BrandMaker

Job Manager and Marketing Data Hub

Administration Manual

Version 7.4

EN

23 March 2023

100600000001

Table of Contents

Copyright

Head office: An der Raumfabrik, Amalienbadstraße 41a, 76227 Karlsruhe, Deutschland,
www.brandmaker.com

Your feedback is important to us!

We would be grateful to be notified of any errors you may discover. Just send us an e-mail at documentation@brandmaker.com.

1 Useful information for getting started

This provides you with tips and information that will help you to efficiently carry out tasks related to the Job Manager or Data Hub. The manual is aimed towards specialist administrators and provides a full overview of the configuration and functions of the Job Manager and Data Hub. It is worth mentioning that the default module name for the Marketing Data Hub is *Data Hub*.

Jobs, processes, and data objects

Jobs and processes are managed in the *Job Manager* module. Jobs are combined with a classic workflow, processes with a BPMN workflow. Jobs and processes represent standardized workflows in your company.
In the *Data Hub* module, users manage data objects, which are always combined with a classic workflow. In data objects, a standardized workflow captures data that can be used, for example, as a data source for other modules.

Classic workflows and tasks

One or more classic *workflows* are assigned to each job and each data object. A classic workflow consists of several *workflow steps*. The administrator assigns a user group to each workflow step when creating the classic workflow. The user selects a responsible person from the user group when forwarding the item to the next workflow step. In addition, *tasks* can be added to each workflow step so that typical activities are created automatically when the job is created.

BPMN workflows

BPMN stands for **B**usiness **P**rocess **M**odel and **N**otation and is used for the graphical representation of business processes. A process is always combined with a BPMN workflow, and the workflow is represented in BPMN notation.

Task Templates

Tasks that are used frequently can be saved as task templates and used again. To use a task template with a workflow, the number of workflow steps must match the number of task steps created in the task template.

Custom objects and structures

You can use custom objects and structures to create boilerplate texts, for example, that you can use to edit a document in the *Brand Template Builder* module. In addition, the values for a dropdown list can be loaded from a custom structure.

Variables

Different types of variables are used to collect and output data and information. The variables are placed on a datasheet using drag and drop. There are two different types of variables:

- Variables created automatically that allow you to use the basic functions, for example, for displaying the creator or the current workflow step.
- Variables that can be created individually (text fields, selection fields, or an asset selector, for example) can be created and configured as required.

Categories

To organize the types, you can create categories in *Administration* > *Data Structures & Workflows* and assign the existing types to one or more categories. When the user creates a new job or data object, they choose the required type from the overview, which is sorted by category.

Jobs/processes and sub-jobs/processes

A *job* or process groups together all the information required to carry out a task (for example, the creation of a new campaign). Additional required tasks can be mapped as *sub-jobs* or sub-processes. A sub-job or sub-process can use a different workflow that is independent of the job. Sub-jobs or sub-processes allow you to structure workflows and make dependencies between different workflows recognizable.

> **Note**
>
> It is possible to create a synchronization between jobs or processes of a certain type and corresponding planning elements in Marketing Planner. This synchronization is set up by BrandMaker.
>
> If you require any further assistance, please speak to your BrandMaker contact person.

Products and sub data objects

A *data object* groups together all the information related to a data object. If the data object consists of multiple parts, they can be represented through *sub data objects*. A sub data object can use a different workflow that is independent of the data object. Sub-data objects allow you to copy the data object structure precisely.

User

The access to and visibility of tabs and the field functions (variables) placed on them can be restricted when you configure the type for each workflow step. For this, users are divided up according to *Assignee/Processor*, *Creator*, *Other participants* and *Anonymous*. The user role to which you belong determines the datasheet tabs and variables that you can view or edit.

User	Description
Creator	You are the creator (owner) of a job, process, or data object if you have created the job or data object. The creator can be changed at a later stage. **Note:** You cannot retroactively adjust the creator for a process type (BPMN workflow).
Assignee/Possessor	You are the (current) editor/assignee of a job, process, or data object if you are responsible for the current workflow step. You can also be a member of an assigned group, where each person has the same rights, but you choose the workflow step to be edited, enter the data and forward it to the next step.
Participant	You are a participant if you are invited to the job, process, or data object as a participant. As a participant, you are not responsible for the current workflow step, but you can follow the progress of the job, process, or data object. Participants can be added to a discussion.
Anonymous	Anonymous users are any users who are not the creator, assignee/processor, or participant, but who can still access a job, process, or data object. In general, anonymous users only have read access to variables.

1.1 Required actions after an update

You can skip this section for a fresh installation.

1.1.1 Migrate types

Due to some radical changes in the types, they have to be migrated after an update of any previous system. After a successful migration, users can open and create jobs, processes, and data objects again.

Important information

Problem: After an update, Job Manager cannot be used immediately, and also new types cannot be created under > *Administration* > *Datasheet Engine* > *Types*.

Solution: After an update, first run > *Administration* > *Datasheet Engine* > *Maintenance* > *Migrate Types* once. This service will update every existing type in the background and publish it in the new format.

Only users with the right MIGRATE_TYPES can initiate a migration. For this purpose, the user needs the permission for the Migrate Types tab, that is a tailored role for the administration with the MANAGE_TYPE_CONFIGURATION permission. If you require any further assistance, please get in touch with your contact person or support.

1.2 Display, technical, and unique name

Each type and each custom variable have three different names:

- *Display name **
- *Technical Name*
- *Unique Name*

Property	Display Name	Technical Name	Unique Name
Usage	The display name is the name that is used to show the type or variable on the interface and that is used for the inheritance. You can create the display name in various languages.	The technical name is used to operate objects via interfaces of external systems (for example, via REST). The technical name is also used for grouping variables.	Must be used when creating formulas in Formula type variables.
Input	You enter the display name when you create a type or variable.	The technical name and unique name are derived and created from the displayed name when you create a type or custom variable.	

Property	Display Name	Technical Name	Unique Name
Change	The display name can be changed and edited in any way.	The technical name can be changed.	The technical name cannot be changed.
Restrictions	There are no restrictions for the display name. However, long names may not be displayed in full under certain circumstances.	The characters [a-z], [0-9], and [_] are used for the name.	
Derivation	—	• If the display name begins with a number, the name "type_" is prefixed to the technical name. • Space characters are replaced with an underscore "_". • Umlauts and special characters are removed. • Uppercase letters are replaced by lowercase letters. • Restricted to a maximum of 255 characters.	• Like technical name; additionally restricted to a maximum of 24 characters.
Uniqueness	The display name can be used multiple times for each type.	The technical name and unique name must be unique for each type. If multiple variables of the same type with the same display name are entered, consecutive numbering is added during the derivation (for example, price_1, price_2).	

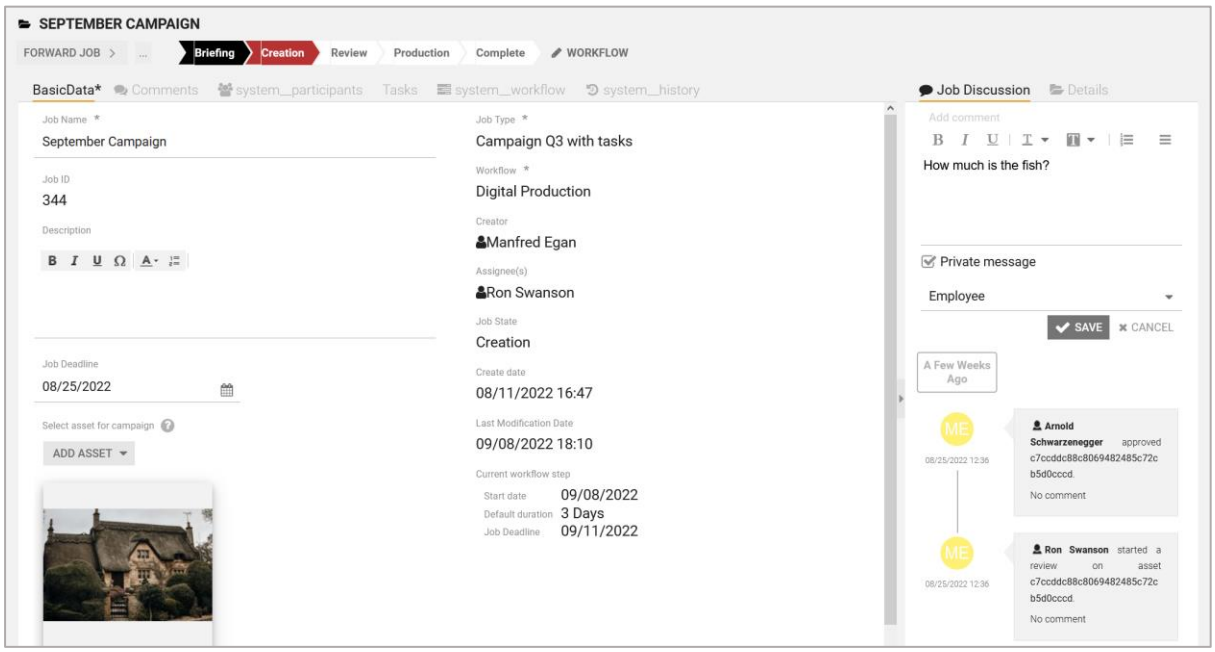
1.3 Datasheets

All the information about a job or data object is collected and mapped on a *datasheet*. You can group together the different variables clearly on various tabs and place them easily using drag and drop.

Among others, the following information is displayed on the datasheet:

- The users who are involved/participating in the job or data object
- The current state (the current workflow step)
- The creator and the current assignee
- The predefined project workflow for the job or data object, including any tasks that have been defined.
- The relevant information for the job or data object

Note: The appearance of the datasheet can be defined to suit the purposes of individual customers. Among others, the appearance is determined by the number of tabs created, their names, and the field functions that are used on them. The *Basic Data* and *Comments* tab are created automatically; they can be renamed but *cannot* be removed. You can hide the *Comments* tab.

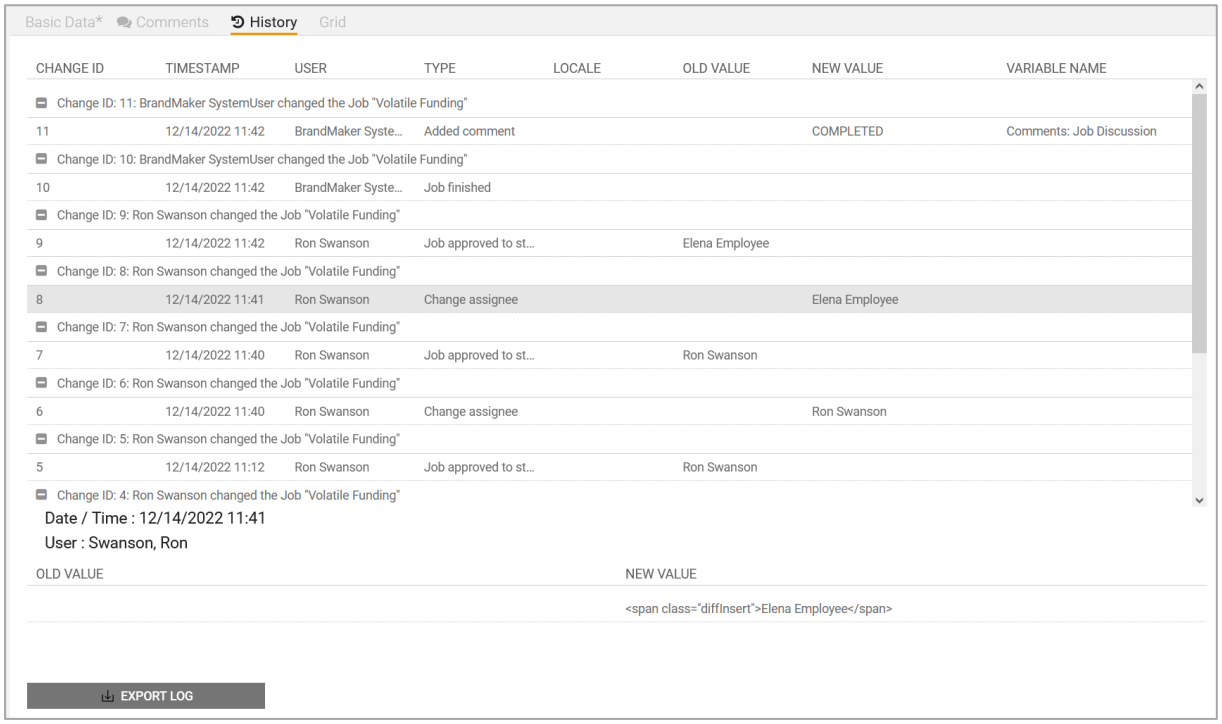


1.3.1 Change history

You can use the change history to monitor when an object was processed or edited. It also records the amount of time it took. Navigate to the *History* tab in the currently open datasheet to open its change history. The following information is displayed in a table overview:

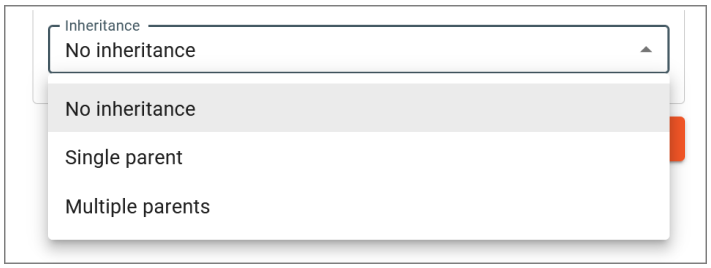
- *Change ID:* Consecutive numbering of the changes (only with Job Manager)
- *Timestamp*: The time at which the change was made
- *User*: The name of the user who added or edited a comment or discussion
- *Type*: The type of editing or change (for example, the addition of a new comment)
- *Locale*: This column is not relevant in Job Manager.
- *Old value*: The original value of the edited variable
- *New value*: The new value of the edited variable
- *Variable name*: The name of the variable or discussion that was edited. Select the line to display the new and old value in the lower area of the dialog window.

Click *Export log* to export the change history as an XLSX file.



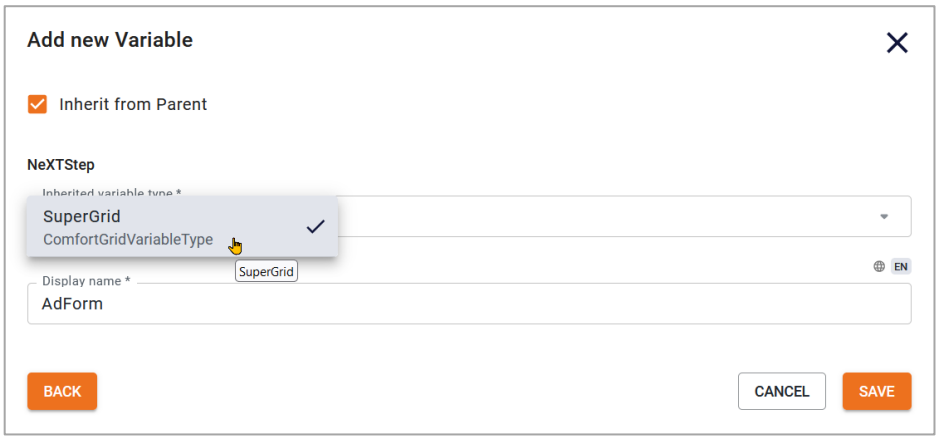
1.4 Inheritance for sub jobs and sub data objects

Sub-jobs and sub data objects can inherit values from a parent job or data object. If the values in the parent job or data object change (for example, a deadline), the value is also updated in the sub-job or sub data object. In the subtype, you can configure the following options using the *Inheritance* selection box:



Note: Note that the inheritance can be created only when you create a new sub-job type or sub data object type. The *Inheritance* field cannot be changed at a later stage!

- *No inheritance*: In this case, variables of the subtype do not inherit a value from a parent job or data object.
- *Single parent*: The variables can inherit; however, only one single type is permitted as the parent type. Define the parent types from which the inheritance is carried out for the sub-job or sub data object. In this case, you are given the option of specifying the parent type variable that must be inherited directly when creating the variables (see screenshot). If you use a large amount of inherited variables, the consistent application of this method provides you with a significant performance boost.



- *Multiple parents*: Select this option if the parent types from which a subtype inherits cannot be clearly defined. For example, the *Translation* sub-job can be permitted for both the parent job types *Brochure*, *Manual*, and *Flyer*. Activate the option *Inherit from parent* in the subtype for the inheriting variable. In this case, the link is established via the display name, which must be identical for the variable in the parent type and subtype.
  Users can click on the inheriting variable on the datasheet to break the inheritance.
- In this case, the last inherited value remains in the sub-job or sub data object, even if the value is changed in the parent job or data object. By clicking , users can restore the inheritance.

1.5 Localization

Note

It is worth noting that you will only be able to use this feature in the Data Hub.

You use the *Localization* function to adapt data objects to the conditions of regional markets in the Data Hub. A localized data object contains a separate datasheet that is adjusted to local conditions and that uses an ID for each locale. You edit each datasheet in a separate workflow and can select different workflow types for each localized datasheet. This is significant, for instance, if there are different legal requirements in the territories and a legal check or certification is required.

You can use different workflow types to define different rights and visabilities for the required fields for each locale.

To enter values that are identical for a data object in each locale just once, flag the variable as language-neutral. Examples here include international data object names or technical data in standardized measurement systems (for instance, measurements in the metric system, weight specifications in kilograms).

You can also create sub data objects for localized data objects that inherit their values. Note that inheritance is only possible between localized parent data objects and localized sub data objects: The sub data object variant for *Spain* inherits its values from the parent data object variant *Spain*.

1.5.1 Configuring localization

Follow the process below to configure the localization. Follow all the required steps for setting up the Data Hub module (see Required work steps, page 18):

1. Create the required locales (see Locales, page 15).
2. Define the required workflows. For more information, see the configuration manual.
3. Create a data object type with the *Enable Localization* checkbox activated (see Types, chapter 2.3).

   Note: Note that the *Localization* field can only be edited when you create a type.
4. Assign all workflows required for the localized data object to the type (for Type configuration, see Classic workflow for job or data object types, page 32).

Create the variables that are used for this data object type. For variables that are identical for each locale, activate the *Shared value* checkbox (for variable descriptions, see chapter 4).

5. Publish the changes (see Publishing changes on page 51).

1.5.2 Locales

You can use the locale to describe the territories that are relevant to you using a small amount of data. You can enter the following values:

- *Name*: The unique ID of the locale.
  Warning! Entering other characters as digits leads to errors! Use consecutive numbering that continues the numbering from the last created territory.
- *Display name **: Include the names that you want to use in the various languages for the locale. Use the following structure: *~{language code}nameserver* multiple names one after the other. Note that you only have to enter the translations for the interface languages of your BrandMaker system.
  - *Language code*: language code according to ISO 639-1 in uppercase (for example, EN for English, DE for German).
  - *Name*: name of the locale that is displayed in the selection list in the datasheet of the localized data object.
  - Example: The *Spain* entry in the languages English, German, and French: ~{EN}Spain~{DE}Spanien~{FR}Espagne
- Attributes:

Name	Function	Edit
uniqueName	Unique name, which must be unique within the locale	Mandatory field
scriptCode	The character set used for print characters (for example, Cyrillic or Simplified Chinese)	—
numbers	The character set used for numbers	—
languageCode	Language code according to ISO 639-1 in lowercase	Mandatory field
image	Load an image to represent the territory visually. If the field is empty, the flag for the selected country code is displayed if it is stored by default.	Optional
default	Standard locale setting: true = the locale is the standard locale. false = the locale is not the standard locale. **Warning!** Only one locale must be flagged as the default locale at all times.	—
currency	Currency entry for the locale	—
countryCode	Enter the country code of the territory	Mandatory field
collationParameter, collation	Parameter for setting the character sorting in the relevant language	—

Name	Function	Edit
calendar	Calendar form, such as the Gregorian calendar	—

1.5.2.1 Configuring or Changing the Locale

1. Choose > *Administration* > *Data Structures & Workflows* > *Custom Objects & Structures* > *Custom Objects.*
2. In the *Select custom structure* selection list, select the structure `L10N_Locale(L10N_Local)`.
3. Choose *Select*.
4. To create a new locale: In the *Create new custom object* field, enter the name of the new locale. Choose *Create*.
   OR
   To change an existing locale: In the *Choose available custom object* selection list, select an existing locale.
5. The following fields are always mandatory:

- *Name*
  Warning! Use consecutive numbering that continues the numbering from the last created territory.
- *Display name **
- Attribute *uniqueName*
- Attribute *language code*
- Attribute *country code*

6. Optional: edit the following fields:

- Attribute *default* for the locale that is the standard locale.
- Optional: Attribute *image*

7. Click the *Save* button at the end of the attribute list.

You have configured the locale. Repeat the process if additional locales are required.

1.5.2.2 Deleting a Locale

Warning! Data loss! Do not delete any locales that are being used in the *Brand Template Builder* module.

1. Choose *> Administration > Data Structures & Workflows > Custom Objects & Structures > Custom Objects.*
2. In the *Select custom structure* selection list, select the structure `L10N_Locale(L10N_Locale)`.
3. Choose *Select* button.
4. In the *Select available custom object* selection list, select an existing locale.
5. Click the *Delete* button below the attributes.

You have deleted the locale.

1.6 Synchronization

Basically, it is possible to set up a job type so that planning elements are created, or data is synchronized in the Marketing Planner when the job is edited. For this purpose, the datasheet of the job type must be configured accordingly so that the position in the element tree as well as basic data is mapped correctly. In addition, for the combination of job type and workflow it is defined in which workflow steps the data is updated in the Marketing Planner when saving the job. You can also define whether the planning element is deleted if the job is canceled or deleted.

Note

This synchronization is set up by BrandMaker. Please don't hesitate to contact your service manager for further information.

2 Required work steps

You must perform a range of work steps for the configuration. It is worth mentioning that the process differs based on whether you are using a conventional workflow created with the previous function or a BPMN workflow.

2.1 Jobs and Data Objects

Prerequisite

- You have created users and a user group.
- You have created a workflow.

Note the detailed descriptions for users, user groups, and workflows in the configuration manual.

Step by step

The following work steps are required or can be carried out as options for the configuration of Job Manager and Data Hub:

1. Create a new type and assign a module (see chapter 2.3).
2. Link the types with one or more workflows (see chapter 2.3.4).
3. Design datasheets for a type (see chapter 2.5).
4. Create and define variables (see chapter 2.6).
5. Optional: Define conditions for displaying a variable and access rights for variables for each workflow step (see chapter 2.6.1).
6. Optional: Define variables for the e-mail notification (see chapter 2.8).
7. Optional: Create a category (see 2.9.1).
8. Optional: Configure the format of the unique object numbers (see chapter 2.9.3).
9. Optional: Define the default type for new jobs and data objects (see chapter 2.9.3.1).
10. Optional: Configure the settings for the Only Briefing type and theme navigation (see chapter 2.9.4).
11. Optional: Create and manage templates for the Task Manager variable (see chapter 2.10).
12. Publish changes (see chapter 2.11).

2.2 Processes

Prerequisite

- You have created users and a user group.
- You have created the custom structures required for your decisions.

Note the detailed descriptions for users, user groups, and workflows in the configuration manual.

Step by step

1. Create new types and assign a module (see chapter 2.3.2.1).
2. Design the datasheets for a type (see Designing the datasheet layout in chapter 2.5).
3. Create and define variables (see Creating and managing variables in chapter 2.6).
4. *Optional*: Define conditions for displaying a variable and access rights for variables for each workflow step (see Rights and visibilities of variables in chapter 2.7).
5. Create a BPMN workflow for the type (see chapter 3).
6. *Optional*: Define variables for the e-mail notification (see E-mail notification in chapter 2.8).
7. *Optional*: Create categories (see Type categories in chapter 2.9.1).
8. *Optional*: Configure the format of the unique object numbers (see Object number configuration in chapter 2.9.3.2).
9. *Optional*: Define the default type for new jobs and data objects (see Default types in chapter 2.9.3.1).
10. *Optional*: Configure the settings for the *Only Briefing* type and theme navigation (see Settings in chapter 2.9.4).
11. *Optional*: Create and manage templates for the *Task Manager* variable (see Managing task templates in chapter 2.10).
12. Validate the created type (see chapter 2.11.2).
13. Publish changes (see Publish changes in chapter 2.11).

2.3 Types

The type is the central element of the datasheet Engine. The type determines which basic data is created for a job, a process, or a data object. You create types under > *Administration* > *Datasheet Engine* > *Types*.

To reach that page, it is required that the right MANAGE_TYPES is assigned to your role.

If you open a type for editing or create a new one, the Properties page opens first. Instead of tabs, a ☰ dropdown menu is used to switch to the other pages with their respective editors.

- Workflow Modeler, see chapter 3.1.1, Workflow Modeler.
- E-Mail Management, see chapter 3.4, E-Mail templates.
- Datasheet-Layout, see chapter 2.5, Managing a datasheet layout.

2.3.1 Properties

The following table shows the properties of the job and data object types. A process has the same properties as a job except for the inheritance function. Note that you set the *Name* (and thus the displayed name), *Type*, *Inheritance,* and for Data Objects, the *Localization* property when you create them. Type and inheritance cannot be changed after their creation.

Name	Description
(Displayed) name	Define the name that is visible to users. You can create the displayed name in different language versions. See Display, technical, and unique name, chapter 1.2.
Type	The type determines which basic data is created for a job, a process, or a data object.
Inheritance	**Note:** that inheritance can only be defined while you create a new type. The *Inheritance* property cannot be changed afterward! The function can exclusively be used for jobs and data objects. Configure the inheritance: • *No inheritance*: No datasheet variable inherits values from a parent datasheet. • *Single parent datasheet*: Only one job type is available as a possible parent datasheet. • *Multiple parents datasheet*: Multiple job types are available as possible parent datasheets. Refer also to the Chapter 1.4.
Localization	**Note:** Can only be activated when creating a new data object type. Select the checkbox at *Enable localization* if you want to create localized variants of a data object. See Localization, chapter 1.5.
Unique name, technical name	**Note:** Only accessible in the editing dialog box. See Display, technical, and unique name, chapter 1.2.
Description	**Note:** Only accessible in the editing dialog box. Enter additional information about the type that is displayed when it is created.
Categories	**Note:** Only accessible in the editing dialog box. Select the categories to which the type is assigned. When the item is being created, the types are displayed in categories.

Name	Description
Type can only be selected by	**Note:** Only accessible in the editing dialog box. Specify which organizational unit, user group or VDB group can select the types. **Note:** Note that this setting does not affect the visibility of jobs based on this type.
Activating an access control.	**Note:** Only accessible in the editing dialog box. If you activate this checkbox, only users who belong to the same organizational unit or an organizational unit in the hierarchy below it, such as the creator and processor of the job or data object can view the jobs derived from this type.
Parent datasheets	**Note**: Only visible in the editing dialog box if *Inheritance* = *single parent datasheet* was specified when created. Select a type.
Jobs/processes and data object type only exist as sub-job/process/data object	Activate the checkbox if the job, process, or data object may only be used as a sub-job, sub-process, or sub-data object.
Manually adding sub-jobs/sub-data objects	**Note:** Effective for Job and Data object types. Only accessible in the editing dialog box. Activate the checkbox if the user is allowed to add sub-jobs or sub-data objects to the job or data object manually.
Permitted sub-job types/data object types	**Note:** Only accessible in the editing dialog box. Only for Job and Data object types. The *Manual addition of sub-jobs* checkbox must be selected. Specify which types can be added as sub-jobs/data objects during creation.
Selected sub-job/data object type	**Note:** Only accessible in the editing dialog box. Only for Job and Data object types. The *Manual addition of sub-jobs* checkbox must be selected. Specify which sub-job or sub-data object is added by default during creation.

Name	Description
Automatically added sub-jobs/data objects	**Note:** Only for job and data object types. Only accessible in the editing dialog box. Specify which sub--jobs or data objects are added automatically when the item is created. Click the plus sign to configure a sub-job or sub-data object. Define the following settings: • *Job/Data object type*: Specify the sub-object type. • *Workflow*: Defines the workflow for the sub-object. • *Processor*: Defines which user is assigned to the sub-object. Depending on the settings of the selected type, further selection fields may open up • *Default job name*: Set a default job name. • *Optional sub-job/data object*: If you activate the checkbox, the user can choose whether the sub-object is created. If the checkbox is deactivated, the sub-object will always be created automatically.
Use for synchronization	**Note:** Only accessible in the editing dialog box for process types. Once synchronization is enabled, the function cannot be disabled. Activate the switch if the type is used for synchronization of a process with a planning element. If you have further questions about synchronization, please contact your BrandMaker contact person.

Note

The "Job Deadline" provided in the parent job is inherited to the sub-job as the default value.

2.3.2 Managing Types

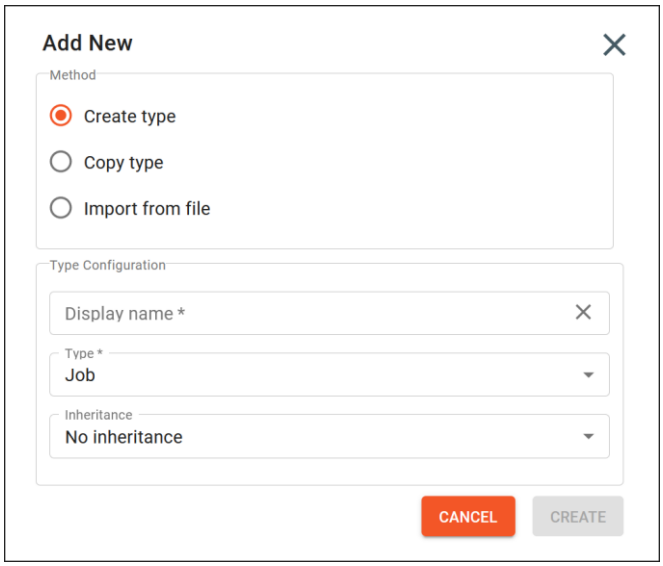
The chapters below describe how to manage types. This includes the following tasks:

- Creating a type, see Chapter 2.3.2.1
- Editing types, see Chapter 2.3.2.2
- Copying a type, see Chapter 2.3.2.3
- Deleting types, see Chapter 0

2.3.2.1 Creating a type

1. Choose *> Administration > Datasheet Engine > Types*.
2. Click *the button Create*.

The following creation wizard is displayed, where *Create type* is already preselected:



3. Enter a name in the *Display name* field.
4. Select which type you want to create: Job, Data Object or Process.
5. For job and data-object types: Specify the inheritance. Note that you cannot change this setting at a later time.
6. For data object types: Select the checkbox if you want to create localized variants of a data object. Note that you cannot change this setting at a later time.
7. Click *Create*.

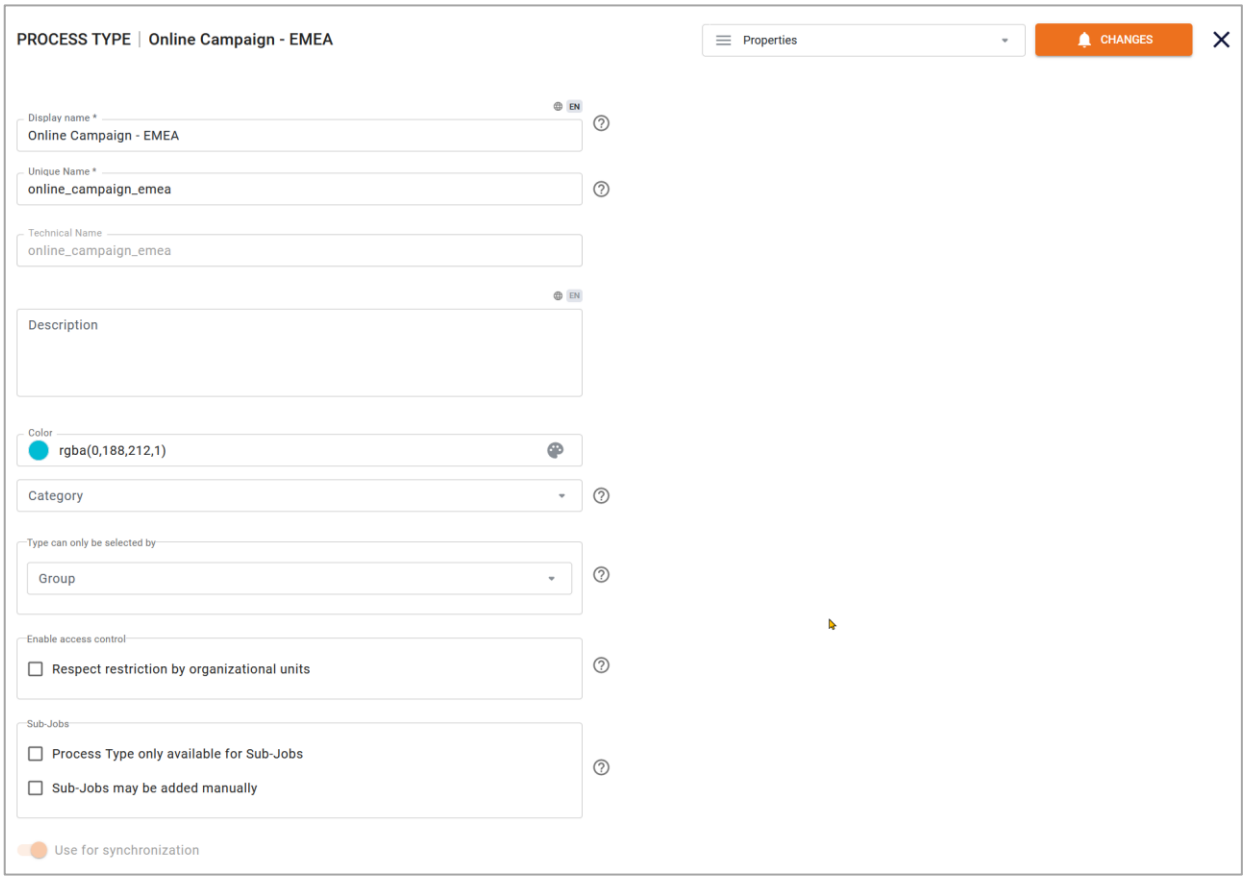
   When you create a new type or open an existing type, the *Properties* page always opens first.
8. Edit or add to the type properties. Please refer to Chapter 2.3.1.
9. Use the *Changes* button to display the sidebar with the change history and any error messages.
10. If no errors or warnings are displayed, click the *Publish* button.

You have created the type and published the changes.

2.3.2.2 Editing types

1. Choose > *Administration* > *Datasheet Engine* > Types.
2. In the table, click the -icon at the end of the row for the type whose properties you want to edit.

