.

The different types of scopes of API keys are as follows:

Type of API	Description	Scope	Type of Scope
Trigger	To read access to API triggers and jobs	trigger_read	Runtime
Trigger	To execute API triggers	trigger_execute	Runtime
Environment	To add API keys for a specific or multiple environments	environment_read	Runtime
BAM	To download business activity monitoring data	bam_download	Monitoring
Archived Job Data	To download archived monitoring data for jobs	job_data_download	Monitoring

Key Points

- Only an `IRPAOfficer` can create or delete API keys in the **API Keys** section. Anyone with edit access can use these API keys. For more information, see Authorizations.
- Once the API keys are defined, all the API keys are displayed in the **API Keys** section of the **Control Tower** tab.
- If an API key is deleted in the configuration page, there will be an impact on the API keys that are replicated on the single or multiple environments.

SAP Cloud ALM Integration

The SAP Cloud Application Lifecycle Management (ALM) platform allows you to monitor your environment backlog and the statuses of various automation processes all in one place.

By integrating SAP Build Process Automation with SAP Cloud ALM, an administrator can monitor unattended jobs, API triggered jobs, and the automation backlog in the lobby with other cloud services like SAP S/4HANA Cloud. For more information on job monitoring with SAP Cloud ALM, see Job and Automation Monitoring.

Configure SAP Cloud ALM Integration

SAP Build Process Automation uses a dedicated BTP destination to send information to SAP Cloud Application Lifecycle Management (ALM) to help you securely manage the lifecycle of your application. The following steps take you through the process. whether you are configuring a new setup, or already have a CALM configuration that you need to migrate towards a destination.

Configuration

Prerequisite

To enable SAP Build Process Automation to send monitoring data to SAP Cloud ALM, you must first:

- Create a CALM destination.
- Have a user with the "Destination Administrator" authorization in the SAP BTP subaccount in which SAP Build Process Automation is subscribed.

- Obtain the SAP Cloud ALM Service Key to connect to the SAP Cloud ALM system.
- For more information about BTP destinations, see Manage Destinations

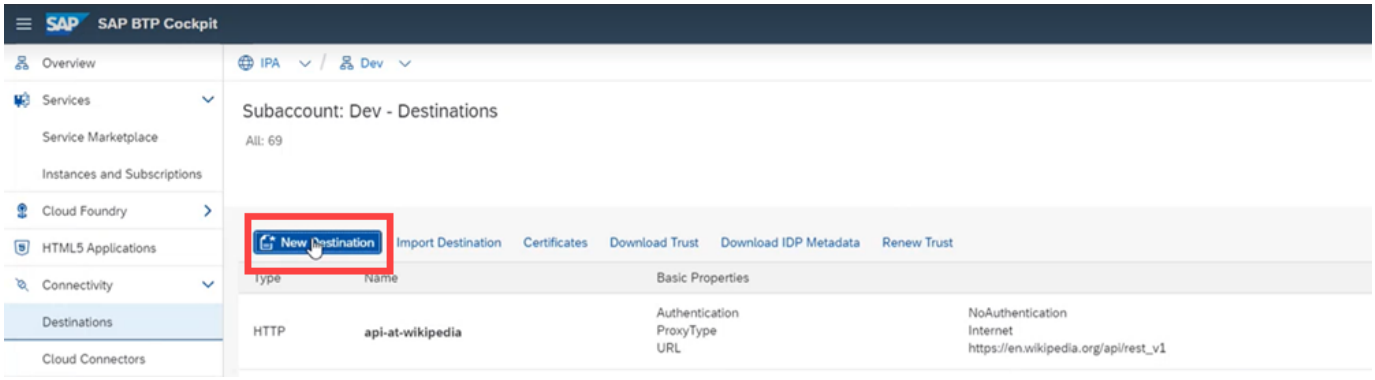
In the BTP Cockpit

1. Open the **Connectivity** drop-down menu in the navigation bar at the top of the screen.

   Click **Destinations**.