The tab *Properties* of the job type is displayed:

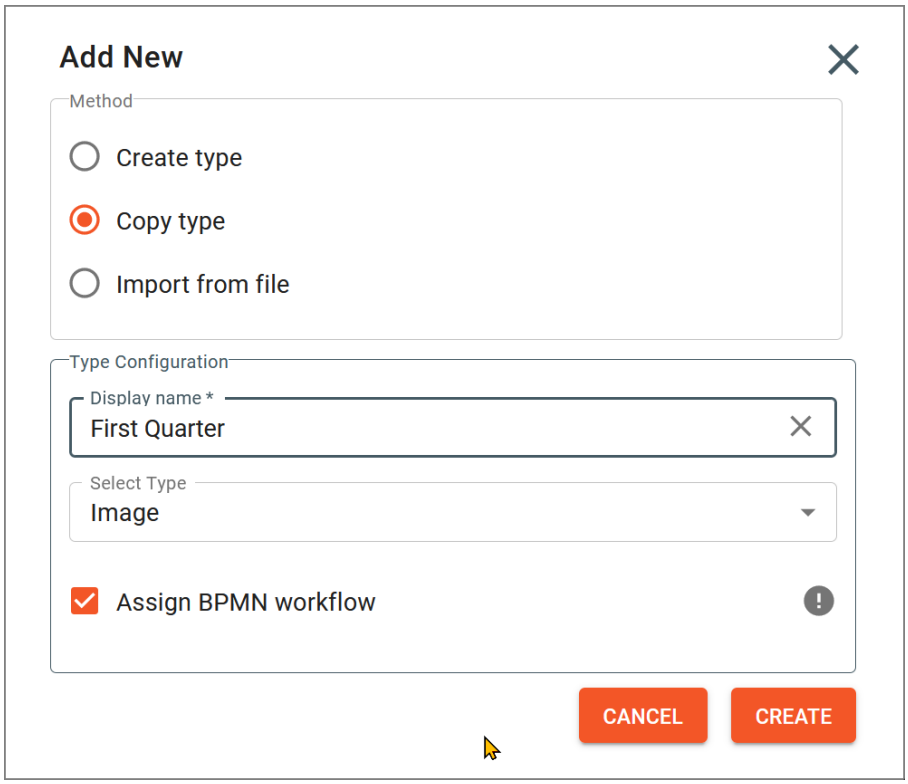


3. Edit or add to the type's properties. Please refer to Chapter 2.3.1. Changes are saved directly, but not published, i.e., they are not yet effective.
4. Use the *Changes* button to display the sidebar with the change history and any error messages.
5. Click *Publish* at the bottom of the *Changes* panel-.

You have edited the type and published the changes.

2.3.2.3 Copying a type

If you copy a job type with classic workflow, the *option Assign BPMN-Workflow* is available. If you have activated the checkbox, the classic workflow will be discarded, and you can then create a new workflow for this type in the BPMN Workflow Modeler.



1. Click > *Administration* > *Datasheet Engine* > *Types*.
2. Click *Create*.

The *Add New* dialog opens.

3. Select the method *Copy type.*
4. Specify a name (displayed name) for the copy.
5. In the *Select Type* menu, search and select the type that will serve as a template for the copy.
6. Click *Create*.

You have created a copy of an existing type.

2.3.2.4 Localize display name

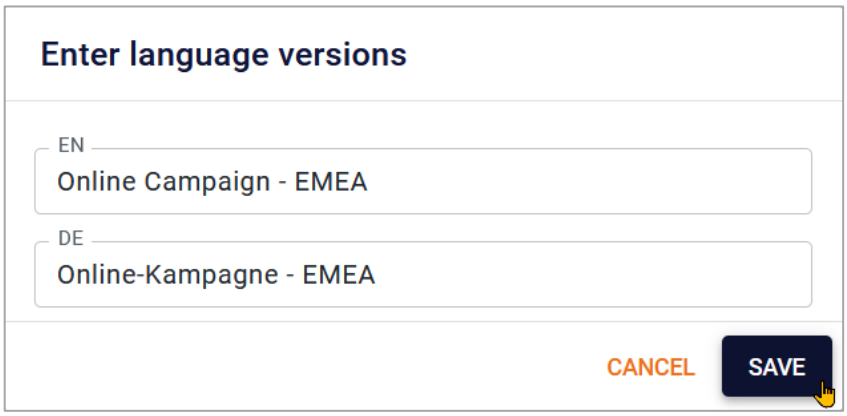
To localize the displayed name of a type, open it by clicking on the -icon in the type list. You create and edit the translations on the *Properties* page. This can also be done at the time of type creation. Click in the field *Display name*.

1. Pay attention to the Display Name field.

   There is a -Icon above the text field.

2. Click this icon to store the job name in multiple languages.

   The dialog *Enter language versions* opens.



3. Fill in the fields for all languages you need a translation for.
4. Click *Save* to apply the changes to the language versions and return to the Properties page.
5. Switch to the tab *Changes* and click the button *Publish*.
6. Optional: Click on an ISO language code next to the globe (e.g., EN) to switch between the available localizations and make that language version the default *Display name*.
7. Use the *Changes* button to display the sidebar with the change history and any error messages.
8. Click *Publish* at the bottom of the *Changes* panel-

You have created the desired language versions. The language versions are now available in the respective interface languages.

2.3.2.5 Deleting types

1. Click > *Administration* > *Datasheet Engine* > *Types*.
2. In the table, at the end of the row for the type you want to delete, click the icon .

The Delete dialog will open.

3. Click *Cancel* to abort the deletion. Press *Confirm* to delete the type permanently.

You have deleted the type.

> **Note**
>
> Deletion of types is only possible if no job or process of this type is currently under way.

2.3.3 Adding a sub-job or sub data object

Sub-jobs allow you to structure workflows and make dependencies between different workflows visible. You can use sub-jobs to subdivide a data object or data object data record. You can specify:

- whether a user can add sub-jobs or sub data objects manually when creating a new job or data object,
- whether certain sub-jobs or sub data objects are added automatically when a new job or data object is created.

Prerequisites:

- You have defined allowed sub-job types or sub data object types.
- You have already created job types or data object types and linked them to a workflow.

In the area *Automatically generated Sub-Jobs* choose *Add Sub-Job* (*Automatically generated Sub-Products* > *Add Sub-Product*) to specify the subtype that is added automatically when the item is created. Enter the required settings in the dialog box that opens.

When configuring a sub-process, the administrator can set up that the current editor of the parent process is automatically entered as creator of the sub-process.

> **Note**
>
> The required settings are identical for adding a sub-job and adding a sub data object.
> The “Job deadline” set in the parent job is inherited to the sub-job as the default value

2.3.4 Export and import of process types

Export and import existing process types to simplify and speed configuration of new systems.

Note that email templates are not taken over during export, and after an import you have to adjust BPMN workflows with a “Send Task” accordingly.

2.3.4.1 Export of a process type

The export is done as a JSON file. The export includes the following features, rights, and parameters:

- Type properties
- Users, comfort tables as well as variables in connection with the sync to Marketing Planner
- BPMN workflows
- Access permissions
- Input/output parameters for service tasks
- Step names of the simplified view

2.3.4.2 Import of a process type

Once you have set up and configured a process with BPMN workflow on your test system, it should also be transferred to live operation at some point after thorough testing. The import is done in the creation wizard:

1. Choose *> Administration > Datasheet Engine > Types*.
2. Click *the button Create*.
3. Select the *Import from file* option.
4. To import, drag the desired JSON file from the file system to the marked upload area. The file name is listed below the area if the file extension is correct, and the process can be imported.
5. Or you can click in the marked area and a file selection dialog opens with a preset file type filter *.json*. Browse the file system for the file you want to import.
6. Enter a name for the process in the *Displayed name* field.
7. The *Create* button executes the import and stores the process in your system under the specified name.

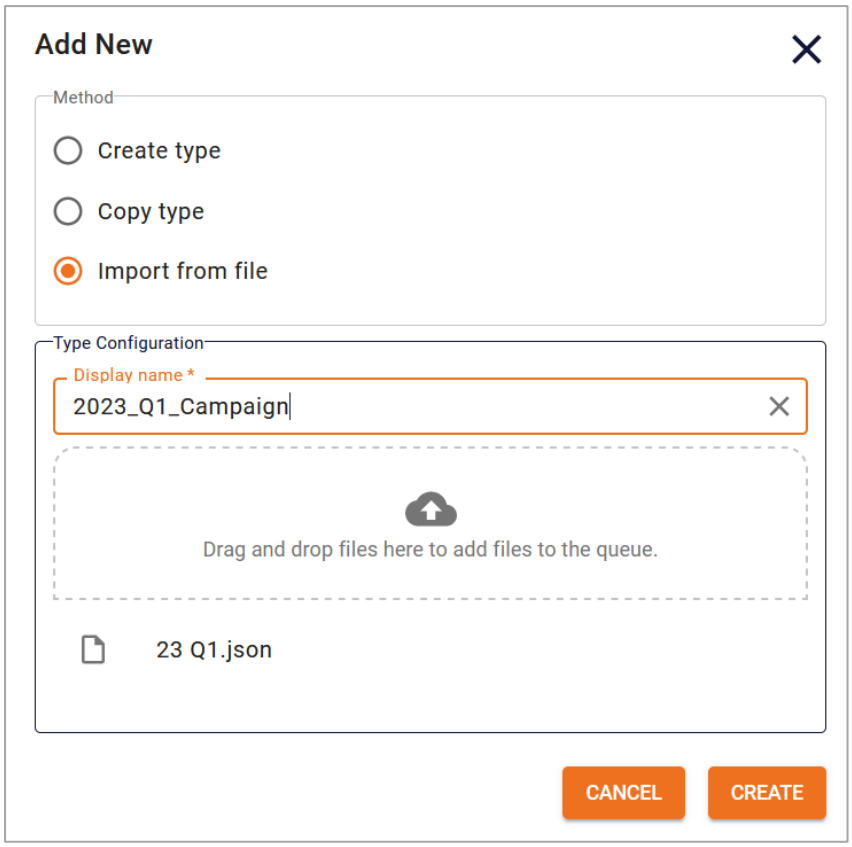
You have successfully imported a process type that was previously exported on another system.

Attention

Imports are performed without the data of a user-defined data structure. If the data structure on the import system is incomplete, you will see error messages on the relevant gateways. You cannot use the process yet. You need to recreate missing data structures and relink them in the imported BPMN workflow.

Note

Imports of processes with synchronization settings currently work only to a limited extent. This is because the settings for dimensions and Planner layers are ID-based, but these IDs differ from one system to another and cannot be re-mapped. Therefore, a manual re-creation is necessary.


Add New
Method
Create type
Copy type
Import from file
Type Configuration
Display name *
2023_Q1_Campaign
Drag and drop files here to add files to the queue.
23 Q1.json
CANCEL
CREATE

2.4 Assigning a workflow

To use a type, you must link the type to a workflow. This ensures that all the steps required for processing a datasheet are carried out.

Note:

- Job and data object types are linked to a classic workflow.
- For a process, you create a BPMN workflow.

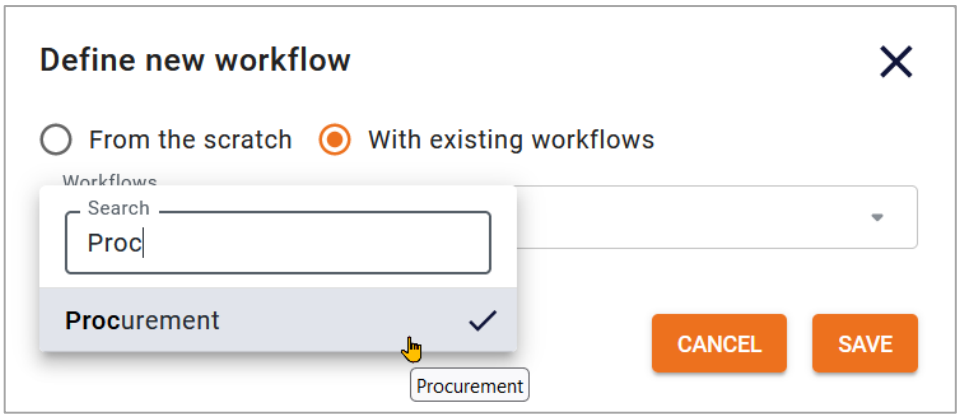
To create a classic workflow, you have two options:

- Under > *Administration* > *Data Structures & Workflows* > *Workflows,* you create classic workflows that other modules can also use. In this case, you have to assign one or more workflows to the job or data object type. To accomplish this, refer to the first of the following sections.
- Under > *Administration* > *Datasheet Engine* > *Types,* you create a classic workflow that is used exclusively by this job or data object type. Please refer to the second of the following sections.

2.4.1 Assigning classic workflows

You can assign workflows directly when creating the type or edit the type's properties later.

1. Select > *Administration* > *Datasheet Engine* > *Types*.
2. Click the *Create* button.
3. After creating a type, switch to the *Workflows* page via the ☰ select menu.
4. Click *Create New* to add a workflow.
5. Select *From the scratch* to define the workflow completely from the ground up or *With existing workflows*, to reuse an existing workflow.



6. When the workflow configuration is complete, click *Save*.
7. Use the *Changes* button to display the sidebar with the change history and any error messages.
8. Click the *Publish* button.

After you have assigned or customized a classic workflow, you may also immediately close the *Workflow* tab (✕) and are not required to take the approach via > *Changes* > *Publish*. In doing so, the changes are accumulated for the time being and published at a later point. How this can be done is explained in detail in section 2.11.

2.4.2 Classic workflow for job or data object types

1. Navigate to > *Administration* > *Datasheet Engine* > *Types*.
2. Click on the pencil icon in the list for a previously created job or data object type.

The Properties page is displayed.

3. Choose *Workflows* via the ☰-select menu.
4. Click *Create new*.

The Define *New Workflow* dialog box opens.

5. If you want to create a new classic workflow:
    a. Click *From the scratch*.
    b. Enter a name and description for the workflow.
6. If you intend to use an existing classic workflow as a starting point:
    a. Click *With existing workflows*.
    b. Select an existing workflow from the drop-down list.
7. Click *Save*.

You have created the workflow.

8. Edit the workflow:

- Click  to configure access rights for each step.
- Click the plus icon to add a subsequent workflow step.
- Double-click a workflow step or select > ⋮ > *Edit* to edit the settings of the workflow step.
- Select > ⋮ > *Copy* to copy the workflow step.
- Select > ⋮ > *Delete* to delete the workflow step.

You can adjust the order of the workflow steps by drag and drop at any time. You can move a workflow step by dragging and dropping the ⁞⁞ handle.

10. Click ✕ to close the *Workflows* page.

You have created a new classic workflow that is configured to suit this type.

2.4.3 Create a BPMN workflow for a process

You can also copy a job type with a classic workflow, remove the existing assignment and assign a BPMN workflow instead. If you select a job type with an existing workflow as described in section 2.3.2.3, the *Assign BPMN workflow* option is displayed. This is a replacement for the legacy *Copy type and assign a BPMN process* function.

The feature *> Administration > Datasheet Engine > Assign Workflow* available in previous versions is no longer present in version 7.2. Therefore, proceed as follows:

1. Click *> Administration > Datasheet Engine > Types*.
2. Locate and select the desired process in the overview list.
3. Click the pencil icon.

The *Properties* page is displayed.

4. Switch to the Workflow Modeler page via the ☰ select menu.

The BPMN Workflow Modeler workspace is opened. This is where you can design a workflow according to the description in Chapter 3.

2.5 Managing a datasheet layout

Note

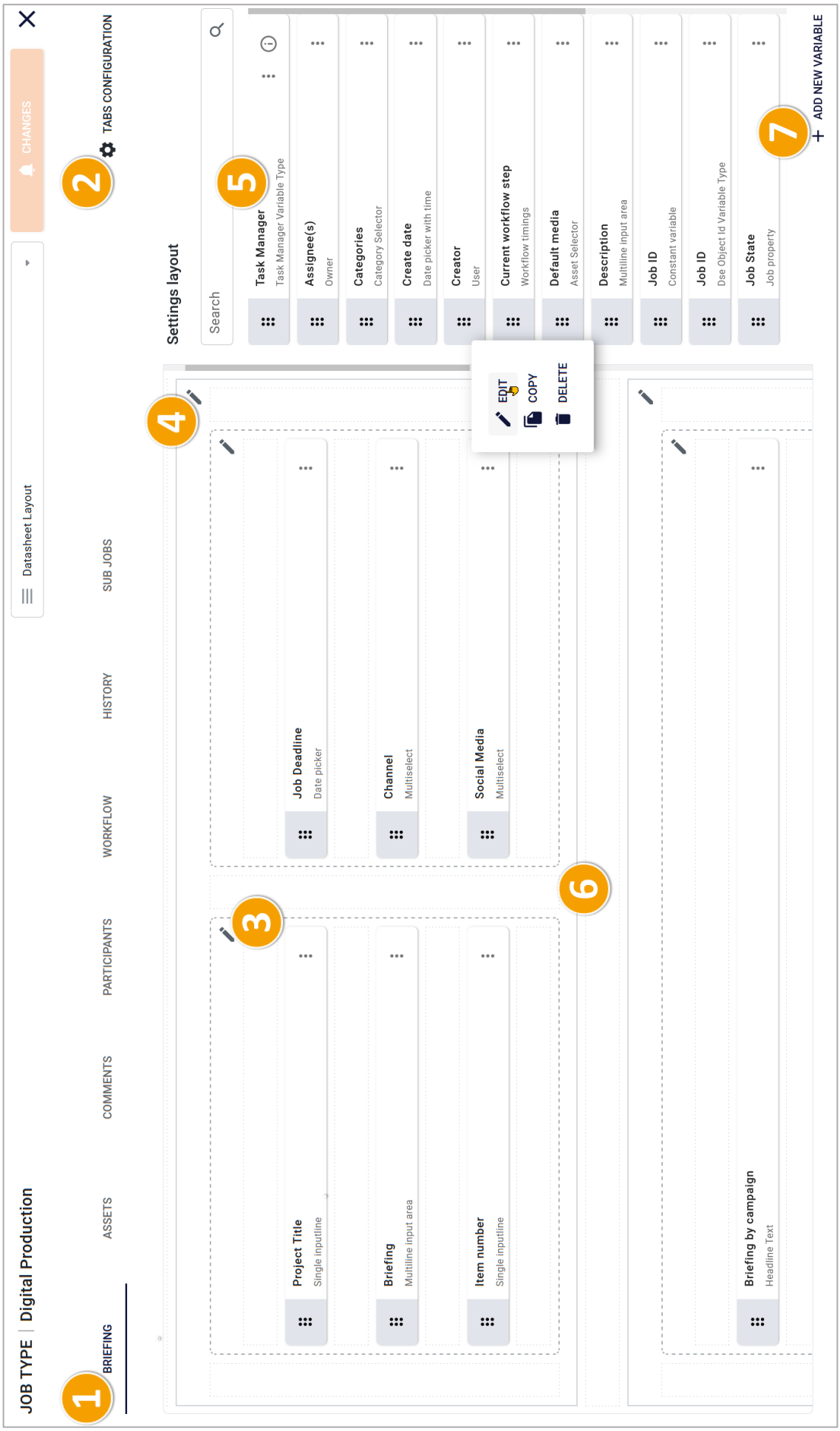
This section describes how to set visibility for job and data object types. A description of the visibilities for processes can be found in chapter 3.7.

You can use existing variables and attributes to design datasheets that are used and filled out by users when creating a job, process, or data object. Note that a process is also managed in the Job Manager module.

To give users the option of hiding already edited areas for better clarity, administrators can configure in the settings (3) of the datasheet layout that columns and rows can be minimized by collapsing them.

1. Go to > Administration > Datasheet Engine > Types.
2. Open a type and switch to the *Datasheet Layout* page via the ☰ selection menu to edit the datasheet of the type.

Note: It should be noted that the display of the datasheet differs in the Job Manager and Data Hub. A datasheet in the Job Manager always includes the tabs *Basic Data*, *Comments*, *Participants*, *History*, and *Workflow*. You can rename, hide and rearrange most of these tabs. Tabs can be hidden using the visibility settings, but you cannot delete them. Additionally, a job datasheet always contains a job discussion, a Data Hub datasheet can switch between language variants and has the product discussion in the right sidebar instead.


JOB TYPE | Digital Production
CHANGES
TABS CONFIGURATION
Datasheet Layout
BRIEFING
ASSETS
COMMENTS
PARTICIPANTS
WORKFLOW
HISTORY
SUB JOBS
Settings layout
Search
Task Manager
Task Manager Variable Type
Assignee(s)
Owner
Categories
Category Selector
Create date
Date picker with time
Creator
User
Current workflow step
Workflow timings
Default media
Asset Selector
Description
Multiline input area
Job ID
Constant variable
Job ID
Dse Object Id Variable Type
Job State
Job property
ADD NEW VARIABLE
EDIT
COPY
DELETE
Job Deadline
Date picker
Channel
Multiselect
Social Media
Multiselect
Project Title
Single inputline
Briefing
Multiline input area
Item number
Single inputline
Briefing by campaign
Headline Text
1
2
3
4
5
6
7

Number	Description
1	This lists all the datasheet tabs that have been created.
2	Configuration of Tabs: Adjust the order of the tabs following General, add new tabs, rename existing ones, or delete them completely. In addition, you can configure the title of the tab in other language versions here.
3	Row settings: Collapsible, Separated. If collapsible, a ⊟ to collapse or a ⊞ to expand, the area appears on the left of the row name in the open datasheet.
4	You can use the pencil icon to access the column settings. Use the slider to adjust the display width of each column individually. Click *Apply* to save the changes.
5	In this pane, you can find all variables that have not yet been placed in the layout. Drag and drop variables from here into the layout onto a vertical placeholder for a new column, or onto a horizontal placeholder for a new row above or below your existing layout. You can filter a long list of variables by searching using the search field. Complex layouts with up to four columns with different column widths are possible.
6	This is the area where the datasheet layout is displayed. To rearrange the order of columns, drag a variable box at the dashed frame and release it via drag-and-drop at the new vertical position. To change the order of the columns, drag a variable field by its dashed frame and drop it at its new vertical position. To clear a column or row, drag-and-drop the dashed frame to the right into the *Settings layout* area. The dropped frame and its variables are removed from the layout. In the same way, drag-and-drop the gray outer frame back to the right into the *Settings layout* area, to clear the layout and redesign it from scratch. You can edit or copy variables from the ⋮ context menu. You can use the ⁝⁝ drag-and-drop handle to reposition a single variable in the layout, or move it back to the *Layout Settings* area to remove it from the layout.
7	Click *+ Add New Variable* to create a new variable for use with this datasheet.

Note: If you want to connect a data object and its data in the *Brand Template Builder* module to the *Smart Group* function, an image must be defined in the *Images* field on the datasheet and the data object must be assigned to at least one theme. Accordingly, the type must provide the system variables *Images* and *Themes*. Take this into account when you create the datasheet.

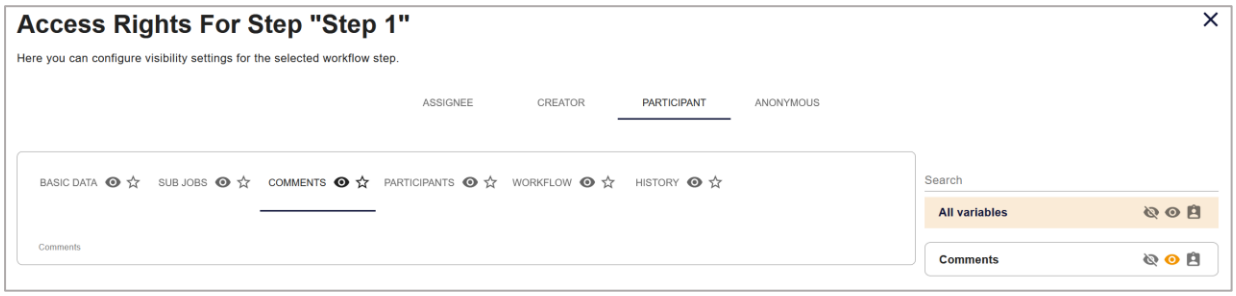
2.5.1 Visibility of a datasheet tab

The access to and visibility of tabs and the variables placed on them can be restricted when you configure the type for each workflow step. For this, users are divided into the categories *Assignee/Processor*, *Creator*, *Other participants* and *Anonymous*. The category that you belong to when opening the datasheet as a user thus determines which tabs and variables on the datasheet can be viewed or edited.

To edit the visibility of a datasheet tab, go to > *Administration* > *Datasheet Engine* > *Types* and open the type. In the *Workflows* / *Workflow Modeler* page, click the icon . The visibility of a datasheet tab can be set per workflow step for classic workflows and per user task for BPNM as follows:

Category	Visibility
Assignee	Visible or not visible
Creator	Visible or not visible
Other participants	Visible or not visible

Note: The visibility of the *Basic Data* tab cannot be changed. This tab is visible to all users.



2.6 Managing Variables

You can edit the variables that are assigned to a type or add new variables. After you have opened a type, the assigned variables can be viewed and managed in the *Datasheet Layout* tab.

> **Note**
>
> If you want to connect a Data object and its data in the *Brand Template Builder* module to the *Smart Group* function, an image must be defined in the *Images* field on the datasheet and the data object must be assigned to at least one theme. Accordingly, the type must provide the system variables *Images* and *Themes*. Take this into account when you create the datasheet.

Prerequisites:

- You have the right MANAGE_VARIABLES.

⋮ Menu/Button	Description
+ ADD NEW VARIABLE	This creates a new variable for the selected type.
	This edits the selected variable.
	This copies the selected variable.
	This deletes the selected variable.

> **Note**
>
> You can also create a new variable when editing the datasheet layout.

2.6.1 Grouping variables

In principle, each variable is created individually for each type. This means that the values of the variables are displayed in different columns in the overview of the module.

You can group variables to compare specific values for each job, process, or data object to each other. You can then display these values in a column in the overview. Examples include:

- The price of data objects that are derived from various data object types
- The completion date of jobs that are derived from various job types

To group variables, the variables in the various types must have an identical technical name and must be of the identical variable type. In addition, the option *For all types* must be activated in the settings of the variables.

If a user wants to display these variables in a column:

1. In the overview page, navigate to *Jobs* > *Add column*.
2. In the ▾ menu, find the name of the job type in question.
3. . Click ⊞ in front of the type name to display the list of its variables.
4. Choose a variable.

The column is displayed and has the variable name in the column title.

A variable column can be deactivated again by hovering over the column title on the job overview page until the ▾ menu is visible on the side. There, you remove the checkbox in front of the column title.

2.7 Visibility of variables

> **Note**
>
> This section describes how to set visibility for job and data object types. For a description of the visibility of processes, see Chapter 3.7.

By defining the visibilities and rights of a variable, you can specify whether a variable is later displayed on the datasheet for certain user roles and can be edited. You can configure these settings separately for each workflow step. There are a number of different user categories:

- *Assignee*: The assignee is the user responsible for the workflow step.
    - *Creator*: The creator is the user who has created the job or data object type.
- *Other participants*: Other participants are users who were invited to the job or data object type.
- *Anonymous*: All other users who do not belong to one of the named user groups are flagged as anonymous.

To define the visibility and editing options for the user categories, you:

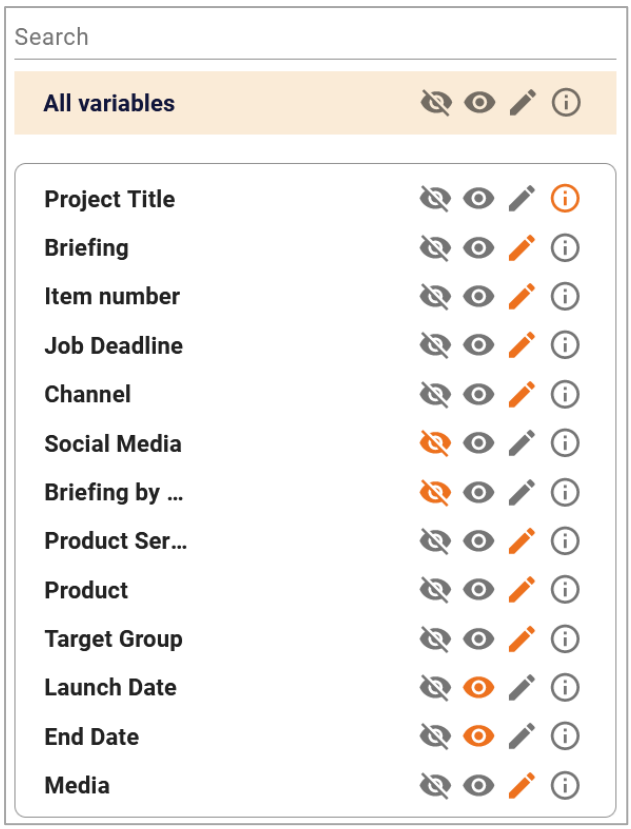
- Choose > *Administration* > *Datasheet Engine* > *Types*.
- Open the desired type then switch to the Workflows page via the ☰-select menu.
- For each workflow step separately, click the icon to configure access rights.
- Access rights are configured individually on the tabs for *Assignee*, *Creator, Participant* and *Anonymous* and for each datasheet tab.

On the right, you now see the list of all variables. If this list is too long, a search field is available above it to narrow down the results.

You can define the visibilities and editing options by activating the following options:

Name/Icon	Description
	The variable is displayed and can be edited.
	The variable is displayed read-only and cannot be edited.
	The variable is defined as a mandatory field and must be edited. You and the person processing the datasheet will recognize this by the * after the variable name in the datasheet.
	The variable is not displayed.

By clicking on one of these icons in the column title, the change will be applied globally to all variables in the tab. However, you will usually want to adjust each variable individually. The active settings are highlighted in a different color.



2.8 Workflow messages

Messages are generated automatically and sent for actions such as forwarding to the next workflow step or inviting a user to a job, for example.

Prerequisites:

- You have the right MANAGE_EMAIL_NOTIFICATION.
  1. Go to *> Administration > Datasheet Engine* and choose a Job type.
  2. Click the pencil icon to open it for editing.
  3. Via the ☰ -selection menu, you are directed to the page *E-Mail Management* > *Manage Standard E-Mails*.

2.8.1 Manage Standard-E-Mails

You can define which users are informed with a message when an action is carried out for a type. To accomplish this, users are divided into different user groups. Select a type from the dropdown list and select *Manage e-mail dispatch*.

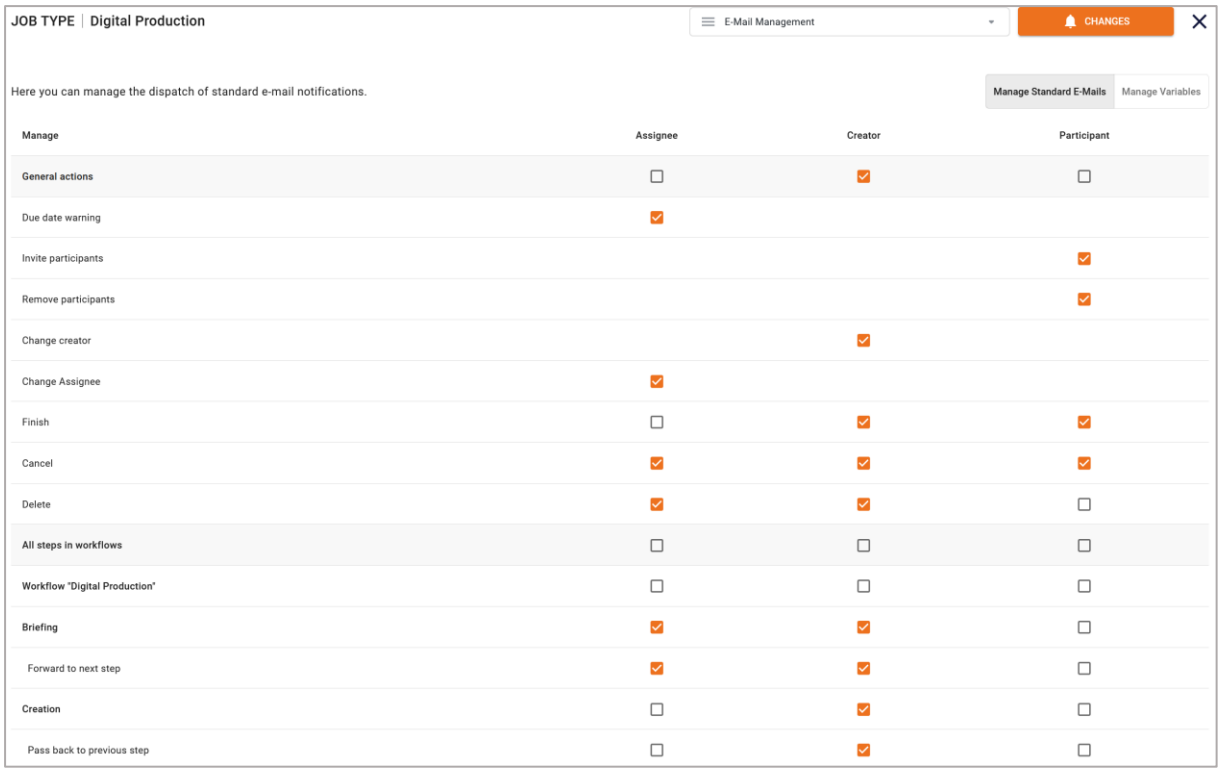
User groups:

- *Assignee*: The user who is responsible for the workflow step.
- *Creator*: The user who created the type.
- *Participant*: Users that have been invited.

Actions that trigger the dispatch:

- *General actions*: Due date warning, Invite participants, Remove participants, Change creator, Finish, Cancel, Delete, Change Assignee
- *Workflow-specific actions*: Forward to next step, Pass back to previous step

Activate the corresponding checkboxes to configure the dispatch of messages. Note that the user that triggers an action does not receive a message.



2.8.2 Manage Variables

The available variables are listed here. You can define which variables appear in which order in the standard messages under the *Job variables* section.

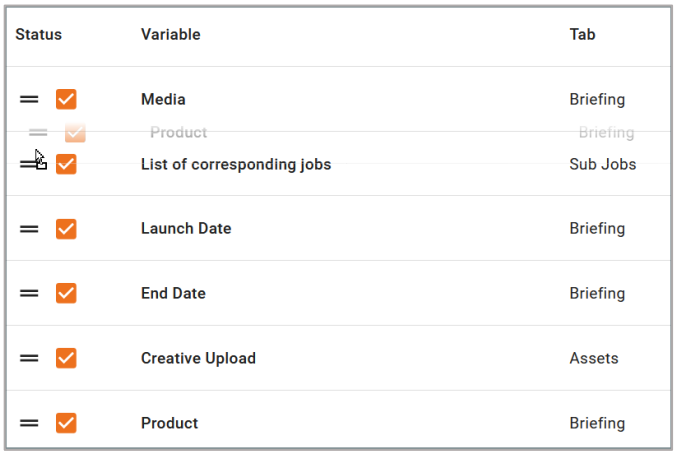
1. Switch to the *Manage variables* tab.

The list contains the following columns:

Name	Description
Status	Activate the checkbox to adopt the variable or its value in the system messages.
Variable	This displays the name of the variable.
Tab	This displays the tab on which the variable is placed.

To sort the variables: For activated variables (checkbox checked), you can change the order individually using the ═ gripper.

2. To rearrange the order, drag the element as shown in the figure and drop it in the desired new position.



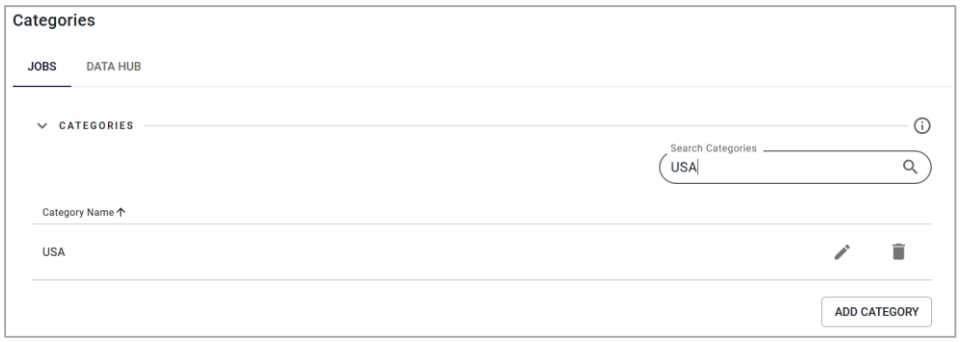
You have defined which variables appear in standard messages under the job variables.