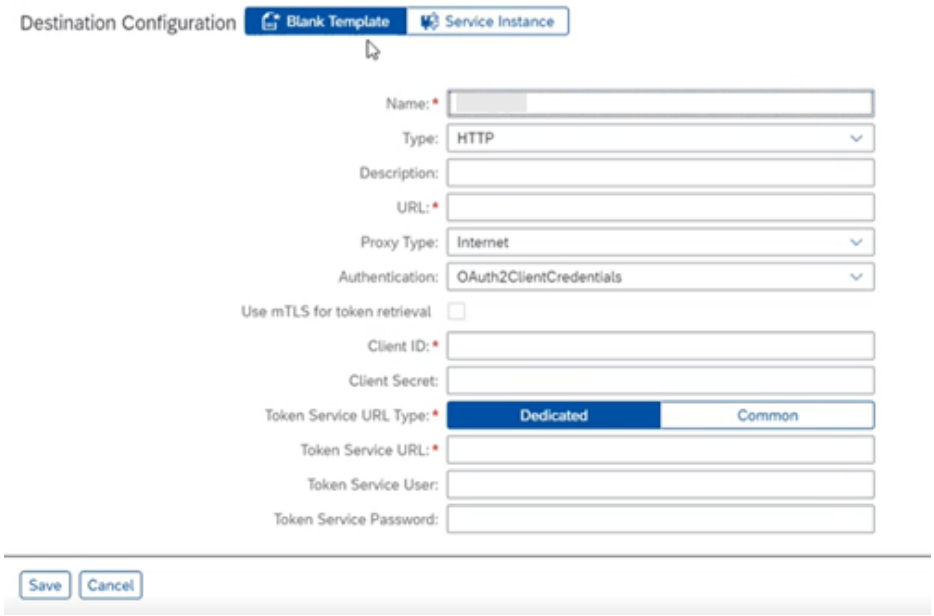
   The **Destinations** page opens.

2. Click **New Destination**.



   A blank template opens at the bottom of the page.

3. In the name field, enter `sap_process_automation_cloud_alm`.



4. Enter the URL given in your service key instance in the cockpit in the **URL** field. For example, `https://>datacenter<.alm.cloud.sap/`.
5. Choose the authentication `OAuth2ClientCredentials`.
6. Enter your CALM Tenant URL, and add `/oauth/token` to the end, in the **Token Service URL** field.
7. Add `clientId` and `clientSecret`, then click **Save**.

Activate Monitoring

1. Click on the drop-down menu next to **Backend Configuration** and select **SAP Cloud ALM**.
2. Enter your SAP Build Process Automation tenant ID as the service name.
3. **Save** and **Register**.
4. Activate the monitoring data collection for the monitoring use-cases as described here: Use SAP Cloud ALM.

Use SAP Cloud ALM

Job Monitoring Configuration

In this section, you select the jobs you want to monitor in SAP Cloud ALM. Dy default, job execution data is not sent to SAP Cloud ALM.

Once you have selected one or more jobs, **Save**.

ℹ Note

A selected job is visible in SAP Cloud ALM as soon as it has been executed.

You can also check **Select new jobs automatically** to automatically send new job execution data to SAP Cloud ALM.

ℹ Note

You can prevent job execution data from being automatically sent to SAP Cloud ALM even if you have checked the **Select new jobs automatically** option. To do so, deselect the jobs of your choice in the table and **Save**.

Health Monitoring Configuration

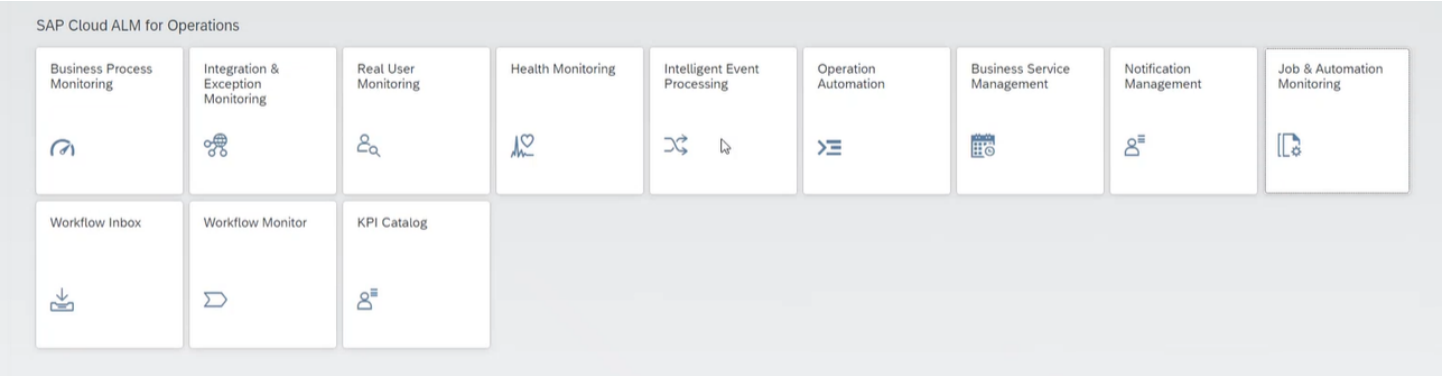
To monitor distribution in a product, you need to select the distribution KPI. By default, distribution related KPIs are not sent to SAP Cloud ALM.