2.9 Other Settings

In the section > *Administration* > *Datasheet Engine* > *Other Settings* and its three subsections *Categories*, *General Settings* and *Other Settings* different settings for types and data objects can be configured globally.

2.9.1 Categories

For a better overview, you can create categories and assign types to one or more categories. When users later create a new job, process, or data object, they select the desired type in an overview organized by category. You can access the categories via > *Administration* > *Datasheet Engine* > *Other Settings*. In the *Jobs* or *Data Hub* tab, expand the *Categories* section with the right arrow › to see all categories, browse and filter them further.



2.9.2 Creating a type category

Use case: You want to create the category `Photo shootings` for the Job Manager module.

Prerequisites:

- You have the right MANAGE_TYPE_CATEGORIES.

Step by step:

1. Click *Navigate to > Administration > Datasheet Engine > Other Settings.*

   You are on the *Jobs* tab.
2. Open *Categories* by clicking the right arrow ›.
3. Select the *Add Category* button at the bottom of the list.

   This will open an *Add Category* dialog box.
4. Enter `Photo shootings` in the *Name* input field.
5. Optional: enter the category name in other languages.
   a. To accomplish this, click on the globe icon. This opens the *Enter language versions* dialog box.
   b. Enter the languages that have not yet been translated.
   c. Finish the input with *Save.*
6. Exit the dialog box with *Save.*

You have created the category `Photo shootings` for the Job Manager module. You can assign existing and newly created job types to this category.

2.9.3 General Settings

2.9.3.1 Default types

Create the default type for the Job Manager and Data Hub modules. The type defined as the default is then displayed under the category *Standard* when you create a new job, process, or data object. You can also define whether the *Only Briefing* type can be used in the individual modules. With the help of the *Only Briefing* type, requirements for a job, process, or data object can be described simply, without having to configure an extensive datasheet or select a workflow.

Prerequisites:

- You have the right MANAGE_DEFAULT_TYPES.

1. Navigate to > Administration > Datasheet Engine > Other Settings.
2. Then select the tab for *Jobs* or *Data Hub*, depending on what you want to define the settings for, and click the right arrow › to expand the *General Settings* rubric.
3. Make a selection under *Default for Jobs*
4. Optional: Check the box for *Hide Type "Only Briefing".*
5. Click the *Save* button.
6. For the changes to take effect, you have to publish them under > *Administration* > *Datasheet Engine* > *Maintenance*.

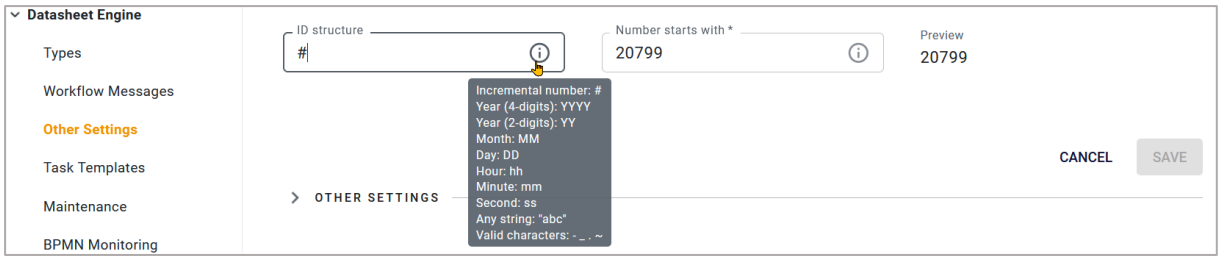
2.9.3.2 Editing the format of the object ID

Select the *ID structure* text field to edit the format of the object ID for a module.

Name	Description
Possible inputs for ID structure	Enter an ID configuration in the box. The provided tool tip (i) provides you with formats and examples. It lists the characters that you can use for the object ID format, such as "YY" for two-digit year numbers or "DD" for days, for example, and special characters permitted.
Number starts with	The next number used (the ID used last + +1) is entered in the input field automatically. The newly defined object ID format is applied from the entered number onwards. You can enter another higher number. Your entry is checked. If you enter an invalid number, you are informed about this by an info message. In the provided tooltip (i) you can find additional information.
Preview	This displays a preview of the current format of the object ID.

2.9.3.3 Configuring the object ID

When you create a datasheet, a unique ID for the job, processes, or data object is assigned automatically. Note that processes are managed like jobs in the *Job Manager* module. Choose > *Administration* > *Datasheet Engine* > *Other Settings* > *Jobs* > *General Settings* to determine the format based on which the object ID for jobs and processes is created. In the *Data Hub* tab, you will find identical settings for data objects.



Prerequisites

- You have the right MANAGE_OBJECT_NUMBERS.

Note: A newly defined object ID format is applied as of the next incremental number (#) that is used. Unique IDs that have already been assigned for datasheets remain unchanged. The newly defined formats (a YYYY-MM-DD time stamp, for example) are attached to these IDs.

2.9.4 Other Settings

For the *Only Briefing* type, you can define the message text when a job or data object is forwarded. Click *> Administration > Datasheet Engine > Other Settings,* and then select whether you want to define the settings for jobs or data objects. Select the tab *Jobs* or *Data Hub* and click the right arrow to expand the *Other Settings* section.

Prerequisites:

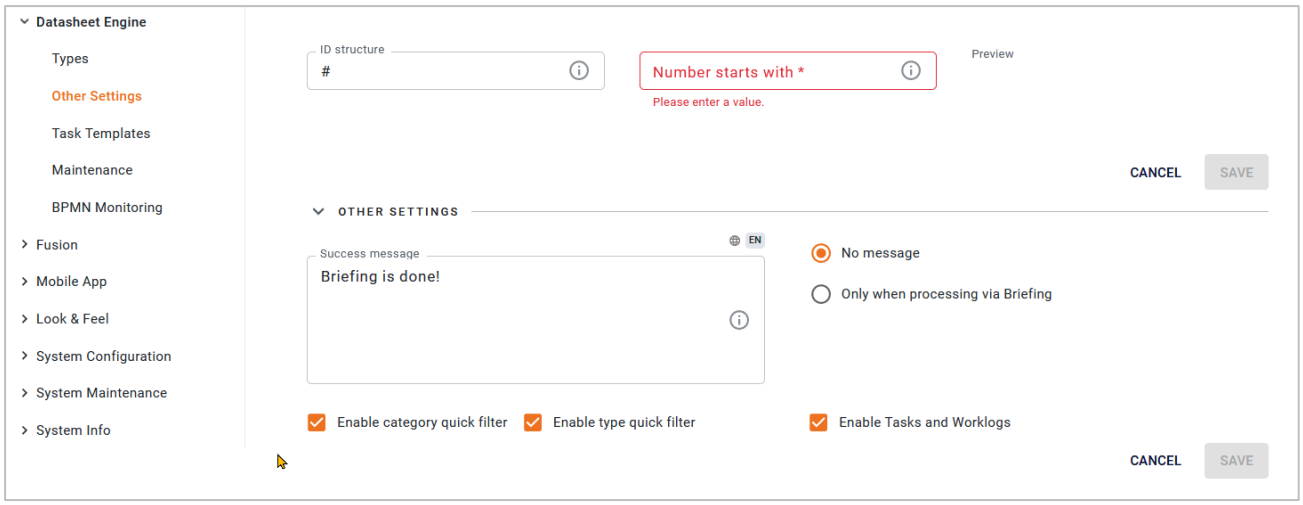
- Use of the *Only Briefing* type is allowed.

From the dropdown list or input field, you can define which message (if any) is sent when jobs or data objects of the type *Only Briefing* are forwarded:

- *No message*: No message is sent when the item is forwarded.
- *Only when processing via Briefing*: The stored message is sent when the item is forwarded.

You can also set the checkbox at *Enable category* quick filters, which allows filtering of type categories in the overview of the respective module for all users. Select the checkbox at *Enable type quick filter* to allow type filters in the overview of the respective module for all users.

Note: If you want to make any changes to the **ID structure**, you need to publish them (go to Administration > DSE > Maintenance > Publish Changes).



If you want to access the navigation for tasks and time management, activate the checkbox at *Enable Tasks and Worklogs*.

The changes are applied with *Save*. Changes to these settings must also be published on the *Administration* > *Datasheet Engine* > *Maintenance* > *Publish changes* page. The `Categories` system variable must have been used at least once on a datasheet for *Filter by Category* to be available in the Job Manager overview.

2.10 Task Templates

You can use a created task schedule again by saving it as a template. In the Task Planner, you can thus quickly and easily access and use a task saved as a template. Click *> Administration > Datasheet Engine > Task Templates* to create a new template or edit an existing template.

Prerequisites:

- You have the right MANAGE_TASK_TEMPLATES.
- Tasks are already saved as templates.

This opens an overview that displays the name of the task, the number of work steps, the description, and the creator of the task. Choose one of the icons to edit a task template or create a new task template.

Name/Button	Description
Neu	This creates a new task template.
	This allows you to edit the task template.
	This copies the task template.
	This deletes the task template.

2.10.1 Creating a task template

You want to create a task template to define the steps for creating a new brochure. You want to define the individual steps `Check image material and texts`, `Assign agency`, `Assign print agency`, and `Send`.

Prerequisites:

- To use the task template, a workflow that covers the work steps defined in the task template is required.
- You have the right MANAGE_TASK_TEMPLATES.

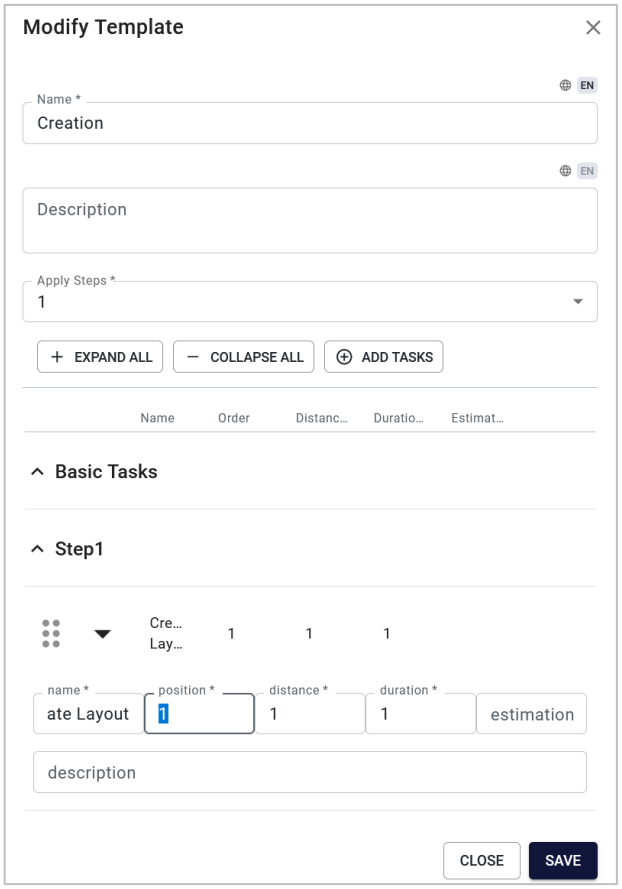
Step by step:

1. Choose *> Administration > Datasheet Engine > Task Templates*.
2. Select *Create new* to create a new task template.

   This opens a new dialog box.
3. Enter the name of the task template in the input field.
4. *Optional*: Enter the name in different language versions.
5. *Optional*: Enter a description of the task template.

6. Select the entry *1* from the *Apply Steps* dropdown list.
7. Choose *Add Tasks*.
8. You activate the edit mask with the right arrow ▸ below the *Step 1* entry.
9. Enter `Check image material and texts` as the name of the task.

Note: In an activated input screen, use the esc key to discard an entry.



10. Double-click the cell in the *Duration* column.

This activates the input screen.

11. Enter the *Duration* in days and the *Estimation* in hours.
12. Select the entry *2* from the *Apply Steps* dropdown list.

This creates the field *Step 2*.

13. Choose *Add Tasks*.

This activates the input screen below the entry *Step 2*.

14. Enter `Assign agency` as the name of the task.
15. Repeat steps 9 and 10.
16. Use the *Apply Steps* dropdown list to create the work steps `Assign print agency` and `Send`.
17. Click *Save*.

The task template has been created and can be used on a datasheet with Task Manager. The names of the work steps are overwritten with the names of the individual workflow steps of the workflow assigned to the job when a template is used.

2.11 Publishing changes

Changes to a type, process, or data object can be displayed and published individually by clicking on the bell icon with the *Changes* button. As long as no publication of changes is pending, the button remains grayed out. After publishing, the floating sidebar disappears again automatically or can be hidden manually at any time by clicking ×..

All the changes made in the type configuration are grouped together and can be carried out simultaneously at a time that you can specify yourself. This does not only mean the changes to the types themselves, but any changes and adjustments within the datasheet engine. This includes, for example, datasheet layout, categories, task templates or the variables.

Prerequisites:

You have the right PUBLISH_DSE_CHANGES.

Attention!

You have to fix all formal errors before you can publish a newly created or modified BPMN workflow. Publishing with formal errors could permanently damage the job type.

2.11.1 Publish changes individually

You can publish the changes for each type separately. You do this under *> Administration > Datasheet Engine > Types* on the *Changes* side panel.

Except for classic workflows, all changes related to this type are published. Note that you cannot plan such a publication.

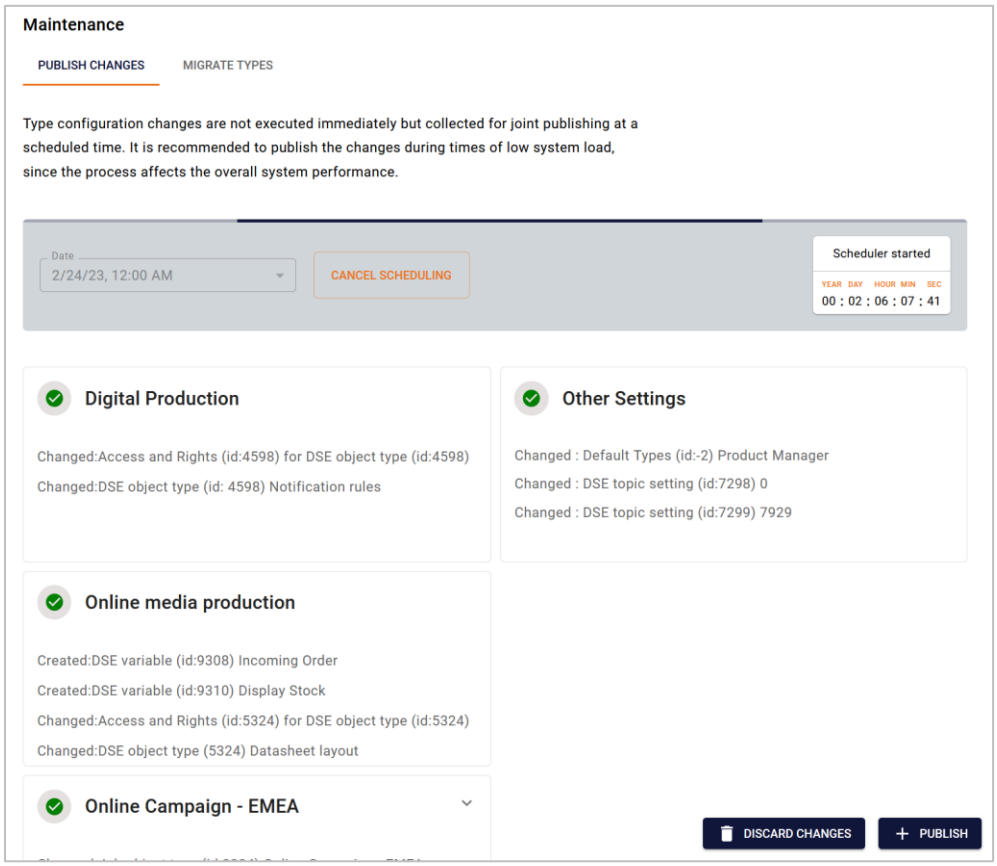
Note

Note that publishing changes may lead to impact performance for other users. Therefore, we recommend publishing the collected changes for a larger number of types at a time when system utilization is low. Schedule this at a time when there are no or only a few users working in the system.

Associated tasks

- *Publishing changes immediately*, page 53: You can publish changes immediately at all times. You can do so even if you have already scheduled a publication.
- *Scheduling a publication*, page 54: If you want to publish the changes at a specific time, you can enter this time. Administration then carries out the publication at this time. You can schedule a publication even if there are still no changes in the type configuration.
- *Changing a scheduled publication*, page 56: If you want to carry out the publication at a different time, you can edit the time.
- *Canceling a scheduled publication*, page 56: If you no longer want to carry out a publication, you can cancel it.

- *Discarding changes*, page 57: If you would rather not publish the changes that have been made, you can delete them. Note that you can only undo all the pending changes, not just one.

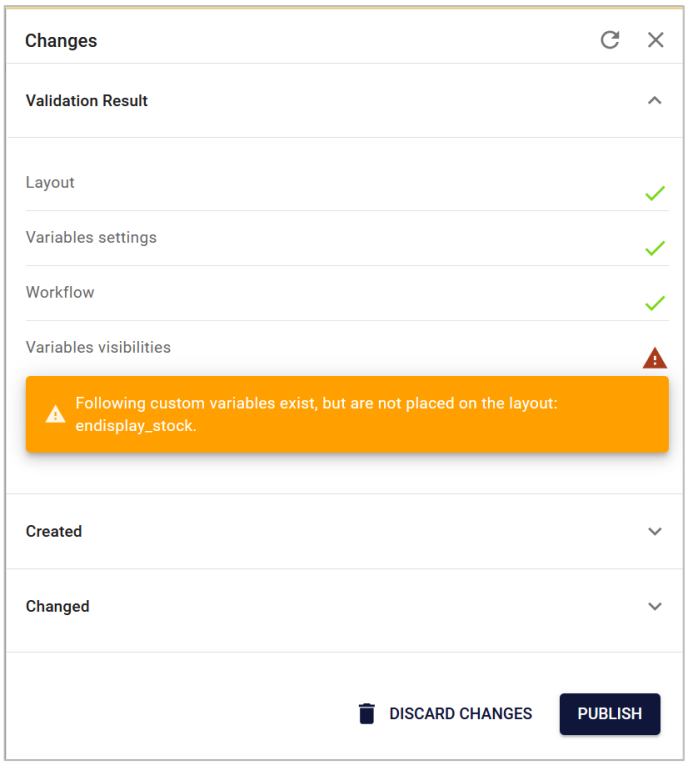


2.11.2 Validation

The validation of the types (Job, Data object or Process) is done automatically within the respective creation dialog. When you create a new type or modify an existing type, the validation information is displayed on the *Changes* panel shown on the side under the collapsible and expandable *Validation results* area. Errors that need to be corrected are briefly described and highlighted in red.

1. Click the *Changes* button if the button is active and unpublished changes are therefore present.

The side panel with the changes will be displayed.



1. After corrections and adjustments, click the icon ↻ under the Changes panel to validate the type again.

You have validated the type and can publish it.

2. Optional: Click the *Publish* button to publish the type immediately.
3. Optional: Click the *Discard changes* button to return to the last saved status or to undo all changes made so far.

   You will be prompted for final confirmation.

Only after all validation errors are fixed, can you publish the type using the *Publish* button.

2.11.3 Publishing changes immediately

You can publish changes immediately at all times. You can do so even if you have already scheduled a publication.

> **Note**
>
> For BPMN workflows, changes to be published are only created when you have changed the way the workflow works. For example, by adding service tasks or user tasks. Nonrelevant changes include changing labels of decisions and step names of simplified workflow steps or moving elements in the editor without changing their order or number.

Prerequisite

- An administrator has added changes that have not been published yet.

Publish changes of a type immediately

1. Navigate to > *Administration* > *Datasheet Engine* > *Types*.
2. Click the pencil icon for the type whose changes you want to publish.
3. The editor opens.
4. Click the *Changes* button.
5. Click *Publish*.

The changes are published. After clicking the *Publish* button, both server-side confirmations and error messages are briefly displayed as browser notifications.

Publishing accumulated changes immediately

1. Navigate to > *Administration* > *Datasheet Engine* > *Maintenance* > *Publish changes*.

The list of pending changes is displayed.

2. Press the *Publish* button.
3. In the dialog *Publish changes,* press *Confirm*.

All pending changes are published immediately.

> **Note**
>
> You will also be notified of existing errors in the type configuration on the overview page under > *Administration* > *Datasheet Engine* > *Maintenance* > *Publish changes*. From there, you can open each type with warnings and errors directly via a deep link to resolve the problem. Only when all pending changes are marked with a green checkmark, can you publish them immediately or at a later time.

2.11.4 Scheduling a publication

If you want to publish the changes at a specific time, you can enter this time. The back-end system then performs this task on schedule. You can schedule a publication even if there are still no changes in the type configuration.

> **Note:** Note that publishing changes may lead to impact performance for other users. We therefore recommend publishing changes at a time when the system load is lower.

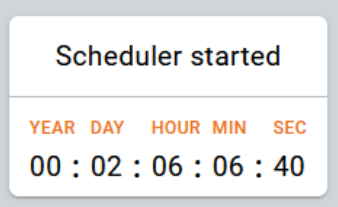
1. Navigate to > *Administration* > *Datasheet Engine* > *Maintenance* > *Publish changes*.

   The list of changes is displayed.

2. Click in the date field.

3. Enter the date and time of the publication and confirm it with the *Apply* button.
4. Choose the button *Schedule Publishing*.
5. In the confirmation dialog, press *Confirm*.

A countdown is displayed.



You have scheduled the publication. The changes will be published at the scheduled date and time.

2.11.5 Changing a scheduled publication

If you want to carry out the publication at a different time, you can edit the time.

Prerequisite

- You have already scheduled a publication.

Changing a scheduled publication

1. Navigate to *> Administration > Datasheet Engine > Maintenance > Publish changes*.

   The list of pending changes and the publishing time are displayed.

2. Press Cancel Scheduling and then press Confirm.

3. Use the date picker to enter the new time of publication and confirm it with the Apply button.

4. Select the button *Schedule Publishing*.

5. In the confirmation dialog, press *Confirm*.

   A countdown is displayed.

You have edited the publishing time. The changes are published at the adjusted date and time.

2.11.6 Canceling a scheduled publication

If you no longer want to carry out a publication, you can cancel it.

Prerequisite

- You have already scheduled a publication.

Changing a scheduled publication

1. Navigate to *> Administration > Datasheet Engine > Maintenance > Publish changes*.

   The list of changes and the publishing countdown are displayed.

2. If you have changed your mind, select *Cancel Publishing*.
3. In the confirmation dialog, press *Confirm*.

The publication is now canceled. The changes therefore do not take effect.

2.11.7 Discarding changes

If you would rather not publish changes that have been made, you can delete them. Note that you can only undo all the pending changes, not just one.

> **Warning! Data loss!**
>
> If you discard changes, all the unpublished changed in the type configuration are discarded. You cannot undo this step!

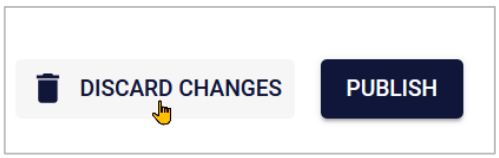
Prerequisite

- An administrator has added changes that have not been published yet.

Discard changes of a single type

1. Click > *Administration* > *Datasheet Engine* > *Types*.
2. Click the pencil icon for the type whose changes you want to publish.

   The editor opens.
3. Switch to the *Changes* tab.



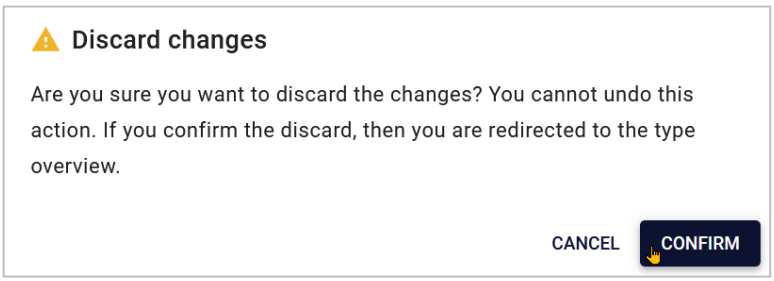
4. Click *Discard changes*.

The changes in the type configuration are deleted.

Discarding all changes

1. Navigate to > *Administration* > *Datasheet Engine* > *Maintenance* > *Publish changes*.

   The list of pending changes is displayed.
2. Choose *Discard Changes*.



3. In the confirmation dialog, press *Confirm*.

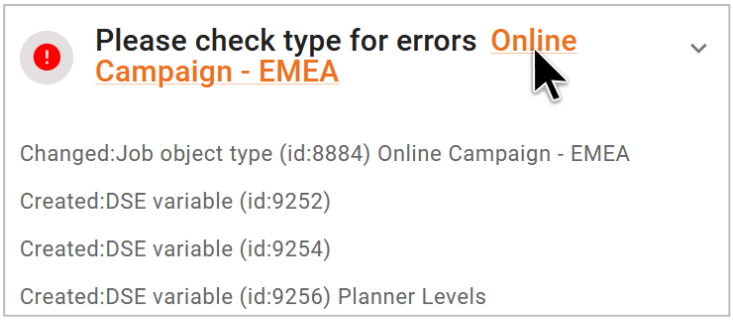
All unpublished changes in the type configuration are deleted. The page displays “No changes”.

2.11.8 Errors prevent publication

When setting up a new system, thousands of database changes quickly accumulate. If several people are responsible for maintaining the types in the system, in such a case you will of course not immediately recognize which and above all, whose changes could not be published because they are incorrect.

The display of remaining validation errors on the *Datasheet Engine* > *Maintenance* > *Publish changes* page supports you in troubleshooting. However, don't let this error list worry you, because it doesn't mean that all type customizations are lost. From here you can open any type with warnings and errors directly via a deep link and edit them to resolve the problem.

However, don't let this message worry you because it doesn't mean that all your type The display of existing validation errors in the *Changes* panel supports you in troubleshooting.



Continue as described above in section 2.11.2.

After a scheduled publication, you can be notified of the publication status on the *Publish changes* page. If the page is empty after execution, and you see the *No changes* notice, everything worked.

3 BPMN workflow

This chapter explains how to create BPMN workflows for processes. You can create a BPMN workflow in the BPMN Editor.

Don't be afraid of the initial complexity. Take your time to learn the ropes. If you had only a few building blocks at your disposal before, entirely new possibilities open up now. With BPMN, you have numerous design levels and elements at your disposal. The following chapters can help you with the conversion from classic workflows to BPMN. A good way to do this is to use the *Copy type* option when creating a new type, select a classic job, but enable the *Assign BPMN Workflow* option. See Chapter 2.3.2.3.

Open BPMN editor

1. Click > *Administration* > *Datasheet engine* > *Types*.
2. Click the pencil icon for editing a process type.
3. Select *Workflow modeler* from the ☰ dropdown menu.

The BPMN editor opens. The following chapter describes the layout of the editor.

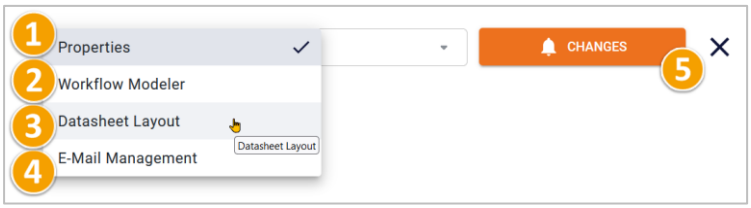
3.1 Layout of the Editor

This section describes the layout and main functions of the editor.

The editor consists of four subpages:

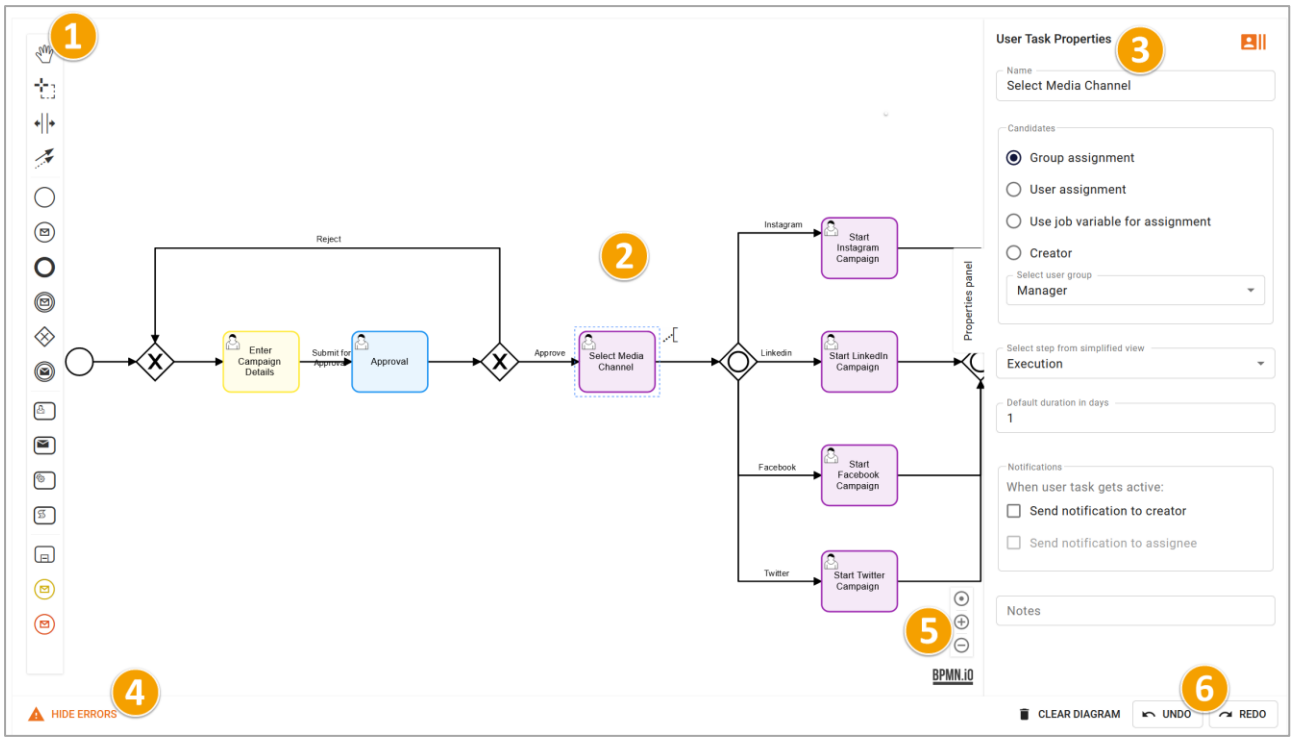
- Properties: All properties of the type are entered or edited here (1). Properties are described in Chapter 2.3.1.
- Workflow Modeler (Chapter 3.1.1, 2)
- Datasheet Layout (Chapter 2.5, 3)
- E-Mail Management: Editor for creating e-mail templates for sending tasks (Chapter 3.4, 4)

Changes button: As soon as changes have been made, the previously grayed out *Changes* button becomes active (5).



A side panel opens where you can validate and publish the changes to the type. Here you can view the changes made to the type and any errors and then publish them separately for each type. (see Chapter 2.11).

3.1.1 Workflow Modeler



1 Toolbar

The toolbar contains the elements you use to build the structure of your workflow. See chapter 3.2 for a detailed element description with usage examples. You can also access the following tools for working with the elements in the drawing area here:

- Hand tool
- Lasso tool
- Add/remove space

2 Drawing area

You add the elements to the drawing area to build your workflow.

3 Workflow Properties

This area shows the properties of the element you have selected. The screenshot above shows the workflow dialog where you activate and set the simplified view. For more information about the simplified view, see chapter 3.6.

4 Button for switching the view

Either the *Hide errors* or *Show errors* button is displayed in this area:

- Show current error messages: If you activate this view, the current formal errors in the workflow design are displayed. This setting is especially recommended when you are finishing the design. If you deactivate the view, no formal errors are displayed.

5 Action buttons

You can access the following actions in the bottom area:

- Delete diagram: This action deletes all the elements from the drawing area.
- Undo: The last action is undone.
- Redo: The last undone action is restored.
  Keyboard shortcut: CTRL + Z / CMD + Z
- *Redo*: The last undone action will be restored.
  Keyboard shortcuts: CTRL + Y or SHIFT + CTRL + Z / CMD + Y or SHIFT + CMD + Z

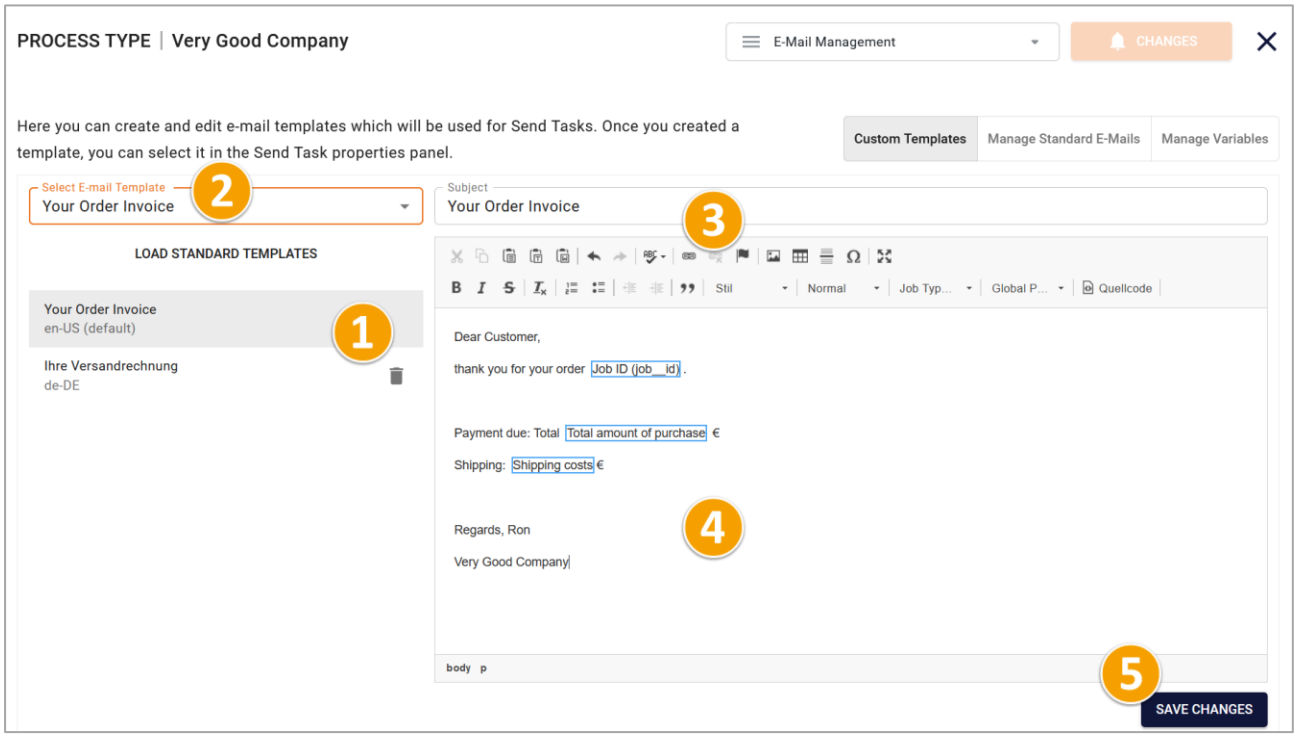
6 Zoom Buttons

These buttons are used to customize the view.

- ⊙: Center the view and reset the zoom to the default value.
- ⊕: With each click, you enlarge the view by zooming in.
- ⊖: With each click, you zoom out to make the view smaller.

3.1.2 Custom templates

In the open process type, navigate to ☰ > *E-mail management* > *Custom templates*. Here you can create and edit the e-mail templates that are to be used later when sending tasks in processes.



1 List of language variants

When you create or edit a template, the language variants are displayed here. In the current version, you can create German and English language variants, whereby the version for US-EN must be created first.

2 *Select E-mail Template* selection list

In the selection list, select the e-mail template you want to edit, or select the *New e-mail template* entry to create a new template. Optional: If you click *Load Standard Templates* below the list, the templates stored in the system are loaded into the selection list. You can then also edit the default templates.

3 Subject

Enter the subject of the e-mail in the selected language variant.

4 Editor

Enter the content of the e-mail in the editor. You have numerous formatting options, and you can add system variables (see Chapter 4.1.14) and job variables using the drop-down lists *Job Type Placeholders* and *Global Placeholders*. Variables of the following types are available:

- Single line and multiline input area (see chapter 4.1.14)
- Single-select and multi-select (see chapter 4.1.14)
- Date picker with/without time (see chapter 4.1.6)
- User (see chapter 4.1.2)
- Numbers (see chapter 4.1.12)
- Combo box (see chapter 4.1.11.1)
- Comfort Grid (see chapter 4.1.4)
- Advanced Grid (see chapter 4.1.2)

Other variable types cannot be inserted.

In the use case with BPMN, only the following variables are filled:

- JM_JOB_CREATOR_NAME
- JM_JOB_CREATOR_EMAIL
- JM_JOB_LIST_VARIABLES
- LINK_JM_JOB_DETAIL

5 Create template

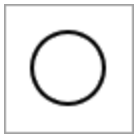
Click the button to save the created template. The text entered under *Subject* is used as the name.

3.2 Available elements

Some concepts of the "Business Process Management Notation" (like Service Tasks, Script Tasks, Events, etc.) are not immediately understandable for casual users. Using this technology requires a comprehensive range of professional experience. If this is present, you will soon be rewarded and amazed at what is possible with it.

3.2.1 Start event

The start event initiates a process instance. Each BPMN workflow requires exactly one start event.



Parameters

You can enter a name for the start event in the properties dialog.

3.2.2 End event

The end event ends a process instance. Each BPMN workflow requires exactly one end event.

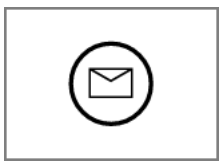
Parameters

You can enter a name for the end event in the properties dialog.

3.2.3 De-archive event

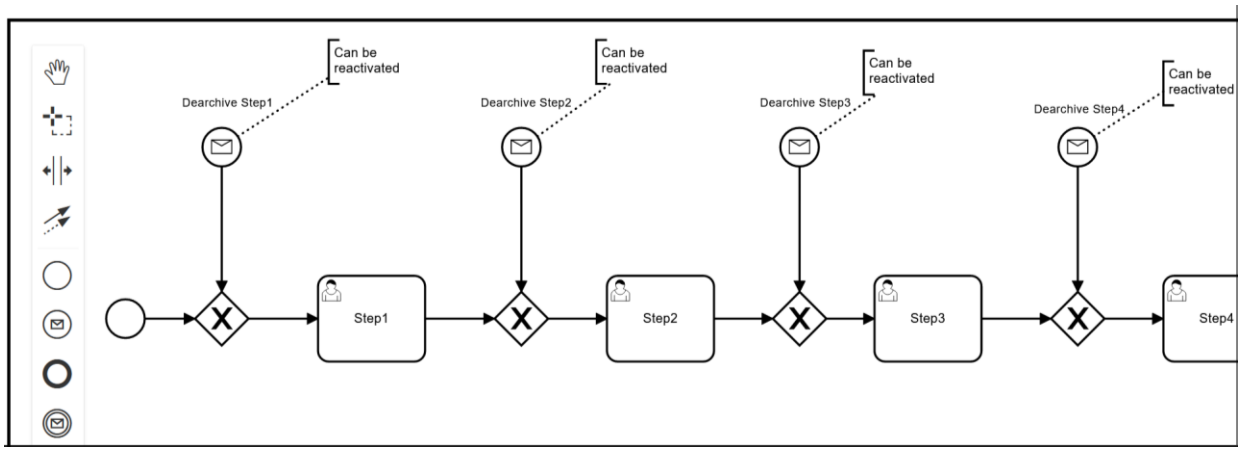
Occasionally it is necessary that terminated or canceled jobs can be reactivated with BPMN workflows. You can adapt your existing BPMN workflows accordingly. A standard BPMN message event of type `StartMessageEvent` is used for this purpose.

The De-archive event must be accessible to the StartMessageEvent. Appropriately prepared jobs can later be restored by users in the Job Manager. If the associated type has been deleted, processes that have already been canceled and terminated cannot be reactivated.



Implementation in the Workflow Modeler

The workflow can be started either by the regular start event or a dearchive event stored as De-Archive-StartMessageEvent. You can theoretically insert a dearchiving event with a gateway before each workflow step.



Notes

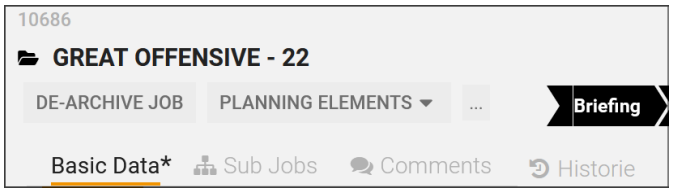
- Currently, a BPMN job can only be reactivated if the entire workflow does not contain fewer steps at the time of reactivation than at the time of job creation.
- Deleted jobs cannot be reactivated, even if they were previously properly terminated or canceled.

In practice, not every entry point proves useful. If a job were reactivated only after an element such as *CreateNode*, the person reactivating the job would be missing the previously uncreated node in the current workflow step. Therefore, careful planning is required for each user scenario.

Job Manager users can reopen a completed or canceled job with BPMN workflow. Prerequisite: Users need the DE_ARCHIVE right in their Job Manager role.

You will then find the button *De-archive Job* where the button *Close* is otherwise located. If several reactivation points exist, a selection dialog appears where the user can choose from a list to reactivate the job in a selectable workflow step.

User interface in the Job Manager to reactivate a canceled or terminated job with BPMN workflow:



3.2.4 User task

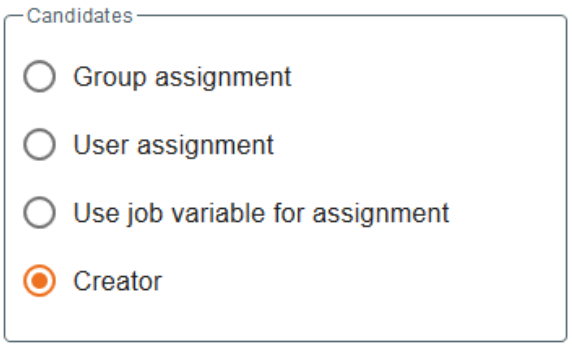
Each step in a process instance must be described as a task. The *User task* element is used to model the work to be performed in the system by a human actor.

Name

For a user task, you have to enter a name in the *Properties panel*.

Assignment

Since user tasks must be executed by real humans, you need to assign them to a group or a specific user. Alternatively, you can also select the responsible person using a variable on the datasheet (e.g., with the *Creator* variable).



- *Group assignment*: When the user task is activated, it is displayed in the *Available tasks* list for the group. Every user who is part of the group can accept the task.
  With this setting, you select a user group in the pick-list.
- *User assignment*: When the user task is activated, it is assigned directly to the selected user. With this setting, you select a user in the pick-list. You can search through the users with access to the module based on logins, e-mail address, first name, and last name. An auto-complete helps you to select the user.

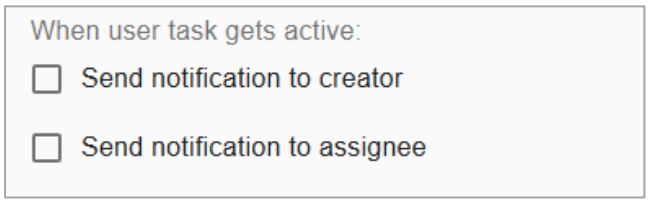
- Use variable for the assignment: When the user task is activated, it is assigned directly to the user who is entered in the selected variable.

> **Note**
>
> The default *Creator* variable is always available. To select a different variable for the assignment, you must first create the datasheet layout with the desired variable.

> **Attention!**
>
> If the variable is not filled out when the user task is activated, the process hangs and cannot be ended. Therefore, ensure that the variable is filled out, for example, by having the variable filled out as a mandatory field in an earlier step (see chapter 1).

- *Creator*: The user task is assigned to the creator of the process.

Based on the assignment, you can control which notifications are sent when a user task is activated. If the task is assigned to a group, a notification is sent only to the creator. If a specific user has been defined as the processor, notifications can be sent to both the creator and the processor.



Decisions

To make it easier to create objects, decisions such as *Approve,* or *Reject* can be enabled and added directly in the properties of the user tasks. Decisions are displayed as buttons in the header of the datasheet. When the user clicks a button, the relevant sequence flow is performed.

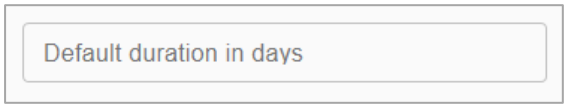
If you activate the *Manage decisions* checkbox, an exclusive gateway is added in the drawing area after the user task. The *Properties panel* for the task displays the default decisions *Forward*, *Approve*, and *Reject*.

You can use these decisions, remove them (recycle bin icon) and enter additional decision options in the *Add decision* text box and save them with ⏎.

You assign the decisions to the sequence flows by connecting the exclusive gateway to the following elements.

Default duration

You can also define a default duration in days for user tasks. The default duration is used to calculate the due date for the task.



Notes

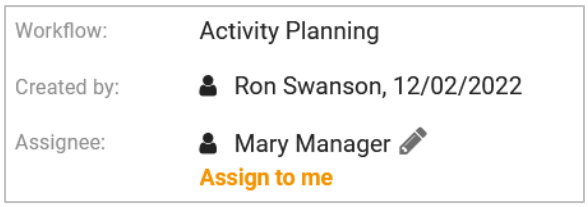
Enter your description in the *Notes* field. This will help you or other colleagues later to track your decisions and thoughts about the workflow structure.

Change Assignee in Workflow step

The editor of the current workflow step can be changed without changing the job status. This option is only available if the user task in the BPMN workflow contains the *group assignment* or the *job variable* for the assignment.

It is not possible to change the editor if they are set as *user assignment* or *creator* in the *User Task Properties*. The new editor will be notified if the notification setting is configured. To accomplish this, scroll down on the right side of the *User Task Properties* panel and activate the checkbox at *Send notification to editor*.

The user assignment of the workflow step can thus be set by editors in the ... menu of the job datasheet, but also in the *Details* area of an open job with the pencil icon in the *Editor* field.

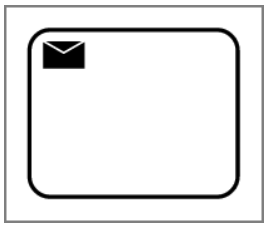


Note

The *Assignee* field must not be empty. The new assignment can be selected only from a user group associated with the active user task, or from the user group corresponding to the user variable in use.

3.2.5 Sending tasks

When a sending task becomes active in the workflow, the system sends an e-mail. The administrator selects a template for the e-mail in the properties dialog and specifies the recipient(s).



In the Workflow Manager, in the *Send Task Properties*, select *Participants* as recipients.

Later, when executed in this step, the send task will send a notification to all current job participants. All participants will also receive notifications under the bell icon in the menu bar.

E-mail template

You can create and select an email template on the *Email Managemen*t page on the *Custom Templates* tab, see chapter 3.1.2. How to create templates is explained in chapter 3.4.

Recipients

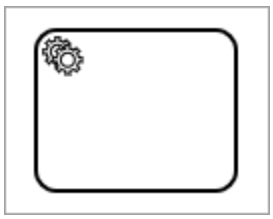
Select one or more recipients of the e-mail.

3.2.6 Service tasks

Note

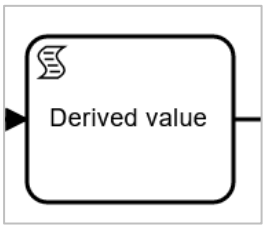
The service tasks are accessible in the interface from version 7.0. Note, however, that these service tasks can only be used to synchronize a job with a planning element in the *Marketing Planner* module. If you require any further assistance, please contact your BrandMaker representative.

All job element actions use the `JobRestService`.



3.2.7 Script task

The JavaScript action, called a *script task* in BPMN, allows developers to add JavaScript code to process execution. Script tasks are used to perform minor calculations and application logic. Applications include data validation, data conversion, and integration with external systems to query or transfer data.



Note

Script tasks are intended for small-scale tasks. For more sophisticated or complex use cases, it is recommended to use either separate Service Tasks or Application Servers.

When a process instance calls up a script task, it creates a corresponding job and waits for its execution. When this job is completed, the process instance proceeds to the next logical step.

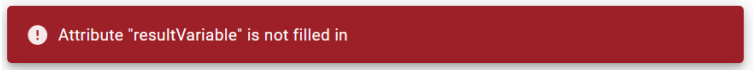
You can use this to quickly add additional custom logic to any business process. First, however, you need to familiarize yourself with the specifics of the script task.

The basic structure of the script template is always static and therefore cannot be changed. Since the editor, apart from the coloring of the syntax, has no real debugging, you can only rely on the reviewing of gross errors before the type is published. The final functional test is then only done with a job based on this process type.

```
1 function calculate(creator__) {
2   var result
3
4
5
6
7   return result
8 }
9 calculate(creator__);
```

You insert your code starting at line 3.

You can use multiple payload variables in a script task. You must select at least the result variable, the payload variable is optional. If the task does not contain the required return value (`resultVariable`), then you will receive an error message during validation before publishing.



Example

You use the `creator__` variable as the payload and want to output the login name of the creator in a job variable at the end of the script task. Subsequently, you will see that more than just the login name is returned:

```
{"name":"Ron Swanson","id":1120,"login":"ron.swanson"}
```

This content must first be parsed. You can do this with the following code:

```
const json = creator__;

const obj = JSON.parse(json);

result = obj.login;
```

```
1  const myJSON = JSON.stringify({
2    name: 'Ron Swanson',
3    id: 1120,
4    login: 'ron.swanson',
5  });
6  const obj = JSON.parse(myJSON);
7  var result = obj.login;
8  console.log(result);
9

CONSOLE ×

ron.swanson
```

You now actually get the login name alone as a return string.

```
ron.swanson
```

Note

Proceed carefully and note that you can also assign the script task *Result* to variables that should not be overwritten at all, such as IDs.

With the appropriate programming skills, you define rules in JavaScript to describe which combinations of events and data lead to predefined results.

In addition to rules, you need to define what happens when the input matches multiple rules or no rules (error handling).

Error handling

In addition to the rules, you must define what happens if input matches more than one rule or none (error handlers). Error handlers can be added to a script task according to the procedure for service tasks. To do this, press the *Add Error Event Handler* button in the opened *script task properties*.

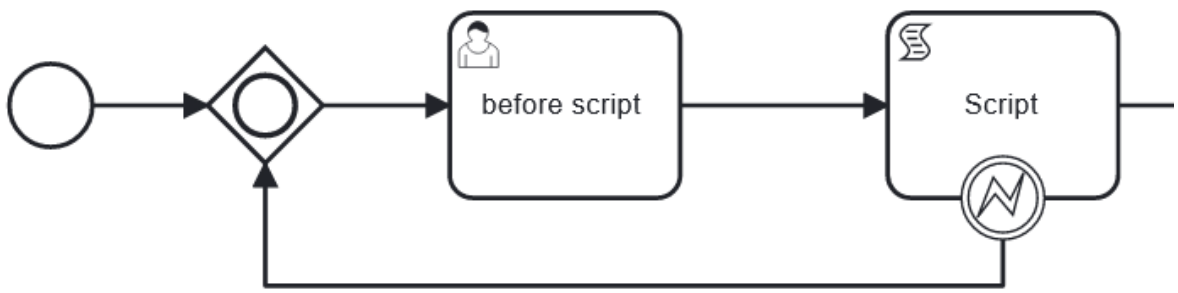


Table with examples of payload variable return values

Displayed payload variable / Unique name	Ungeparste Payload nach Abfrage durch Script Task

Assignee:
`ctx__assignee`
	`{`
	`name=Elena Employee,`
	`id=1119,`
	`login=elena.employee`
	`}`

Job-ID: `job__id`	`437`
Description: `description__`	`"String"`

Creator:
`creator__`
	`{`
	`"name":"Ron Swanson",`
	`"id":1120,`
	`"login":"ron.swanson"`
	`}`

Job status:-`job__state`	`ACTIVE`
Job type: `job__type__pseudo__variable`	`7877`
Current workflow step: `workflow__timing`	`{duration=null, dueDate=null, startDate=null}`
Job name: `job__name`	`Disney Summer Campaign 2023`
Workflow: `workflow__object__id`	`2829`

Return value of the script

Here you can specify a variable on the datasheet as target. This serves as a buffer for the result of the script task and can be evaluated elsewhere. It is sometimes necessary to create separate fields for this on the datasheet layout, perhaps hidden on their own tab. This way, users are not confused and are not tempted to briefly fill in such a standard field. These variables serve as a buffer for the `result` of the script task, which can then be evaluated later and used as a pre-filled parameter.

Convert numerical value to string

You have performed a calculation and want to write the result into a text variable. Only line 1 alone does nothing. The result is not written into the variable. It is only through the conversion in line 2 that the result now appears in the filled text variable in the next process step.

```
1    var num = (23*11*65);

2    var result = num.toString();
```

Convert string to floating-point number

The variables for `Total amount of purchase` and `Shipping` costs are created as text fields (Single inputline) in the datasheet, but contain numerical values. The conversion of the variable is quite simple.

```
1    var price = parseFloat(system__price);
```

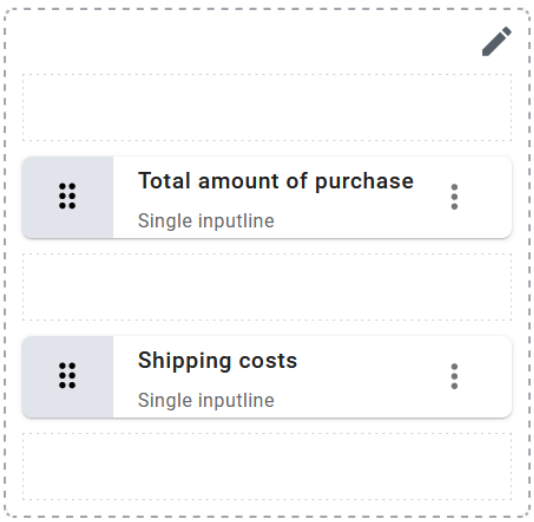
This allows you to perform calculations.

Note

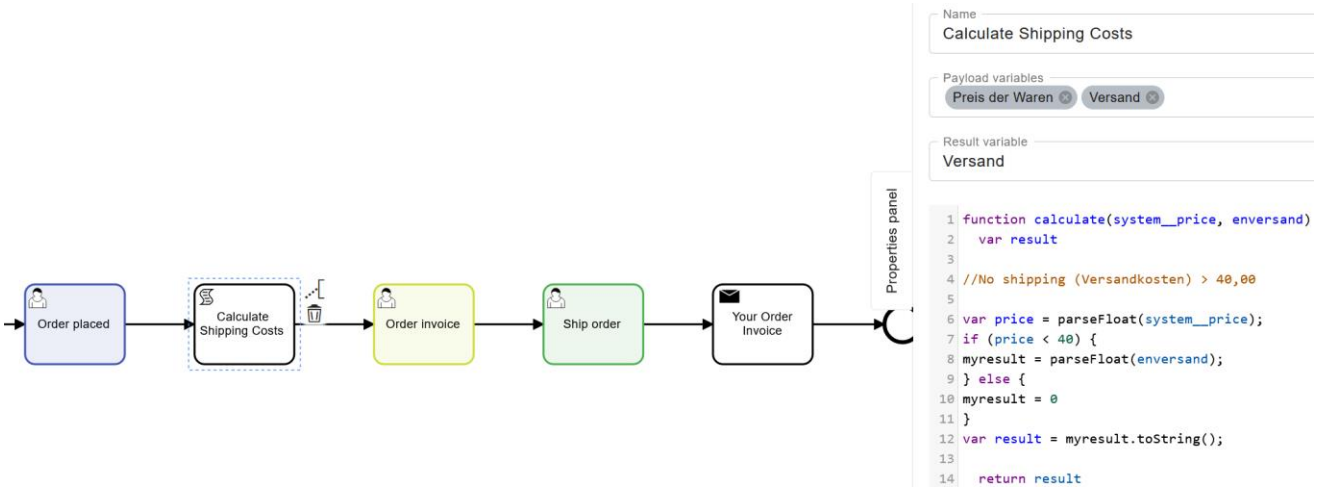
It is worth noting that only the last value declared in the script is returned. The content of `result` can only be assigned to one variable at a time. If you have any questions, please contact your BrandMaker contact person.

3.2.7.1 Process with script task

Our example workflow is to determine whether shipping costs are incurred for the value of goods in an order. The calculation is done via *Script Task*.



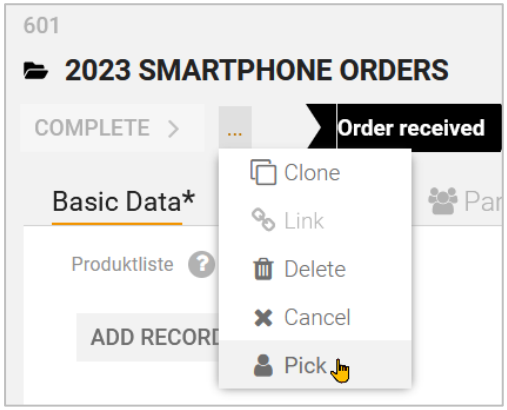
The default shipping costs are prefilled in the datasheet in the variable `Shipping costs`. Whether an invoice is issued without shipping costs is not difficult to calculate.



The Payload Variables (`Total amount of purchase, shipping costs`) are text variables and their content must be converted to a floating-point number. The condition is that no shipping costs are calculated starting from a goods value of €40.00. How much the shipping costs are below the limit is predefined in the `Shipping` variable. The script task evaluates the price of the goods from the order in the previous step, the condition for shipping costs is checked in an `if/else` statement. If there are none, the `Shipping Costs` variable is set to 0.

```
//No shipping (Shipping costs) > 40,00

var price = parseFloat(system__price);

if (price < 40) {

myresult = parseFloat(enversand);

} else {

myresult = 0

}

var result = myresult.toString();
```

An e-mail with the calculated shipping costs is sent as a preliminary invoice at the end of this application example.



Note

For workflow steps that are not permanently assigned to a person, but must be actively accepted by the editor, the value calculated by the script task in the previous intermediate step is only updated in the datasheet after the step has been accepted for editing. (Menu command: *Pick*).

3.2.8 Intermediate Event Throw Message

There are two types of intermediate events. An intermediate event in the sequence flow can either trigger an event or respond to the receipt of an event. Intermediate events in the sequence flow have exactly one incoming and one outgoing sequence flow.

Intermediate event messages are used to send messages from one process to another process and trigger an action there. After the event is triggered, the process continues.

3.2.9 Intermediate Event Catch Message

An intermediate event catch message is used in processes to catch messages from a parent process and trigger appropriate actions. The sequence is paused until the event occurs.

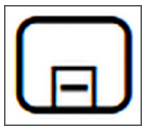
For example, the flow is interrupted until a message is received. The process continues after the event has occurred.

3.2.10 Sub-Actions

Sub-actions are actions independent of the central BPMN workflow that are executed in parallel. Event-based subprocesses in the Workflow Modeler are identified by a dotted outline.

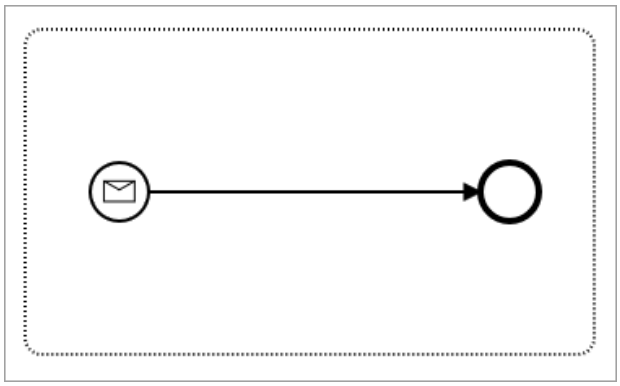
The event used to trigger one of these subprocesses is configured via a start event.

Create extended subprocess



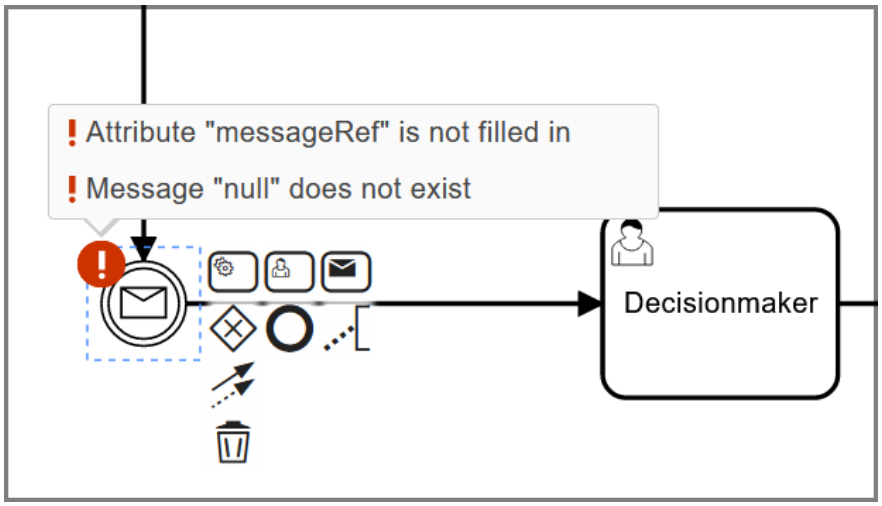
The extended subprocess always looks as shown below and always starts with a message start event. For example, such a process responds to MAPL update events.

Message events are used to send and receive messages between two separate processes. Message events serve as a mechanism for communication between processes.



After you created the subprocess, add an intermediate event catch message to the main process. The Workflow Modeler still reports errors at first as necessary attributes and referencing are missing.

After you create the subprocess, add an intermediate event catch message to the main process. The Workflow Modeler still reports errors at first. Necessary attributes and referencing are missing.



To resolve this issue, highlight the placed intermediate catch event message symbol and select the option `Mapl Update Event` in *the Intermediate Catch Event Properties* panel in the *Message* field.

This fixes the errors.

The following two workflow elements can be used to configure which actions are initiated after the deletion or cancellation of an associated job.

Message events are used for these use cases.

Create Subprocess „Cancel Message“

You create an event subprocess that responds to a `Cancel Message` type event. Example: A user has cancelled a job prematurely. You can specify what should happen when the job is cancelled.

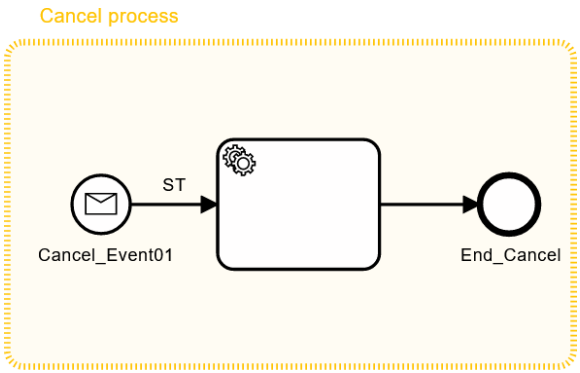
You can:

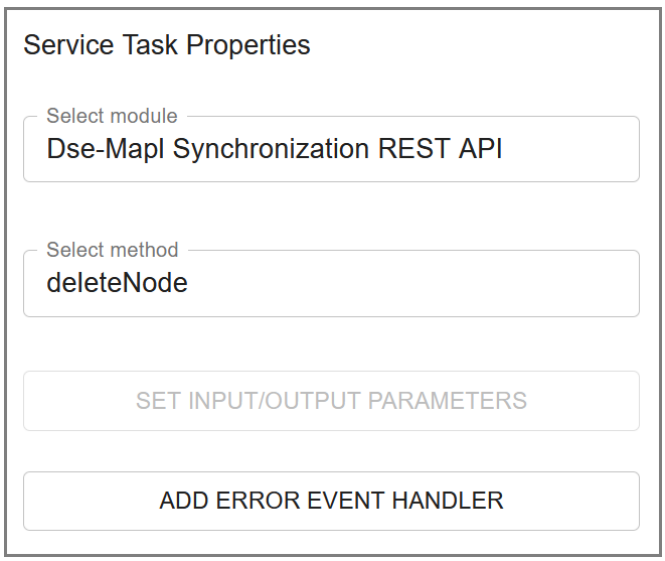
- Delete synchronized planning elements
- Cancel sub-processes

The planning element in Planner is to be removed by the event.

The following methods for the *MaplNodeSyncRestService* are available:

- deleteNode
- createNode
- moveNode
- updateNode





Adjust the properties in the service task of the sub-process accordingly. The method *deleteNode* is needed to delete the planning element in Planner on termination.

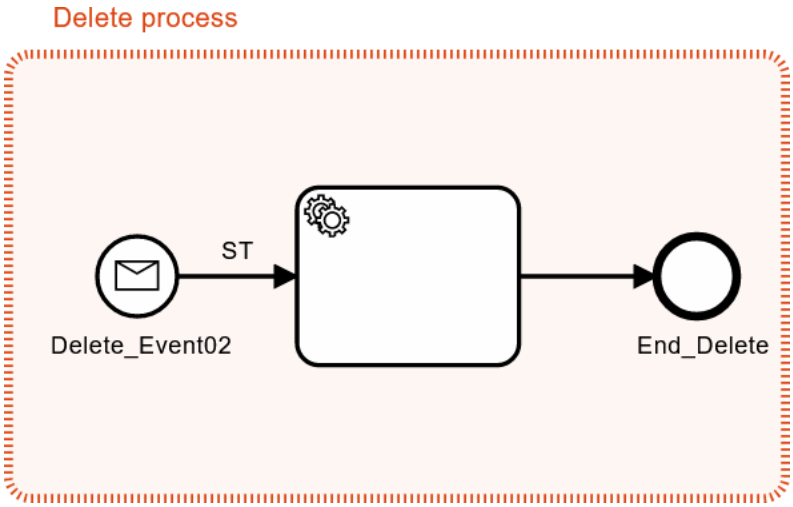
1. In the drop-down menu, select the module: *Dse-Mapl Synchronization REST API*.
2. From the drop-down menu, select *deleteNode*.
3. After you create and configure the subprocess, add an Intermediate Event Catch Message to the main process.

Create Subprocess "Delete Message"

You create an event subprocess that responds to an event of type `Delete Message`.
Example: A user has deleted a running job.

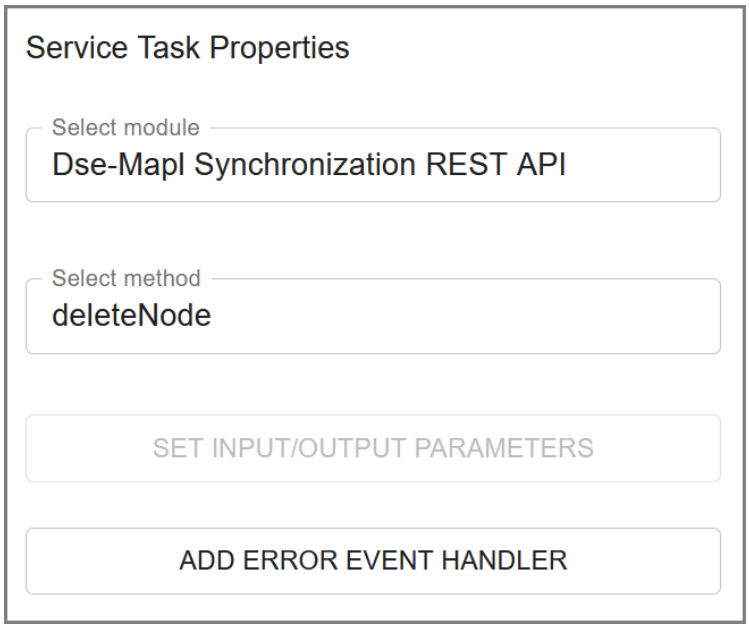
You can specify what happens after the job is deleted. For example, if the associated node in Planner is also to be deleted, you must create a corresponding service task.

Adjust the properties in the service task of the sub-process accordingly. Here, too, the method *deleteNode* is used to delete the planning element in Planner in the same way when it is cancelled.



Adjust the properties in the service task of the sub-process accordingly.

1. In the drop-down menu, select the module: *Dse-Mapl Synchronization REST API*.
2. From the drop-down menu, select *deleteNode*.
3. After you create and configure the subprocess, add an Intermediate Event Catch Message to the main process.



After publishing the changes, this job will be synchronized with the related elements in Planner. The planning element in the Planner is removed by an event with the above settings for the two subprocesses presented.

It is also possible to process several service tasks one after the other in one subprocess. In this way, it is also possible to delete sub-jobs as part of a subprocess.

3.2.11 Gateways

Gateways are used to control the process sequence: If the condition is true, the process takes the appropriate course. The available gateways are described below. For information about setting up conditions with gateways, see chapter 3.5.

Exclusive (OR)

All the conditions are evaluated, but only one condition can be true; therefore, only one sequence flow is performed. The conditions can be established using user decisions or *Number* or *Single select*-type variables.

Inclusive (AND)

All the conditions are evaluated, and anything from one condition to all the conditions can be true; each sequence flow with a true condition is performed. The conditions can be established using *Multiselect*-type variables.

Parallel

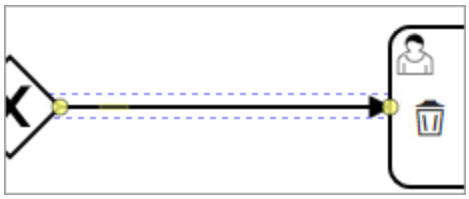
All the subsequent sequences are performed; conditions are therefore not required.

Parameters

You can enter a name for the gateways in the *Properties panel*. Other settings are based on the conditions that you establish. For more information, see chapter 3.5.

3.2.12 Sequence flow

A sequence flow connects tasks, gateways, and events and creates the process flow between the start and end element. Conditions for decisions at gateways are also described on the sequence flows.



Default sequence flow

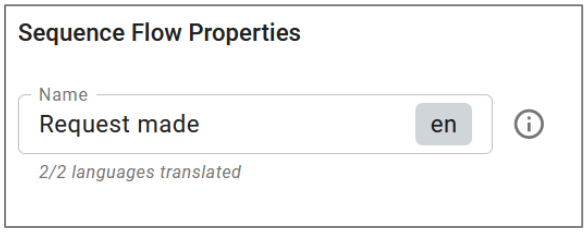
For sequence flows departing from a gateway, you have the option to specify a default sequence flow. The default sequence flow is executed if none of the other conditions are met.
How to set a default sequence flow is described in Chapter 3.3.2.