To monitor agent status, select the corresponding checkbox.

Once the distribution KPI is selected, SAP Cloud ALM service receives information each 5 minutes.

Monitor SAP Build Process Automation Notifications in SAP Cloud ALM

When a notifier attached to a job that is monitored in SAP Cloud ALM fails, a notification is sent to the SAP Cloud ALM for Operations.



Click on the **Integration and Exception Monitoring** module where you can monitor the failing notifier. The tenants with exceptions are marked with red.

To display more information like status and error messages, click on the exceptions.

Use My Inbox

Process your tasks from the My Inbox application. You can use My Inbox on your desktop or mobile device.

Depending on your role, you can access My Inbox as follows:

- For citizen developers – from the My Inbox icon (□) in the **Lobby**. For more information, see My Inbox for Citizen Developers.

- For process participants – from the **My Inbox** tile in your SAP Build Work Zone home page. For more information, see My Inbox for Process Participants.

My Inbox for Citizen Developers

As a citizen developer, you can access and process your tasks from the My Inbox application in the **Lobby**.

Prerequisites

You must have completed the initial setup. For more information, see Initial Setup.

Key Features

- View all tasks that are assigned to you.
- Claim tasks.

  > **Note**
  >
  > When you claim a task, you become its processor and its other recipients no longer see it in My Inbox. The status of a claimed task changes from **Ready** to **Reserved**.

- Sort tasks by priority, due date, task title, and the user who created the task.
- Filter your tasks by priority, due date, status **Ready**, and creation date.
- Group tasks by task title, priority, status, and by task type.
- View task-specific details.
- Release tasks for which you are the processor.

  > **Note**
  >
  > When you release a task, you are no longer assigned as a processor of this task and it becomes visible in My Inbox for its other recipients. The status of the task changes from **Reserved** to **Ready**.

- View **Workflow Log**.

  View details about the workflow for a selected task and events, relevant to it chronologically.

  > **Note**
  >
  > Only the user ID is displayed. Additional user details are not available.

- View the priority of the task (**Low**, **Medium**, **High**, **Very High**).
- Execute and complete tasks.

  > **Note**
  >
  > When you select a task from the **List** view, the task details are displayed in the **Details** view. The available actions appear at the bottom of the screen.

- Share a task via e-mail.

Rate Limits

To ensure optimal operation of the service, workflow capability execution is subject to resource limits, for example, regarding the number of requests per second. If the limit is exceeded, My Inbox displays the following error message: **" Your action could not be performed because the usage limits are reached. Please retry later or contact your help desk for assistance."**. The client should then reduce the number of calls.

My Inbox for Process Participants

As a process participant, you can process tasks, create and manage substitution rules from the My Inbox application on your desktop or mobile device.

Prerequisites

- Your administrator must have completed the initial setup. For more information, see Initial Setup.
- Your administrator must have configured the My Inbox application on SAP Build Work Zone. For more information, see Configure SAP Build Work Zone Content, Assign Permissions, and Create and Access Site.

Key Features

- View all tasks that are assigned to you.
- Claim tasks.

  > **Note**
  >
  > When you claim a task, you become its processor and its other recipients no longer see it in My Inbox. The status of a claimed task changes from **Ready** to **Reserved**.

- Sort tasks by priority, due date, task title, and the user who created the task.
- Filter your tasks by priority, due date, status **Ready**, and creation date.
- Group tasks by task title, priority, status, and by task type.
- View task-specific details.
- Release tasks for which you are the processor.

  > **Note**
  >
  > When you release a task, you are no longer assigned as a processor of this task and it becomes visible in My Inbox for its other recipients. The status of the task changes from **Reserved** to **Ready**.

- View **Workflow Log**.

  View details about the workflow for a selected task and events, relevant to it chronologically.

  > **Note**
  >
  > Only the user ID is displayed. Additional user details are not available.

- View the priority of the task (**Low**, **Medium**, **High**, **Very High**).
- Execute and complete tasks.

  > **Note**
  >
  > When you select a task from the **List** view, the task details are displayed in the **Details** view. The available actions appear at the bottom of the screen.

- Share a task via e-mail.
- Add and manage your substitutions in the **Substitution Management** view. For more information, see Create and Manage Substitution Rules.

Rate Limits

To ensure optimal operation of the service, workflow capability execution is subject to resource limits, for example, regarding the number of requests per second. If the limit is exceeded, My Inbox displays the following error message: `" Your action could not be`

performed because the usage limits are reached. Please retry later or contact your help desk for assistance.”. The client should then reduce the number of calls.

Create and Manage Substitution Rules

You can use My Inbox to create substitution rules to manage your tasks in your absence. The substitution rules can be created for planned and unplanned absences.

Prerequisites

You have the **Manage My Substitutes** option, displayed in the user action menu.

Planned Substitution

Planned substitution is usually targeted for a scenario where you know the start date and the end date for your absence. Your substitute will then see your tasks directly displayed in their inbox for the period defined by you.

To create a substitution rule for a planned absence, proceed as follows:

1. In My Inbox, choose the user action menu and select **Manage My Substitutes**.
2. Make sure you have selected the **Planned** tab and choose **Add New Substitute** in the footer of the screen.
3. In the **Add Substitute** dialog, enter the user ID of the substitute you want to nominate.

   To retrieve the expected user ID format, select a task in My Inbox, open the user action menu and select **Support Information**. The value of the `CreatedBy` property is in the expected user ID format, for example **username@sap.com**.
4. Choose a period for the substitution and choose **Save**.

   > **Note**
   > If you do not select a substitution period and save the rule, it is planned from the day of the creation of the rule for an undefined period.

   The planned substitution rules are automatically activated on the start date you have selected, and are automatically deactivated on the end date, respectively. On the start date of the substitution rule, your substitute will receive the tasks you have defined in the substitution rule automatically. On the end date of the substitution rule, your substitute will stop receiving the tasks you have defined in the substitution rule automatically.

   Tasks, which have been already claimed by the substitute prior to the end date will stay in the substitute's inbox.

As a result, the successful creation of a planned nominee is confirmed and you can see the entry in the **Planned** substitution tab.

After you have created the new substitution rule, make sure that the user ID of your substitute is spelled correctly in the list of planned substitutions.

You will see **Active**, or **Inactive** status for each substitution rule you have created.

Unplanned Substitution

To create a substitution rule for an unplanned absence, proceed as follows:

1. In My Inbox, choose the user icon and select **Manage My Substitutes**.
2. Select the **Unplanned** tab.
3. Choose **Add New Substitute** in the footer of the screen.
4. In the **Add Substitute** dialog, enter the user ID of the substitute you want to nominate.

   To retrieve the expected user ID format, select a task in My Inbox, open the user action menu and select **Support Information**. The value of the `CreatedBy` property is in the expected user ID format, for example **username@sap.com**.

5. Choose **Save**.

   Tasks, which have been already claimed by the substitute prior to the end date will stay in the substitute's inbox.

   As a result, the successful creation of an unplanned nominee is confirmed and you can see the entry in the **Unplanned** substitution tab.

   After you have created the new substitution rule, make sure that the user ID of your substitute is spelled correctly in the list of unplanned substitutions.

   > **Note**
   > In this case, your substitute will need to accept the substitution in order to see your tasks in their inbox.

Take Over Tasks as an Unplanned Substitute

You can take over or stop receiving tasks from users, who have nominated you as their unplanned substitute.

To do so, proceed as follows:

1. In My Inbox, choose the user icon and select **Substitute For**.

   You will see the list of users who have nominated you as their unplanned substitute. By default, you will not be receiving their tasks. If you want to take over, you will need to activate the substitution for that particular user.

2. Click the switch button to activate the substitution rule for the selected user.
3. After you activate or deactivate the user, choose **Done**.

If you deactivate the substitution rule, you will stop receiving tasks from the selected user.

Use Preferences

Change your preferences to customize the display of your user interface.

Context

The SAP Build Lobby user interface can be changed from the default theme to the high contrast theme, letting you control how your user interface looks and behaves. You can choose a theme from a list, which remains in effect until you change it. Set up your preferences for theme and language according to your needs.

You can change the following settings:

Preferences	Description
Theme	The theme that you want to use for the application UI.
Language	The language in which you want the UI texts to appear.

Procedure

1. To view and edit your preferences, in the upper right part of the lobby screen, click your user name and choose **Preferences**.
2. Choose **Theme** to select the theme as per your preferences.
3. Choose **Language** to change the language.
4. Choose **Apply** to change the settings.
5. **Optional:** To restore the original settings, choose **Reset**.