Parameters

In the case of sequence flows that connect exclusive and inclusive gateways with subsequent elements, you define the decisions in the *Properties panel*.

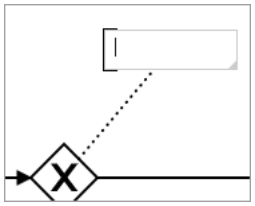
Customizable button name

You can customize the button names that appear in the open job datasheet in the Workflow Modeler. The name entered under *Sequence Flow Properties* will later be used as the button name. If no name is stored, the default name from the text resources appears.



3.2.13 Comments

You can use comments to add explanatory details about the elements in the process.



Parameters

You can also enter and modify comments in the properties dialog.

3.3 Creating a BPMN workflow

This chapter describes how you create BPMN workflows using the Workflow Modeler.

Attention!

You have to fix all formal errors before you can publish a newly created or modified BPMN workflow. Publishing with formal errors can permanently damage the job type.

3.3.1 Adding elements

You can add a start or end event, a user task, or a gateway in two ways:

Toolbar

1. Click the relevant icon in the toolbar.
2. Click the point in the drawing area at which you want to add the element.

The element is added.

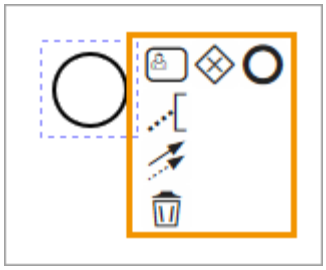
Adding related elements

You use this option when you wish to add elements that are directly related to one another.

Prerequisite: You have already added one element. This element must be before the element to be added in the process sequence.

1. Click the added element after which you wish to add a related element.

   A toolbar is displayed to the right of the element:



2. In the toolbox, click the symbol for the next element.

The selected element is added. It is connected to the previous element through a sequence flow.

3.3.2 Connecting elements

Once you have positioned elements, you must connect them using sequence flows. Please refer to the following sections, which explain how to connect using the toolbar or using elements that have already been inserted. In addition, you can define a default sequence flow for sequence flows that leave from a gateway, see the last section.

Toolbox

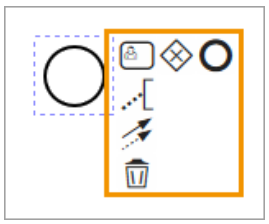
1. In the toolbox, click the symbol for sequence flows.
2. In the drawing area, click the element at which the sequence flow starts.
3. In the drawing area, click the element at which the sequence flow ends.

The sequence flow connects the elements.

On elements that have been added already

1. Click an element that you have already added to the drawing area.

   A toolbox is displayed to the right of the element:



2. Click the sequence flow symbol.
3. Click the element to which you want to connect the starting element.

The sequence flow connects the elements.

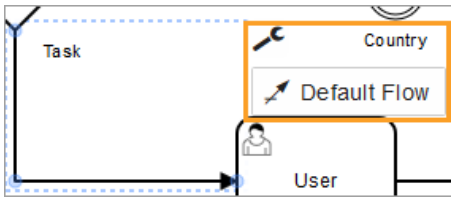
Set default sequence flow

> **Note**
>
> You can only set a default sequence flow that originates from a gateway.

1. Click the sequence flow you wish to set as default.

   The icon is displayed on the frame of the sequence flow.

2. Click > > *Default flow*.



You have set the sequence flow as the default sequence flow.

3.3.3 Positioning elements

To display all elements in a clear way, you can reposition individual elements of the diagram.

> **Note**
>
> If you move connected elements, the connections are not removed; instead, they are updated and adapted to the desired position.

Positioning an individual element

1. Click the element and hold the mouse button.
2. Drag the element to the desired position and release the mouse button.
3. Alternatively, you can fine-position a selected element or group using the keyboard with the four direction keys or move it in larger steps with SHIFT + direction key.

You have positioned the element.

Positioning multiple elements

1. Hold down CTRL + SHIFT / CMD + SHIFT and click the elements you want to move together.

   The elements are highlighted together.
2. Click one of the highlighted elements and hold down the mouse button.
3. Drag the elements to the desired position and release the mouse button.

You have positioned the elements.

Positioning an area

If you wish to position an area of the diagram containing several elements, proceed as follows.

1. In the toolbox, click the  symbol.
2. Move the mouse pointer over a free drawing area and hold down the mouse button to select the area that you wish to position.
3. and hold down the mouse button to select the area that you wish to position.
4. Click one of the highlighted elements and hold down the mouse button.
5. Drag the elements to the desired position and release the mouse button.

You have positioned the elements.

Adding/removing space

1. In the toolbox, click the symbol.
2. Move the mouse pointer over a free drawing area and hold down the mouse button to select the area that you wish to position.
3. Move the cursor horizontally or vertically.

> **Note**
>
> Depending on the use of the function, you can remove or add space only in one direction.

The space is added or removed.

3.3.4 Editing elements

To edit elements, you open the Properties dialog box and edit the properties there. To do so, click an element. The Properties dialog box is then displayed to the right of the drawing area. You may have to expand the dialog box.

Editing multiple user tasks

You can select multiple user tasks at the same time and assign them to a user or user group for editing.

1. Hold down the CTRL / CMD key and click the user tasks that you want to edit at the same time.

   You have selected the user tasks. The fields for the assignment are displayed in the Properties dialog box.

2. Assign the tasks. For information about performing the assignment, see chapter 3.2.4.

You have assigned multiple user tasks simultaneously.

Changing the gateway type

> **Note**
>
> If you have created conditions for a gateway, the options for changing the gateway type may be limited.

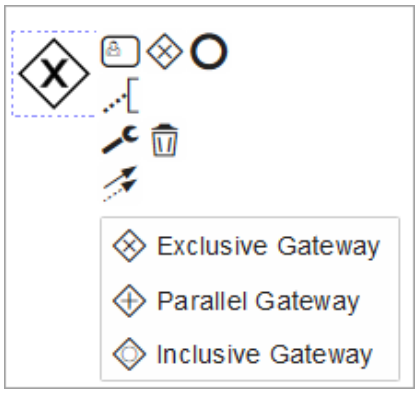
When you add a gateway, you always add an exclusive gateway to start with. To get a parallel or inclusive gateway, you have to change the gateway type.

1. Click the gateway whose type you want to change.

   A toolbar is displayed to the right of the element:

2. Click the symbol.

A pick list is displayed:



3. Select the type.

You have changed the gateway type.

3.3.5 Editing a view

You can adjust the image section with the three zoom buttons above the BPMN.iO logo. Or you can use the following key combinations to adjust the zoom factor:

Windows/Mac key combination	Function
CTRL + 0 / CMD + 0	Reset zoom factor to default
CTRL + + / CMD + +	Zoom in
CTRL + - / CTRL + -	Zoom out
SHIFT + Scroll wheel	Scroll horizontally

Trackpad owners can use the pinch-zoom gesture to adjust the area with two fingers by moving them apart and bringing them together.

The zoom functions support the mouse scroll wheel. The current position of the mouse pointer over the workflow is considered. If you use such a mouse, hold down the CTRL / CMD key and scroll up to zoom in. Scroll in the opposite direction to zoom out.

Hold down the SHIFT key and use the scroll wheel to move the image section to the right and left. Without pressing another key, you can scroll up and down the visible area. Press the esc key to exit the selected BPMN tool. You can now click and hold directly in the white area of the workflow modeler until the mouse pointer turns into a hand, and freely move the image section. This method saves the detour through the toolbar. You do not need to select the hand tool separately.

Additional keyboard shortcuts for the Workflow Modeler

Windows/Mac key combination	Function
CTRL + Z / CMD + Z	Undo the last action
CTRL + Y / CMD + Y or SHIFT + CTRL + Z / SHIFT + CMD + Z	Redo the state as it was before pressing STRG + Z / CMD + Z
CTRL + A	Select all elements
E	Quickly switch to text mode to enter a name for the selected workflow item
H	Deselect the current tool and switch to the hand tool
L	Lasso tool
S	Space tool

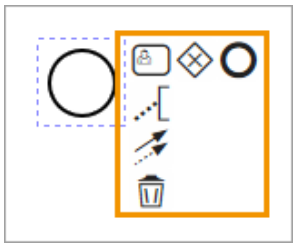
3.3.6 Deleting elements

> **Note**
>
> If you have rejected decisions for a gateway, the gateway can no longer be deleted. To delete the gateway, you must first remove the subsequent user tasks.

Deleting an individual element

1. Click the element.

A toolbox is displayed to the right of the element:



2. You can optionally click on the trash icon.
3. Or press the Del key or Backspace key on your keyboard to delete.

The element is deleted.

Deleting multiple elements

1. Hold down the CTRL/ CMD keys and click the items you want to delete.
2. Press the Del or Backspace key.

The elements are deleted.

Deleting the elements in an area

1. In the toolbox, click the symbol.
2. Click the drawing area and hold down the mouse button to select the area that you wish to delete.
3. Press the Del or Backspace key.

The elements are deleted.

Deleting all elements

4. In the Workflow Modeler, click the *Delete diagram* button in the bottom-right corner.

All elements are deleted. You must completely set up the whole workflow again.

3.4 E-Mail templates

3.4.1 Creating an e-mail template

1. In the ☰ menu of the datasheet editor, select *E-Mail Management* > *Custom Template E-Mail Management* > *Custom Templates*.
2. Enter a subject. The subject is used as the name for the English version and the template.
5. Enter the content for the English variant.
6. Click *Create template*.

The English variant is created.

7. Click on the German variant on the left.
8. Enter the German subject.
9. Enter the content for the German variant.
10. Click *Create template*.

You have created the e-mail template.

3.4.2 Editing an e-mail template

1. In the ☰ menu of the datasheet editor, select E-Mail Management > Custom Template E-Mail Management > Custom Templates.
2. Select the template you want to edit.
3. select the language variant you wish to edit.
4. Optional: Edit the subject.
5. Optional: Edit the content of the template.
6. Click *Save Changes*.
7. Optional: Repeat steps 3 to 6 for other language variants.

You have edited the e-mail template.

3.4.3 Deleting a German e-mail template

For technical reasons, you can only delete the German variant and existing other languages of an e-mail template. The en-US variant is the default and therefore has no Recycle Bin icon.

1. In the ☰ menu of the datasheet editor, select E-Mail Management > Custom Template E-Mail Management > Custom Templates.
2. Select the template whose German variant you want to delete.
3. Click the recycle bin icon for the German variant.

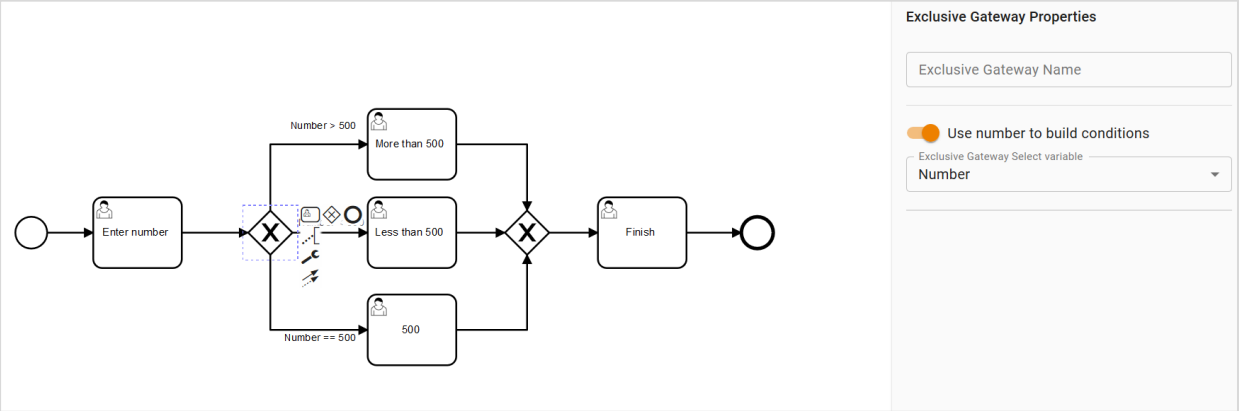
You have deleted the German version of the e-mail template.

3.5 Establishing conditions

This chapter explains how to establish conditions using exclusive or inclusive gateways.

3.5.1 Exclusive gateway: Using a number

You wish to establish a condition in which a number is evaluated, such as:



Prerequisites

- You have assigned a number variable to the type (see chapter 4.1.12).
- You have added an exclusive gateway to the workflow.

Setting up a condition

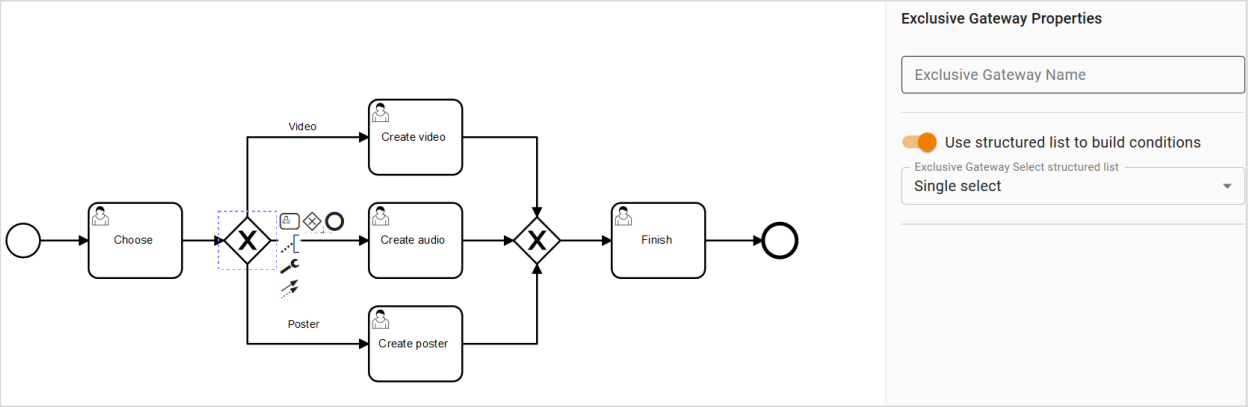
1. Click the gateway.
2. In the Exclusive Gateway *Properties panel* on the right, activate the function *Use number to build conditions*.
3. In the pick-list below that, select the variable that you want to use to create the condition.
4. Select all the outgoing sequence flows one after the other, and enter the operator and value of the number variables in the *Properties panel*.

> **Note**
>
> Negative numbers are permitted.
>
> If the number ranges for the various sequence flows overlap, the Workflow Modeler reports an error.

You have set up the condition.

3.5.2 Exclusive gateway: Using a custom structure

You want to establish a condition in which the selection is evaluated in a list. In this case, only one path is permitted to be executed.

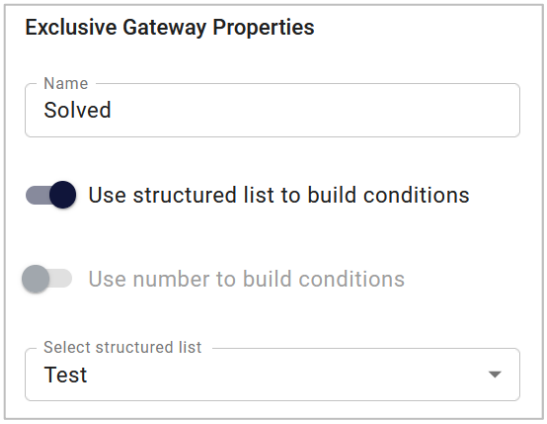


Prerequisites

- You have assigned a single select variable to the type (see chapter 4.1.14). A custom structure is assigned to the variable.
- You have added an exclusive gateway to the workflow.

Setting up a condition

1. Click the gateway.
2. In the Exclusive Gateway Properties panel on the right, activate the function *Use structured list to build conditions*.



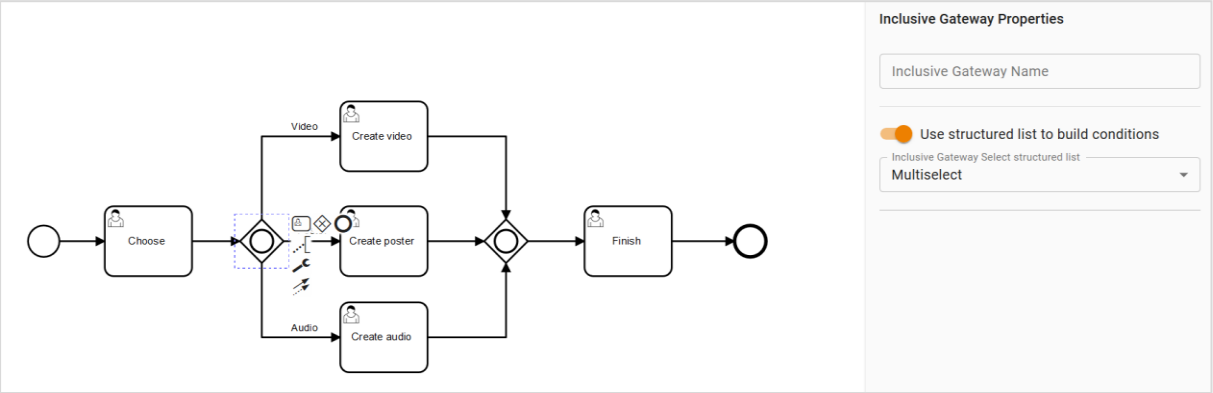
3. In the pick-list below that, select the variable that you want to use to create the condition. In our example above, the selected variable is called *Test*.
4. Select all the outgoing sequence flows one after the other and enter the list value with which the applicable path is to be executed in the *Properties panel*.

> **Note**
>
> If not all the list values are assigned to a sequence flow, the Workflow Modeler reports an error.

You have set up the condition.

3.5.3 Inclusive gateway: Using a custom structure

You want to establish a condition in which the selection is evaluated in a list. In this case, multiple paths are allowed to be executed.

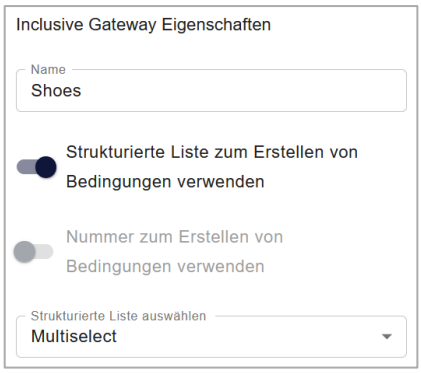


Prerequisites

- You have assigned a multiselect variable to the type (see chapter 4.1.14). A custom structure is assigned to the variable.
- You have added an inclusive gateway to the workflow.

Setting up a condition

1. Click the gateway.
2. In the Properties dialog box, activate the function *Use structured list to build conditions*.



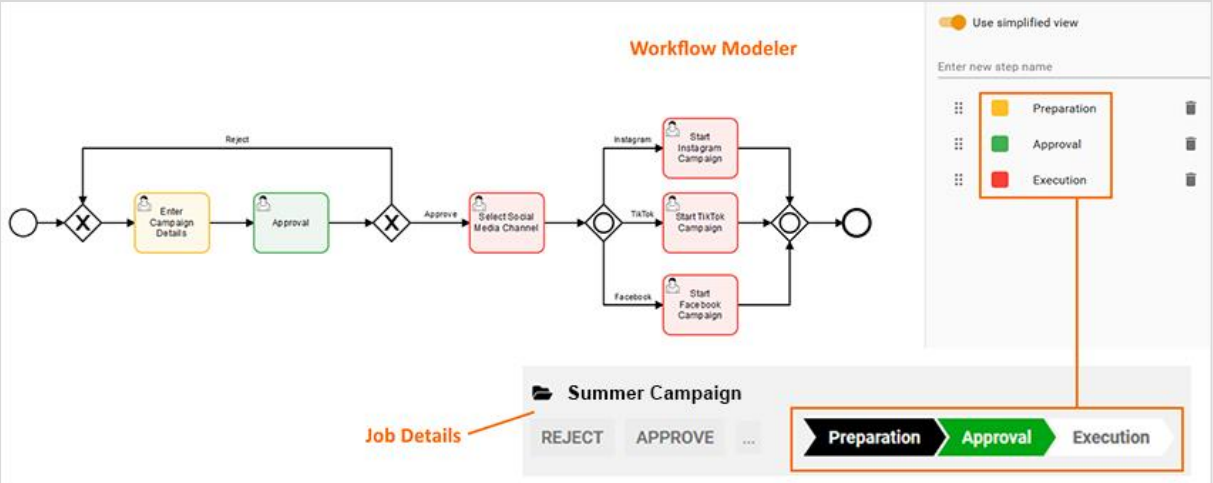
3. In the pick list below that, select the variable that you want to use to create the condition. In our example above, the selected variable is called *Multiselect*.
4. Select all the outgoing sequence flows one after the other and enter at least one list value with which the applicable path is to be executed in the Properties dialog box.

> **Note**
>
> If not all the list values are assigned to a sequence flow, the Workflow Modeler reports an error.

You have set up the condition.

3.6 Simplified view

You can use the simplified view to display the status of the job or data object in the workflow on the datasheet. In this case, you assign one or more user tasks to a process step. These process steps are displayed sequentially in the datasheet of a job or data object and show the progress made in the workflow:



> **Note**
>
> Processes are shown in the module's Kanban representation in read-only mode. This means that they cannot be dragged and dropped into another workflow step. The datasheet, on the other hand, can also be opened and edited in the Kanban view.

Prerequisite

- You have added at least one user task on the drawing area.

Configuring the simplified view

> **Note**
>
> If the simplified view is used, the Workflow Modeler checks if all steps are assigned to the view. If not all user tasks are assigned, the modeler reports an error.

1. Click on a free space in the drawing area.

The *Properties panel* for the workflow is displayed.

2. Activate the *Use simplified view* checkbox.
3. Enter a name for the workflow step in the *Enter new step name* field.
4. Press ⏎ or the Enter key.

The step has been created.

5. Optional: Click the colored field next to the step name and select a different color.
6. Select the tasks that you want to assign to the step.

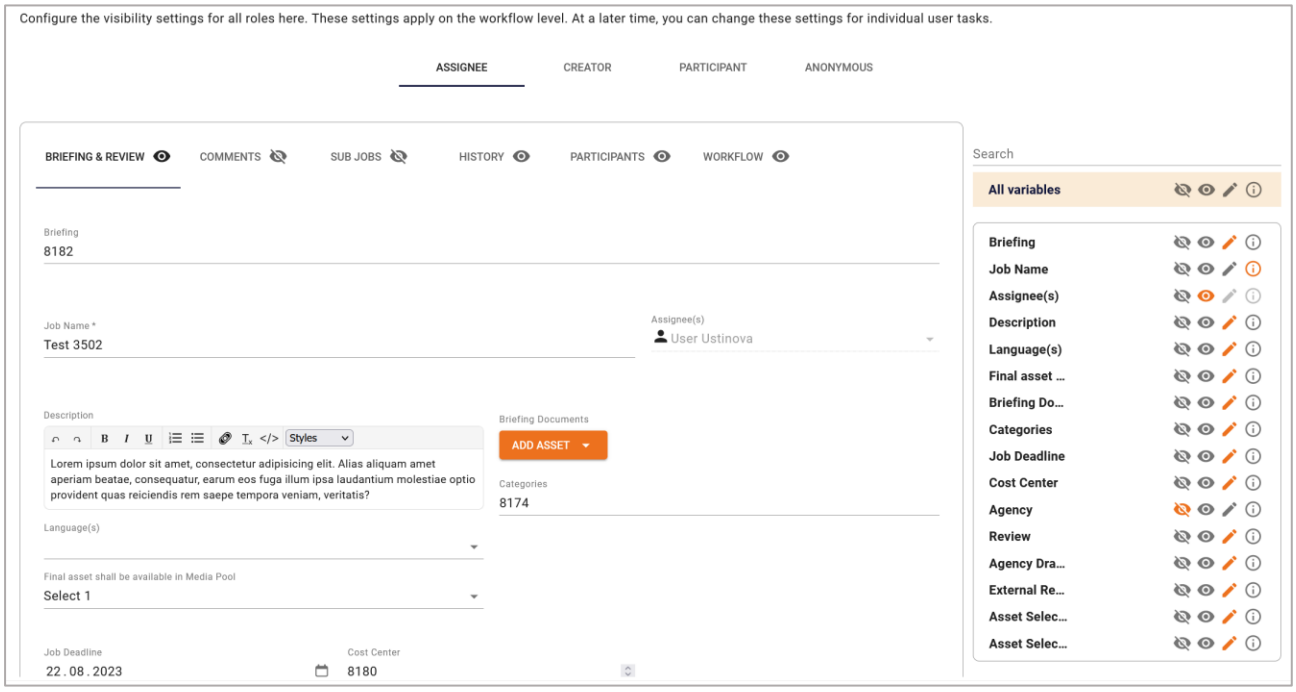
7. In the Properties panel, select the step in the *Select step from simplified view* pick-list.
8. Repeat steps 3 to 6 until all the required workflow steps are created and all the user tasks are assigned to a step.

You have set up the simplified view.

3.7 Setting up visibility and editability

You define the visibilities of tabs, the datasheet and single variables in the editor for the process. After calling up the page or clicking in an empty area, you select a workflow. You can also select a user task. Click the button next to the *Workflow Properties* or *User Task Properties* to open the Access Rights dialog.

Depending on what you activate, you either set up the visibilities for the overall workflow or the corresponding user task.



At the top of the dialog, first select the user role for which you want to set visibility. Then activate the visibility of the datasheet tabs below. For the visible tabs, you can then specify for the variables whether they are visible, editable, or a required field.

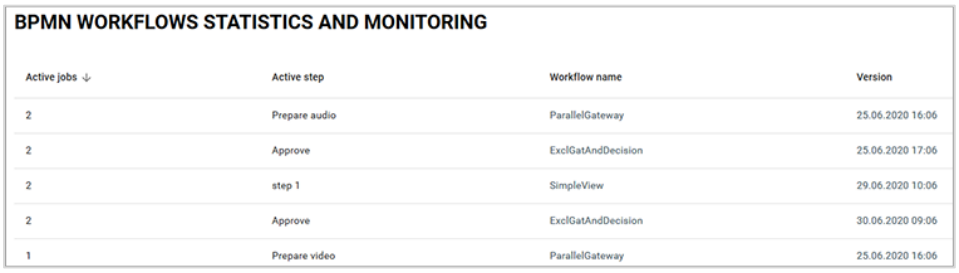
Mandatory fields to be filled in are marked with * in the Job Manager interface of the datasheet. These access rights can also be set for system variables.

When you set up the visibilities of a user task, you can also define a default tab for the datasheet. The default tab is displayed directly when you open the datasheet. To set a tab as default, click in the empty star icon ☆ next to the tab name. The new default tab gets a filled star ★.

Tip: If you are editing a process with several variables, you can search for variables in the right pane above the list of variables.

3.8 Monitoring

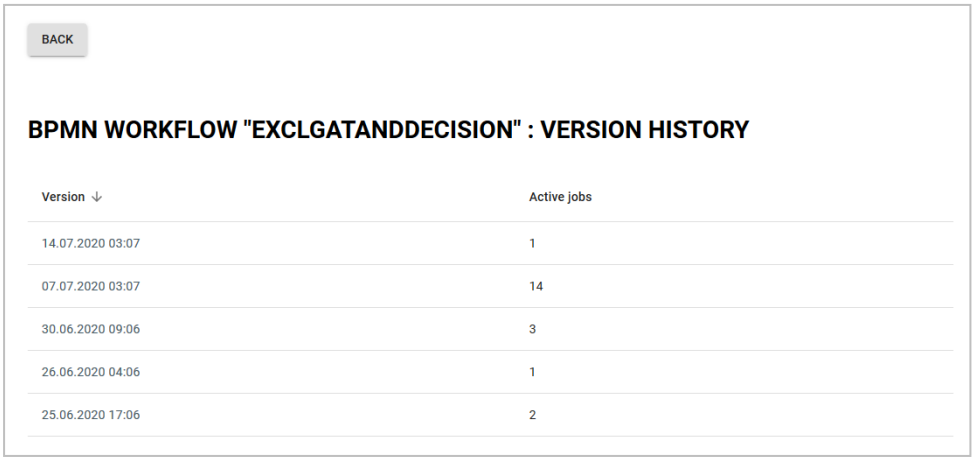
Administrators can access an overview of the running BPMN processes under *> Administration > Datasheet Engine > BPMN Monitoring*.



BPMN WORKFLOWS STATISTICS AND MONITORING

Active jobs	Active step	Workflow name	Version
2	Prepare audio	ParallelGateway	25.06.2020 16:06
2	Approve	ExclGatAndDecision	25.06.2020 17:06
2	step 1	SimpleView	29.06.2020 10:06
2	Approve	ExclGatAndDecision	30.06.2020 09:06
1	Prepare video	ParallelGateway	25.06.2020 16:06

The table lists the ongoing processes by amount and active step. It also lists the name and version of the type. When you click the type name, the version history of the workflow opens:

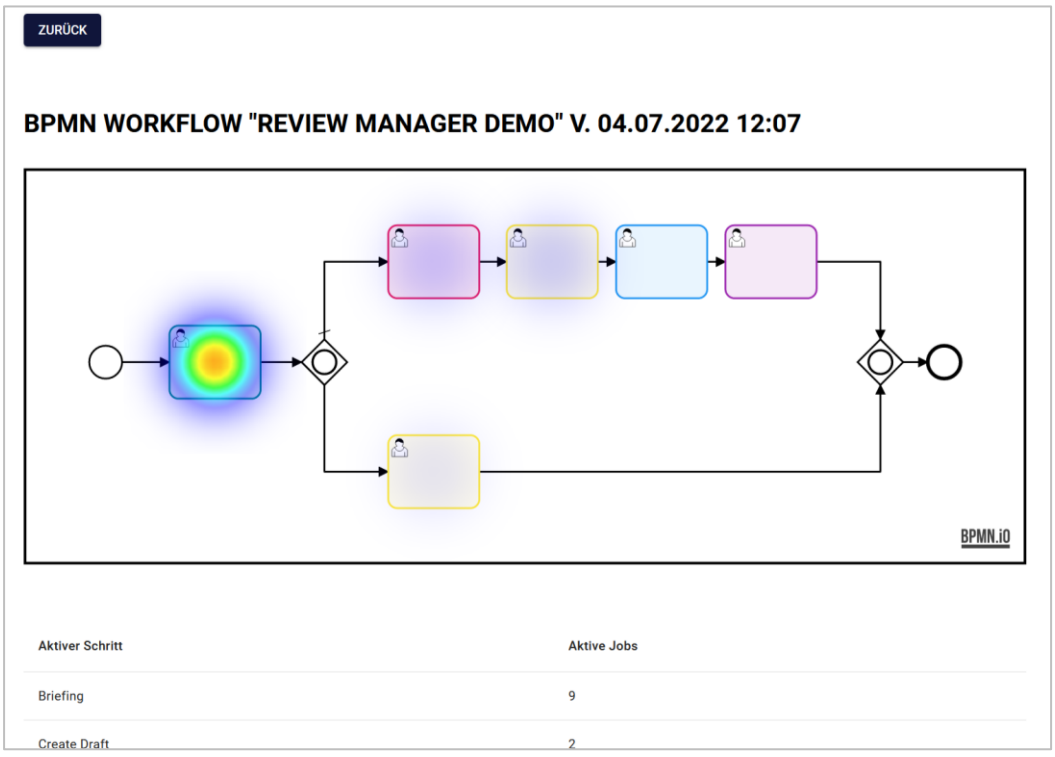


BACK

BPMN WORKFLOW "EXCLGATANDDECISION" : VERSION HISTORY

Version	Active jobs
14.07.2020 03:07	1
07.07.2020 03:07	14
30.06.2020 09:06	3
26.06.2020 04:06	1
25.06.2020 17:06	2

When you click the *Version* column, the workflow is displayed with a superimposed thermal image:



ZURÜCK

BPMN WORKFLOW "REVIEW MANAGER DEMO" V. 04.07.2022 12:07

BPMN.iO

Aktiver Schritt	Aktive Jobs
Briefing	9
Create Draft	2

The more intense the color saturation of the hotspots, the more processes are in the respective process step. The steps that do not have an active process at the current time are displayed without a color.

4 Available variables

Different variable types are used to allow you to group together and output data and information on a datasheet. The variables are placed on a datasheet using drag and drop.

There are two different types of variables:

- System variables that allow you to use the basic functions, for example, for displaying the creator or the current workflow step. When you create a type, system variables are created and – if absolutely necessary – placed on the datasheet automatically.
- Custom variables (text fields, selection fields, or an asset selector, for example) can be created as required.

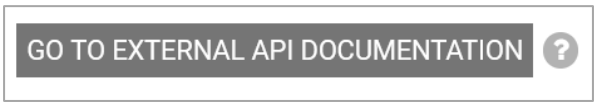
Note: When you create a variable, fields that are mandatory are flagged with a * on the interface. Pre-filling means that users see already filled values in the datasheet but are allowed to overwrite the content.

4.1 Custom variables

4.1.1 Action button

With the variable named *Action button,* you integrate a button on a datasheet that opens a URL. The processor of the datasheet can use the button to access programming that is generally customer-specific. Use the button *Close* at the bottom of the page or click in the Close box × at the top right to navigate back to Job Manager.

Name	Description
Unique name, technical name	**Note:** This is visible only if the variable is changed. None of the fields are visible when you create the variable. For details, see chapter 1.1.
Display Name	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. For details, see chapter 1.1.
URL	URL that is opened when you click the button.
Help text	Define the help text that can be displayed to users.



4.1.2 Advanced Grid

Use an *Advanced Grid* to place information or input fields clearly on the datasheet. The variables selected for the advanced grid are displayed as the grid columns. You can group the following variable types together in an advanced grid:

- *Single inputline*
- *Date picker*
- *Single select*
- *Reference/Relation*
- *Relation property*: Use a *Relation* to connect a specific variable from a different datasheet (an Asset Selector, for example).

  Note: Note that you can only add a *Relation* if you have already created a *Relation* variable in the table.

- *Organizational units*

  Note: You can only add the *Organizational units* variable type to the advanced grid once. If you add other variable types, the *Organizational units* entry is no longer displayed for selection.

- *Multiline input area*
- *Asset Selector*
- *Multiselect*
- *Combo box*
- *Action*: An action button can be integrated into an extended table. Via the button, users call customer-specific functions. The button is not displayed in a column like other variables, but the user reaches the button when a row is in edit mode. Clicking the button invokes a URL stored in the variable.

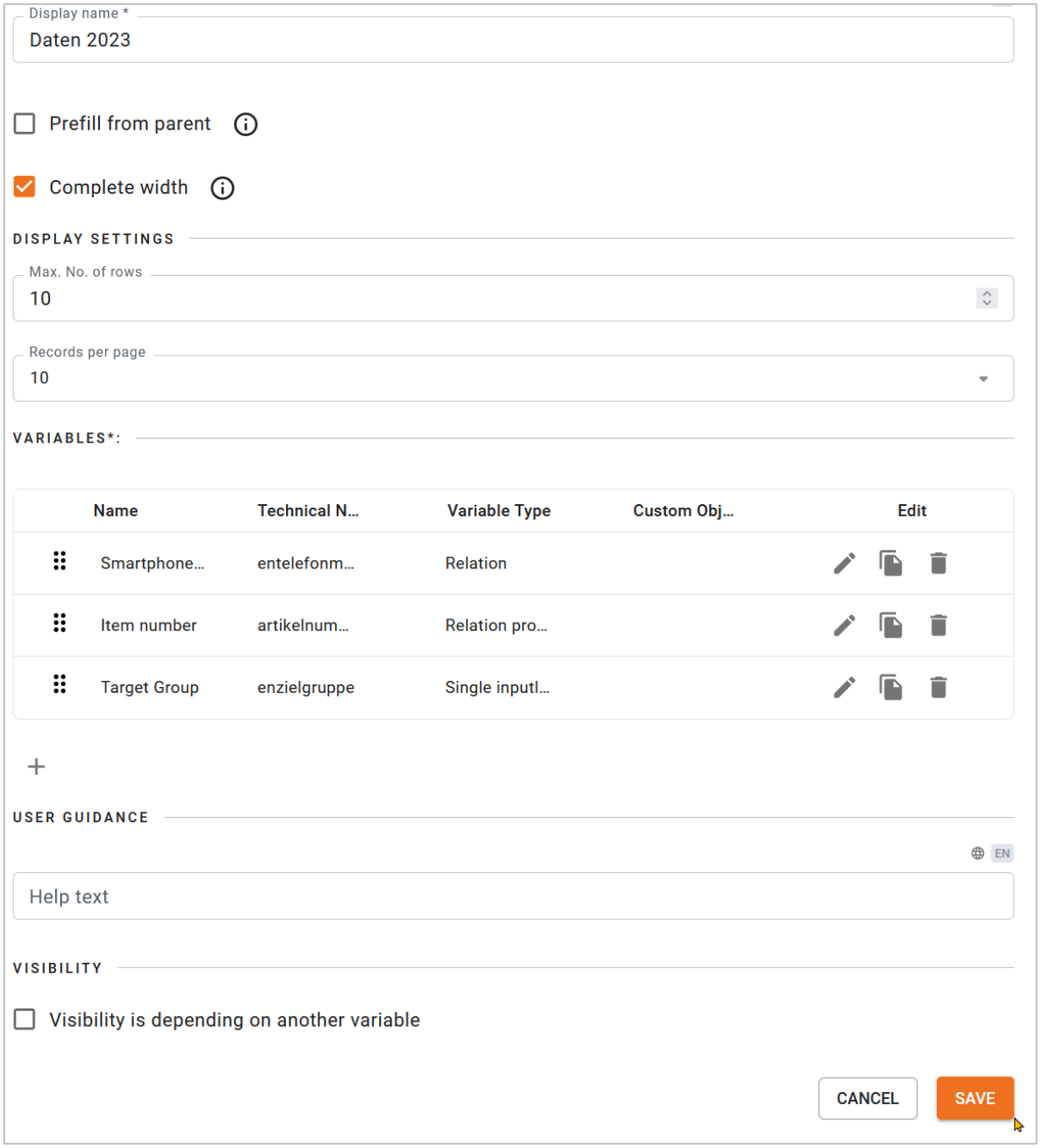
Each variable type can be configured as a mandatory field, which means that the user must edit the field when populating the table. To accomplish this, set the checkbox at *Mandatory field* when creating.

The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. None of the fields are visible when you create the variable. See Display, technical, and unique name, page 9. **Note:** If you want the values of a parent table to be inherited by the table, the technical name of the parent and child tables must be identical.

Name	Description
*Display name **	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.
Max. No. of rows	Define how many maximum rows the table can contain.
Table height	Enter the table height in pixels.
Complete width	Activate this checkbox to utilize the entire width of the datasheet. **Note:** If this checkbox is activated, the variable can only be placed on a one-column datasheet layout.
Records per page	Define with how many records per page the table is paginated: 10, 20, 50 or 100 records per page or without pagination. **Note:** Note that the performance for displaying large tables with pagination improves considerably.
Prefill from parent	Activate this checkbox if you want the table to be filled by a table in a higher-level job or data object. The user can edit the table in the child job or child data object and delete and add data records. The user can also refresh the data filled by the parent element. All changes are lost in this case. **Note:** For a successful inheritance, the parent and child tables must have an identical technical name.
Help text	Define the help text that can be displayed to users.
Shared value	**Note:** This is visible only if the type is a data object type and localization is enabled (see Localization, page 14). **Note:** Note that the variable can be selected as language-neutral only while you create it. The *Shared value* field cannot be changed at a later stage! Activate the checkbox if you want the variable value to receive the same value in each localized data object version.
	You can add an additional variable to the grid.

Name	Description
Variables	Your selected variables are listed in the *Variables* area. You can: • Change the order of the variables. • Open a variable for editing. • Copy a variable. • Delete a variable and remove it from the table.



4.1.3 Asset Selector

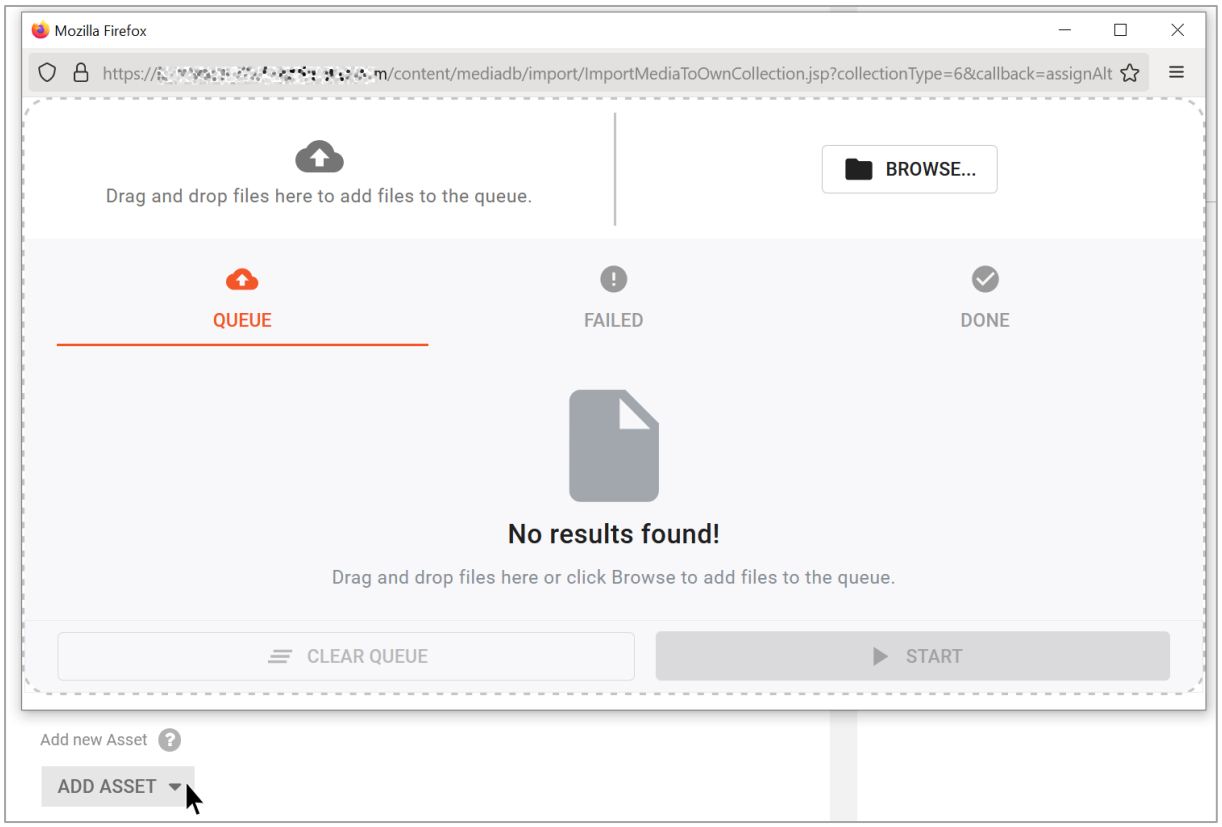
Use an *Asset Selector* to make the buttons *Upload new assets* and *Select from Media Pool* available on a datasheet. You can use the *Upload new assets* function to add both locally saved files and assets from a collection to the datasheet. You can also specify that assets are added to a datasheet automatically based on specific criteria.

Note: the assets that can be selected and edited are determined by the permissions in the user role.

The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. None of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
*Display name**	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.
Type spanning	Select this checkbox to display variables with an identical technical name and identical variable type in one column in the filter view. See Grouping variables, page 38.
Complete width	Checkbox, activate to utilize the entire width of the datasheet.
Help text	Define the help text that can be displayed to users.
Shared value	**Note:** This is visible only if the type is a data object type and localization is enabled (see Localization, page 14). **Note:** Note that the variable can be selected as language-neutral only while you create it. The *Shared value* field cannot be changed at a later stage! Activate the checkbox if you want the variable value to receive the same value in each localized data object version.
Inherit from parent	**Note:** This is visible only if the type permits inheritance (*Inheritance* dropdown list = `Multiple parents`). Activate this checkbox if you want the data record to adopt the values of a parent job or data object as a sub-job or sub data object. **Note:** The variables must have identical display names.

Name	Description
Prefilling from custom structure	Select a custom structure and a created attribute. By choosing *Key*, you can configure the display of an asset based on the affiliate ID. You can use the option *Override with* to link the display of an asset to a single selection that is linked to the same custom structure: in the single selection, you choose one of the custom objects (for instance, the data object). The text field is linked to the attribute *Product image*. If a different data object is set in the single selection, the corresponding data object image is automatically displayed in the text field.
Select media automatically	Activate this checkbox to add assets with the following properties to a datasheet automatically: • *Show all assets with the same 'Item Number'* • *Show all media that include ID in 'free text field'*: Specify the free text field in which the job ID or data object ID must be entered. **Note:** If one of these options is activated, users cannot select assets manually.
Free text field	**Note:** This can only be used when the checkbox *Show all media that include ID in 'free text field'* is activated. You can specify the free text field in which the job ID or data object ID must be entered.
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the variable only after a user has selected a specific value for a different selection field on the datasheet.
*Variable**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the selection field upon whose value you want to make the visibility dependent.
*Value**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected in order to display the variable.



4.1.4 Combo box

Use a *Combo box* to combine the features of a *Single select* and *Single inputline* with each other. In a combo box, users can:

- Select from values that are defined as a data source using a custom structure.

  Note: Stored values are proposed using Auto-Complete.

- Enter other values that are different from the stored values if required.

Prerequisites:

- You require a custom structure that you can select as the data source.

The following parameters are provided when you create or change the variables:

Name	Description
Unique Name *Technical Name*	**Note:** This is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
*Display name **	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.

Name	Description
Type spanning	Select this checkbox to display variables with an identical technical name and identical variable type in one column in the filter view. See Grouping variables, page 38.
*Data Source**	Select an existing custom structure as data source for the values of the selection field.
Help text	Define the help text that can be displayed to users.
Shared value	**Note:** This is visible only if the type is a data object type and localization is enabled (see Localization, page 14). **Note:** Note that the variable can be selected as language-neutral only while you create it. The *Shared value* field cannot be changed at a later stage! Activate the checkbox if you want the variable value to receive the same value in each localized data object version.
Inherit from parent	**Note:** This is visible only if the type permits inheritance (*Inheritance* dropdown list = `Multiple parents`). Activate this checkbox if you want the data record to adopt the values of a parent job or data object as a sub-job or sub data object. **Note:** The variables must have identical display names.
Default value	Enter the value to use as the default for the field. **Note:** Note that the default value is adopted only when you create an object (data object/job). Changing the default value does not have any effect on objects that have been created already.
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the selection field only after a user has selected a specific value for a different selection field on the datasheet.
*Variable**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the selection field upon whose value you want to make the visibility dependent.
*Value**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected to display the variable.


Enter
ZX 81
E10 Series
Q Series
ZX Series

4.1.5 Comfort Grid

Users can also use the comfort grid to enter data in a clear tabular form. When users fill in the grid on the datasheet, numerous keyboard shortcuts in particular make editing easier. Please refer to the user manuals for details.

4.1.5.1 Data Types

You can use the following data for a comfort grid:

- One-line text field
- Numbers
- Single select: selection of a value from a structured list.
- Date
- Formula, for details see chapter

Each variable type, except formula. can be configured as a mandatory field, i.e., the user must edit the field when filling the table. To achieve this, set the checkbox at *Mandatory field* when creating.

Numerical and date values are displayed and entered localized according to the selected UI language:

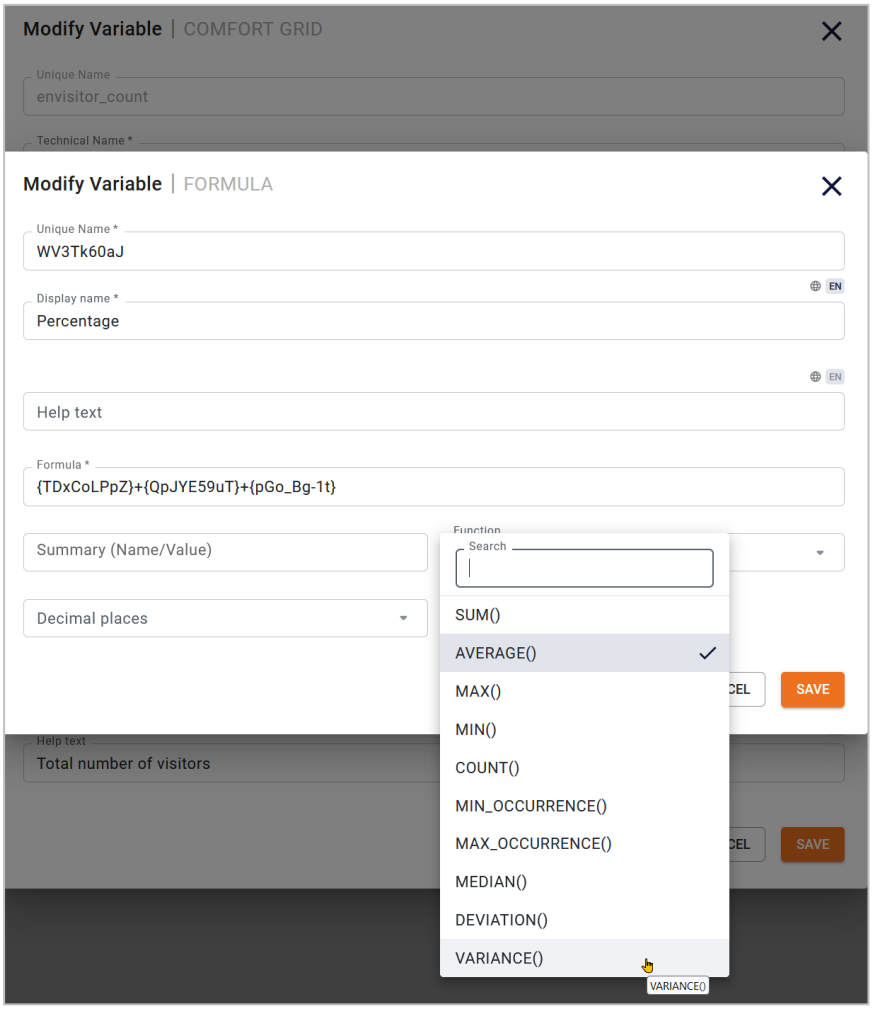
4.1.5.2 Summary row

You can set up a summary for each table variable in the footer of the table. To do this, enter a name and select one of the following functions:

- SUM(): Sum of all numbers in the column
- AVERAGE(): arithmetic mean of all numbers in the column
- MAX(): highest value in the column
- MIN(): lowest value in the column
- COUNT(): number of values in the column
- MIN_OCCURRENCE():Value that occurs least in the column.
  Example
  The column is a variable of type Single-Selection with the values A, B, C and D. The column contains 5 times A, 1 time B, 0 times C and 3 times D. The result in the footer is C (0).
  Comma-separated values are displayed if the result contains multiple values.
- MAX_OCCURRENCE(): Value that occurs most frequently in the column.
  Example
  The column is a variable of type Single-Selection with the values A, B, C and D.

The column contains 5 times A, 1 time B, 0 times C and 3 times D. The result in the footer is A (5).
If the result is not unique, several values are displayed, separated by commas.

- MEDIAN(): Median of the values in the column
- DEVIATION(): Deviation, rounded to two decimal places
- VARIANCE(): Variance, rounded to two decimal places



You can combine the functions with table variable types as follows

	Text	Date	Single Select	Number	Formula
SUM()				✓	✓
AVERAGE()				✓	✓
MAX()		✓		✓	✓
MIN()		✓		✓	✓
COUNT()	✓	✓	✓	✓	✓

	Text	Date	Single Select	Number	Formula
MIN_OCCURRENCE()		✓	✓	✓	✓
MAX_OCCURRENCE()		✓	✓	✓	✓
MEDIAN()				✓	✓
DEVIATION()				✓	✓
VARIANCE()				✓	✓

4.1.5.3 Formula variables

With a formula variable, you calculate values in a column according to an entered formula. The following operators and operands are available:

- +, -, /, * and (and)
- Numbers
- Other variables of the *Number* type within the table. You address the variables as follows: {*Name of the variable*}. Then the value of the variable is copied from the same row into the formula. Note Example 1 below.
- The following summaries for other columns. Note Example 2 below:
  - SUM(): sum of all numbers in the column
  - AVERAGE(): arithmetic mean of all numbers in the column
  - MAX(): highest value in the column
  - MIN(): lowest value in the column
  - COUNT(): number of values in the column
  - MEDIAN(): Median of the values in the column
  - DEVIATION(): Deviation, rounded to two decimal places
  - VARIANCE(): Variance, rounded to two decimal places
- Mathematical operations and constants according to the following Library; note Example 3 below.

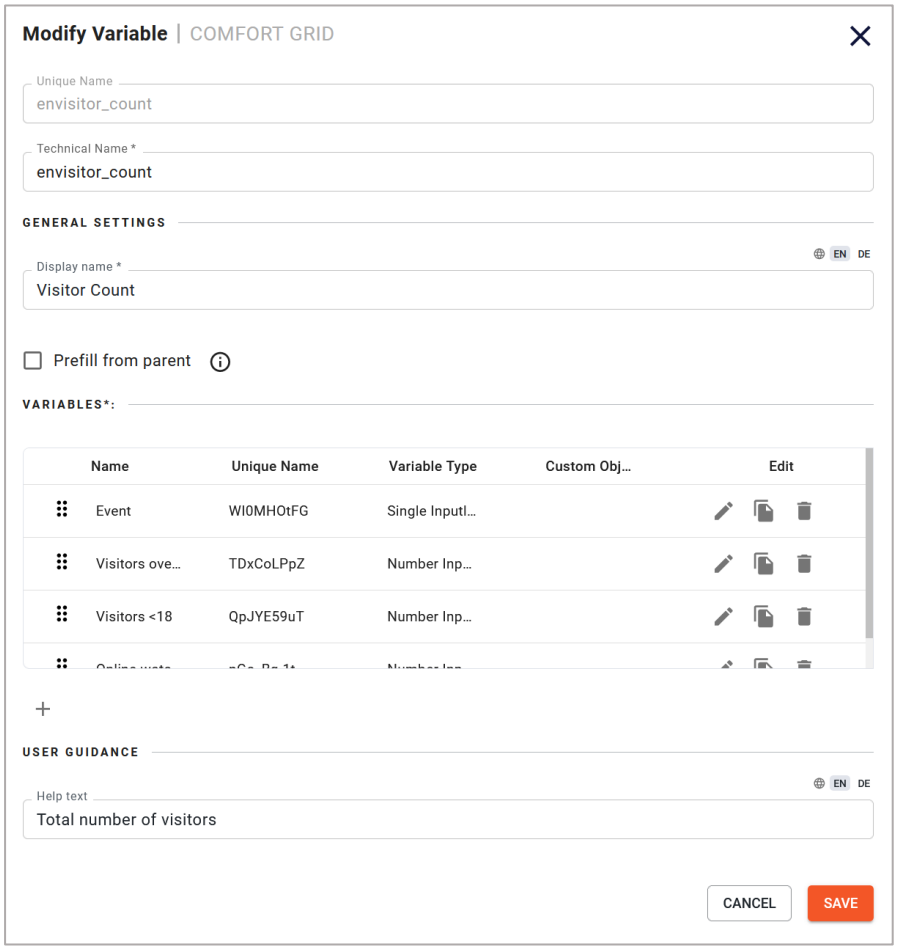
Example 1

A table is to be used to enter visitor data for events, categorized by visitors over 18 years of age, visitors under 18 years of age, and visitors who participated in the event online. For this, you need the average in the respective category over all events.
Additionally, you want to calculate the following values:

- Total number of visitors
- Percentage of online visitors

For these values, you also want to know the average across all events.

Implementation: Create a variable of type *Comfort Grid* on the job type datasheet. Add the following column variables to the table:

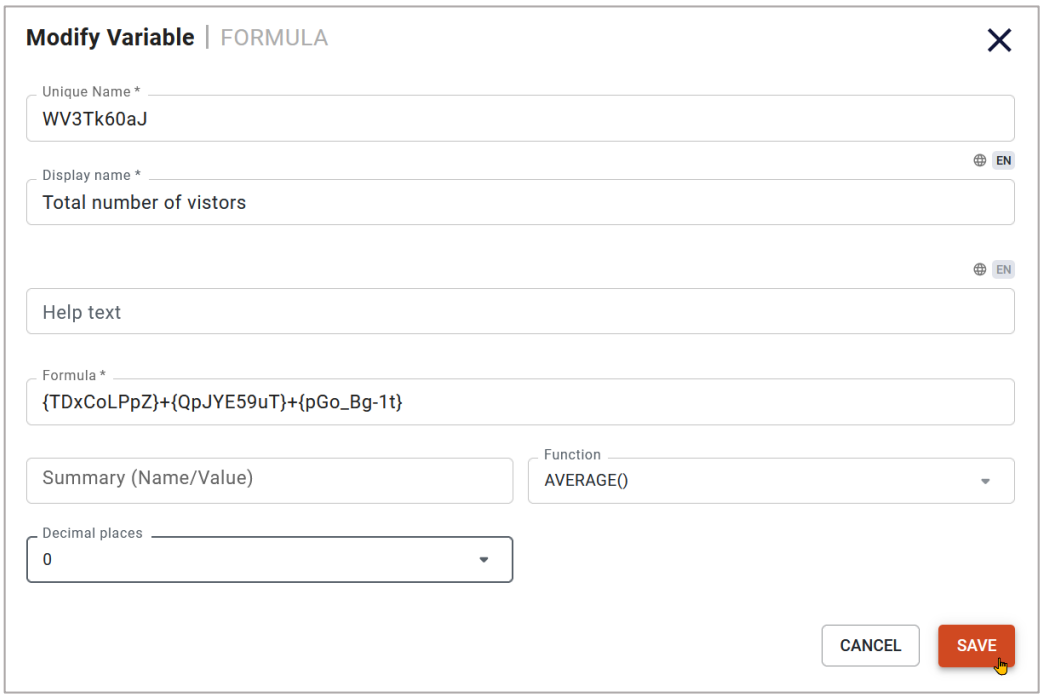


- Event: simple text field, summary function: COUNT() to get the total number of events.
- Visitors over 18: simple number field to enter the number of visitors over 18 years old, summary function: AVERAGE()
- Visitors U18: simple number field to enter the number of visitors under 18 years old, summary function: AVERAGE()
- Online Visitors: simple number field to enter the number of visitors who participated online; summary function: AVERAGE()

These fields are edited for each event by the user on the datasheet. To calculate the other values, create two formula variables in the comfort grid:

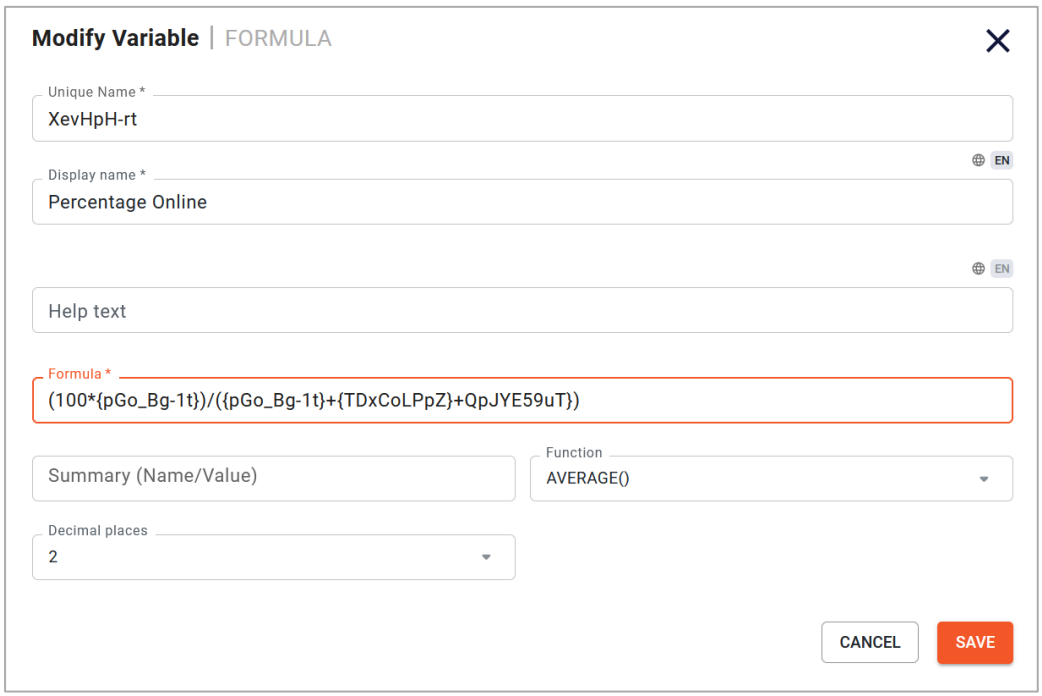
- Total number of visitors: The formula is
  {Visitors over 18}+{Visitors U18}+{Online Visitors}.
  However, you must store formulas with unique names (e.g.,: WI0MHOtFG) for the variables. To achieve this, first create a human-readable formula and then replace the names to avoid unintentional errors. There is no syntax check. If a formula is not correct, you will only recognize this when created jobs cannot be opened.
  The formula sums up the entered values in the number fields for each event in the

respective line. Note that the unique names in the curly brackets must correspond to the names of the number field variables described above.

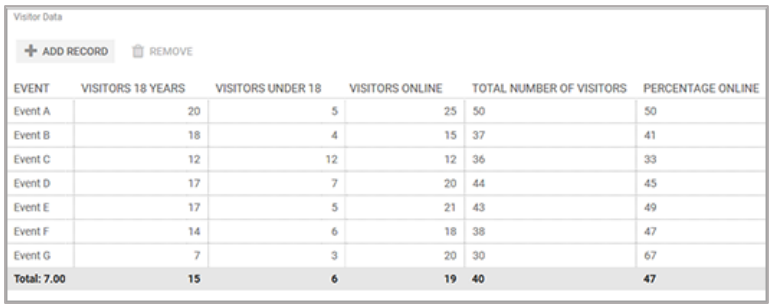


- Share Online: The formula is (100*{Visitors Online})/({Visitors Online}+{Visitors O18}+{Visitors U18}).

  Again, replace all names in curly brackets with the unique names.



The following screenshot displays how a correspondingly filled table is displayed on the job datasheet.


Visitor Data

ADD RECORD REMOVE

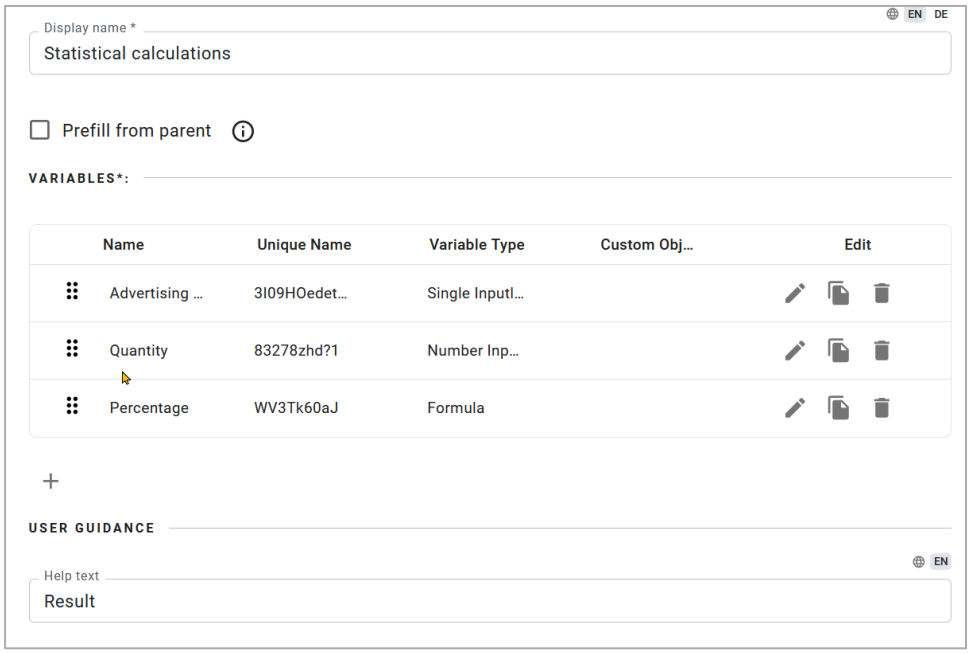
EVENT	VISITORS 18 YEARS	VISITORS UNDER 18	VISITORS ONLINE	TOTAL NUMBER OF VISITORS	PERCENTAGE ONLINE
Event A	20	5	25	50	50
Event B	18	4	15	37	41
Event C	12	12	12	36	33
Event D	17	7	20	44	45
Event E	17	5	21	43	49
Event F	14	6	18	38	47
Event G	7	3	20	30	67
Total: 7.00	**15**	**6**	**19**	**40**	**47**

In this case, the user only fills in the first four columns. The last two columns are calculated automatically.

Example 2

You want to compile and evaluate the number of distributed advertising media in a table.

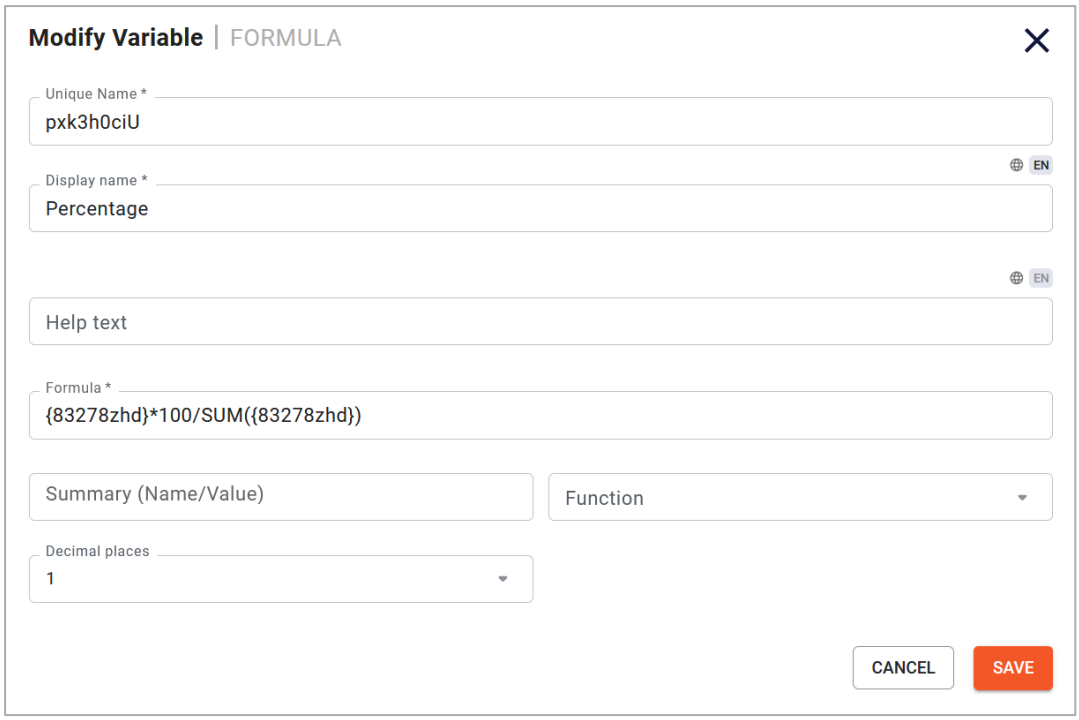
Implementation: Create a variable of type *Comfort Grid* on the job type datasheet. Add the following column variables to the table:



- Advertising media: Simple text field, without summary function
- Number: Simple number field to enter the number of ad media, summary function: SUM()

These fields are edited for each ad media by the user on the datasheet. To calculate the percentage, create a formula variable in the comfort grid:

- Ratio: The human-readable formula is {Number}*100/SUM({Number})
  Replace *Number* in curly brackets with the unique name. The formula takes the entered number of the ad media in the respective row and calculates the share of the sum of all values entered in the Number column. The result is given with one decimal place.



Example 3

With the following Library (external link) you can insert mathematical functions and constants in the formulas. You can insert the functions and constants into the formulas as follows:

- Example Pi: You want to multiply a value from the *Diameter* column by Pi. The entry in the formula line is: Math.PI{Diameter}
- Example Powers: You wish to take the square root of the sum of two column values: Math.pow({column A}+{column B}, 0.5)

4.1.5.4 Grid Parameters

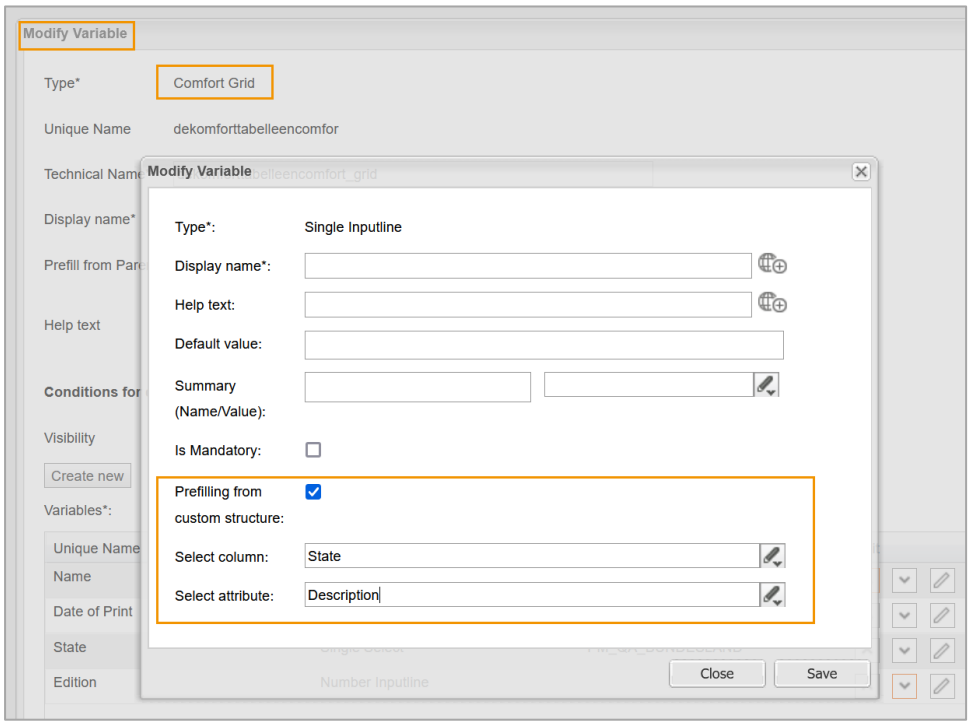
The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** this is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See chapter 1.1.
Display Name	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. You can create required language versions. See chapter 1.1.
Prefill from parent	Activate this checkbox if you want the table to be filled by a table in a higher-level job or data object. The user can edit the table in the child job or child data object and delete and add data records. The user can also refresh the data filled by the parent element. All changes are lost in this case. **Note:** For a successful inheritance, the parent and child tables must have an identical technical name.
Help text	Define the help text that can be displayed to users. You can create required language versions.
Create new	You can add an additional variable to the grid. Simple text fields, number fields and date fields have the option for *Pre-filling from custom structure* within the comfort grid. See above in the table under 4.1.14.
Variables	Your selected variables are listed in the *Variables* area. You can: • Change the order of the variables. • Open a variable for editing. • Copy a variable. • Delete a variable and remove it from the table. For a description of the parameters for the various table variables, see the following chapter 4.1.5.5.
Prefilling from custom structure	Activate the checkbox to be able to select already existing values of the type *Single select* here. After you select a column, you can select an attribute in the field below it. There, only attributes of type date, text, floating-point number and integer can be selected.

4.1.5.5 Grid Variables

Single inputline variable

Name	Description
Unique name *Technical Name*	**Note:** this is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See chapter 1.1.
Display Name	Define the name with which the variable is displayed in the table. You can create required language versions.
Help text	Enter the help text that can be displayed for a user. You can create required language versions.
Default value	Enter a default value.
Summary	Enter a designation in the left field. Select a function for the summary in the right field, see chapter 4.1.5.2.
Mandatory	Activate the checkbox if the user has to edit the column.
Prefilling from custom structure	Activate the checkbox to be able to select already existing values of the type “Single selection” here. After you select a column, you can select an attribute in the field below it. There only attributes of type date, text, floating-point number and integer can be selected.



Simple text fields, simple number fields and date selections can now be prefilled in a comfort grid.

Date picker variable

Name	Description
Unique name *Technical Name*	**Note:** this is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See chapter 1.1.
Display Name	Define the name with which the variable is displayed in the table. You can create required language versions.
Help text	Define the help text that can be displayed to users. You can create required language versions.
Summary	Enter a designation in the left field. Select a function for the summary in the right field, see chapter 4.1.5.2.
Mandatory	Activate the checkbox if the user has to edit the column.
Prefilling from custom structure	Activate the checkbox to be able to select already existing values of the type “Single selection” here. Select an existing custom structure and a created attribute whose values you want to use to prefill the date picker.

Single select variable

Name	Description
Unique name *Technical Name*	**Note:** this is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See chapter 1.1.
Display Name	Define the name with which the variable is displayed in the table. You can create required language versions.
Source	Choose a custom structure as source for the dropdown list.
Dependent on parent selection	**Note:** Only visible if a single selection has already been created in the table, which is linked to the custom structure superior to the data source. Activate the checkbox if the display of the selections in this field depends on the setting in another selection field, for example, the display of street names on the selection of a town in another selection field.
Parent selection	**Note:** This is visible only if the *Dependent on parent selection* checkbox is activated. Select the parent field containing the setting that the displayed values of the selection depend on.

Name	Description
Help text	Enter the help text that can be displayed for a user. You can create required language versions.
Default value	Select a default value.
Summary	Enter a designation in the left field. Select a function for the summary in the right field, see chapter 4.1.5.2.
Mandatory	Activate the checkbox if the user has to edit the column.

Single number field variable

Name	Description
Unique name *Technical Name*	**Note:** this is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See chapter 1.1.
Display Name	Define the name with which the variable is displayed in the table. You can create required language versions.
Help text	Define the help text that can be displayed to users. You can create required language versions.
Default value	Enter a default value.
Summary	Enter a designation in the left field. Select a function for the summary in the right field, see chapter 4.1.5.2.
Decimal places	Set the number of decimal places with which the data in the column will be displayed. Possible are 0 to 10 decimal places.
Mandatory	Activate the checkbox if the user has to edit the column.
Prefilling from custom structure	Activate the checkbox to be able to select already existing values of the type “Single selection” here. Select an existing modifiable structure and a created attribute with whose values the number field will be prepopulated.

Formula variable

Name	Description
Unique name *Technical Name*	**Note:** this is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See chapter 1.1.
Display Name	Define the name with which the variable is displayed in the table. You can create required language versions.
Help text	Define the help text that can be displayed to users. You can create required language versions.
Formula	Enter the formula.
Summary	Enter a designation in the left field. Select a function for the summary in the right field, see chapter 4.1.5.2.
Decimal places	Set the number of decimal places used to display the data in the column. Possible are 0 to 10 decimal places.

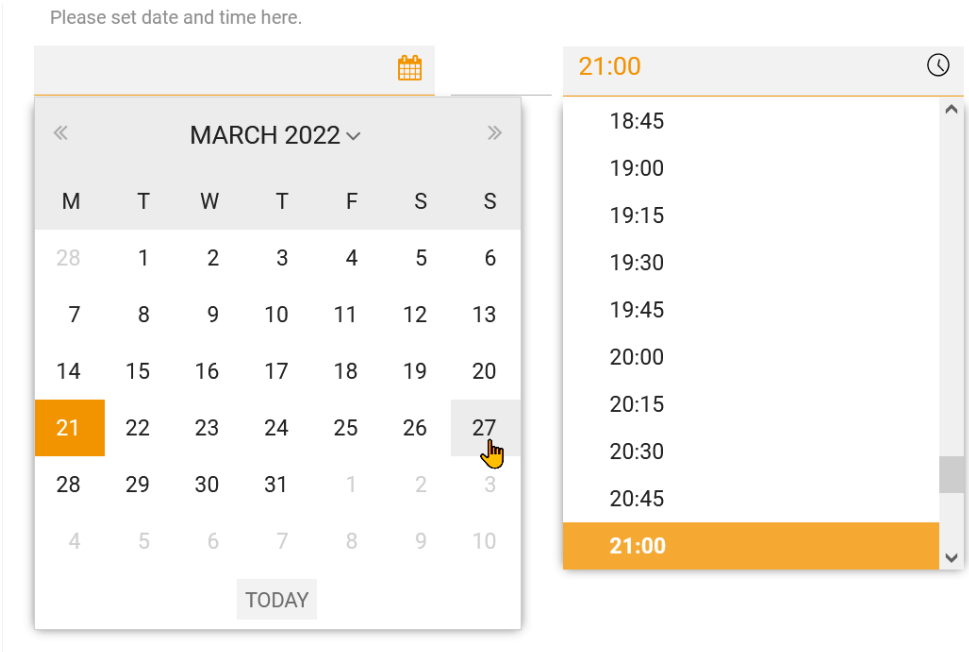
4.1.6 Datepicker with and without time

Use the *Datepicker* variable to create a date field with a date picker on the datasheet. The variable *Datepicker with time* also provides an additional field for the time.

The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
*Display name **	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.
For all types	Select this checkbox to display variables with an identical technical name and identical variable type in one column in the filter view. See Grouping variables, page 38.
Validity	Select the checkbox *Allow only valid dates* to define conditions for the date selection validity.
Number of offset days	**Note:** This is visible only if the *Validity* checkbox is activated. Specify the minimum number of days into the future the date must be in relation to the *Reference* date.
Reference	**Note:** This is visible only if the *Validity* checkbox is activated. Define the date to which the Number of offset days refers. Choose: • *Creation date* • *Current date* • *Transfer date from initial creation to any other workflow step*
Help text	Define the help text that can be displayed to users.
Shared value	**Note:** This is visible only if the type is a data object type and localization is enabled (see Localization, page 14). **Note:** Note that the variable can be selected as language-neutral only while you create it. The *Shared value* field cannot be changed at a later stage! Activate the checkbox if you want the variable value to receive the same value in each localized data object version.

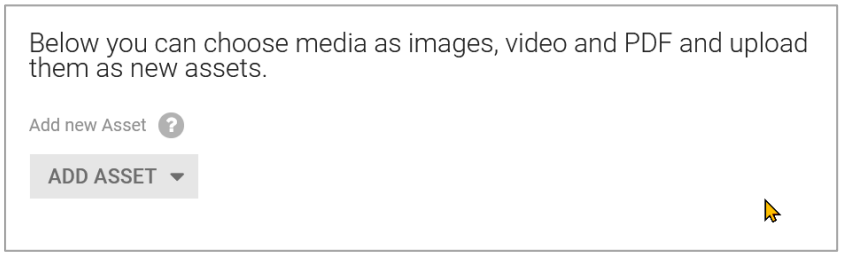
Name	Description
Inherit from parent	**Note:** This is visible only if the type permits inheritance (*Inheritance* dropdown list = `Multiple parents`). Activate this checkbox if you want the data record to adopt the values of a parent job or data object as a sub-job or sub data object. **Note:** The variables must have identical display names.
Prefilling from custom structure	**Note:** This option can only be used for the *Datepicker* variable. Select an existing custom structure and a created attribute whose values you want to use to prefill the date picker.



4.1.7 Description text

Use a *Description text* to enter additional information on a datasheet, for example.

Name	Description
Unique name, technical name	**Note:** This is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
Text	This displays the title used to display the headline text on the datasheet. You can create any language versions that are required. The text is used as the display name in the system; see Display, technical, and unique name, page 9.
Visibility	Activate the checkbox *Visibility is depending on another variable* to display the field only after a user has selected a specific value for a different selection field on the datasheet.
Variable	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the dropdown list upon whose value you want to make the visibility dependent.
*Value**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected in order to display the variable.



4.1.8 Document Selector

Use a *Document Selector* to make the *Select from Brand Template Builder* button available on a datasheet. Users can use this button to open a search in the *Brand Template Builder* module and add a document (finalized or still in progress) to the datasheet.

A document that is still in progress can be loaded to the document wizard for further processing directly from the datasheet. Users can open the detailed view of a finalized document and call additional functions (to adjust it to suit their own purposes, for example).

Note: Which documents in *Brand Template Builder* module can be selected and edited is determined by the permissions in the user role.

The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
*Display name**	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.
Type spanning	Select this checkbox to display variables with an identical technical name and identical variable type in one column in the filter view. See Grouping variables, page 38.
Help text	Define the help text that can be displayed to users.
Shared value	**Note:** This is visible only if the type is a data object type and localization is enabled (see Localization, page 14). **Note:** Note that the variable can be selected as language-neutral only while you create it. The *Shared value* field cannot be changed at a later stage! Activate the checkbox if you want the variable value to receive the same value in each localized data object version.
Inherit from parent	**Note:** This is visible only if the type permits inheritance (*Inheritance* dropdown list = `Multiple parents`). Activate this checkbox if you want the data record to adopt the values of a parent job or data object as a sub-job or sub data object. **Note:** The variables must have identical display names.

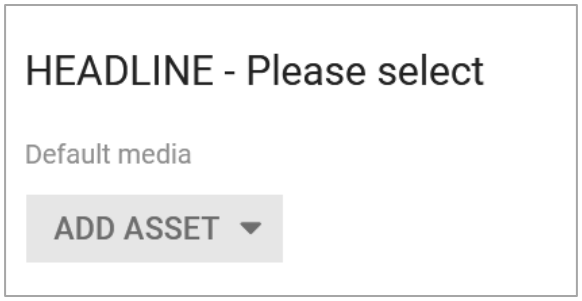
Name	Description
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the selection field only after a user has selected a specific value for a different selection field on the datasheet.
*Variable**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the selection field upon whose value you want to make the visibility dependent.
*Value**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected to display the variable.

4.1.9 Headline Text

Use a *Headline Text* to name the sections of a datasheet and structure the placed variables, for example.

The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. None of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
Text	This displays the title used to display the headline text on the datasheet. You can create any language versions that are required. The text is used as the display name in the system; see Display, technical, and unique name, page 9.
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the field only after a user has selected a specific value for a different selection field on the datasheet.
Variable	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the dropdown list upon whose value you want to make the visibility dependent.
Value	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected to display the variable.

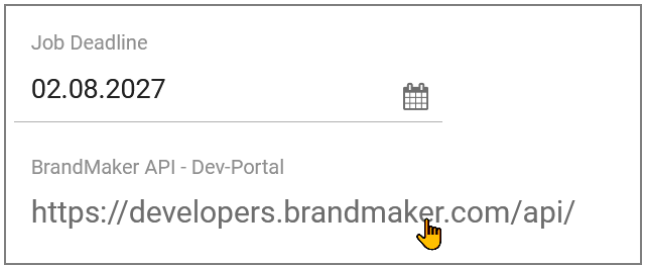


4.1.10 Link

You use a *Link* to create a reference to another page in the Marketing Efficiency Cloud (e.g., a Marketing Shop page) or an external website.

The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
*Display name **	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.
Type	You can create two different types of links: • *Relative*: This creates a link to an existing page in the Marketing Efficiency Cloud (for example, a Marketing Shop page. Do not enter the entire address as the link; instead, enter only the part that is relevant to the link. The “front” part of the link (the URL of your system) is generated and added automatically. • *Absolute*: This creates a link to an external web page.
Help text	Define the help text that can be displayed to users.
URL	Enter the address of the (web) page for which you want to create a link. **Note:** You must enter the full address for an external web page (including "http://" or "https://").
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the selection field only after a user has selected a specific value for a different selection field on the datasheet.
*Variable**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the selection field upon whose value you want to make the visibility dependent.
*Value**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected to display the variable.



4.1.11 Multiple values

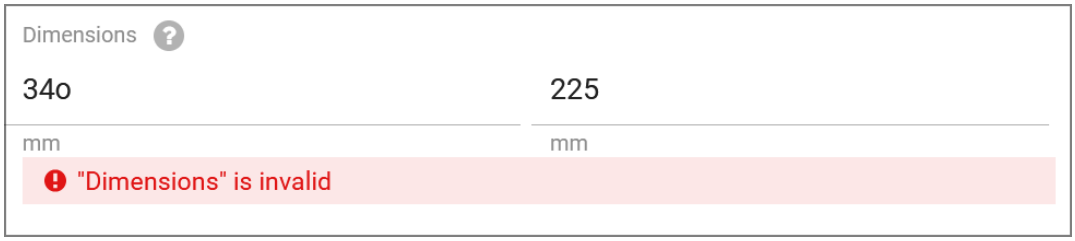
Use the variable *Multiple values input lines* to place a field for two related values (height and width, for example) next to each other on a datasheet.

The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. None of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
*Display name **	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.
Type spanning	Select this checkbox to display variables with an identical technical name and identical variable type in one column in the filter view. See Grouping variables, page 38.
Shared value	**Note:** This is visible only if the type is a data object type and localization is enabled (see Localization, page 14). **Note:** Note that the variable can be selected as language-neutral only while you create it. The *Shared value* field cannot be changed at a later stage! Activate the checkbox if you want the variable value to receive the same value in each localized data object version.
Percentage	Distribute the width of both input fields in percentage to the total width of the datasheet.
Max. Characters	Define the maximum number of characters that can be entered in the field.
Reg. Exp. Validator	Enter a regular expression in the field to check the entry for validity. For example, you can then ensure that bank codes or e-mail addresses have been entered in a specific format.

Name	Description
Input Size	Define the width of the variables. If so many characters are entered that cannot be displayed in the configured width, the field is provided with a scroll bar.
Suffix	You can enter a suffix (`Millimeter` or `%`, for example) that is attached to the field.
Help text	Define the help text that can be displayed to users.
Inherit from parent	**Note:** This is visible only if the type permits inheritance (*Inheritance* dropdown list = `Multiple parents`). Activate this checkbox if you want the data record to adopt the values of a parent job or data object as a sub-job or sub data object. **Note:** The variables must have identical display names.
Default value	Enter the value to use as the default for the field. **Note:** Note that the default value is adopted only when you create an object (data object/job). Changing the default value does not have any effect on objects that have already been created.
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the field only after a user has selected a specific value for a selection field on the datasheet.
*Variable**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the selection field upon whose value you want to make the visibility dependent.
*Value**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected to display the variable.

In this example, the input is invalid because the regex default, which only allows three-digit numbers, is not met and the lowercase letter “o” was entered instead of zero.



4.1.11.1 Examples of regular expressions

When creating or modifying the expressions, you can use an online tool such as https://regexr.com/ for testing. The following table provides you with some proven examples for a successful start. For further support, please visit the above website.

Name	Description
E-mail address	`^[\w.+-]{2,64}\@[\w.-]{2,249}\.[a-z]{2,6}$`
SWIFT BIC Code	`/^[A-Z]{6}[A-Z0-9]{2}([A-Z0-9]{3})?$/`
IBAN without spaces	`[a-zA-Z]{2}[0-9]{2}[a-zA-Z0-9]{4}[0-9]{7}([a-zA-Z0-9]?){0,16}`
Three-digit number	`\d{3}`
German zip code	`[0-9]{5}`
IPv4 address	`^(?:(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)\.){3}(?:25[0-5]|2[0-4][0-9]|[01]?[0-9][0-9]?)$`

4.1.12 Numbers

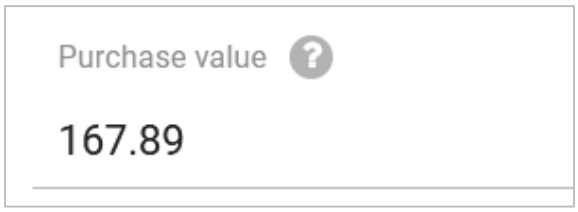
Use the *Numbers* variable to make an input field for numbers available to users on the datasheet.

The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
*Display name **	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.
Type spanning	Select this checkbox to display variables with an identical technical name and identical variable type in one column in the filter view. See Grouping variables, page 38.
Help text	Define the help text that can be displayed to users.
Use Unit	Select this checkbox to add a unit to the field.

Name	Description
Shared value	**Note:** This is visible only if the type is a data object type and localization is enabled (see Localization, page 14). **Note:** Note that the variable can be selected as language-neutral only while you create it. The *Shared value* field cannot be changed at a later stage! Activate the checkbox if you want the variable value to receive the same value in each localized data object version.
Unit of measure	**Note:** This is visible only if the *Use Unit* checkbox is activated. You can select whether a unit for *Length* or *Weight* is displayed.
Default unit	**Note:** This is visible only if the *Use Unit* checkbox is activated. You can define a default entry for the unit of measure, such as centimeter (cm) or kilogram (kg).
Inherit from parent	**Note:** This is visible only if the type permits inheritance (*Inheritance* dropdown list = `Multiple parents`). Activate this checkbox if you want the data record to adopt the values of a parent job or data object as a sub-job or sub data object. **Note:** The variables must have identical display names.
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the variable only after a user has selected a specific value for a different selection field on the datasheet.
Variable	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the dropdown list upon whose value you want to make the visibility dependent.
Value	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected to display the variable.

If the number field is enabled with the option *Use Unit*, it can also run conversions between units for you. For comparison, you can see a simple number field below.



4.1.13 Relation

Use a *Relation* to link datasheets to each other. For example, users select a job from a selection list. Users can then open the relevant datasheet directly from the displayed link.

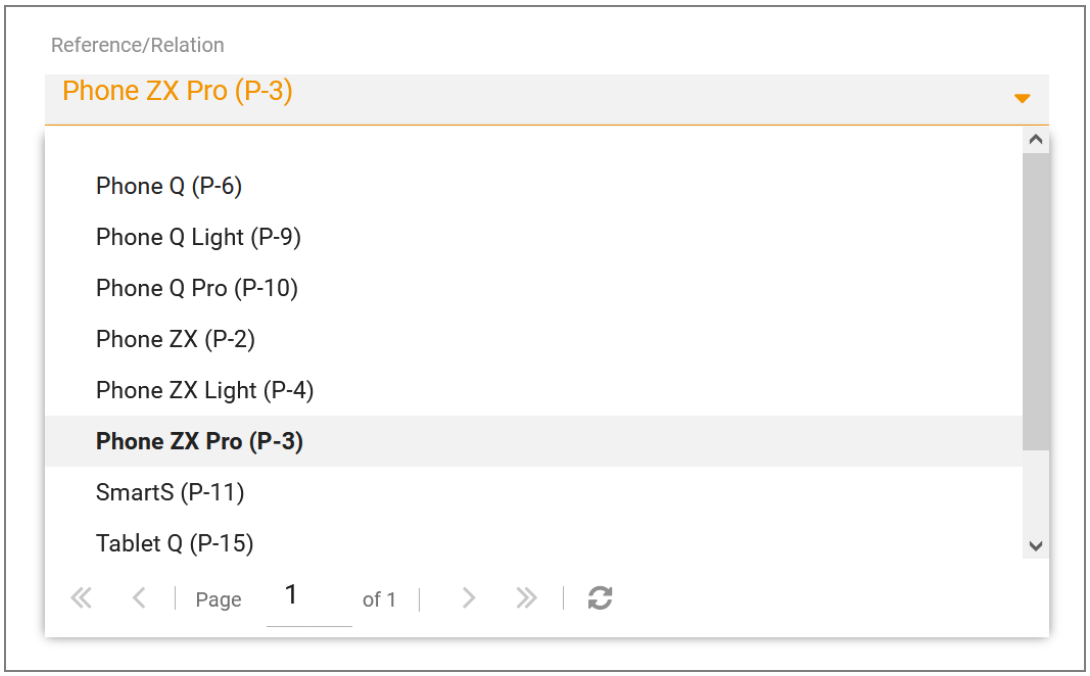
Prerequisites:

- You require at least one type that you can reference.

The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. None of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
*Display name **	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.
Type spanning	Select this checkbox to display variables with an identical technical name and identical variable type in one column in the filter view. See Grouping variables, page 38.
*Module**	This selects the module containing the types to which you want to make the reference.
Help text	Define the help text that can be displayed to users.
Shared value	**Note:** This is visible only if the type is a data object type and localization is enabled (see Localization, page 14). **Note:** Note that the variable can be selected as language-neutral only while you create it. The *Shared value* field cannot be changed at a later stage! Activate the checkbox if you want the variable value to receive the same value in each localized data object version.
Type	All types available based on the module you selected are displayed. You can select the types you want to reference.
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the variable only after a user has selected a specific value for a different selection field on the datasheet.

Name	Description
*Variable**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the dropdown list upon whose value you want to make the visibility dependent.
*Value**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected to display the variable.



4.1.14 Single inputline vs. Multiline input area

Use single inputline and/or multiline input area *text fields* to allow users to enter text on the datasheet.

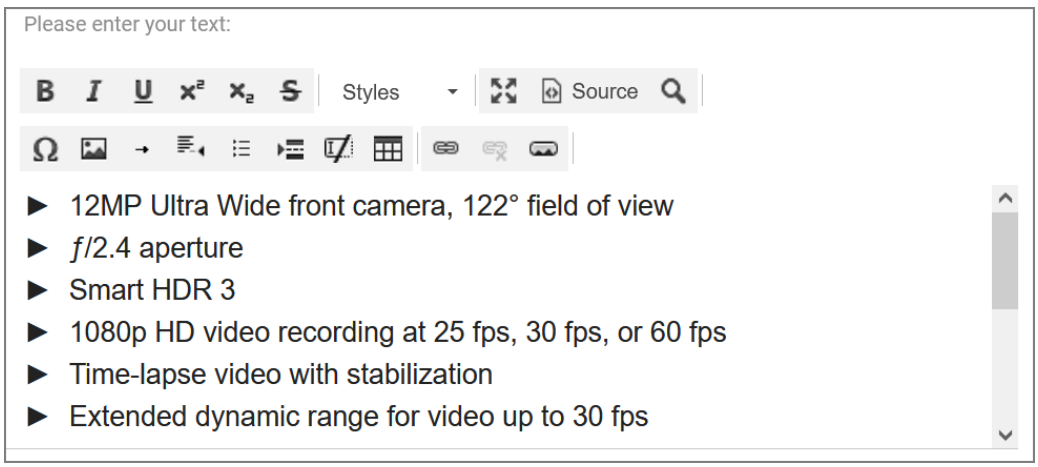
Note: You can enter a maximum of approximately 500,000 characters in a multiline input area.

The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
*Display name **	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.
Type spanning	Select this checkbox to display variables with an identical technical name and identical variable type in one column in the filter view. See Grouping variables, page 38.
Max. Characters	Define the maximum number of characters that can be entered in the field.
Columns Visible	**Note:** Visible only for multiline input areas. You can define the width of the input area by specifying the number of columns.
Lines Visible	**Note:** Visible only for multiline input areas. You can define the height of the input area by specifying the number of lines.
Complete width	**Note:** This is visible only for multiline input areas. Activate this checkbox to utilize the entire width of the datasheet for the input area in a one-column layout. If you activate this checkbox, leave the *Columns Visible* field empty.
Allow formatting	**Note:** Visible only for multiline input areas. This checkbox can only be activated while you create the variables. This displays a rich-text editor that allows users to format the text (for example, bold, italics, underline, and so on).

Name	Description
Editor configuration	**Note:** This is visible only for multiline input areas and if the *Allow formatting* checkbox is activated. You can select an editor configuration from the selection list. This allows you to define which formatting the user can use in the text field. For more information, see the configuration manual.
Reg. Exp. Validator	**Note:** Visible only for single input lines. Enter a regular expression in the field to check the entry for validity. For example, you can ensure that entries for BIC codes or e-mail addresses are provided in a specific format.
Input Size	**Note:** Visible only for single input lines. Defines the maximum number of characters displayed. If more characters are entered, the field gets a scroll bar.
Suffix	**Note:** Visible only for single input lines. You can enter a suffix (`Millimeter` or `%`, for example) that is attached to the field.
Help text	Define the help text that can be displayed to users.
Shared value	**Note:** This is visible only if the type is a data object type and localization is enabled (see Localization, page 14). **Note:** Note that the variable can be selected as language-neutral only while you create it. The *Shared value* field cannot be changed at a later stage! Activate the checkbox if you want the variable value to receive the same value in each localized data object version.
Inherit from parent	**Note:** This is visible only if the type permits inheritance (*Inheritance* dropdown list = `Multiple parents`). Activate this checkbox if you want the data record to adopt the values of a parent job or data object as a sub-job or sub data object. **Note:** The variables must have identical display names.
Default value	Enter the value to use as the default for the field. **Note:** Note that the default value is adopted only when you create an object (data object/job). Changing the default value does not have any effect on objects that have been created already.

Name	Description
Prefilling from custom structure	Select a custom structure and a defined attribute of this structure. By choosing *Key*, you can configure the pre-assignment of a value based on the branch number of the current user or on the name of the user. In the process, the unique names of the custom objects must match the branch number or the username. You can use the option *Override with* to link the display in the text field to a single selection that is connected to the same custom structure: In the single selection, you select one of the custom objects (for instance, the branch office). The text field is linked to the attribute *Address*. If a different affiliate is set in the single selection, the corresponding address is automatically displayed in the text field. If you activate the checkbox *Activate a fixed connection to the selected "custom object"*, the text field content is updated automatically if the custom structure is changed at a central location. Note that to do so, a value must be selected in the *Override with* selection list.
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the field only after a user has selected a specific value for a selection field on the datasheet.
Variable	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the selection field upon whose value you want to make the visibility dependent.
Value	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected to display the variable.



4.1.15 Single select and Multiselect

Use a *Single select or Multiselect* to allow users to select one or more predefined values.

Prerequisites:

- You require a custom structure that you can select as the data source.

The following parameters are provided when you create or change the variables:

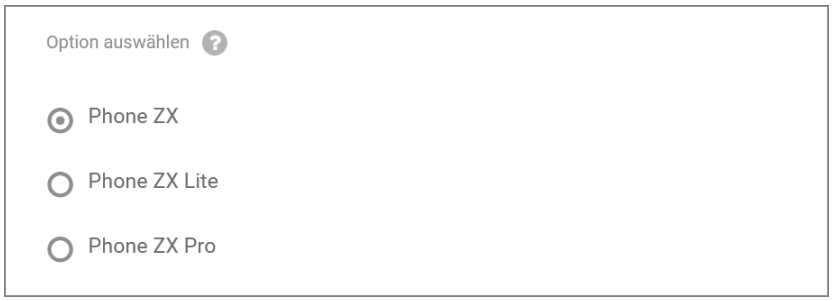
Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
*Display name **	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.
Type spanning	Select this checkbox to display variables with an identical technical name and identical variable type in one column in the filter view. See Grouping variables, page 38.
Selection Type	You can define the layout for the selection field. *Single Select:* • *Dropdown*: This creates a dropdown list. • *OptionBoxArea*: This creates an option box. • *OptionBoxAreaWithImages*: This creates an option box. The existing preview images for the custom structure can be displayed. *Multiselect:* • *Selection Box*: This creates a selection list. • *OptionBoxArea*: This creates an option box. • *OptionBoxAreaWithImages*: This creates an option box. The existing preview images for the custom structure can be displayed.
Data source	Select an existing custom structure as data source for the values of the selection field.

Name	Description
Dependent on parent selection	**Note:** Only visible if there is a single selection of the *Dropdown* type or a multiple selection of the *Selection Box* type and if a parent selection has already been created on the datasheet. Activate the checkbox if the display of the selections in this field depends on the setting in another selection field, for example the display of street names on the selection of a town in another selection field.
Parent selection	**Note:** This is visible only if the *Dependent on parent selection* checkbox is activated. Select the parent field containing the setting that the displayed values of the selection depend on.
Help text	Define the help text that can be displayed to users.
Shared value	**Note:** This is visible only if the type is a data object type and localization is enabled (see Localization, page 14). **Note:** Note that the variable can be selected as language-neutral only while you create it. The *Shared value* field cannot be changed at a later stage! Activate the checkbox if you want the variable value to receive the same value in each localized data object version.
Inherit from parent	**Note:** This is visible only if the type permits inheritance (*Inheritance* dropdown list = `Multiple parents`). Activate this checkbox if you want the data record to adopt the values of a parent job or data object as a sub-job or sub data object. **Note:** The variables must have identical display names.
Default value	Enter the value to use as the default for the field. **Note:** Note that the default value is adopted only when you create an object (data object/job). Changing the default value does not have any effect on objects that have been created already.
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the selection field only after a user has selected a specific value for a different selection field on the datasheet.
*Variable**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the selection field upon whose value you want to make the visibility dependent.

Name	Description
*Value**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected in order to display the variable.

Selection with *Multiselect*, display type: *SelectionBox*

Selection with *Singleselect*, display type: *OptionBoxArea*



4.1.16 Template

Use a *Template* variable to add a template from *Brand Template Builder* module to the datasheet. You can link the text boxes from the template to the variables on the datasheet to fill the corresponding text boxes automatically when editing the documents. Choose *Edit document* to open the document. If the values of the variables linked to the text boxes have been changed, a separate dialog box is displayed in which you can decide which content elements from the document are to be updated.

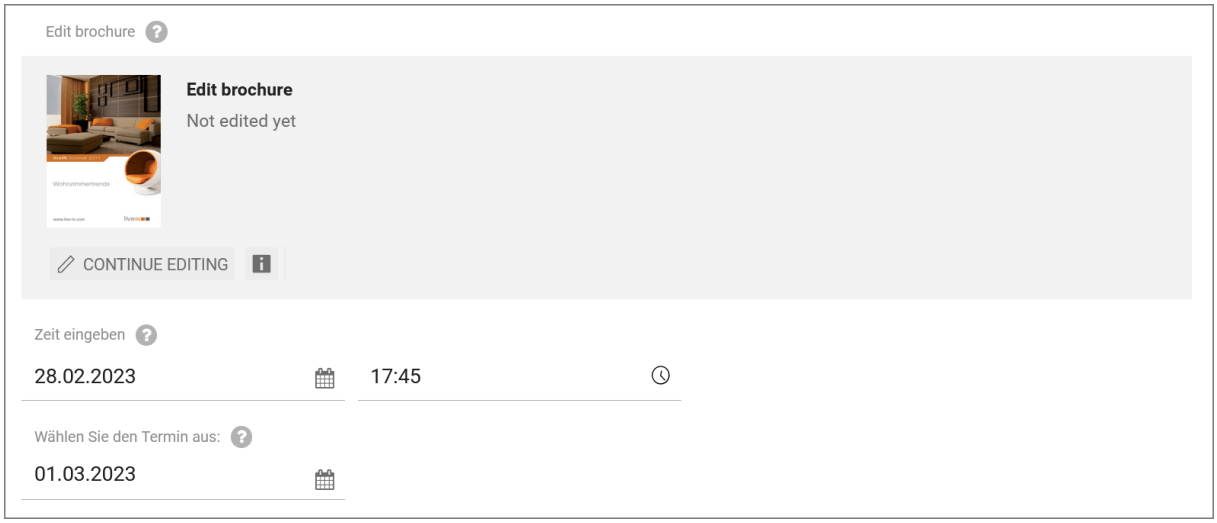
Note: A *BTB template* can be placed only on a one-column datasheet layout. To edit a document, you must have the corresponding rights.

The following parameters are provided when you create or change the variables:

Name	Description
Unique name *Technical Name*	**Note:** This is visible only if the variable is changed. Neither of the fields are visible when you create the variable. See Display, technical, and unique name, page 9.
*Display name**	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. See Display, technical, and unique name, page 9.
Type spanning	Select this checkbox to display variables with an identical technical name and identical variable type in one column in the filter view. See Grouping variables, page 38.
*Template ID**	Enter the ID number (excluding T-) for the template that you want to use.
Help text	Define the help text that can be displayed to users.
Shared value	**Note:** This is visible only if the type is a data object type and localization is enabled (see Localization, page 14). **Note:** Note that the variable can be selected as language-neutral only while you create it. The *Shared value* field cannot be changed at a later stage! Activate the checkbox if you want the variable value to receive the same value in each localized data object version.
Inherit from parent	**Note:** This is visible only if the type permits inheritance (*Inheritance* dropdown list = `Multiple parents`). Activate this checkbox if you want the data record to adopt the values of a parent job or data object as a sub-job or sub data object. **Note:** The variables must have identical display names.

Name	Description
Preview	As soon as you have entered a valid ID of a template, a preview image of the template is displayed.
Assign variables to boxes	You can assign a datasheet variable to the text boxes of the template that is in use.
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the variable only after a user has selected a specific value for a different selection field on the datasheet.
*Variable**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the dropdown list upon whose value you want to make the visibility dependent.
*Value**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected in order to display the variable.

Note: As soon as a new Web-to-Publish document is created, a preview of the created document is displayed on the datasheet.



4.1.17 User

You can use the *User* variable to ensure that the user can select another system user on the datasheet. If necessary, you can limit the selection to users in a user group.

Name	Description
Unique name, technical name	**Note:** This is visible only if the variable is changed. Neither of the fields are visible when you create the variable. For details see chapter 1.1.
Display Name	Define the name with which the variable is displayed on the datasheet. You can create any language versions that are required. For details see chapter 1.1.
User group	Specify a user group. Once defined, the processor can only select users from this group on the datasheet. If you do not define a group, the user can choose from all the available users.
Default value	Enter the value to use as the default for the field. **Note:** Note that the default value is adopted only when you create an object (data object or job). Changing the default value does not have any effect on objects that have been created already.
Help text	Define the help text that can be displayed to users.
Visibility	**Note:** This is visible only if a single-select or multi-select is already created for the type. Activate the checkbox *Visibility is depending on another variable* to display the selection field only after a user has selected a specific value for a different selection field on the datasheet.

4.2 System variables

When you create a new type, variables are created and placed on the datasheet automatically. Absolutely vital system variables include:

- Job/data object name
- Job/data object type
- Workflow

Other system variables can be removed from the datasheet but cannot be deleted.

Note: The technical names of the system variables are used in the list. There may be a different name in the *Display name* field.

Technical Name	Variable type/Description
WorkflowTiming	Workflow timings: This variable is used to display the start date, duration, and due date of the current workflow step.
ArticleNumber	Single inputline: This variable can be used to display an existing item number (for an asset, for example).
Owner	Bean property: This variable is used to display the assignee.
Description	Multiline input area: This variable provides an input field. By default, the maximum number of characters is limited to 1000 and the height is defined as three lines.
CreateDate	Datepicker: This variable is used to display the creation date.
Creator	Bean property: This variable is used to display the creator of the job or data object.
Deadline	Datepicker: This variable is used to make a date field available for the due date.
CurrentStepOverdueDate	Dynamic date: This variable is used to display the time (number of days) since the job or data object has been in a workflow step.
WorkflowOverdueDate	Dynamic date: This variable is used to display the number of days that the job or data object is overdue (in relation to the defined due date for the workflow step).
Job/data object name	Single inputline: This variable is used to provide an input field for the job or data object name.
JobIdFormatted	Constant variable: This variable is used to display the unique ID of the job or data object.

Technical Name	Variable type/Description
JobTypePseudoVariable	Job type: This variable is used to display the job type or data object type that is in use.
Comments	Chat desc from props: This variable is used to make the comment function available.
LastModificationDate	Datepicker: This variable is used to display the date of the last change.
SubJobs	Subjobs: This variable is used to make the functions required for sub-jobs or sub data objects available.
SystemPrice	Single inputline: This variable can be used to assign fixed or variable prices to the Service item types for the Marketing Shop.
default_media	Asset Selector: This variable is used to make the *Default media* variable available.
CurrentStepStartDate	Dynamic date: This variable is used to display the start date of the current workflow step.
JobState	Bean property: This variable is used to display the current workflow step.
TaskManager	Task Manager: This variable is used to make the functions for the Task Manager available; for Task Manager, page 143.
Themes	Description text: This variable can be used to assign one or more themes to the job or data object.
WorkflowObjectId	Workflow: This variable is used to display the workflow that is in use.
WorkflowStartDate	Dynamic date: This variable is used to display the start date of the workflow.

4.2.1 Task Manager

You can use tasks to plan and subdivide jobs or data objects in more detail. You can use the *Task Manager* and:

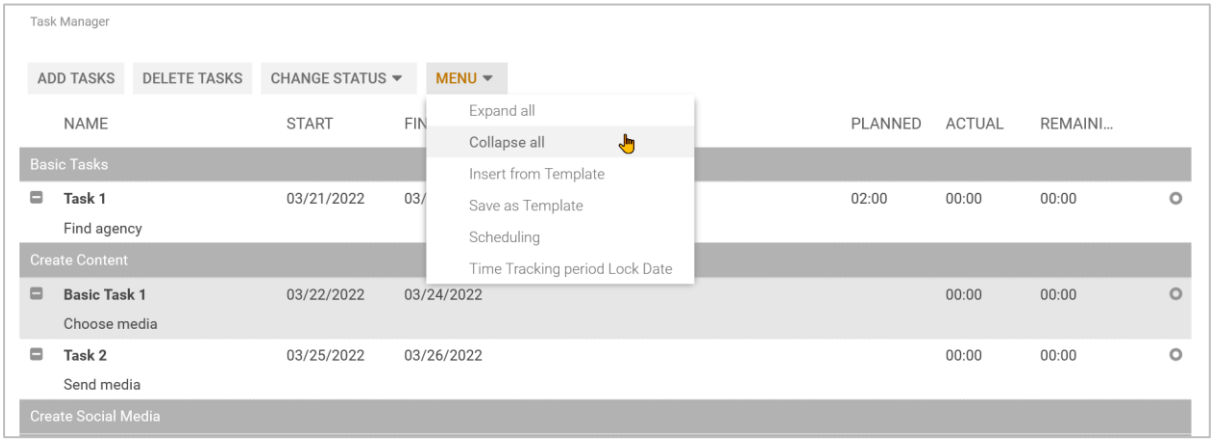
- Create the appropriate tasks for the workflow steps automatically using task templates.
- Enter the planned time required for a task.
- Post the actual time required for a task.
- Define the start and end date for the tasks and the assigned workflow steps.
- Assign individual tasks to a user for processing.

Note: The variable for the Task Manager is created by default. To use this function, you must place the element *Task Manager* on a single-column datasheet layout. Note that it is not possible to schedule individual workflow steps if the Task Manager is in use for that job. The start and end date of the workflow steps and tasks are then provided from the Task Manager.

When you create or configure the *Task Manager*, you define:

Name	Description
Unique name, technical name	See Display, technical, and unique name, page 9.
*Display name **	Define the name with which the Task Manager is displayed on the datasheet. See Display, technical, and unique name, page 9.
Include weekends	Activate this checkbox to take weekends into account for the automatic recalculation of dates.
Help text	Define the help text that can be displayed to users.
Default Task Templates	Select a task template that is suitable for the workflow. When you create a job, the individual task steps for the assigned task are created automatically. **Note:** To use a task template together with a workflow, the number of task steps must be identical to the number of workflow steps.
Visibility	Activate the checkbox *Visibility is depending on another variable* to display the variable only after a user has selected a specific value for a different selection field on the datasheet.
*Variable**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can choose the selection field upon whose value you want to make the visibility dependent.

Name	Description
*Value**	**Note:** This is visible only if the checkbox *Visibility is depending on another variable* is activated. You can specify which value must be selected in order to display the variable.



5 Appendix

5.1 Tips: Roles and rights

The rights management of the user roles for the Job Manager is an important component to provide administrators and users with different abilities and functions or just explicitly forbid and hide them in the user interface.

It is advisable to know exactly which rights to use and to test them thoroughly before putting the system into operation. This is because the internal designations of the rights do not always use meaningful names and sometimes only the combination of rights unlocks a function in the user interface. All rights to the Job Manager are completely tabulated in Chapter 11.1.7 of the Administration manual.

The following formerly used rights are still visible and selectable under *> Administration > Users & Groups > Rights & Roles*, but their assignment to a role has no functional meaning anymore:

- COMSUME_JMS_MESSAGES
- SHOW_BLUEPRINT_JOBS

Example of a "Clerk" role

The user can see and access all organizational units and their objects. The user may edit jobs in the workflow steps assigned to him and also create new jobs from the stored types/processes. They are not allowed to skip workflow steps. This user is not allowed to *complete*, *cancel*, *delete* jobs nor has access to the *Export* button or the *Total Filter* menu.

The *Advanced Search* is available (MANAGE_FILTERS) and a Public Filter may be saved from it. The BPMN workflow tab is not active. In the *Workflow* tab for standard jobs, only the next step may be selected and none may be skipped. The user is allowed to start a review on the asset in the datasheet.

- ACCESS_ALL_ORGS
- CREATE_JM_REQUEST
- EDIT_JM_REQUEST
- MANAGE_FILTERS
- MANAGE_REVIEW
- MANAGE_VIEWS
- MODULE_ACCESS
- SELECT_TYPE

Example of an "Assistant" role

Here the rights are even more restricted. Users with this role are only allowed to fill in the datasheet from jobs assigned to them and then forward them to the next workflow step. No new jobs can be created other than the preset default job type and the optional *Only Briefing*.

- CHANGE_JM_REQUEST
- CREATE_JM_REQUEST
- EDIT_JM_REQUEST
- MODULE_ACCESS

5.1.1 Important rights

See all (finished, deleted, canceled) jobs

Prerequisite: To be able to select and open any jobs of others in the filter menu, the user role in the Job Manager must have the ACT_LIKE_CREATOR right. This gives the user the right to also access running jobs/processes with the same authorizations of a creator.

Advanced search

Prerequisite: The user role has the MANAGE_FILTERS right so that the advanced search can be used.

Saving and publishing filters

Prerequisite: The user role has the PUBLISH_FILTER right.

Finishing a job

Prerequisite: The user role has the FINISH_JM_REQUEST right.

Copying a job

What is meant here is the user command in the datasheet menu: ... *> Clone and not Copy type in the DSE administration.*

Prerequisite: The user role has the COPY_JOB and CREATE_JM_REQUEST rights. For users without this combination of rights, the menu command remains hidden.

Canceling a job

Prerequisite: The user role has the CLOSE_JM_REQUEST right.

Delete a job

Prerequisite: • You are the creator of the job/process or have been assigned the appropriate authorization in your role in the Job Manager with the DELETE_JM_REQUEST right.

Undo deletion

You can reactivate jobs that have been marked as deleted.

Prerequisite: You are the creator of the job/process or have been assigned the corresponding authorization in your role in the Job Manager with the UNDELETE right.

De-activate/reactivate

Prerequisite: The user role has the DE_ARCHIVE right.

Publish filter

Prerequisite: The user role has the MANAGE_FILTERS and PUBLISH_FILTER rights.

Changing the creator of a job

Prerequisite: The user role has the CHANGE_JM_REQUEST right.

Customize column view

The *Add column* dropdown menu is available in the overview.

Prerequisite: The user role has the MANAGE_VIEWS right.

Save tasks as a template

Prerequisite: Users need the CREATE_TASK_TEMPLATES right in their user role to be able to save tasks as task templates.

Forwarding over multiple steps

Prerequisite: The SKIP_WORKFLOW_STEPS right is required for forwarding over multiple steps for the user role.

BPMN workflow tab visible

Prerequisite: Users require the ACCESS_WORKFLOW_TAB_BPMN right in their user role to see the BPMN-specific workflow tab.

Export

After the XML import has been completely removed from the product, this part still needs to be manually hidden to clean up the interface.

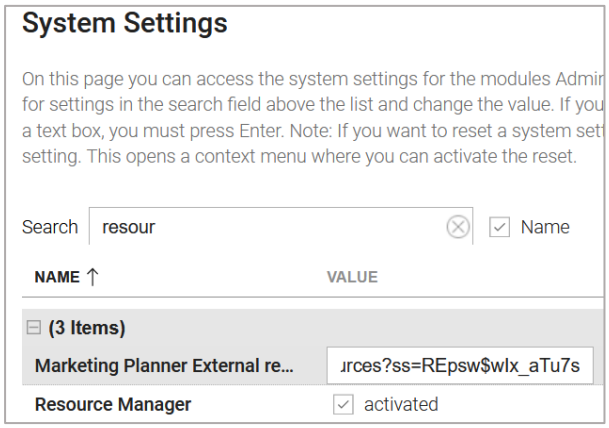
Prerequisite: You require the EXPORT_TO_XML right to initiate exports. For users without this right, the Export button and the *Exports* page remain hidden. Without the SEE_ALL_EXPORTS right, even administrators cannot see and download others' exports. Since the *Export Jobs* command also stands

in the *All Filters* dropdown menu, SML export can currently only be completely disabled for those roles that do not have either the CLOSE_JM_REQUEST or `DELETE_JM_REQUEST right.`

.

5.1.2 Activate Resource Management

To use the resource management, the entry `Resource Manager` must be activated under > *Administration* > *System configuration* > *System settings*.



This displays the *Resources* entry in the menu bar of the Job Manager module..

5.2 Visibility by Management Level

5.2.1 Job view with restriction by organizational unit

Organizing data based on the rights and roles system is not always easy. We will show you on the basis of a use case how to make it possible for supervisors to see only the jobs of their team members, and supervisors above them in the hierarchy to see the jobs of a correspondingly larger group of people.

Goal of the configuration

The CMO wants to be able to see and edit their own jobs and also have read access to all the jobs that the marketing managers in their sub-departments have created. According to the organization chart above, i.e. "Head of Marketing DACH" and "Head of Marketing Germany".

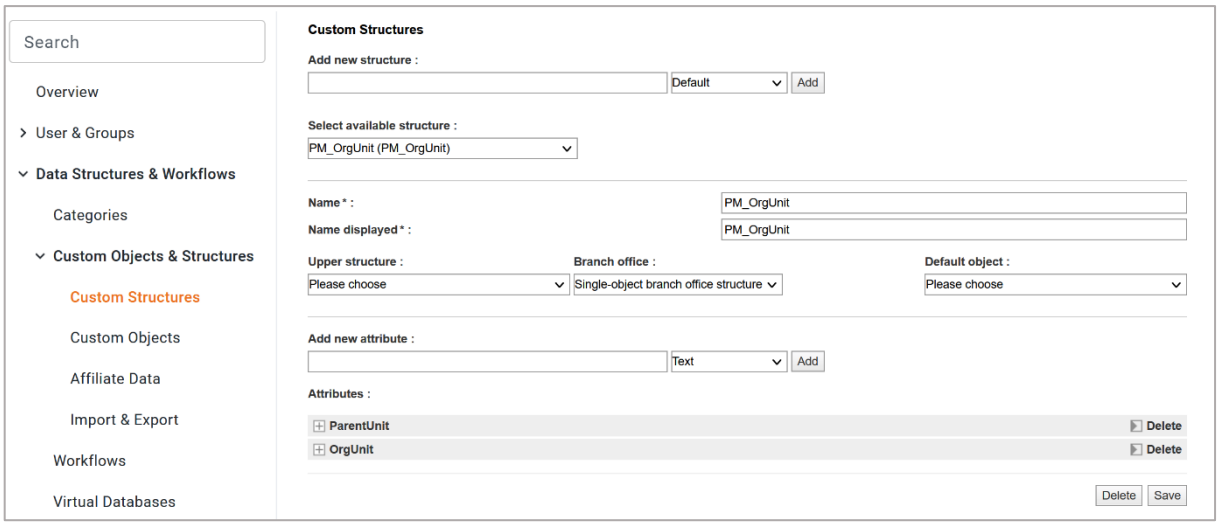
Managers who do not head any other sub-departments are only allowed to see and edit the jobs that are in their area of responsibility.

Configuration steps

You need a new modifiable structure with two attributes to uniquely assign users to an organizational unit.

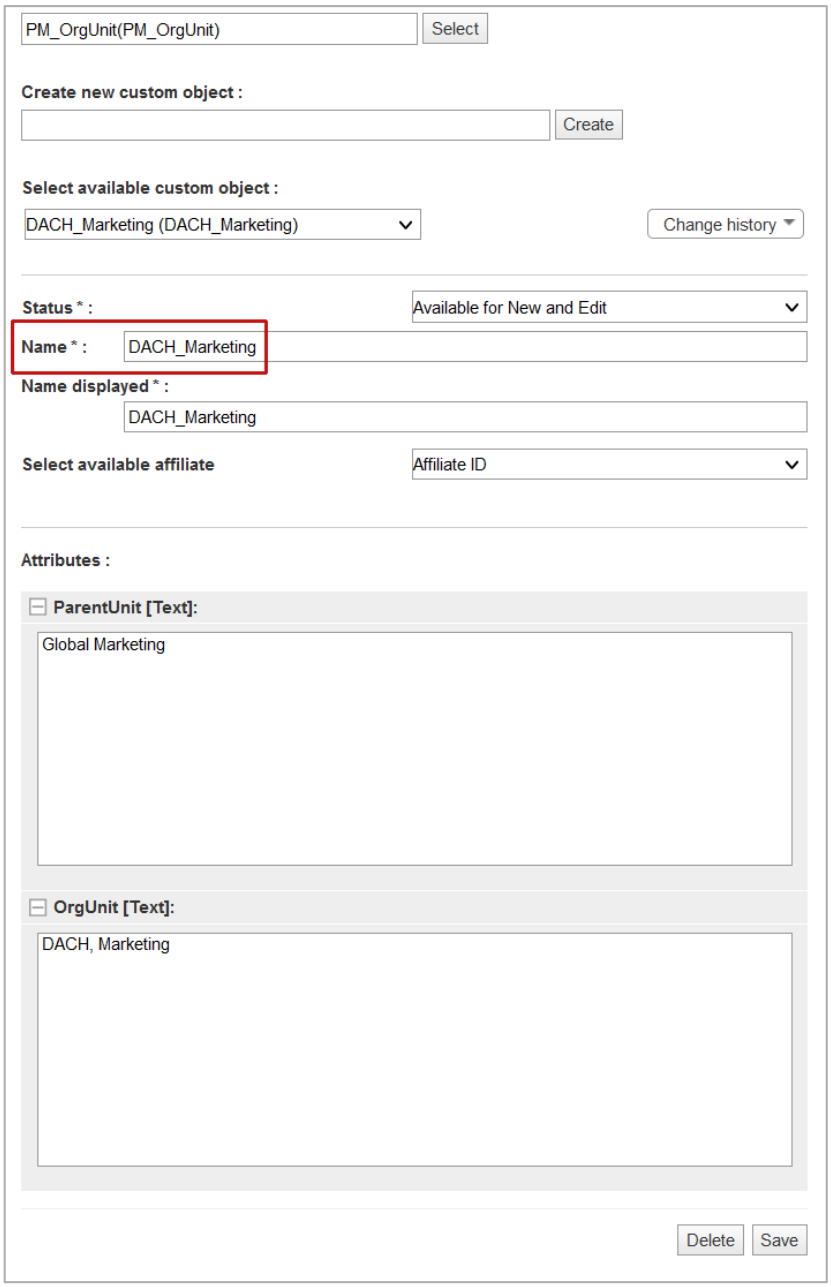
1. Navigate *to Administration > Data Structures & Workflows > Modifiable Structures.*
2. Press Create.
3. Create the `OrgUnit` attribute.
4. To be able to assign users to a parent unit, create the `ParentUnit` attribute.

> **Note**: Normally, the assignment of an organizational unit would already suffice. In order to map the hierarchical classification of the user in their organization, it makes sense to additionally specify `ParentUnit`. In order to be able to filter even more finely, it is conceivable to use further attributes (e.g., cost center).



For each organizational unit, you must now create a modifiable object and fill in the attributes accordingly.

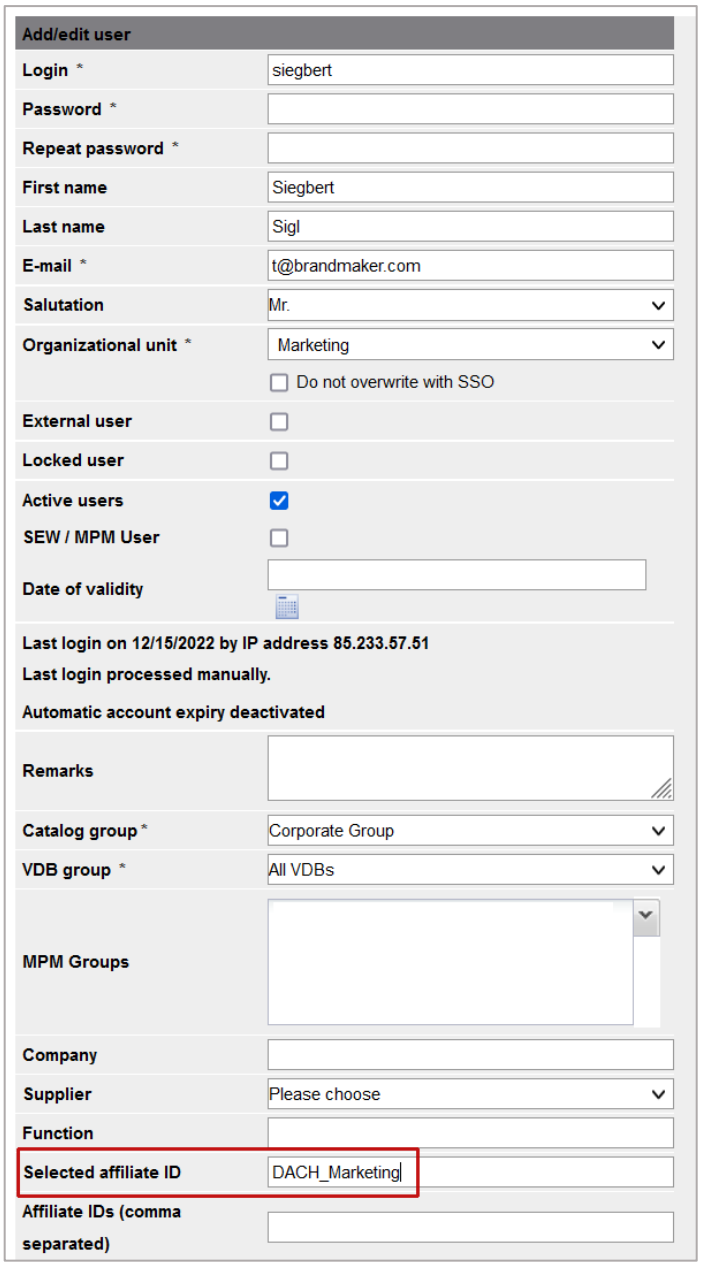
1. Navigate to *Administration > Data Structures & Workflows > Modifiable Objects*.
2. Create the modifiable objects for the modifiable structure you just created.
3. Also fill in the `OrgUnit` and `ParentUnit` fields.



The technical name outlined in red above will be stored in all user accounts that belong to this organizational unit in the following step.

4. Copy the technical name from the name field to avoid input errors.
5. Store the OrgUnit by pasting it from the clipboard in the user account under *Administration > Users & Groups > Users in the Selected Affiliate ID field and click Update to save*.

   After saving, the entry appears in the field below *Affiliate IDs (comma separated).*
6. Store this value for all users of the OrgUnit "DACH, Marketing".



Note: The field content for *Selected Affiliate ID* (here `DACH_Marketing`) must correspond exactly to the technical name of the custom object previously created for this department, and must not be the displayed name (*Display name* * field).

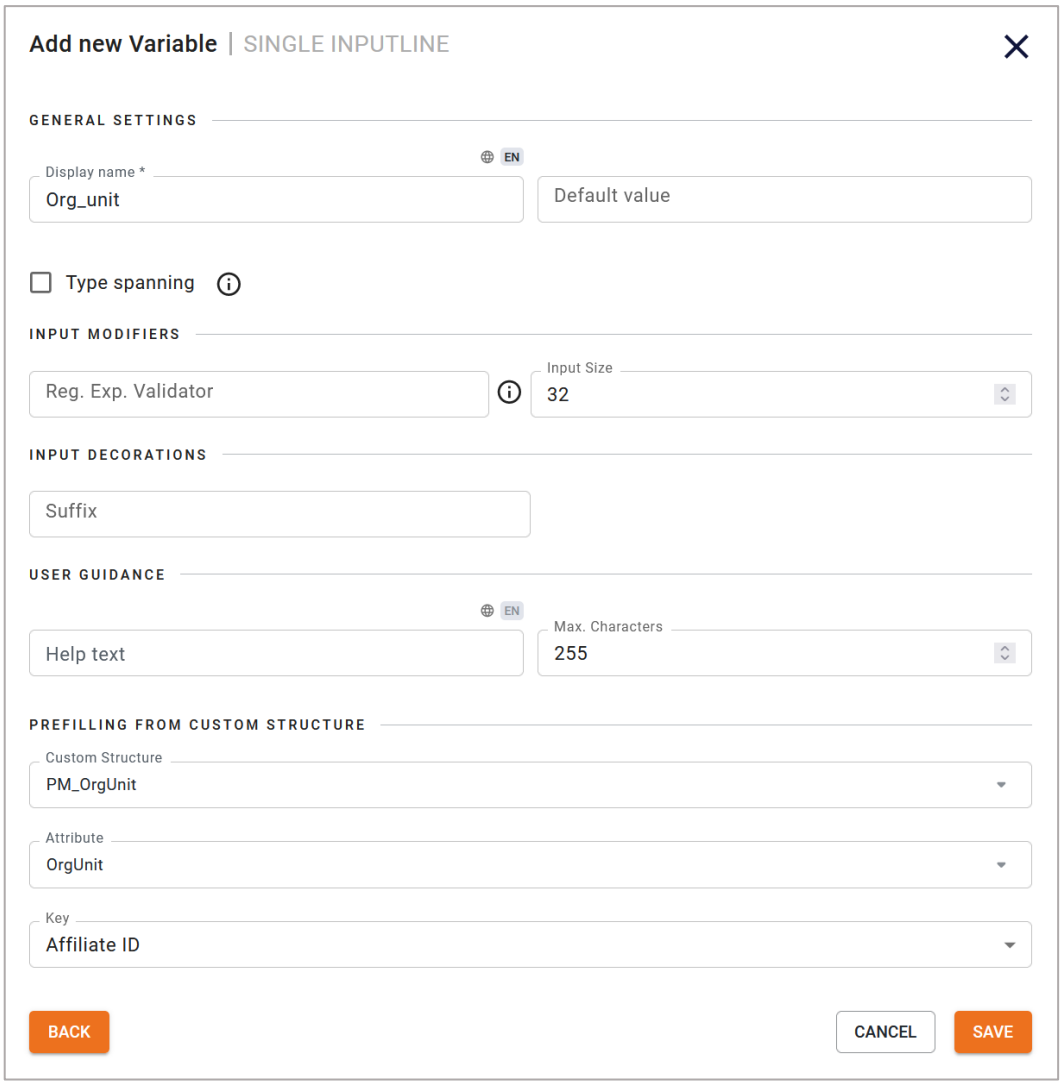
7. Extend all datasheet layouts of eligible job types with two fields that automatically pre-populate the attributes from the modifiable object when a new job is created later.

Customize datasheet

1. Create a variable (Type: Single Inputline) for the OrgUnit.

The variable name can be freely chosen, but when pre-populating from the modifiable structure, you must stick to the previously defined values.

2. Place the variable on the datasheet.

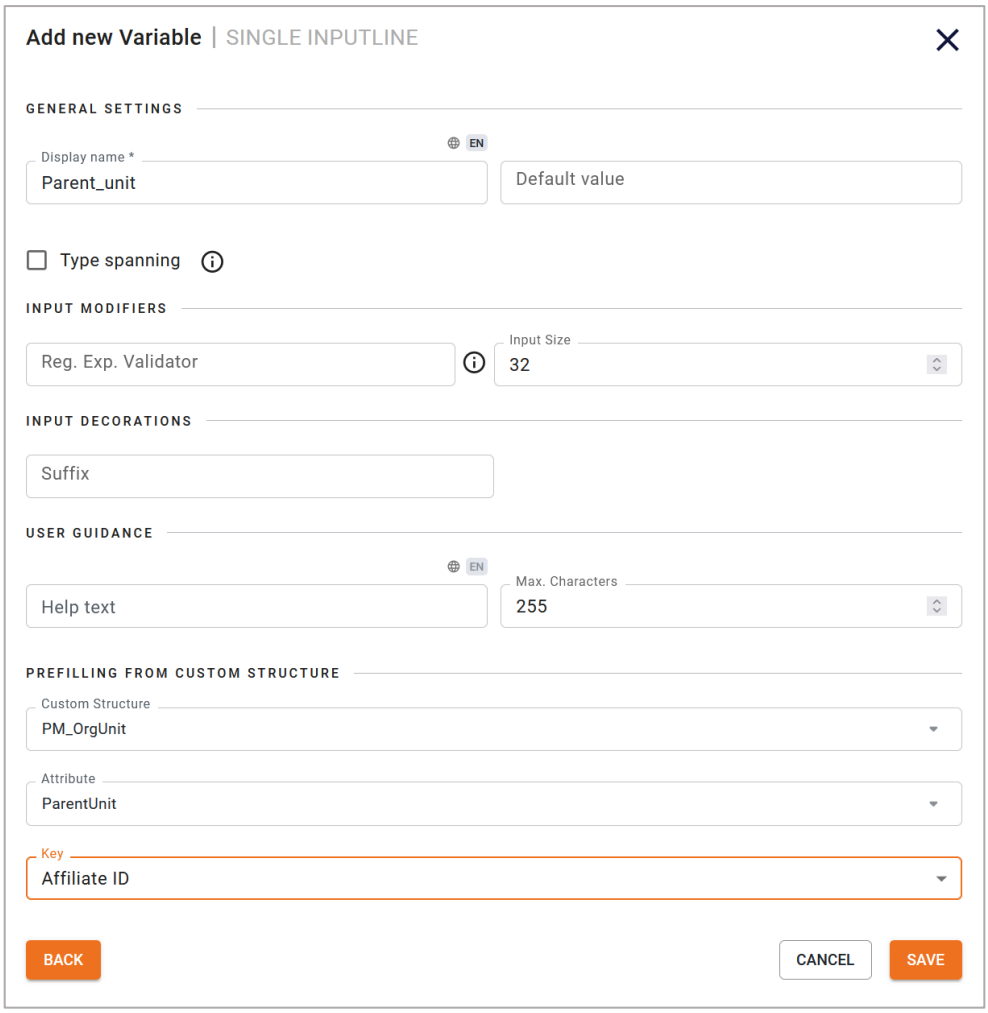


The figure above shows the variable for the OrgUnit.

3. Now create a variable (Single Inputline) for the ParentUnit of `DACH_Marketing` according to the same principle.

The variable name can be freely chosen, but when pre-populating from the modifiable structure, you must stick to the previously defined values.

The figure on the following page shows the creation of the variable for the ParentUnit of the OrgUnit.

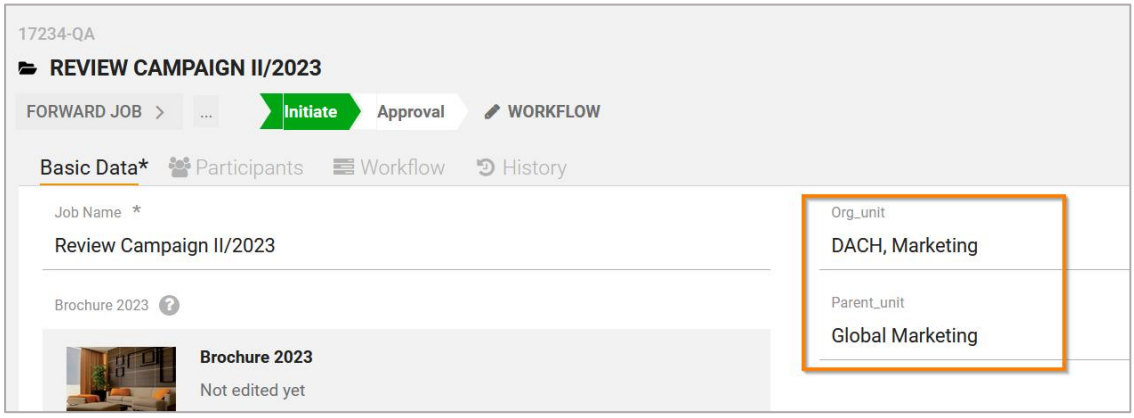


4. Publish the changes and create a new job based on this job type.

Each time a new job is created for the department head for whom a value has been stored in the *Selected Affiliate ID* field in the account, the open job datasheet will display the attribute values of their organizational unit and the ones above it.

As a result of the previous steps, such a job is uniquely assigned to both a user and the organizational unit to which it belongs.

Example in datasheet



5.2.2 Follow-up

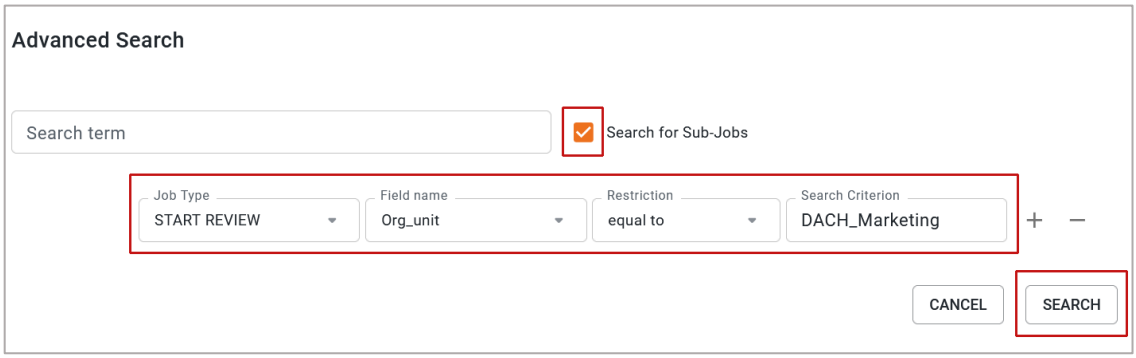
You need to make sure that a user can only see and edit their own jobs and those of their subordinate departments. For this purpose, a filter is set for this user, which displays only the jobs created by users from the subordinate organizational units.

Prerequisite:

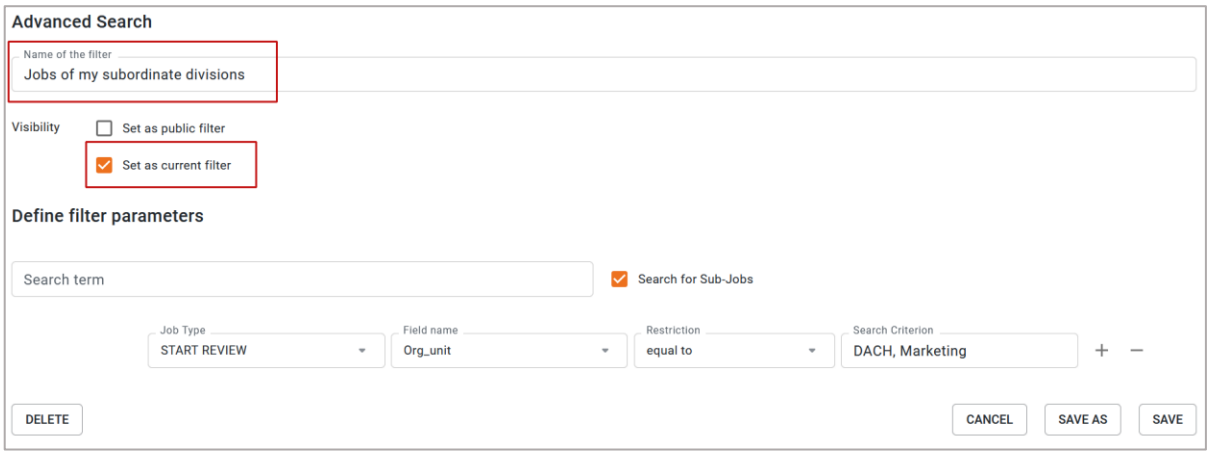
- To be able to set up this filter, the user must initially have the MANAGE_FILTERS right in their role.

Note: The MANAGE_FILTERS right must be revoked from the user after the filter has been created and tested to prevent them from creating their own filters for jobs from other organizational units.

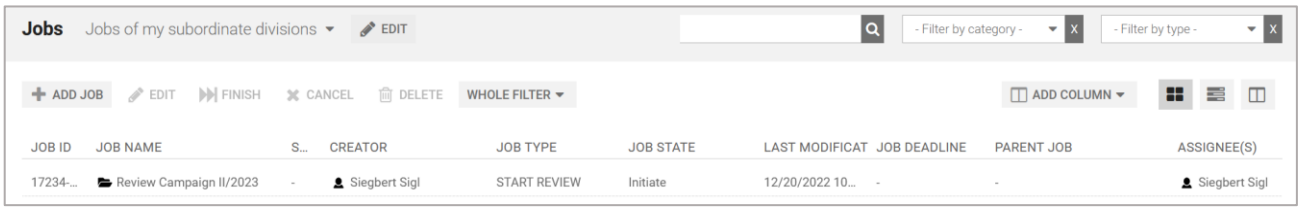
1. To create the filter, log in as this user.
2. On the Job Manager home page, start an *Advanced Search*, in the Filter menu and click *Edit*.



3. Fill in the search as shown in the figure above without a search term.
   The search criterion here is the displayed name and not the technical name as before, i.e., `DACH, Marketing`.
4. Click *Search*.
5. Set the checkbox at *Set as current filter and press Save*.



Desired result: The logged-in CMO now sees the appropriately prepared jobs of the subordinate organizational unit "DACH, Marketing" with this filter.



6. Optional: If you extend the filter shown above with further search strings with OR, which each contain the visible name of another organizational unit as a search criterion, then further jobs of other, subordinate organizational units will also be displayed, provided they have been created accordingly.
7. Remove MANAGE_FILTERS from the role.

To prevent the user from setting filters on jobs and organizational units that they should not see, the MANAGE_FILTERS right must finally be removed from their role.

Click *Update* after removing it.